AGRICULTURE CROP RECOMMENDATION SYSTEM USING MACHINE LEARNING

Submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering Degree in Computer Science and Engineering

By

KANIKA BHATNAGAR
(Reg. No. 38110232)
MAMILLA JAAHNAVI
(Reg No. 38110292)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600119, TAMILNADU

MARCH 2022

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC
(Established under Section 3 of UGC Act, 1956)
JEPPIAAR NAGAR, RAJIV GANDHI SALAI
CHENNAI– 600119
www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **KANIKA BHATNAGAR (38110232) and MAMILLA JAAHNAVI (38110292)** who carried out the project entitled "**Agriculture Crop Recommendation System using Machine Learning**" under my supervision from December 2021 to March 2022.

Internal Guide:
Dr. B. U. Anu Barathi, M.E., PhD

Head of the Department:

Dr. L. Lakshmanan M.E., Ph.D.

Dr. VIGNESHWARI, M.E., Ph.D.

Submitted for Viva voce Examination held on_____

Internal Examiner **External Examiner**

DECLARATION

I **Kanika Bhatnagar (38110232) and Mamilla Jaahnavi (38110292)** hereby declare that the Project Report entitled **Agriculture Crop Recommendation System using Machine Learning** is done by me under the guidance of **Dr. B. U. Anu Barathi., M.E., PhD** professor, Dept of CSE at **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE:

PLACE: Chennai **SIGNATURE OF THE CANDIDATE**

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph.D.**, **Dean**, School of Computing, **Dr. L. Lakshmanan M.E., Ph.D. and Dr. S. Vigneshwari M.E., Ph.D.** Heads of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. B. U. Anu Barathi, M.E., Ph.D.,** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

In Indian economy and employment agriculture plays major role. The most common problem faced by the Indian farmers is they do not opt crop based on the necessity of soil, as a result they face serious setback in productivity. This problem can be addressed through precision agriculture. This method takes three parameters into consideration, viz: soil characteristics, soil types and crop yield data collection based on these parameters suggesting the farmer suitable crop to be cultivated. Precision agriculture helps in reduction of non-suitable crop which indeed increases productivity, apart from the following advantages like efficacy in input as well as output and better decision making for farming. Crop yield prediction incorporates forecasting the yield of the crop from past historical data which includes factors such as temperature, relative humidity, ph., rainfall and area (Hectares). This method gives solutions like proposing a recommendation system through an ensemble model with majority voting techniques using Random Forest and K Nearest Neighbor as learner to recommend suitable crop based on soil parameters with high specific accuracy and efficiency.

LIST OF ABBREVIATIONS

SR. NO	ABBREVIATION	EXPANSION
1	KNN	K Nearest Neighbor

LIST OF TABLES

LIST OF FIGURES

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 OUTLINE

Agriculture is the one amongst the substantial area of interest to society since a large portion of food is produced by them. Currently, many countries still experience hunger because of the shortfall or absence of food with a growing population. Expanding food production is a compelling process to annihilate famine. Developing food security and declining hunger by 2030 are beneficial critical objectives for the United Nations. Hence crop protection; land assessment and crop yield prediction are of more considerable significance to global food production.

This project uses Python 3.6 for the programming in a scientific development environment called the PyCharm. Various data manipulation, machine learning and visualization packages are used to create and analyze the dataset using a traditional machine learning model. A Data Visualization tool called Tableau is used to interpret the results provided by the model after analysis to represent and act as a proof for the intended result.

1.2 MODEL IDE

PyCharm is an integrated development environment (IDE) used in computer programming, specifically for the Python programming language. It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems (VCSes), and supports web development with Django as well as data science with Anaconda.

1.3 PROBLEM STATEMENT

Problem # 1. Instability:

Agriculture in India is largely depending on monsoon. As a result, production of food-grains fluctuates year after year. A year of abundant output of cereals is often followed by a year of acute shortage. This, in its turn, leads to price income and employment fluctuations. However, for the thirteen years, in successive (1987-88 to 1999-00) a normal monsoon has been observed.

Problem # 2. Cropping Pattern:

The crops that are grown in India are divided into two broad categories: food crops and non-food crops. While the former comprises food-grains, sugarcane and other beverages, the latter includes different kinds of fibers and oilseeds. In recent years there has occurred a fall in agricultural production mainly due to fall in the output of non-food articles. Moreover, rabi production has become as important as kharif production in the late 1990s. In 1999-2000, for example, of the total grain production of 209 mn. tones, rabi accounted for 104 mn. tones. This indicates a structural change in agricultural production.

Problem # 3. Land Ownership:

Although the ownership of agricultural land in India is fairly widely distributed, there is some degree of concentration of land holding. Inequality in land distribution is also due to the fact that there are frequent changes in land ownership in India. It is believed that large parcels of land in India are owned by a- relatively small section of the rich farmers, landlords and money-lenders, while the vast majority of farmers own very little amount of land, or no land at all. Moreover, most holdings are small and uneconomic. So, the advantages of large-scale farming cannot be derived and cost per unit with ‘uneco-nomic’ holdings is high, output per hectare is hectare is low. As a result, peasants cannot generate sufficient marketable surplus. So, they are not only poor but are often in debt.

Problem # 4. Sub-Division and Fragmentation of Holding:

Due to the growth of population and breakdown of the joint family system, there has occurred continuous sub-division of agricultural land into smaller and smaller plots. At times small farmers are forced to sell a portion of their land to repay their debt. This creates further sub-division of land. Sub-division, in its turn, leads to fragmentation of holdings. When the size of holdings become smaller and smaller, cultivation becomes uneconomic. As a result, a major portion of land is not brought under the plough. Such sub-division and fragmentation make the efficient use of land virtually impossible and add to the difficulties of increasing capital equipment on the farm. All these factors account for the low productivity of Indian agriculture.

Problem # 5. Land Tenure:

The land tenure system of India is also far from perfect. In the pre-independence period, most tenants suffered from insecurity of tenancy. They could be evicted any time. However, various steps have been taken after Independence to provide security of tenancy.

Problem # 6. Conditions of Agricultural Labourers:

The conditions of most agricultural labourers in India are far from satisfactory. There is also the problem of surplus labour or disguised unemployment. This pushes the wage rates below the subsistence levels.

Problem # 7. Other Problems:

There are various other problems of Indian agriculture.

These are related to:

(i) The systems and techniques of farming,

(ii) The marketing of agricultural products and

(iii) The indebtedness of the farmers.

1.4 PROPOSED SYSTEM

- Our proposed system is an application which predicts name of the crop as well as calculate its corresponding yield.
- Name of the crop is determined by several features like temperature, humidity, wind-speed, rainfall etc. and yield is determined by the area and production.
- In this project KNN and Random Forest is used for prediction. It will attain the crop prediction with best accurate values.

1.5 OBJECTIVE

In Indian economy and employment agriculture plays major role. The most common problem faced by the Indian farmers is they do not opt crop based on the necessity of soil, as a result they face serious setback in productivity.

This problem can be addressed through precision agriculture. This method takes three parameters into consideration, viz: soil characteristics, soil types and crop yield data collection based on these parameters suggesting the farmer suitable crop to be cultivated.

Precision agriculture helps in reduction of non-suitable crop which indeed increases productivity, apart from the following advantages like efficacy in input as well as output and better decision making for farming.

This method gives solutions like proposing a recommendation system through an ensemble model with majority voting techniques using Random Forest and K - Nearest Neighbor as learner to recommend suitable crop based on soil parameters with high specific accuracy and efficiency.

CHAPTER 2

LITERATURE SURVEY

Dr. Y. Jeevan Nagendra Kumar depicts the utilization of different ML methods to speed up. This provides us with a thought of a very much arranged harvest that can be planted in the mountains.

Sujata Kullur clarified that past yield data doesn't anticipate gathering and reaping. It likewise gives ranchers the data they need to handle climate data about crops that can be reaped, and to deal with the substance of the harvest at the perfect opportunity.

The conduct of the mentors of **Judah Khan**, the numerous choice trees, was obvious, and the division or issuance turned into a penance as indicated by the quantity of divisions.

Amrita Vishwa Vidyapeetham clarifies the likely advantages of IoT casualties to the climate at a decent cost, and clarifies the expected effects of its utilization and use. They utilized a Hadoop sound card. Data, research, and related data don't assist with recognizing bugs and bugs.

Pavan Patil discloses to us that "Testament Tree and KNN" utilized the power calculation. This strategy assists with guaranteeing that the plants in a specific region are typically full grown.

Fatima Sadatulla considered mathematical creation and clarified the various approaches to digging for the advancement of time. Since a great deal of information assortment can be troublesome, K implies that it utilizes registering to deal with a ton of

data. Genuine numbers are utilized to choose the collect as a long-lasting issue. Furthermore, they center around authority abilities and forefront.

5

Yash Sanghvi states the usage of agricultural information with data processing and visual data processing techniques are delineate. Data processing in agriculture is employed for analyzing the varied organic phenomenon and abiotic factors.

Nirupama Mallick clarified that it assumes a significant part in data handling in the rural area. They need to decide specific computational abilities, for example, K normal, SVM, ANN. Collect is anticipated predominantly founded on the climate, with a genuine score of 95%, C4.5.

Rakesh Kumar discusses the factors deciding the crop choice like production rate, value and government policies are mentioned. The projected methodology could improve web yield rate of crops.

Use of ecological variables, emphasis examination (RA), direct relapse calculation (LR), and collected item reviles.

Satish Babu states the wants and designing required for developing a computer code model for exactitude agriculture is mentioned. This approach can end up in the farmer and crop-level support advisories through devices like mobile phones and tablets.

Abhay V. Raorane explains how to boost harvest productivity by applying several data mining tactics. The processes they used for order, like as ANN, SVM, and K implies, among others.

Anshal Savla explains during this paper, varied algorithms are used associated with classification techniques of information mining. A comparative analysis is finished to indicate that classification algorithmic rule is best fitted to predicting the yield with relevance classification techniques.

Aakunuri Manjula states the need for crop yield prediction and it facilitate in a very nation’s strategic political opinions in exactitude agriculture.

Yash Sanghvi states the usage of agricultural information with data processing and visual data processing techniques are delineate. Raw information/data/information collected from the statistics analysis has helped in crucial the information and by victimization self-organizing and multi-dimensional maps to scale back data.

CHAPTER 3

METHODOLOGY

3.1 INTRODUCTION TO MACHINE LEARNING

Machine learning is a branch of computer science that employs statistical techniques to enable computer systems to "learn" (i.e., progressively improve performance on a specific task) from data without being explicitly programmed. Arthur Samuel coined the term "machine learning" in 1959. Machine learning, which evolved from the study of pattern recognition and computational learning theory in artificial intelligence, investigates the study and construction of algorithms that can learn from and make predictions on data – such algorithms overcome strictly static programme instructions by making data-driven predictions or decisions, by building a model from sample inputs.

Machine learning is used in a variety of computing tasks where designing and programming explicit algorithms with high performance is difficult or impossible; examples include email filtering, network intruder detection, and computer vision.

Machine learning is closely related to (and frequently overlaps with) computational statistics, which is also concerned with making predictions using computers. It has strong ties to mathematical optimization, which provides the field with methods, theory, and application domains. Machine learning is frequently confused with data mining, the latter of which focuses on exploratory data analysis and is referred to as unsupervised learning.

Machine learning is a method used in the field of data analytics to create complex models and algorithms that lend themselves to prediction; in commercial use, this is

known as predictive analytics.

Through learning from historical relationships and trends in the data, these analytical models enable researchers, data scientists, engineers, and analysts to "produce reliable, repeatable decisions and results" and uncover "hidden insights."

3.2 TRAINING THE DATA

There are basically two widely-used types of training that can be done to create a model:

i. Supervised Learning
ii. Un-supervised Learning

3.2.1 SUPERVISED LEARNING

The machine learning task of learning a function that maps an input to an output based on example input-output pairs is known as supervised learning. It derives a function from labelled training data, which consists of a set of training examples. Each example in supervised learning is a pair consisting of an input object (typically a vector) and a desired output value (also called the supervisory signal). A supervised learning algorithm examines the training data and generates an inferred function that can be used to map new examples. In an ideal scenario, the algorithm will be able to correctly determine the class labels for unseen instances. This necessitates that the learning algorithm generalize from the training data to previously unseen situations in a "reasonable" manner.

3.2.2 UNSUPERVISED LEARNING

The machine learning task of inferring a function that describes the structure of

"unlabeled" data is known as unsupervised machine learning (i.e., data that has not been classified or categorized). Because the examples provided to the learning algorithm are unlabeled, there is no simple way to assess the accuracy of the structure produced by

the algorithm—a feature that distinguishes unsupervised learning from supervised learning and reinforcement learning.

The type of training used in this model is **SUPERVISED LEARNING.**

3.3 METHODS IN SUPERVISED LEARNING

Supervised Learning mainly consists of two methods,

- Classification
- Regression

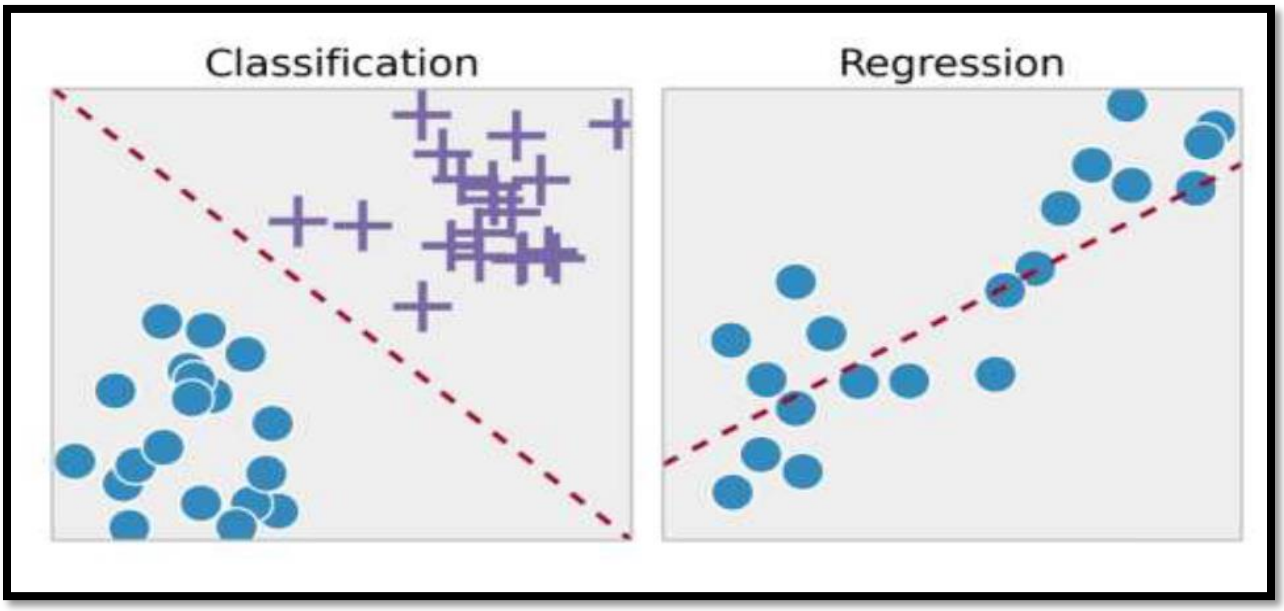


Fig 3.1 Classification vs Regression

3.3.1 CLASSIFICATION

Classification is the problem in machine learning of determining which of a set of categories (sub-populations) a new observation belongs to, based on a training set of data containing observations (or instances) whose category membership is known.

Examples include categorizing an email as "spam" or "nonspam," and assigning a diagnosis to a patient based on observed characteristics (gender, blood pressure, presence or absence of certain symptoms, etc.). Pattern recognition is demonstrated by classification. A classifier is an algorithm that implements classification, particularly in a concrete implementation. Clustering is the corresponding unsupervised procedure, and it involves categorizing data based on some measure of inherent similarity or distance.

3.3.2 REGREESSION

Regression analysis calculates the dependent variable's conditional expectation given the independent variables – that is, the average value of the dependent variable when the independent variables are held constant. Less frequently, the emphasis is on a quantile or other location parameter of the dependent variable's conditional distribution given the independent variables. The regression function, which is a function of the independent variables, must be estimated in all cases. Regression analysis is widely used for forecasting and prediction. It is also used to determine which independent variables are related to the dependent variable and to investigate the nature of these relationships. Regression analysis can be used to infer causal relationships between independent and dependent variables in limited circumstances. This, however, can lead to illusions or false relationships, so exercise caution; for example, correlation does not prove causation.

The type used in this model is **CLASSIFICATION** and so, more focus will be given on it.

3.4 SYSTEM ARCHITECTURE

Crop yield is extremely useful information for farmers. Understanding the yield can help you save money by lowering your losses. Crop yields were previously predicted by experienced farmers. The proposed system works in a similar manner. It uses previous data to forecast future yields. Crop productivity is most affected by weather and fertilizers. The accuracy of this prediction is determined by the accuracy of the information provided.

As a result, the proposed method predicts yield and reduces loss. The expected system assumes the role of an experienced farmer. It is, however, more precise and takes into account a number of additional parameters. There are several factors to consider, including soil condition, weather forecast, pH, humidity, and yield.

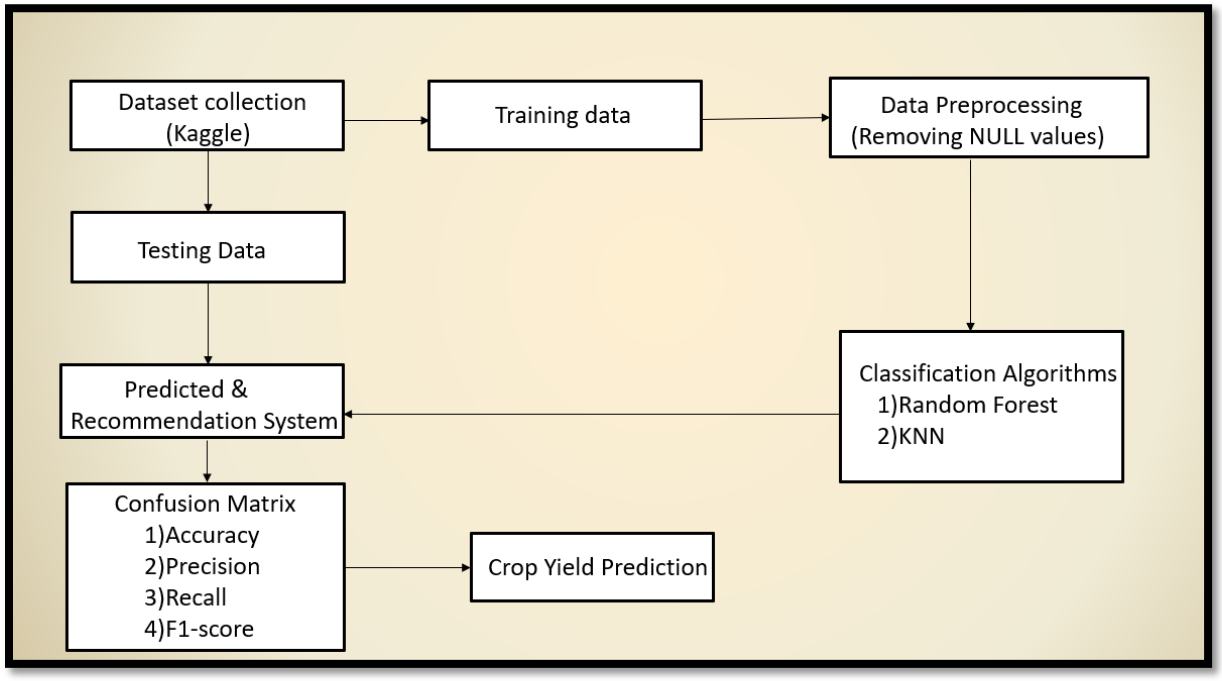


Fig 3.2 System Architecture

CHAPTER 4

ALGORITHMS

4.1 RANDOM FOREST

Random Forest is a supervised machine learning technique that can be applied to Classification and Regression problems. It is based on the concept of ensemble learning, which is the process of combining multiple classifiers to solve a complex problem and improve the model's performance.

Random Forest is a classifier that uses the average of a number of decision trees from a given dataset to improve the predictive accuracy of the data set. Instead of relying on a single decision tree, the random forest takes the predictions from each tree and predicts the final output based on the majority vote of predictions.

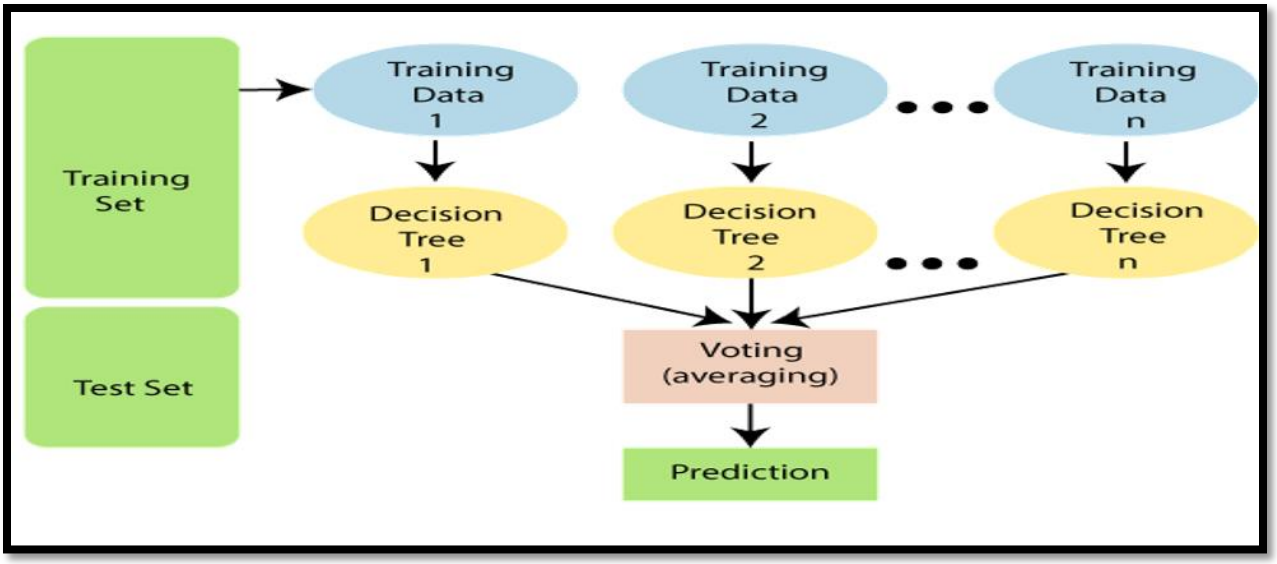


Fig 4.1 Random Forest

4.2 K NEAREST NEIGHBOUR

The k-nearest neighbors' algorithm (K-NN) is a nonparametric method for classification that is used in N pattern recognition. The result is class membership. A majority vote of its neighbors classifies an object, with the object assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). If k = 1, the object is simply assigned to the class of the object's single nearest neighbor.

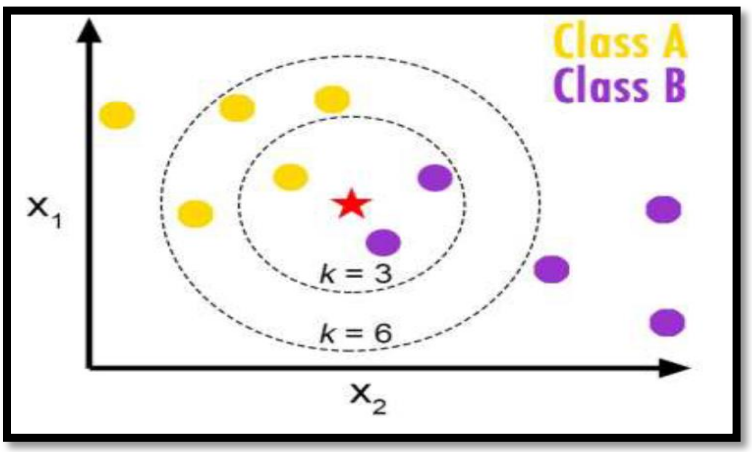


Fig 4.2 KNN

4.3 DATASETS

The dataset comprising the soil specific attributes which are collected from Kaggle. In addition, similar online sources of general crop data were also used. The crops considered in our model include rice, maize, chickpea, kidney beans, pigeon peas, moth beans, mungbean, black gram, lentil, pomegranate, banana, mango, grapes, watermelon, muskmelon, apple, orange, papaya, coconut, cotton, jute, coffee gives an analysis of the dataset. The number of instances of each crop available in the training dataset is depicted. The attributes considered where Nitrogen(N), Potassium(K), Phosphorus(P), Temperature, Humidity, Ph and Rainfall.

The above stated parameters of soil play a major role in the crop's ability to extract water and nutrients from the soil. For crop growth to their fullest potential, the soil must provide a satisfactory environment for it. Soil is the anchor of the roots. Nitrogen is largely responsible for the growth of leaves on the plant. Phosphorus is largely responsible for root growth and flower and fruit development. Potassium is a nutrient that helps the overall functions of the plant perform correctly. Temperature is a key factor in plant growth and development. Along with the levels of light, carbon dioxide, air humidity, water and nutrients, temperature influences plant growth and ultimately crop yields. Humidity directly influences the water relations of plant and indirectly affects leaf growth, photosynthesis, pollination, occurrence of diseases and finally economic yield. The level of acidity or alkalinity (Ph) is a master variable which affects the availability of soil nutrients. The activity of microorganisms presents in the soil and also the level of exchangeable aluminum can be affected by PH. rainfall can also determine how fast a crop will grow from seed, including when it will be ready for harvesting. A good balance of rain and proper irrigation can lead to faster-growing plants, which can cut down on germination time and the length between seeding and harvest. Hence for the following

reasons the above stated parameters are considered for choosing a crop.

4.4 PACKAGES

The packages used in this model include:

- Pandas
- Scikit-learn
- Scikit-plot
- Matplotlib’s pyplot
- Seaborn

4.4.1 DATA MANIPULATION PACKAGES:

- ***Pandas***

- Pandas is a Python package that provides fast, flexible, and expressive data structures that make it simple and intuitive to work with structured (tabular, multidimensional, potentially heterogeneous) and time series data. It intends to be the fundamental high-level building block for performing practical, real-world data analysis in Python. Furthermore, it aspires to be the most powerful and adaptable open-source data analysis and manipulation tool available in any language. It is already well on its way to accomplishing this goal.

- Pandas' two primary data structures, Series (1-dimensional) and Data Frame (2-dimensional), handle the vast majority of common use cases in finance, statistics, social science, and many fields of engineering. Data Frame gives R users access to all of R's data. Frame offers and much more. Pandas is built on top of NumPy and is designed to work well in a scientific computing environment

alongside many other third-party libraries.

- Written in: Python, Cython and C.

➢ *NumPy*

- NumPy is a Python library that adds support for large, multidimensional arrays and matrices, as well as a large collection of high-level mathematical functions for working with these arrays. NumPy is open-source software with numerous contributors.

- NumPy is designed to work with Python's CPython reference implementation, which is a non-optimizing bytecode interpreter. Algorithms written for this version of Python are frequently much slower than compiled equivalents.

- NumPy addresses the slowness issue in part by providing multidimensional arrays as well as functions and operators that operate efficiently on arrays, which necessitates rewriting some code, primarily inner loops, in NumPy.

- Because they are both interpreted, NumPy in Python provides functionality comparable to MATLAB, and they both allow the user to write fast programmes as long as most operations work on arrays or matrices rather than scalars.

- In comparison, MATLAB has a plethora of additional toolboxes, most notably Simulink, whereas NumPy is inextricably linked with Python, a more modern and comprehensive programming language. Additionally, there are complementary Python packages available; SciPy is a library that adds more MATLAB-like functionality, and Matplotlib is a plotting package that provides MATLAB-like plotting functionality.

- Written in: Python and C

4.4.2 MODEL BUILDING PACKAGE

- **Scikit-learn**

- Scikit-learn (formerly scikits. learn) is a free software machine learning library written in Python. It includes support vector machines (svm), random forest, gradient boosting, kmeans, and DBSCAN as classification, regression, and clustering algorithms, and is designed to work with the Python numerical and scientific libraries NumPy and SciPy.

.

- It was built on NumPy, SciPy, and Matplotlib.

- Written in: Python, Cython, C and C++.

4.4.3 DATA VISUALIZATION PACKAGES:

- **Scikit-plot**

- Scikit-plot is the result of a dreadful realization by an unartistic data scientist that visualization is one of the most important components of the data science process, not just an afterthought.

- When you're looking at a colored heatmap of a confusion matrix complete with class labels rather than a single-line dump of numbers enclosed in brackets, it's much easier to gain insights. Furthermore, if you ever need to present your results to someone 15 (virtually any time anyone hires you to do data science),

you show them visualizations, not a bunch of Excel numbers. Overall, it is a simple library for adding plotting functionality to a scikit-learn object.

- Written in: Python, Cython, C and C++.

- **Matplotlib's pyplot**

- Matplotlib is a Python 2D plotting library that generates high-quality figures in a variety of hardcopy and interactive formats across platforms. Matplotlib can be used in Python scripts, as well as the Python and IPython libraries.

- Shells, the Jupyter notebook, web application servers, and four graphical user interface toolkits are all available.

- It provides an object-oriented API for integrating plots into applications that use general-purpose GUI toolkits such as Tkinter, wxPython, Qt, or GTK+.

- matplotlib. Pyplot provides a MATLAB-like plotting framework

- Pylab is a namespace that combines pyplot and NumPy. This is convenient for interactive work, but it is recommended that the namespaces be kept separate for programming.

- Written in: Python.

4.5 SYSTEM REQUIREMENTS

- **HARDWARE**

  System: Pentium i3 Processor.

  Hard Disk: 500 GB.

Monitor: 15'' LED

Input Devices: Keyboard, Mouse

Ram: 2 GB

- **SOFTWARE**

  Operating system: Windows 7.

  Coding Language: python

  Tool: PyCharm, visual studio code

  Libraries: OpenCV

4.6 BACKGROUND STUDY

4.6.1 OVERVIEW

- Precision agriculture allows for the precise application of inputs such as seed, water, pesticides, and fertilizers to crops at the optimal time to maximize productivity, quality, and yields.

- Farmers can better understand their fields by deploying sensors for data collection and mapping fields, allowing them to conserve resources and reduce negative environmental effects.

- The majority of farmers use traditional farming patterns to determine which crops to cultivate in a field. Farmers, on the other hand, do not believe crop yield is affected by soil characteristics and climatic conditions.

- As a result, our Precision Agriculture solution can propose a crop recommendation system that assists farmers in determining the best crop to sow in their field based on weather, moisture, and season.

- Machine learning techniques provide an effective framework for making data-driven decisions. Using the ML recommendation engine, this application also assists in determining the best pesticide, seed spacing, and seed depth.

4.6.2 YIELD PREDICTION

One of the most important topics in precision agriculture is yield prediction, which is critical for yield mapping, yield estimation, matching crop supply with demand, and crop management to increase productivity.

4.6.2.1 HOW YIELD PREDICTION WILL HAPPEN

- When it comes to crop yield prediction, there are several factors to consider. Among these are studies of climate data, satellite imagery, soil conditions, and the possibility of pest attacks.

- All of this adds up to provide a comprehensive picture of the best times for crop production. There are also what-if scenarios and alternative action plans in place to deal with any unforeseen issues.

- Businesses in the AgTech industry are now using neural network algorithms to forecast crop yield.

- Backpropagation algorithms aid in determining the appropriate yield weight value for calculating the error derivative. Crop yield estimation accuracy is important for agronomic production.

- As a result, crop yield prediction is critical for the global food production ecosystem. With better data, it is possible to make more informed decisions.

- Crop yield prediction data is also useful to government agencies so that they can plan for national food security.

4.6.2.2 WHEN YIELD PREDICTION WILL BE USEFULL

- Low yield remains a problem, particularly in developing countries such as India. Poor farming infrastructure, small farm sizes, and ineffective use of technology and pesticides are a few of the causes.

- Because farms are small in size, resources such as irrigation and financial assistance are limited.

- A change in weather conditions is enough to affect the yield. It has a direct impact on cash flow and prevents further investments to improve productivity and mitigate risks.

- AgTech companies must focus on solutions that help predict yield based on factors such as climate.

- Climate factors affecting agricultural production include changes in average temperatures, rainfall levels, heatwaves, CO2 levels in the atmosphere, and ozone concentrations at ground level.

- Climate change affects countries in different ways. Crop production will most likely be more difficult in low-altitude countries.

- There is a need for technological solutions that can predict changing climatic conditions. Statistics-based weather forecasts use historical data to establish a pattern between seasons.

- As a result, predicting summer temperatures based on winter data is a possibility. If there is a change in summer temperatures, it is likely that there will be a change in winter temperatures as well.

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1 RESULTS

The result of the classification model prepared using Python may be divided into three parts in itself:

I. Pre-model Data Visualization
II. Model Information Visualization
III. Post-model Data Visualization

5.1.1 PRE-MODEL DATA VISUALIZATION

The pre-model data visualization section includes all data visualizations, such as

visualizations created with the pure data set to understand the features and distribution of data in the data set.

Visualization:

Plots are essentially one-way circuits. It is also a method for determining the appropriation of each trademark in a size. The Density Plot is a simple histogram detail. Thickness addresses an unsteady measurement of dissemination. It only recognizes passages as a numerical list.

A density plot is used to track the distribution of at least one factor. The main thing to do when recovering new information is to independently check the dissemination of the factors. It provides a wealth of information.

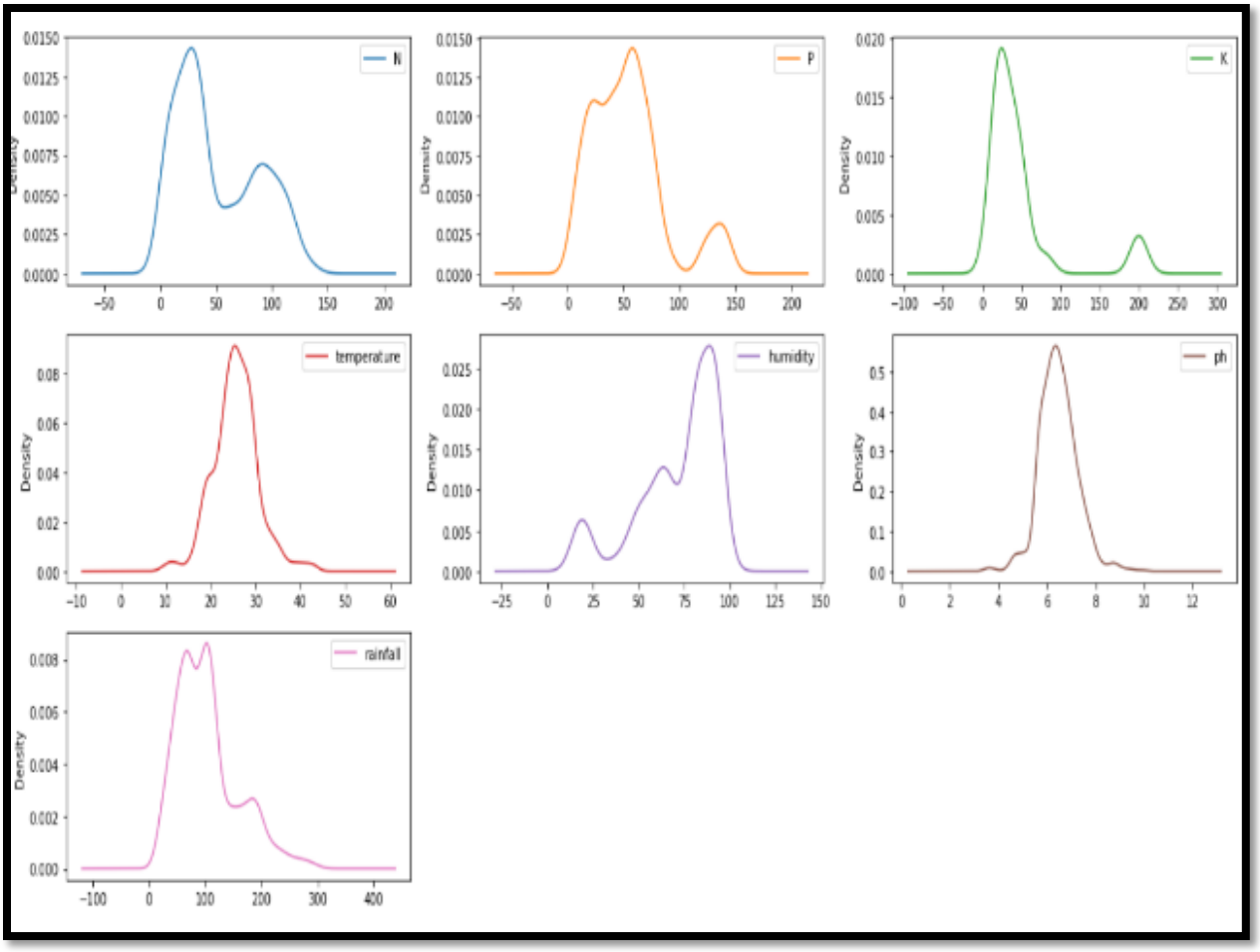


Fig 5.1: Density Diagram

The above figure shows a comparison between density and the seven parameters i.e., Nitrogen(N), Potassium(K), Phosphorus(P), Temperature, Humidity, Ph and rainfall.

5.1.2 MODEL INFORMATION VISUALIZATION

In this part of the chapter, the visualizations with respect to the model will be discussed.

Confusion Matrix:

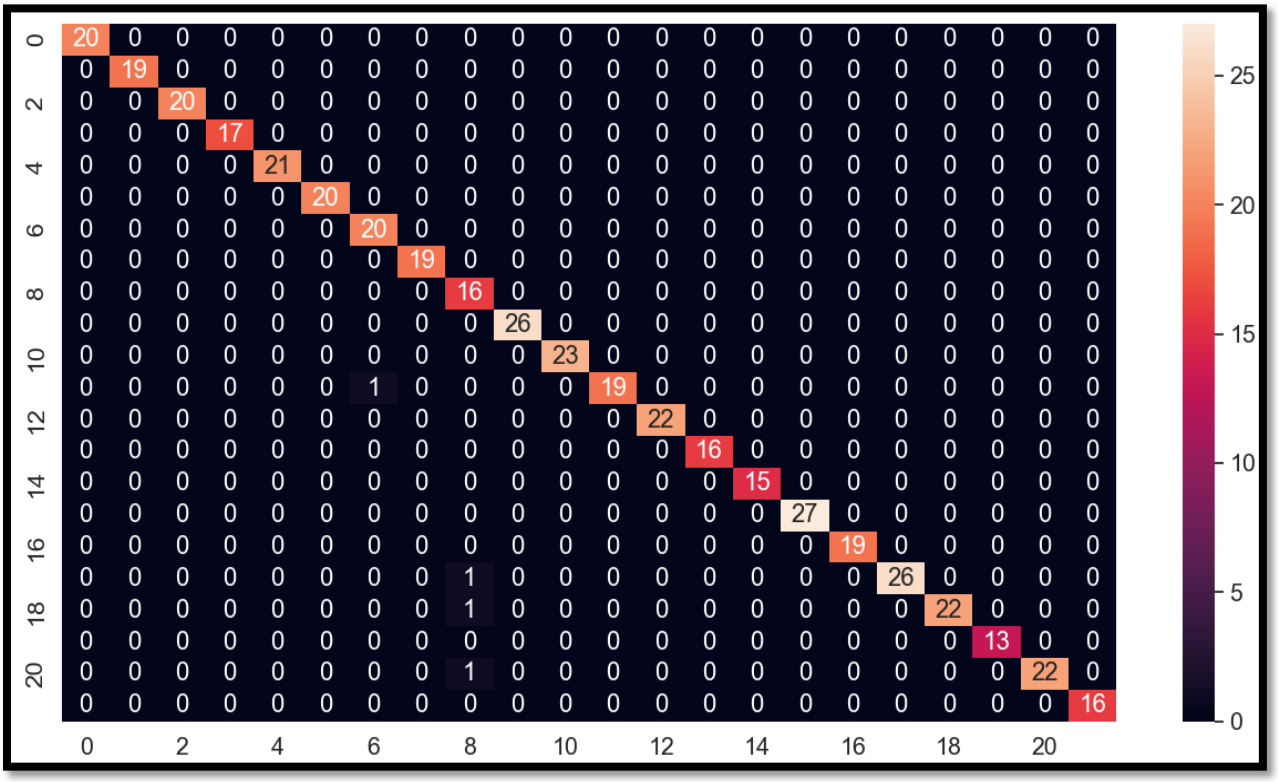


Fig 5.2: Confusion Matrix for Random Forest

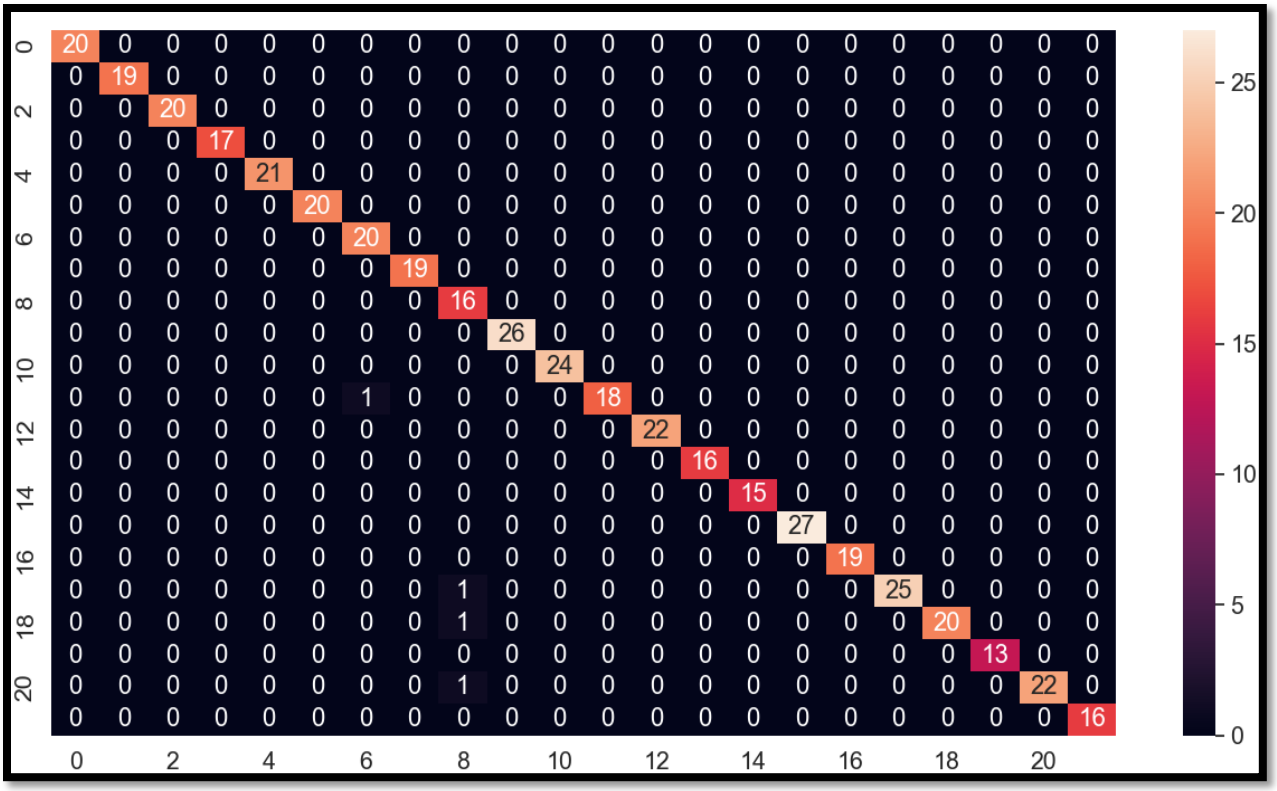


Fig 5.3: Confusion Matrix for KNN

The graph above shows the correct and incorrect classifications. It misclassified 22 times out of the 22 instances of training data used. This is clearly due to the bias that was previously mentioned. There have been no other misclassifications, demonstrating the model's integrity and reliability.

Correlation:

Correlation between attributes allows us to determine how strongly or weakly they are related to one another. The positive relationship between the variables is represented by the numeric value 1. The darker the colour of the numeric values 0, the more negative the relationship between the variables.

```
[6]  # Set up the matplotlib figure
     f, ax = plt.subplots(figsize=(11, 9))
     plt.title('Correlation Analysis with Original Data')
     # Draw the heatmap with the mask and correct aspect ratio
     ca = sns.heatmap(corr, cmap='coolwarm',center=0, vmin = -1,
                      square=True, linewidths=1, cbar_kws={"shrink": .8}, annot = True)
```

Fig 5.4: Correlation Diagram

Correlation Matrix Plot:

A correlation plot matrix can be formed for a collection of variables with each other variables will be plotted against each other. Here have seven columns where normally distributed with random values and column names are: Nitrogen(N), Potassium(K), Phosphorus(P), Temperature, Humidity, Ph and rainfall.

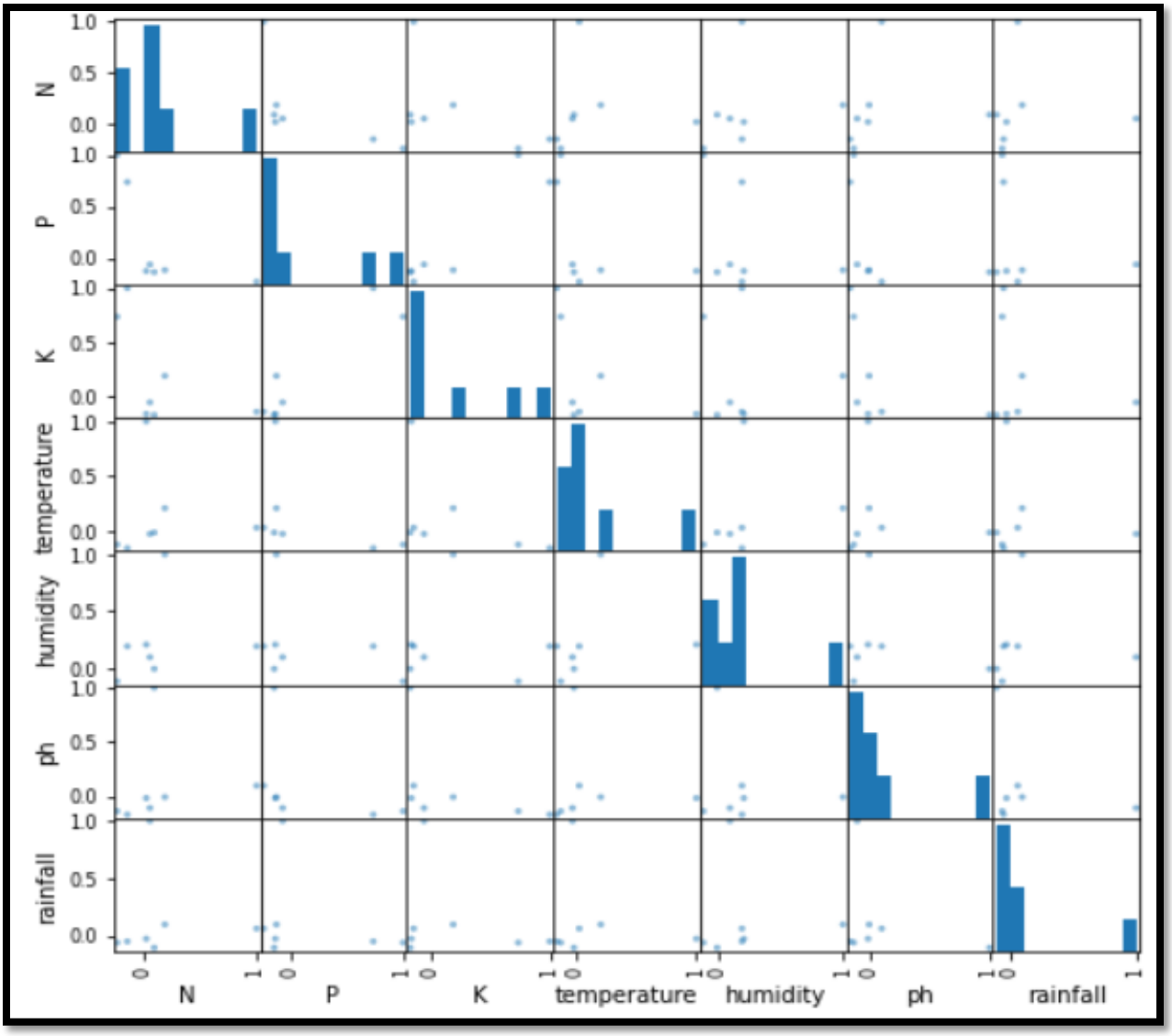


Fig 5.5: Correlation Matrix Plot Diagram

The above figure provides data about the attributes Phosphorus(P), Potassium(K), temperature and humidity may have an exponential distribution. The attribute Nitrogen(N) is easy to notice the distribution is skewed very much to left and the attribute rainfall is easy to notice the distribution is skewed very much to right. The attribute ph. has a Gaussian or nearly Gaussian distribution.

Classification Report:

- ***Accuracy:***

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions}$$

Fig 5.6: Accuracy Formula

Accuracy is a classification problem metric that indicates the percentage of correct predictions. We compute it by dividing the total number of predictions by the number of correct predictions. This formula provides a simple definition based on a binary classification problem. (In the second part of this article, we discuss multiclass and multilabel problems.) In the case of binary classification, accuracy can be expressed as True/False Positive/Negative values.

- ***Precision:***

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

$$= \frac{True\ Positive}{Total\ Predicted\ Positive}$$

Fig 5.7: Precision Formula

Precision is defined as the fraction of positive examples that are actually positive among all positive examples predicted by us. It can also be defined as the number of true positives divided by the total number of true positives plus false positives. False positives occur when the model incorrectly labels something as positive when it is actually negative,

or in our case, when the model incorrectly labels someone as a terrorist when they are not.

- ***Recall:***

$$\text{Recall} = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

$$= \frac{True\ Positive}{Total\ Actual\ Positive}$$

Fig 5.8: Recall Formula

In statistics, the metric our intuition tells us we should maximise is known as recall, or a model's ability to find all relevant cases within a dataset. The number of true positives divided by the number of true positives plus the number of false negatives is the precise definition of recall. True positives are data points classified as positive by the model that are actually positive (meaning they are correct), whereas false negatives are data points classified as negative by the model that are actually positive (meaning they are correct) (incorrect).

- ***F1-score:***

$$F_1 = 2 * \frac{precision * recall}{precision + recall}$$

Fig 5.9: F1-score Formula

It is traditionally defined as the harmonic mean of precision and recall. It's also known as the F Score or the F Measure. In other words, the F1 score conveys the balance between precision and recall. It is thought to be a better measure than Precision and

Recall

separately because the trade-off between the two is difficult to achieve.

	Random Forest	**KNN**
Accuracy	0.995	0.988
Precision	0.996	0.996
Recall	0.995	0.989
F1-score	0.995	0.989

Table 5.1: Algorithm Comparison

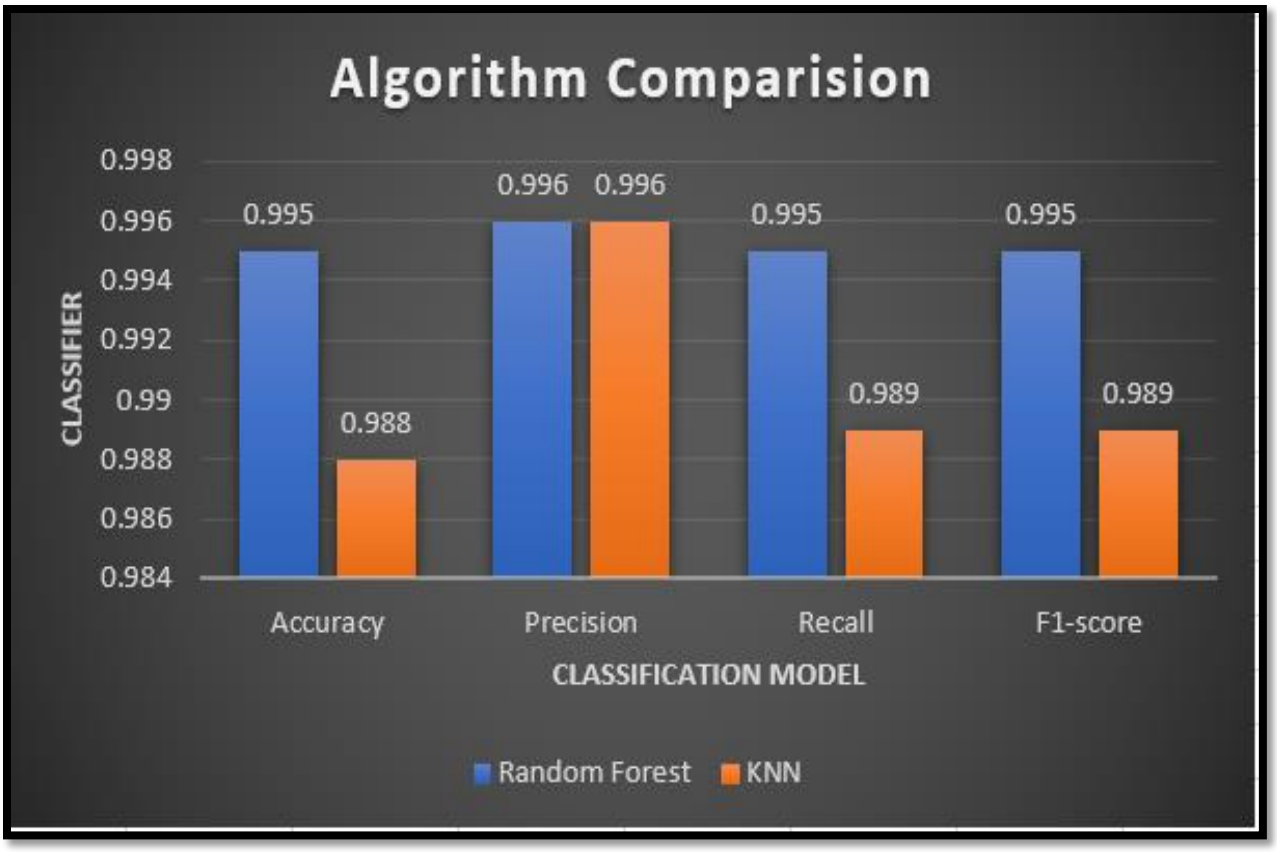


Fig 5.10: Algorithm Comparison

The above figure represents all the necessary factors taken to determine whether a model is good or bad. They are technically known as ***Model Evaluation Metrics***.

5.1.3 POST-MODEL DATA VISUALIZATION

Data visualisation is the depiction of interpreting data by displaying it in a graphical environment, allowing for the detection and exposure of patterns, inclinations, and connections.

The most popular charting libraries are:

- Matplotlib: Low-level, gives the user a lot of freedom.
- Pandas Visualization: This border is easy to use. Matplotlib can be used to create it.
- Plotly: Allows you to make an interactive plot for visualisation.

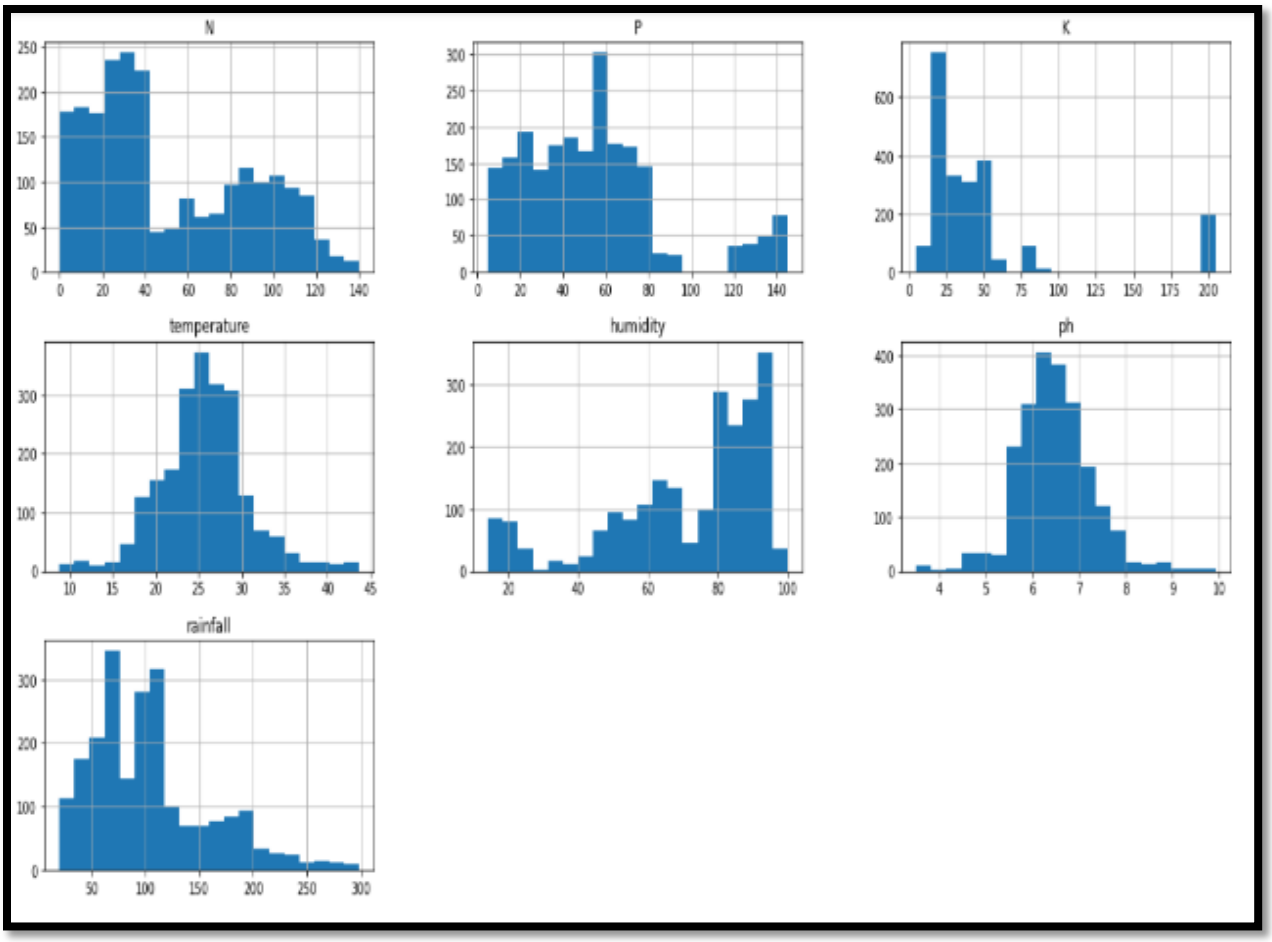


Fig 5.11: Hist Diagram

The attributes Nitrogen(N) and temperature may have an exponential distribution. The attribute Phosphorus(P), Potassium(K) and rainfall is easy to notice the distribution is skewed very much to left. The attribute humidity is easy to notice the distribution is skewed very much to right. The attribute ph. has a Gaussian or nearly Gaussian distribution

5.2 ANALYSIS

5.2.1 PROPOSED SYSTEM

For farmers, crop yield is extremely useful information. Knowing the yield can help you save money by reducing your losses. In the past, experienced farmers predicted crop

yields. In a similar way, the proposed system operates. It takes the prior data and

applies it to forecast future yields. The weather and fertilizers have the greatest impact on crop productivity. The accuracy of the information provided determines the accuracy of this prediction. As a result, the suggested approach anticipates yield and reduces loss. The expected system takes on the role of a seasoned farmer. However, it is more precise and takes into account a variety of additional parameters. Soil condition, weather forecast, pH, humidity and yield are all factors to consider.

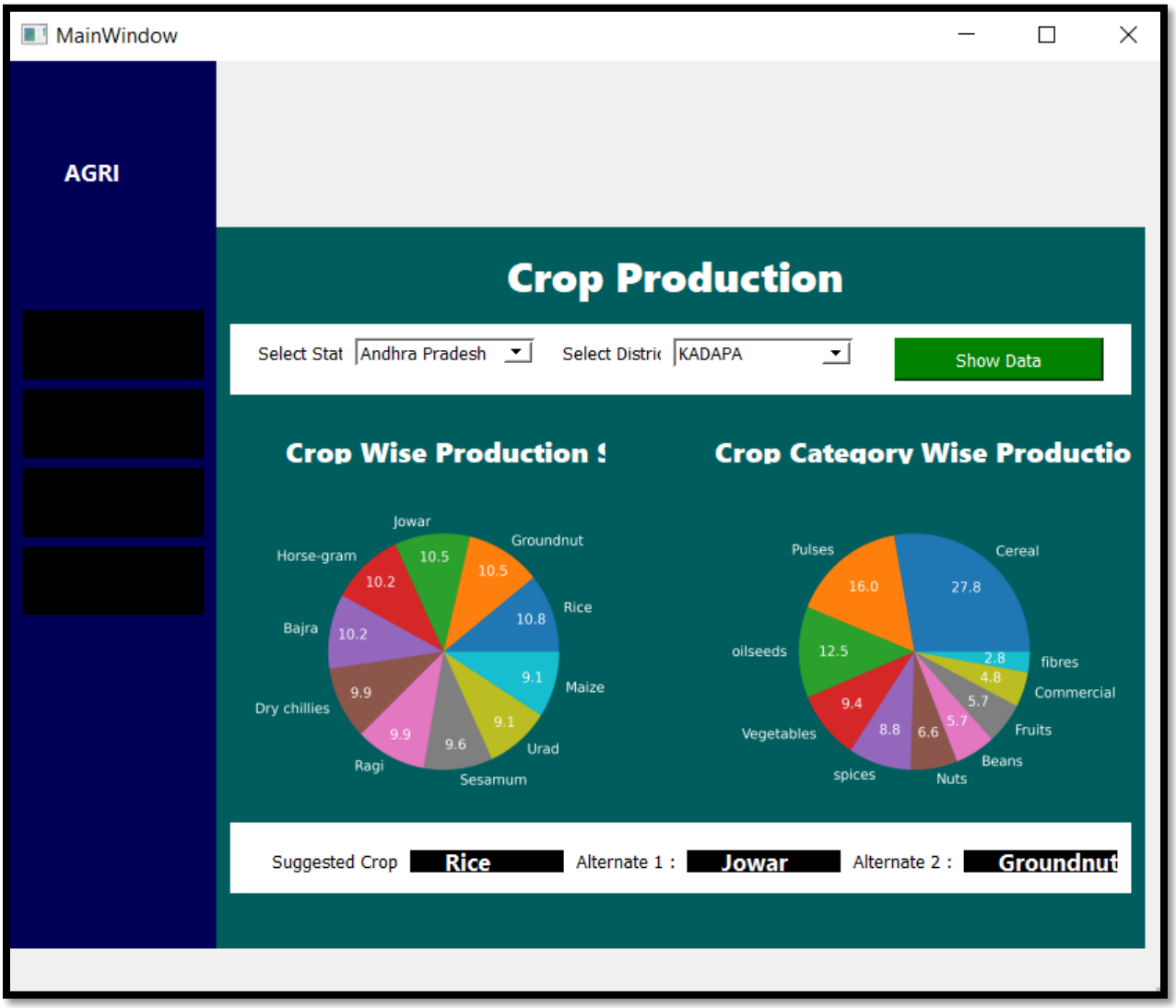


Fig 5.12: State Prediction

The above figure shows Crop creation arranging is accessible in all Indian states. In this figure, we will enter the fixings into districts and zones, which will show the plants filling as a pie chart in light of the info. This item will assist you with deciding the best plants to develop on explicit soils in the Indian state.

Fig 5.13: Crop Prediction

The above figure shows to discover, we will utilize seven unique fixings: nitrogen (N), phosphorus (P), potassium (K), heat, temperature, pH, and precipitation. Presently you want to get a decent gather to get a decent reap.

Contrasted with the past task, we got 9% more on KNN.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

Machine learning techniques are applied in various fields of agriculture. Implement a system to predict crop production from the collection of past data. Carry out an arrangement to conjecture generally upheld rural creation. Plant creation is gotten ready for data handling strategies. Today, random forest is utilized to characterize a fascinating yield as a genuine harvest. In agribusiness, it is for the most part consistent with anticipate yields. The higher the yield, the higher the yield. The proposed procedures assist ranchers with being careful about various harvest necessities and costs. This assists ranchers with picking what harvests to plant. This work is frequently used to distinguish extra plants that can be gathered monetarily and proficiently. This innovation can grow a wide assortment of yields. Indian ranchers might enjoy the benefit of precisely foreseeing yields in various pieces of India.

6.2 FUTURE SCOPE

This research work can be enhancing to the high level by building a recommender system of agriculture production and distribution for farmer. India may be a country wherever agriculture is extremely vital. The prosperity of the farmers ends up in the prosperity of the state. Thus, our work would assist farmers in sowing the acceptable seed supported soil necessities so as to extend productivity and exploit such a way. As a result, farmers will plant the acceptable crop, increasing their yield and therefore the nation's overall productivity. Our future work can concentrate on associate degree improved knowledge set with an oversized variety of attributes, in addition as yield prediction.

REFERENCES

[1] Aruvansh Nigam, Saksham Garg, Archit Agrawal “Crop Yield Prediction using ML Algorithms “, 2019 Fifth International Conference on Image Information Processing (ICIIP)

[2] LeoBrieman, “Random Forests”, 2019 IEEE Transactions on Instrumentation and Measurement

[3] Priya, P., Muthaiah, U., Balamurugan, M.” Predicting Yield of the Crop Using Machine Learning Algorithm”,2020 International Journal of Engineering Sciences & Research Technology (IJESRT)

[4] Mishra, S., Mishra, D., Santra, G. H.,“Applications of machine learning techniques in agricultural crop production”,2020 Indian Journal of Science and Technology

[5] Dr.Y Jeevan Kumar,”Supervised Learning Approach for Crop Production”, 2020 5th International Conference on Communication and Electronics Systems (ICCES)

[6] RameshMedar,Vijay S, Shweta, “Crop Yield Prediction using Machine Learning Techniques”, 2018 International Journal of Engineering Sciences & Research Technology (IJESRT)

[7] Ranjini B Guruprasad, Kumar Saurav, Sukanya Randhawa,”Machine Learning Methodologies for Paddy Yield Estimation in India: A CASE STUDY”, IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium

[8] https://en.wikipedia.org/wiki/Crop_yield

[9] https://en.wikipedia.org/wiki/Random_forest

[10] https://en.wikipedia.org/wiki/Knearest_neighbors_algorithm

[11] N. Suresh, Nvk Ramesh, Syed Inthiyaz, P. Poorna Priya, KurraNagasowmika,Kota V. N. Harish Kumar, Mashkoor Shaik, B. Naresh Kumar Reddy,“CropYieldPrediction Using Random Forest Algorithm”,2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS)

[12] Mamunur Rashid, Bifta Sama Bari, YusriYusup, Mohamad Anuar, NuzhaKhan, “A

Comprehensive Review of Crop Yield Prediction Using Machine Learning Approaches with Special Emphasis on Palm Oil Yield Prediction",2021 IEEE Journal

[13] DhivyaElavarasan, P. M. Durairaj Vincent, "Crop Yield Prediction Using Deep Reinforcement Learning Model for Sustainable Agrarian Applications",2020 IEEE Journal

[14] Roberto Luciani, Giovanni Laneve, MunzerJahjah, "Agriculture Monitoring, an Automatic Procedure for Crop Mapping and Yield Estimation: The Great Rift Valley of Kenya Case",2019 IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing

[15] Nida Rasheed, Shoab Ahmed Khan, Ali Hassan, Saria Safdar, "A Decision Support Framework for National Crop Production Planning",2021 IEEE Jounal

[16] Li He, Craig A, Zhi-Jie Wang, Wei Feng, Tian-Cai Guo, "Reduced Prediction Saturation and View Effects for Estimating the Leaf Area Index of Winter Wheat",2019 IEEE Transactions on Geoscience and Remote Sensing

[17] Alex F. Haufler, John H. Booske, Susan C. Hagness, "Microwave Sensing for Estimating Cranberry Crop Yield: A Pilot Study Using Simualted Canopies and Field Measurement Testbeds",2022 IEEE Journal

[18] Laura Martínez-Ferrer, Maria Piles, Gustau Camps-Valls, "Crop Yield Estimation and Interpretability with Gaussian Processes",2021 IEEE Geoscience and Remote Sensing Letters

[19] Abhinav S, Arpit Jain, Prateek Gupta, Vinay C, "Machine Learning Applications for Precision Agriculture: A Comprehensive Review",2021 IEEE Geoscience and Remote Sensing Letters

[20] Emily Myers, John Kerekes, Craig Daughtry, Andrew Russ, "Effects of Satellite Revisit Rate and Time-Series Smoothing Method on Throughout-Season Maize Yield Correlation Accuracy",2021 IEEE Journal Of Selected Topics In Applied Earth Observations And Remote Sensing

[21] Anup K. Prasad a, Lim Chai b, Ramesh P. Singh a,b,*, Menas Kafatos b. (2005). Crop yield estimation model for Iow a using remote sensing and surface parameters. Elsevier. 0 (0), p26-33. Sciencedirect Jorunal

[22] Mutlu Ozdogan. (2010). The spatial distribution of crop types from MODIS data: Temporal unmixing using Independent Component Analysis. Elsevier. 0 (0), p 1190-1204. Sciencedirect Journal

[23] P.Vossen. (1993). Forecasting of National Crop Production:the Methodologies Developed in theJoint research Centrein Supportto the commisionof European Countries. EARSEL. 2 (2), P 158-166. Springerlink Journal

[24] D Rameshl , B Vishnu Vardhan2. (2013). Data Mining Techniques and Applications to Agricultural Yield Data. IJARCCE. 2 (9), p3477-3480. International Journal of Engineering Research Technology (IJERT)

[25] Paul C. Doraiswamya , Bakhyt Akhmedovb , Larry Beardc , Alan Sterna and Richard Muellerc. (2006). OPERATIONAL PREDICTION OF CROP YIELDS USING MODIS DATA AND PRODUCTS. ISPRS. 0 (0), p45-49. Scribd Conference

APPENDIX

A. CODE

```
from PyQt5 import QtCore, QtGui, QtWidgets
import seaborn as sns
import matplotlib.pyplot as plt


class Ui_MainWindow(object):
    def setupUi(self, MainWindow)    :
        MainWindow.setObjectName("MainWindow")
        MainWindow.resize(839, 678)
        font = QtGui.QFont()
        font.setUnderline(False)
        MainWindow.setFont(font)
        MainWindow.setContextMenuPolicy(QtCore.Qt.DefaultContextMenu)
        self.centralwidget = QtWidgets.QWidget(MainWindow)
        self.centralwidget.setObjectName("centralwidget")
        self.label = QtWidgets.QLabel(self.centralwidget)
        self.label.setGeometry(QtCore.QRect(150, 0, 671, 121))
        font = QtGui.QFont()
        font.setBold(True)
        font.setUnderline(False)
        self.label.setFont(font)
        self.label.setAutoFillBackground(True)
        self.label.setStyleSheet("border-color: rgb(0, 0, 0);")
        self.label.setText("")
        self.label.setPixmap(QtGui.QPixmap("../images/mergeimg.jpg"))
        self.label.setScaledContents(True)
        self.label.setWordWrap(False)
        self.label.setObjectName("label")
        self.frame = QtWidgets.QFrame(self.centralwidget)
        self.frame.setGeometry(QtCore.QRect(0, 0, 151, 641))
        self.frame.setStyleSheet("background-color: rgb(0, 1, 86);\n"
"")
        self.frame.setFrameShape(QtWidgets.QFrame.StyledPanel)
        self.frame.setFrameShadow(QtWidgets.QFrame.Raised)
        self.frame.setObjectName("frame")
        self.label_2 = QtWidgets.QLabel(self.frame)
        self.label_2.setGeometry(QtCore.QRect(40, 10, 71, 61))
```

```
        self.label_2.setStyleSheet("")
        self.label_2.setText("")
        self.label_2.setPixmap(QtGui.QPixmap("../images/logo2.png"))
        self.label_2.setScaledContents(True)
        self.label_2.setObjectName("label_2")
        self.label_3 = QtWidgets.QLabel(self.frame)
        self.label_3.setGeometry(QtCore.QRect(40, 70, 81, 21))
        self.label_3.setStyleSheet("color: rgb(255, 255, 255);\n"
"font: 700 10pt \"Segoe UI\";")
        self.label_3.setObjectName("label_3")
        self.layoutWidget = QtWidgets.QWidget(self.centralwidget)
        self.layoutWidget.setGeometry(QtCore.QRect(10, 180, 131, 220))
        self.layoutWidget.setObjectName("layoutWidget")
```



```


        self.verticalLayout = QtWidgets.QVBoxLayout(self.layoutWidget)

        self.verticalLayout.setContentsMargins(0, 0, 0, 0)
        self.verticalLayout.setObjectName("verticalLayout")
        self.pushButton = QtWidgets.QPushButton(self.layoutWidget)
        self.pushButton.setMinimumSize(QtCore.QSize(35, 50))
        font = QtGui.QFont()
        font.setBold(True)
        font.setUnderline(False)
        self.pushButton.setFont(font)
        self.pushButton.setAutoFillBackground(False)
        self.pushButton.setStyleSheet("background-color: rgb(0, 0, 0);\n"
"color: rgb(255, 255, 255);")
        self.pushButton.setObjectName("pushButton")
        self.verticalLayout.addWidget(self.pushButton)
        self.pushButton_3 = QtWidgets.QPushButton(self.layoutWidget)
        self.pushButton_3.setMinimumSize(QtCore.QSize(35, 50))
        font = QtGui.QFont()
        font.setBold(True)
        font.setUnderline(False)
        self.pushButton_3.setFont(font)
        self.pushButton_3.setAutoFillBackground(False)
        self.pushButton_3.setStyleSheet("background-color: rgb(0, 0, 0);\n"
"color: rgb(255, 255, 255);")
        self.pushButton_3.setObjectName("pushButton_3")
        self.verticalLayout.addWidget(self.pushButton_3)
        self.pushButton_2 = QtWidgets.QPushButton(self.layoutWidget)
        self.pushButton_2.setMinimumSize(QtCore.QSize(35, 50))
        font = QtGui.QFont()
        font.setBold(True)
        font.setUnderline(False)
        self.pushButton_2.setFont(font)
        self.pushButton_2.setAutoFillBackground(False)
        self.pushButton_2.setStyleSheet("background-color: rgb(0, 0, 0);\n"
"color: rgb(255, 255, 255);")
        self.pushButton_2.setObjectName("pushButton_2")
        self.verticalLayout.addWidget(self.pushButton_2)
        self.pushButton_4 = QtWidgets.QPushButton(self.layoutWidget)
        self.pushButton_4.setMinimumSize(QtCore.QSize(35, 50))
        font = QtGui.QFont()
```

```
        font.setBold(True)
        font.setUnderline(False)
        self.pushButton_4.setFont(font)
        self.pushButton_4.setAutoFillBackground(False)
        self.pushButton_4.setStyleSheet("background-color: rgb(0, 0, 0);\n"
"color: rgb(255, 255, 255);")
        self.pushButton_4.setObjectName("pushButton_4")
        self.verticalLayout.addWidget(self.pushButton_4)
        self.frame_2 = QtWidgets.QFrame(self.centralwidget)
        self.frame_2.setGeometry(QtCore.QRect(150, 120, 671, 521))
        self.frame_2.setStyleSheet("background-color: rgb(0, 93, 93);\n"
"")
```



```
        self.frame_2.setFrameShape(QtWidgets.QFrame.StyledPanel)
        self.frame_2.setFrameShadow(QtWidgets.QFrame.Raised)
        self.frame_2.setObjectName("frame_2")
        self.templabel = QtWidgets.QLabel(self.frame_2)
        self.templabel.setGeometry(QtCore.QRect(210, 10, 251, 51))
        self.templabel.setStyleSheet("color: rgb(255, 255, 255);\n"
"font: 700 10pt \"Segoe UI\";")
        self.templabel.setObjectName("templabel")
        self.frame_3 = QtWidgets.QFrame(self.frame_2)
        self.frame_3.setGeometry(QtCore.QRect(10, 70, 651, 51))
        self.frame_3.setStyleSheet("background-color: rgb(255, 255, 255);")
        self.frame_3.setFrameShape(QtWidgets.QFrame.StyledPanel)
        self.frame_3.setFrameShadow(QtWidgets.QFrame.Raised)
        self.frame_3.setObjectName("frame_3")
        self.label_5 = QtWidgets.QLabel(self.frame_3)
        self.label_5.setGeometry(QtCore.QRect(20, 10, 61, 21))
        self.label_5.setObjectName("label_5")

        self.statecombobox = QtWidgets.QComboBox(self.frame_3)
        self.statecombobox.setGeometry(QtCore.QRect(90, 10, 131, 21))
        self.statecombobox.setObjectName("statecombobox")
        #------------- Here I have put the skeleton of the state
        for i in range (len(state_list)):
                self.statecombobox.addItem("")

        self.showdata = QtWidgets.QPushButton(self.frame_3)
        self.showdata.setGeometry(QtCore.QRect(480, 10, 151, 31))
        self.showdata.setStyleSheet("background-color: rgb(0, 130, 0);\n"
"color: rgb(255, 255, 255);")
        self.showdata.setObjectName("showdata")

        self.districtcombobox = QtWidgets.QComboBox(self.frame_3)
        self.districtcombobox.setGeometry(QtCore.QRect(320, 10, 131, 22))
        self.districtcombobox.setObjectName("districtcombobox")
        for i in range (len(district list)):
                self.districtcombobox.addItem("")
                # --------------Here you have put the district's skeleton


        self.label_6 = QtWidgets.QLabel(self.frame_3)
        self.label_6.setGeometry(QtCore.QRect(240, 10, 71, 21))
        self.label_6.setObjectName("label_6")
        self.label_4 = QtWidgets.QLabel(self.frame_2)
```

```
        self.label_4.setGeometry(QtCore.QRect(20, 160, 231, 171))
        self.label_4.setObjectName("label_4")
        self.piechartcrop = QtWidgets.QLabel(self.frame_2)
        self.piechartcrop.setGeometry(QtCore.QRect(10, 190, 301, 231))
        self.piechartcrop.setText("")
        self.piechartcrop.setScaledContents(True)
        self.piechartcrop.setWordWrap(False)
        self.piechartcrop.setObjectName("piechartcrop")
        self.piechartcategory = QtWidgets.QLabel(self.frame_2)
```



```
        self.piechartcategory.setGeometry(QtCore.QRect(350, 190, 301, 231))
        font = QtGui.QFont()
        font.setPointSize(12)
        font.setBold(False)
        self.piechartcategory.setFont(font)
        self.piechartcategory.setCursor(QtGui.QCursor(QtCore.Qt.ArrowCursor))
        self.piechartcategory.setAutoFillBackground(False)
        self.piechartcategory.setText("")
        self.piechartcategory.setScaledContents(True)
        self.piechartcategory.setObjectName("piechartcategory")

        self.frame_4 = QtWidgets.QFrame(self.frame_2)
        self.frame_4.setGeometry(QtCore.QRect(10, 430, 651, 51))
        self.frame_4.setStyleSheet("background-color: rgb(255, 255, 255);")
        self.frame_4.setFrameShape(QtWidgets.QFrame.StyledPanel)
        self.frame_4.setFrameShadow(QtWidgets.QFrame.Raised)
        self.frame_4.setObjectName("frame_4")
        self.label_9 = QtWidgets.QLabel(self.frame_4)
        self.label_9.setGeometry(QtCore.QRect(30, 20, 91, 16))
        self.label_9.setObjectName("label_9")
        self.label_10 = QtWidgets.QLabel(self.frame_4)
        self.label_10.setGeometry(QtCore.QRect(250, 20, 71, 16))
        self.label_10.setObjectName("label_10")
        self.label_11 = QtWidgets.QLabel(self.frame_4)
        self.label_11.setGeometry(QtCore.QRect(450, 20, 71, 16))
        self.label_11.setObjectName("label_11")
        self.suggestedcrop = QtWidgets.QLabel(self.frame_4)
        self.suggestedcrop.setGeometry(QtCore.QRect(130, 20, 111, 16))
        self.suggestedcrop.setObjectName("suggestedcrop")
        self.alternate1 = QtWidgets.QLabel(self.frame_4)
        self.alternate1.setGeometry(QtCore.QRect(330, 20, 111, 16))
        self.alternate1.setObjectName("alternate1")
        self.alternate2 = QtWidgets.QLabel(self.frame_4)
        self.alternate2.setGeometry(QtCore.QRect(530, 20, 111, 16))
        self.alternate2.setAutoFillBackground(False)
        self.alternate2.setObjectName("alternate2")
        self.cropwiselabel = QtWidgets.QLabel(self.frame_2)
        self.cropwiselabel.setGeometry(QtCore.QRect(50, 150, 231, 21))
        self.cropwiselabel.setStyleSheet("color: rgb(255, 255, 255);")
        self.cropwiselabel.setObjectName("cropwiselabel")
        self.categorywiselabel = QtWidgets.QLabel(self.frame_2)
        self.categorywiselabel.setGeometry(QtCore.QRect(360, 150, 301, 21))
        self.categorywiselabel.setStyleSheet("color: rgb(255, 255, 255);")
        self.categorywiselabel.setObjectName("categorywiselabel")
        self.frame.raise_()
```

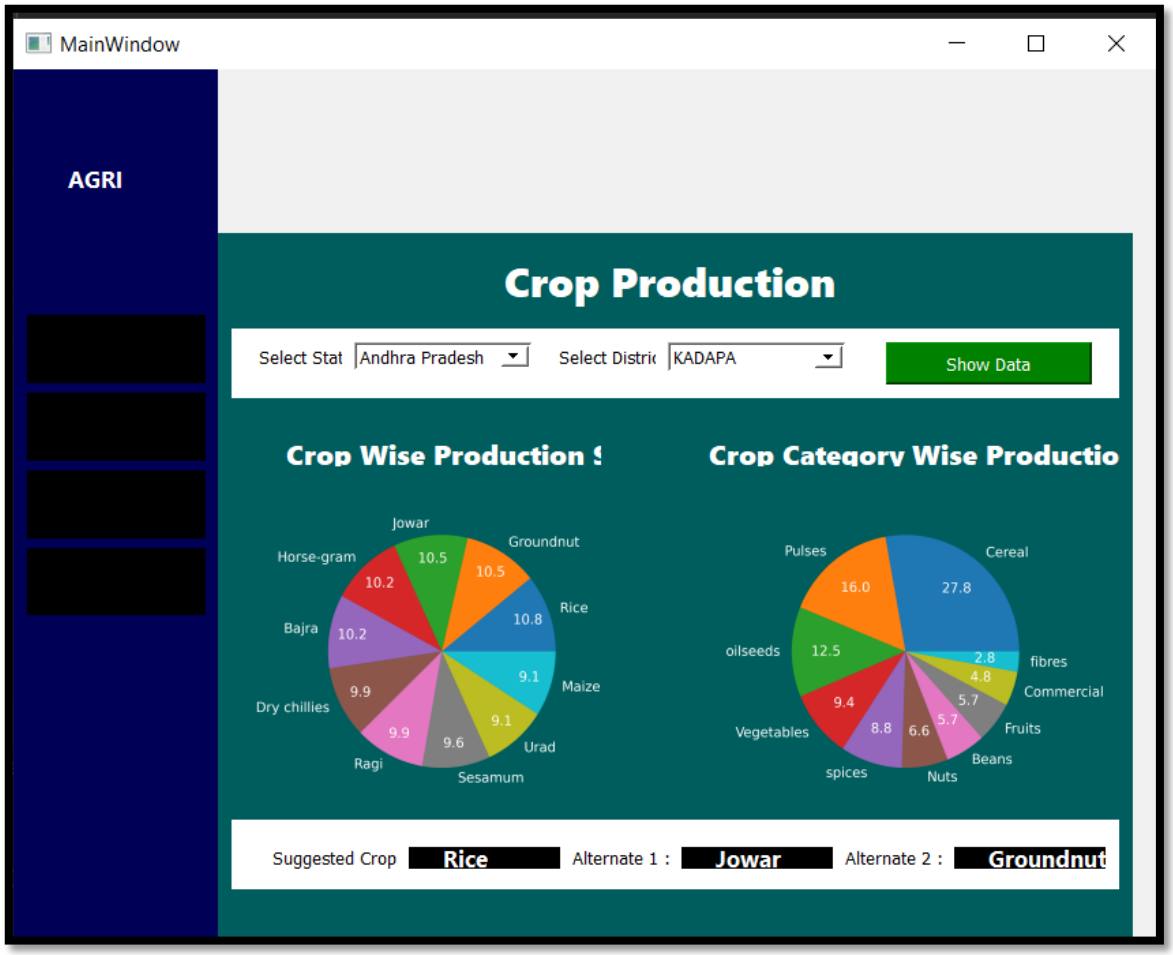
```
        self.label.raise_()
        self.layoutWidget.raise_()
        self.frame_2.raise_()
        MainWindow.setCentralWidget(self.centralwidget)
        self.statusbar = QtWidgets.QStatusBar(MainWindow)
        self.statusbar.setObjectName("statusbar")
        MainWindow.setStatusBar(self.statusbar)
```



```
self.retranslateUi(MainWindow)
        QtCore.QMetaObject.connectSlotsByName(MainWindow)

    def retranslateUi(self, MainWindow):
        _translate = QtCore.QCoreApplication.translate
        MainWindow.setWindowTitle(_translate("MainWindow", "MainWindow"))
        self.label_3.setText(_translate("MainWindow", "AGRI"))
        self.pushButton.setText(_translate("MainWindow", ""))
        self.pushButton_3.setText(_translate("MainWindow", ""))
        self.pushButton_2.setText(_translate("MainWindow", ""))
        self.pushButton_4.setText(_translate("MainWindow", ""))
        self.templabel.setText(_translate("MainWindow",
"<html><head/><body><p><span style=\" font-size:18pt;\">Crop Production
</span></p></body></html>"))
        self.label_5.setText(_translate("MainWindow", "Select State"))
        self.showdata.setText(_translate("MainWindow", "Show Data"))
        self.label_6.setText(_translate("MainWindow", "Select District"))
        self.label_4.setText(_translate("MainWindow", "TextLabel"))
        self.label_9.setText(_translate("MainWindow", "Suggested Crop :"))
        self.label_10.setText(_translate("MainWindow", "Alternate 1 :"))
        self.label_11.setText(_translate("MainWindow", "Alternate 2 :"))
        self.suggestedcrop.setText(_translate("MainWindow",
"....................................."))
        self.alternate1.setText(_translate("MainWindow",
"....................................."))
        self.alternate2.setText(_translate("MainWindow",
"....................................."))
        self.cropwiselabel.setText(_translate("MainWindow",
"<html><head/><body><p><br/></p></body></html>"))
        self.categorywiselabel.setText(_translate("MainWindow",
"<html><head/><body><p><br/></p></body></html>"))

#-----------------------------------------------------------------Your work starts
from here-----------------------


        for i in range (len(state_list)):
                self.statecombobox.setItemText(i,  translate("MainWindow",
state_list[i]))
                #Here I have entered the state data

        for i in range (len(district_list)):
                self.districtcombobox.setItemText(i, _translate("MainWindow",
district_list[i]))
                #Here I have entered the data of the district
```

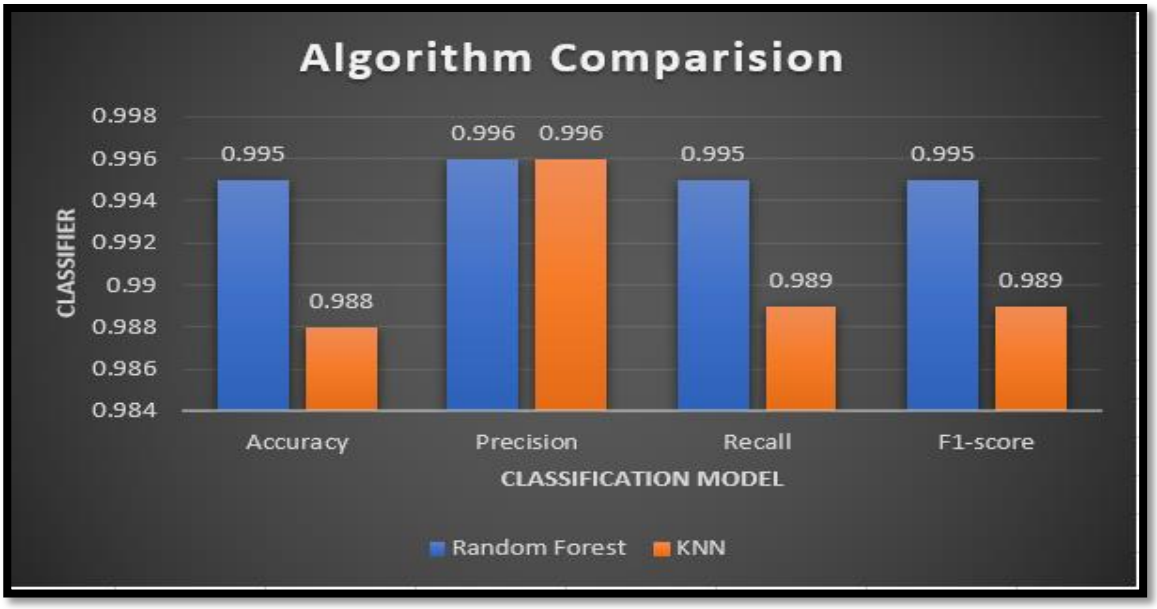
```
        self.showdata.pressed.connect(self.func)
    def func(self):
        state=self.statecombobox.currentText()
        dist=self.districtcombobox.currentText()
        self.find(state,dist)
    def find(self,state,dist):
            self.cropwiselabel.setText("Crop Wise Production Status")
```

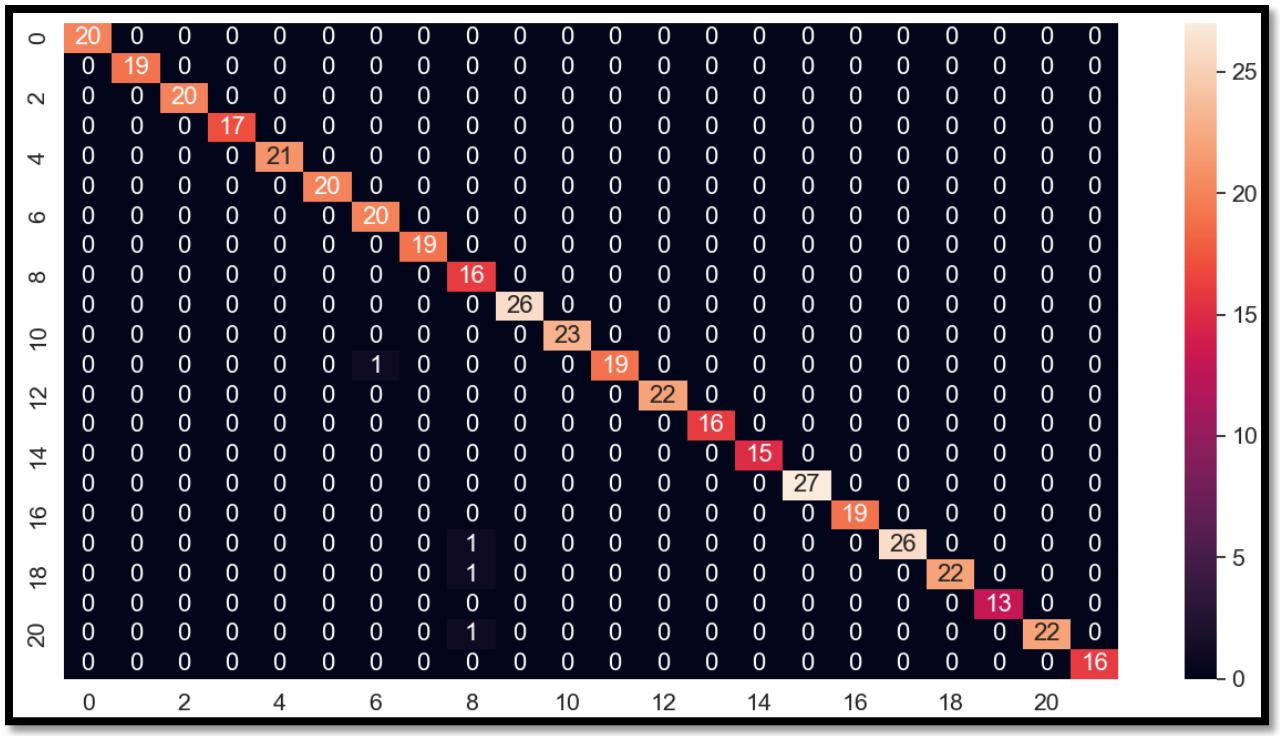

B. SCREENSHOTS

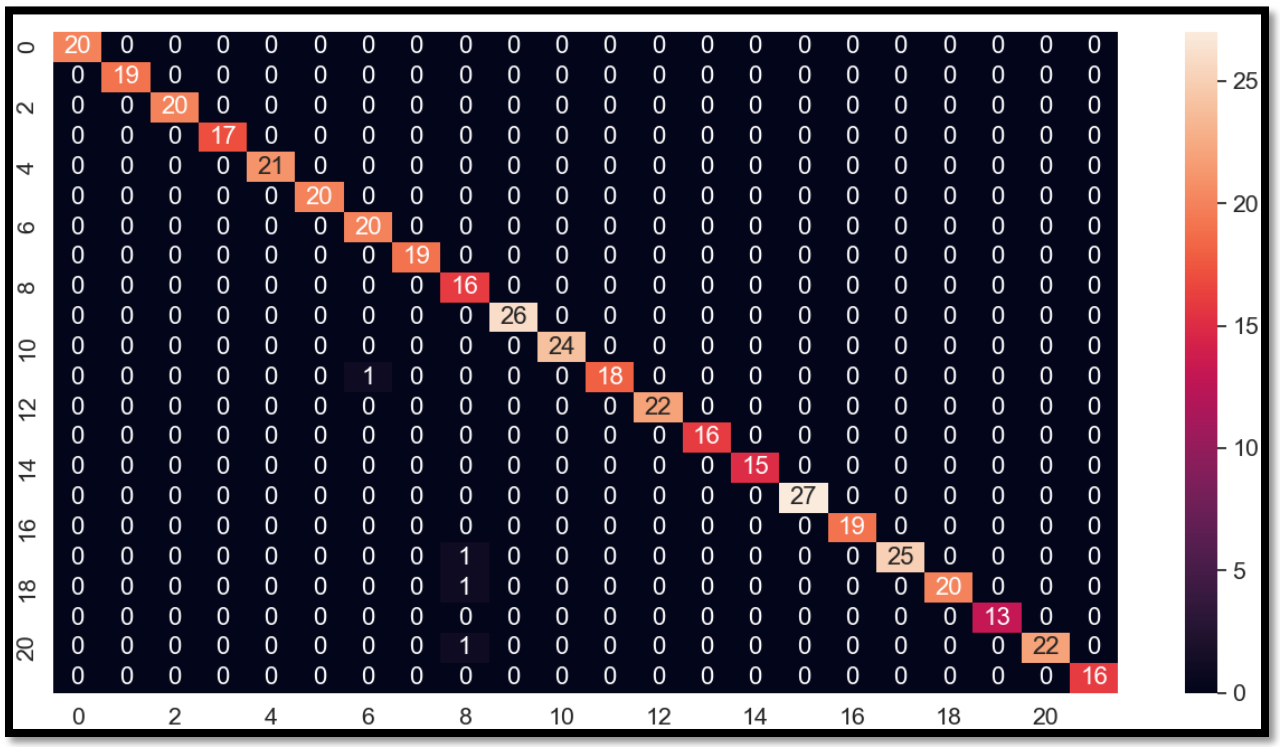



Agricultural Crop Recommendation
Nitrogen(N): 8
Phosphorus(P): 30
Potassium(K): 5
Temperature: 11
Humidity: 20
PH Level: 20
Rainfall: 100
Recommend
Project Member:
KANIKA BHATNAGAR
JAAHNAVI MAMILLA


Agricultural Crop Recommendation
Nitrogen(N):
Phosphorus(P):
Potassium(K):
Temperature:
Humidity:
PH Level:
Rainfall:
Recommend
Reccommended crop is : ['kidneybeans']
Project Member:
KANIKA BHATNAGAR
JAAHNAVI MAMILLA


Algorithm Comparision
CLASSIFIER
0.998
0.996
0.994
0.992
0.99
0.988
0.986
0.984
0.995
0.988
0.996
0.996
0.995
0.989
0.995
0.989
Accuracy
Precision
Recall
F1-score
CLASSIFICATION MODEL
Random Forest
KNN


0 2 4 6 8 10 12 14 16 18 20
25 20 15 10 5 0


0 2 4 6 8 10 12 14 16 18 20
25 20 15 10 5 0

C.PUBLICATION WITH PLAGIARISM REPORT

Paper ID 094.docx

by Paper Id 094.docx Paper Id 094.docx

Submission date: 17-Mar-2022 11:36AM (UTC-0400)
Submission ID: 1786411942
File name: Paper_ID_094.docx (1.24M)
Word count: 2959
Character count: 17064

4%	3%	3%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Kumar Kunal, Md. Azhar Hussain, N Srinivasan, J. Albert Mayan. "Smart Irrigation and Tank Monitoring System", IOP Conference Series: Materials Science and Engineering, 2019 Publication	1%
2	Submitted to UT, Dallas Student Paper	1%
3	A. Viji Amutha Mary, Emmaneni Venkata Naga Sai Prem, Sri Hari Jujjavarapu, P. Asha. "Securing Data by Detecting Multi Channel Attacks Using Deep Learning", 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), 2021 Publication	1%
4	Madiraju Sairam Asish, R. Aishwarya. "Cyber Security at a Glance", 2019 Fifth International Conference on Science Technology Engineering and Mathematics (ICONSTEM), 2019 Publication	1%