A MACHINE LEARNING BASED MODEL FOR PREDICTING THE BEST CROP TO HARVEST

A PROJECT REPORT

Submitted by,

Mr. UDAY V	**-20191COM0211**
Mr. KISHAN CHAND T	**-20191COM0203**
Mr. VELLAMPALLI VISHNU SAI	**-20191COM0220**
Mr. SIDDAREDDYGARI DILLI	**-20191COM0189**

Under the guidance of,

Dr. Alamelu Mangai Jothidurai ,Professor
School of Computer Science & Engineering
Presidency University

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER ENGINEERING

At

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

SCHOOL OF COMPUTER SCIENCE & ENGINEERING
PRESIDENCY UNIVERSITY
BENGALURU
JUNE 2023

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the Project report **“A MACHINE LEARNING BASED MODEL FOR PREDICTING THE BEST CROP TO HARVEST”** being submitted by “UDAY V, KISHAN CHAND T , VELLAMPALLI VISHNU SAI, SIDDAREDDYGARI DILLI” bearing roll number(s) “20191COM0211, 20191COM0203, 20191COM0220, 20191COM0189” in partial fulfilment of requirement for the award of degree of Bachelor of Technology in Computer Engineering is a bonafide work carried out under my supervision.

Dr. Alamelu Mangai Jothidurai
Professor
School of CSE&IS
Presidency University

Dr. Md. Sameeruddin Khan
HOD
School of CSE&IS
Presidency University

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE & ENGINEERING

DECLARATION

We hereby declare that the work, which is being presented in the project report entitled **A MACHINE LEARNING BASED MODEL FOR PREDICTING THE BEST CROP TO HARVEST** in partial fulfilment for the award of Degree of **Bachelor of Technology** in **Computer Engineering**, is a record of our own investigations carried under the guidance of **Dr. Alamelu Mangai Jothidurai, Professor, School of Computer Science & Engineering, Presidency University, Bengaluru.**

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

Mr. UDAY V	**-20191COM0211**
Mr. KISHAN CHAND T	**-20191COM0203**
Mr. VELLAMPALLI VISHNU SAI	**-20191COM0220**
Mr. SIDDAREDDYGARI DILLI	**-20191COM0189**

Signature(s) of the Students

ABSTRACT

Agribusiness is an important component of the Indian economy, but it is facing challenges due to a variety of factors such as soil erosion, climate change, and inefficient resource use. Farmers can overcome these obstacles by leveraging cutting-edge technologies like machine learning (ML) and the Internet of Things (IoT). This paper proposes a system that employs machine learning to predict which plant to harvest based on variables such as NPK soil nutrients, pressure, temperature, wind speed, area, production, yield, crop year, season names and soil type. Data is collected from the state of Maharashtra, and the system uses this information to recommend crops for specific soil types and environmental conditions. Farmers can use the system to make informed decisions about resource allocation and planting techniques by receiving accurate crop growth data. Several algorithms are used in this case, including Decision Tree, Random Forest, KNN, naive bayes, and XG Boost techniques. The accuracy, precision, recall, and AUC score of the classifier will be used to evaluate its performance. Based on the above-mentioned criteria XG Boost is better performing with an accuracy score of 70% and AUC of 0.95. Also, a website will be created where users may enter their information and choose the optimal crop for harvesting.

Keywords:- *Machine Learning, Internet of Things, Decision Tree, Random Forest, KNN, Navies Bayes, XG Boost.*

ACKNOWLEDGEMENT

First of all, we indebted to the **GOD ALMIGHTY** for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Dean, School of Computer Science & Engineering, Presidency University for getting us permission to undergo the project.

We record our heartfelt gratitude to our beloved Associate Dean **Dr. C. Kalaiarasan,** Professor **Dr. T K Thivakaran,** University Project-II In-charge, School of Computer Science & Engineering, Presidency University for rendering timely help for the successful completion of this project.

We would like to convey our gratitude and heartfelt thanks to the University Project-II Co-Ordinators **Mr. Mrutyunjaya MS, Mr. Sanjeev P Kaulgud, Mr. Rama Krishna K and Dr. Madhusudhan MV**.

We are greatly indebted to our guide **Dr. Alamelu Mangai Jothidurai ,Professor,** School of Computer Science & Engineering, Presidency University for her inspirational guidance, valuable suggestions and providing us a chance to express our technical capabilities in every respect for the completion of the project work.

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

UDAY V
KISHAN CHAND T
VELLAMPALLI VISHNU SAI
SIDDAREDDYGARI DILLI

List of Tables

List of Figures

TABLE OF CONTENTS

CHAPTER-1

INTRODUCTION

India is one of the world's top producers of agricultural goods, and the sector accounts for over 17% of its GDP. However, the industry has several difficulties, such as poor productivity, limited infrastructure, knowledge and resource gaps, and climate change. The Internet of

Things (IoT) and machine learning have emerged as potent technologies that have the potential to revolutionize India's agricultural industry by giving farmers access to real-time data and insights. Statistics on agriculture in India show that the average production of the main crops in India is substantially lower than the average yield worldwide. Over 60% of the people in the agricultural nation of India relies on agriculture as their main source of income, and the Economic Survey of India 2021 predicts that the agriculture sector would increase at a pace of 3.4% in 2020–21.

Precision agriculture has great promise, and the Indian government has started measures to encourage its use. By giving farmers access to real-time information on soil moisture, weather patterns, and crop health, machine learning and the internet of things have the potential to completely change the Indian agriculture industry. Sensors can track soil moisture levels and provide real-time information on whether to water their crops. Machine learning algorithms can offer ideas on how to optimize irrigation schedules to increase agricultural yields. Precision agriculture maximizes crop yield while minimizing waste, using sensors and machine learning algorithms to track crop health, soil moisture, and weather patterns. For smallholder farmers in India, precision agriculture may be very helpful, as it can enhance their livelihoods, boost output, and save expenses.

CHAPTER-2

LITERATURE SURVEY

Plenty research had gone to find the problem in Indian agriculture and many more research are going along with the time to predict the solution for the issue.

Using data acquired from the Madurai area, the Crop Suggestions System for Precision Agriculture [1] was created to assist farmers in planting the proper seed according to soil conditions. The main goal is to find a solution for the classifier selection issue in ensemble learning and to have the greatest accuracy possible.

[2]The pre-processed data includes a number of variables that can be used to forecast agricultural yield in Kerala state. Additionally, it gathers information about the local weather conditions using the weather API, forecasts which crops will produce a high yield, and determines the crop's yield price depending on the current market.

[3]The study includes several crop-growing parameters, with a focus on deep learning techniques like ANN and CNN and pre-training the model with gathered data to forecast which crop will do best under the given conditions.

The authors of [4] have suggested a model that uses data from the Government of India's repository website, data.govt.in. The dataset primarily includes the 4 crops, totalling 9000 samples, of which 6750 are utilized for training and the remaining 2250 for testing. Following pre-processing, ensemble-based learners such as Random Forest, Navies Bayes, and Linear SVM are utilized, and the majority voting technique is used to get the greatest accuracy.

The technique for determining which crop is most suited for harvesting is suggested in the study [5]. They utilized many algorithms, including Decision tree, Random Forest, KNN, and neural network, on the Indian Agricultural and Climate Data set in order to obtain the highest level of accuracy.

The authors of [6] proposed a model that uses previous farmland data as the data set. It consists of various attributes such as county name, state, humidity, temperature, NDVI, wind speed, and yield. The model is trained to identify the soil requirements necessary for yield prediction. Algorithms applied to the dataset are random forest, decision tree, and polynomial regression. Among all three algorithms, Random Forest provides better yield prediction compared to other algorithms.

In the paper [7], the factors used by the proposed system include soil pH, temperature, humidity, rainfall, nitrogen, potassium, and phosphorus. Various crops are also included in the dataset. After utilizing the dataset to train and test the model. A variety of algorithms, including Decision Tree, Random Forest, XGBOOST, Naïve Bayes, and LR, are used to forecast a specific crop under specific environmental conditions and parameter values that aid in growing the best crop. Thus, evaluating the accuracy of algorithms and selecting the greatest accuracy will assist farmers in selecting the appropriate seed and aid in boosting agricultural yield.

Authors of [8] implemented precision farming, where a variety of internet of things (IOT) sensors and devices are used to collect data on environmental conditions for farming, the amount of fertilizer to be used, the amount of water needed, and the levels of soil nutrients. Through wired or wireless connectivity, the data gathered by the numerous IOT sensors at the end node is then saved in the cloud or on remote servers. Afterward, relevant meanings and interpretations are inferred from the data using a variety of data analytic techniques, which are then applied to the data to make precise and correct decisions. Then, several algorithms are used to select crops, and the data analysed can be used to understand agricultural conditions and whether they are favourable as well as forecast crop yields with the highest yield.

[9] The study examines the value of climatic and meteorological elements in influencing agricultural choices and proposes a district-by-district forecasting model for the Tamil Nadu state. To raise the quality of incoming data, the paper suggests employing pre-processing and clustering techniques. Furthermore, it recommends employing artificial neural networks (ANN) to predict agricultural productivity and daily precipitation using meteorological data. In order to improve the system’s success rate, the study article suggests a hybrid recommender system that makes use of Case-Based Reasoning (CBR). The effectiveness of the proposed hybrid technique is evaluated against conventional collaborative filtering.

The authors of this research [10] present a framework that uses machine learning and deep learning techniques to suggest the best crop based on soil and climate parameters. Area, Relative Humidity, PH, Temperature, and Rainfall are the predictive variables in the dataset. once the dataset has been pre-processed. The information is then divided into a training set and a test set. The response is then depicted graphically for each of the parameters, including fertilizer use, pesticide use, area, UV exposure, and water, using the above-mentioned algorithms, and the yield is forecasted using the data for these parameters. Thus, with little loss and a high yield, the results can assist farmers in growing suitable crops.

This suggested approach in [11] produced a crop recommendation system for smart farming. This study report analysed several machine learning methods, including CHAID, KNN, K-means, decision trees, neural networks, naive bayes, C4.5, LAD, IBK, and SVM algorithms. The complex computations in this study were performed using the Hadoop framework, which improved the system's precision.

In [12] The right crop using the proposed approach based on details like soil PH, temperature, humidity, rainfall, nitrogen, potassium, and phosphorus.The historical data with the aforementioned parameters are included in the dataset. To eliminate outliers and missing values, the gathered data is pre-processed. The model is subsequently tested and trained. The method utilizes a variety of machine learning classifiers, including Deep Sequential Model, KNN, XGB, Decision Tree, and Random Forest, to accurately and effectively select a crop for site-specific factors. Farmers will be assisted in growing appropriate crops with the highest yield thanks to this research report.

[13]The research suggests a user-friendly recommender system to help farmers choose appropriate crops based on economic and environmental criteria. By examining variables including rainfall, temperature, humidity, soil nutrients, and pH level, the suggested model forecasts crop yields. The technology also helps farmers monitor soil nutrient levels and spot plant problems. The study discusses the topic of farmers' low income, poor crop selection, and declining agricultural yields as a result of unpredictable weather patterns and fluctuating market pricing.

[14] An overview of the paper's applications of data mining methods to the agricultural sector, with a particular emphasis on yield prediction. The use of different data mining approaches, including K-Means, K-Nearest Neighbour (KNN), Artificial Neural Networks (ANN), and Support Vector Machines (SVM), for yield prediction is covered in this paper. The goal of the research is to identify appropriate data models with high generality and accuracy for predicting crop yield production based on available data sets. To address the issue of yield prediction in agriculture, the authors compare various Data Mining approaches on distinct data sets.

[15] The article examines the use of artificial neural networks (ANNs) in agriculture for a variety of tasks, including crop yield prediction, seed categorization, soil moisture estimate, and other tasks.The study examines the impact of geometric characteristics on the process of classifying seeds in India. With the aid of datasets from several Indian areas, the authors employ a machine learning model based on ANNs to predict seed classifications.

[16]The study, which uses regression analysis, examines how to predict crop yields in agriculture. With a particular concentration on the North Western zone of Tamil Nadu state, the focus is on constructing a predictor model for agricultural production in tonnes. According to the study, banana, maize, ragi, turmeric, coconut, cotton, and jowar are the next top yielding crops in this zone after sugarcane and tapioca. Based on regression analysis, the predictor

model can assist in crop production forecasting, which is crucial for the agricultural industry and food management.

[17]The article advocates using an algorithm called the Crop Variety Selection Method (CVSM) to select the optimal crop and variety based on physical, economic, and environmental factors. The algorithm’s main prediction mechanism for agricultural yield rates is an Artificial Neural Network (ANN) with six input neurons and five hidden layers. The crop picker first chooses the crop that is suited for the designated soil type by comparing the current time with crop sowing time. The crop with the best profit margin based on yield rate and the current year’s Minimum Support Price (MSP) is chosen after the chosen crops have been sorted by yield rate and market price. Both inexperienced and seasoned farmers may find the suggested strategy useful.

[18]The research focuses on how important crop yield forecasting is for helping farmers maximise crop production. The paper mentions several techniques and algorithms for calculating crop yield, including the Random Forest algorithm. Additionally discussed is the application of Big Data analytics to agriculture. The report highlights several issues and challenges that have arisen as a result of the use of modern technologies and practises in agriculture. In order to anticipate agricultural yield and improve forecast accuracy, the research recommends employing machine learning techniques. It also aims to provide an easy-to-use user interface while analysing a number of meteorological characteristics, such as cloud cover, precipitation, and temperature.

[19]The significance of predicting West Bengal’s rice production is covered in the study. The authors model and predict rice production using the autoregressive integrated moving average (ARIMA) methodology. The research emphasises the significance of precise rice production forecasts in order to effectively plan crop production, particularly in light of population expansion. The authors also point out that stochastic time series models, like ARIMA, are effective forecasting tools because they can characterise observed data and produce predictions with little forecast error. Overall, the research offers insights into the application of statistical

modelling approaches in agriculture and emphasises the significance of precise crop output forecasting for planning and policy-making.

[20]The dataset has one million records with 22 attributes, including the districts in Tamil Nadu, the pH level, the Temperature, the amount of sunlight, and minerals like phosphorus, potassium, boron, carbon, nitrogen, sulphur, calcium, magnesia, manganese, zinc, iodine, and copper. The prediction method is carried out for the crops of rice, wheat, and maize. This study proposes a machine learning-based crop yield forecast model that makes use of various linear regression and clustering techniques (E-DBSCAN, DBSCAN, CLARA, and K-Means). Ecological factors had a stronger impact on crop yield than other factors, according to the model's testing on datasets for crops (rice, wheat, and maize). In comparison to DBSCAN, CLARA, and K-means, the E-DBSCAN clustering technique was found to be more efficient. In order to predict agricultural yield, multiple linear regression was found to be less error-pron. Using a variety of performance indicators, the model's accuracy was evaluated, and the findings were confirmed. Based on location and growth characteristics, the suggested approach can help farmers choose the optimal crop for harvesting to Maximise output.

[21]In this study, a number of forecasting models were put to the test using historical/large data spanning 46 years to see which one would be most accurate in projecting Pakistan's wheat production during the ensuing three years. The best model in this aspect has been found to be ARIMA. It is suggested that different crop data can be used to both fit models and generate forecasts for provinces and districts using more complex forecasting models.

CHAPTER-3

REQUIREMENT ANALYSIS

3.1: Programming Language used:

3.1.1: Python: Python is a general-purpose high-level programming language sometimes known as an interpreter language. Python is a programming language that Guido van Rossum created and released for the first time in 1991. Its design philosophy places a strong emphasis on code readability and a syntax that allows programmers to express concepts in less code. It includes designs that make simple programming possible at both small and big scales. Python has an automatic memory management system and a dynamic structure. It supports a variety of programming paradigms, including object-oriented, imperative, functional, and procedural, and has a large and comprehensive standard library.

3.2: Libraries used:

3.2.1: NUMPY

The most crucial Python package for scientific computing is called NumPy. A multidimensional array object, derived objects (like masked arrays and matrices), and a variety of routines for carrying out quick-array operations, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation, and more are all included in this Python library.

3.2.2: PANDAS

Pandas is a popular open source Python library for data science, data mining, and machine learning activities. It is based on Numpy, a library that enables multi-dimensional arrays.

Pandas makes it straightforward to complete many of the tedious, time-consuming activities involved in working with data, including:

- Data cleansing

- Data filling
- Data visualization
- Data normalization
- Merges and joins
- Statistical analysis
- Loading and saving data
- Data inspection

According to respected data scientists, you can actually accomplish anything, making Pandas the finest data analysis and manipulation tool accessible.

3.2.3: SEABORN

A Python module called Seaborn is used to make statistical visuals. It has a matplotlib foundation and works seamlessly with pandas data structures. You can explore and comprehend the details with Seaborn's help. Its charting functions work with dataframes and arrays holding full datasets, internally executing the necessary statistical aggregation and semantic mapping to produce useful graphs. You may focus on the meaning of your plots rather than the specifics of how to construct them thanks to its declarative, dataset-oriented API.

3.2.4: Matplotlib

A Python module called Matplotlib can be used to produce interactive, animated, and static visualisations. It is a thorough library that makes both easy and difficult things feasible. Many different types of people use Matplotlib, including artists, data scientists, and students. For the Python programming language and its NumPy numerical mathematics extension, Matplotlib is a graphing library. For integrating plots into programmes utilising all-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK+, it offers an object-oriented API.

3.2.5: Scikit learn

Scikit-learn provides a dependable Python framework for a selection of supervised and unsupervised learning methods. The Python programming language's free machine learning package is called Scikit-learn (formerly known as Scikits.learn and also known as Sklearn). Support vector machines, random forests, gradient boosting, k-means, and DBSCAN are among the classification, regression, and clustering algorithms that are included in it. It is also built to work with the Python scientific and numerical libraries NumPy and SciPy.

3.2.6: Streamlit

An open-source Python tool called Streamlit makes it simple to develop interactive web apps for data research. Engineers and data scientists who want to share their work with others can use Streamlit. Streamlit may be used to construct a wide range of web apps and is simple to learn and use.

3.2.7: PICKLE

Python's Pickle module allows you to serialise and deserialize Python objects. Deserialization is the process of transforming a stream of bytes back into an object once they have been converted from an object via serialisation. Python objects can be sent over a network or saved to a file using Pickle.

CHAPTER-4

WORK FLOW

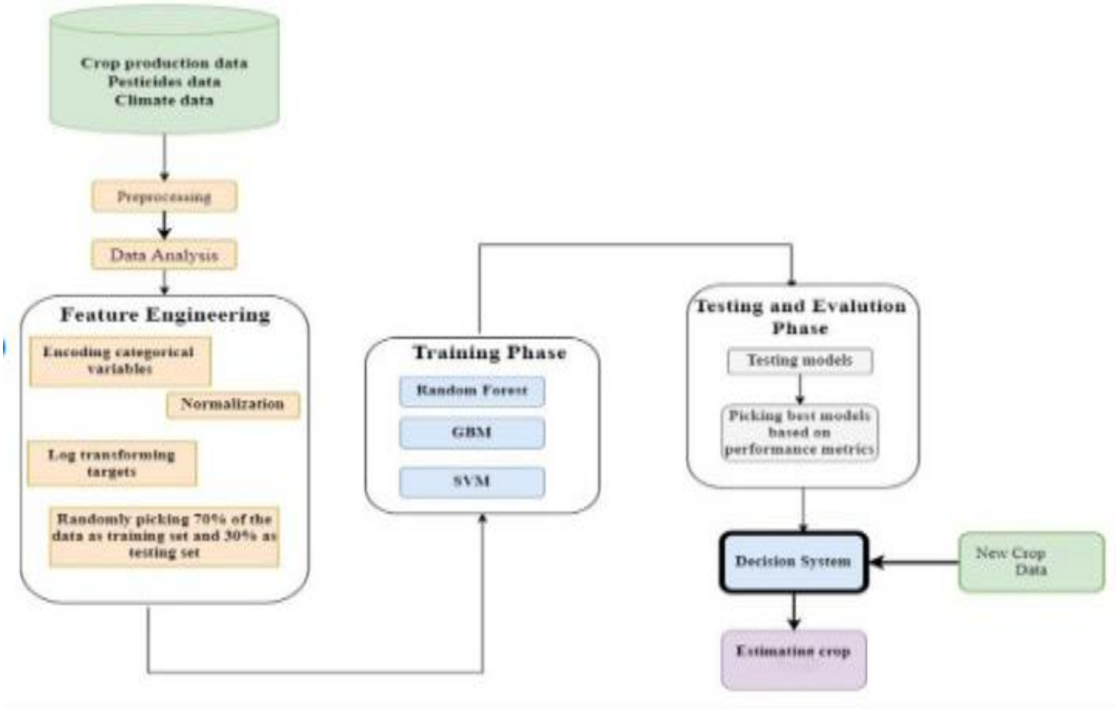


Fig 1. Workflow diagram

4.1 Data Collection :Data collection is the process of gathering relevant information or data from various sources to be used for analysis, decision-making, or research purposes. It is a crucial step in any data-driven project, including machine learning, analytics, or business intelligence. Here's a brief explanation about the data collection process where we can know the various ways and process to collect and organise the data to train our model.

- By Defining Objectives, which means we have to Start clearly by defining the objectives and goals of our data collection efforts.
- By Identifying potential Data Sources which provides Required info which can include structured data from databases, spreadsheets, or APIs, as well as unstructured data from text documents, images, social media platforms, or web scraping.
- By Selecting appropriate methods to collect the required data, This can include surveys, interviews, questionnaires, experiments, observations, or automated data retrieval processes. And also ensure the collected data is accurate and reliable.

4.2 Pre-Processing : Preprocessing, basically refers to the preparatory steps performed on raw data before it is used for analysis, Modeling, or other data-driven tasks. It involves transforming and organizing the data in a way that makes it suitable for further processing and analysis.

Here are some key steps involved in data Preprocessing:

- Data Integration, which is the process of combining multiple data sources or datasets into a unified format. This involves resolving inconsistencies in attribute names, data types, and formatting across different sources to ensure data compatibility and consistency.
- Data Cleaning, which involves identifying and handling missing values, outliers, and inconsistencies in the data.
- Data Transformation, which is performed to normalize or scale the data, by making it more suitable for analysis. And also Common transformation techniques which include normalization, standardization, log transformation, or power transformation. These techniques makes the data into a common scale, by removing skewness, and satisfy the assumptions of statistical models

4.3 Feature Engineering : Feature engineering is the process of creating new features or transforming existing features in a dataset to necessary features which can improve the performance and predictive power of machine learning models. It involves selecting, combining, or deriving features that capture relevant information from the data and make it easier for the models to learn patterns and relationships.by Performing Feature Selection, Feature Extraction where it can be done through various techniques like Principal Component Analysis, Linear Discriminant Analysis etc.. , Also by handling Categorical Variables with the techniques called as one-hot encoding or Label Encoding, And also by Performing Feature Scaling which normalises the data to common scale which makes the model training easier. finally , The goal is to create a set of features that best represents the underlying patterns and relationships in the data, which leads to improve the model performance and make more accurate predictions.

4.4 Training Phase and Testing Phase : Training and testing are two fundamental phases in the development and evaluation of machine learning models. Here's a brief explanation of each phase:

In training phase, basically the machine learning model learns from the available data to identify patterns, relationships, and statistical dependencies. The process involves presenting the model with a labelled dataset, where the input data (features) and the corresponding output (labels or target variable) are known. Where the model adjusts its internal parameters or weights based on the provided data to it So, the objective in the training phase is to find the optimal set of parameters that minimize the difference between the predicted output and the true output. Finally, The training process continues until a stopping criterion is met, such as reaching a predefined number of iterations or achieving a satisfactory level of performance.

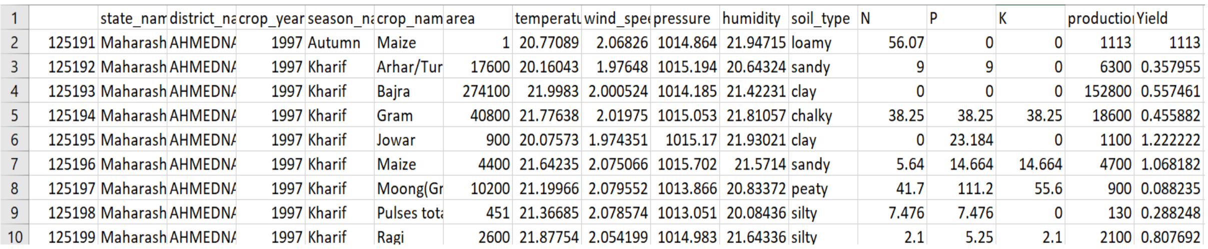
In Testing Phase Basically, it needs to be evaluated to assess its performance and generalization ability. By using a separate dataset, called the testing or validation dataset, which was not used during the training phase. So, during the testing phase, the trained model takes the input data from the testing dataset and generates predictions or classifications. These predictions are then compared with the true labels to evaluate the model's performance. Common evaluation metrics include accuracy, precision, recall, F1 score, or area under the receiver operating characteristic curve (ROC AUC).And also the testing phase provides insights into how well the trained model performs on unseen data. It helps assess the model's ability to generalize and make accurate predictions on data that it hasn't encountered before. By evaluating the model on a separate dataset, we can estimate its performance on real-world scenarios and determine if any overfitting or underfitting has occurred during the training process.

Finally, training and testing phases are iterative and often involve fine-tuning the model, by adjusting hyperparameters, and conducting cross-validation to ensure robustness and optimal performance. So, We can say that these phases are very crucial for developing reliable and accurate machine learning models by that it can effectively generalize to new, unseen data.

CHAPTER-5

PROPOSED METHOD

5.1 Data Description **:** The dataset was gathered from the Smart AI Technologies website. Crop data comprises season names, crop names, area, temperature, wind speed, pressure, humidity, soil type, NPK nutrients, production, and yield for 35 different districts in Maharashtra State for a period of 18 years (i.e., 1997-2014). Below was displayed a brief data sample [fig. 1]. where area is measured in hectares, crop production is measured in tonnes per hectare, and crop yield is measured as the amount of crop produced per unit of harvested or planted land (in kilogrammes). The former use of this dataset was for crop yield prediction; however, we are using it now for crop recommendation using machine learning.



1		state_nam	district_na	crop_year	season_na	crop_nam	area	temperatu	wind_spe	pressure	humidity	soil_type	N	P	K	productio	Yield
2	125191	Maharash	AHMEDNA	1997	Autumn	Maize	1	20.77089	2.06826	1014.864	21.94715	loamy	56.07	0	0	1113	1113
3	125192	Maharash	AHMEDNA	1997	Kharif	Arhar/Tur	17600	20.16043	1.97648	1015.194	20.64324	sandy	9	9	0	6300	0.357955
4	125193	Maharash	AHMEDNA	1997	Kharif	Bajra	274100	21.9983	2.000524	1014.185	21.42231	clay	0	0	0	152800	0.557461
5	125194	Maharash	AHMEDNA	1997	Kharif	Gram	40800	21.77638	2.01975	1015.053	21.81057	chalky	38.25	38.25	38.25	18600	0.455882
6	125195	Maharash	AHMEDNA	1997	Kharif	Jowar	900	20.07573	1.974351	1015.17	21.93021	clay	0	23.184	0	1100	1.222222
7	125196	Maharash	AHMEDNA	1997	Kharif	Maize	4400	21.64235	2.075066	1015.702	21.5714	sandy	5.64	14.664	14.664	4700	1.068182
8	125197	Maharash	AHMEDNA	1997	Kharif	Moong(Gr	10200	21.19966	2.079552	1013.866	20.83372	peaty	41.7	111.2	55.6	900	0.088235
9	125198	Maharash	AHMEDNA	1997	Kharif	Pulses tota	451	21.36685	2.078574	1013.051	20.08436	silty	7.476	7.476	0	130	0.288248
10	125199	Maharash	AHMEDNA	1997	Kharif	Ragi	2600	21.87754	2.054199	1014.983	21.64336	silty	2.1	5.25	2.1	2100	0.807692

Fig 2. Small snippet of data.

5.2 Pre-Processing: It is crucial to undertake a data pre-processing step before developing a model for any data. In this step, raw data is cleaned and converted to produce high-quality data for analysis. This csv dataset contained a total of 17 attribute columns, 12 of which were numerical and the remaining five were categorical. By using these 17 attribute columns, 12,628 records of agricultural data were included in the dataset.

5.3 EDA: EDA plays a Crucial role in Model Building, which is one of the major task in data science life cycle. After doing Analysis for this csv dataset, we found out couple of major Aspects which makes sense and shape to a dataset, to build the models efficiently and effectively.

As mentioned before, this dataset had been used to predict the crop yield, where we noticed a **High Spike,** which was shown in below [Fig 3]. Yield Values, In Initial 3-4 years, later the yield was Stabilized. So, we were Considering the data from the year 2000, By this we can get rid off high Variance and low bias which may cause of Overfitting of data which leads to wrong predictions of the model

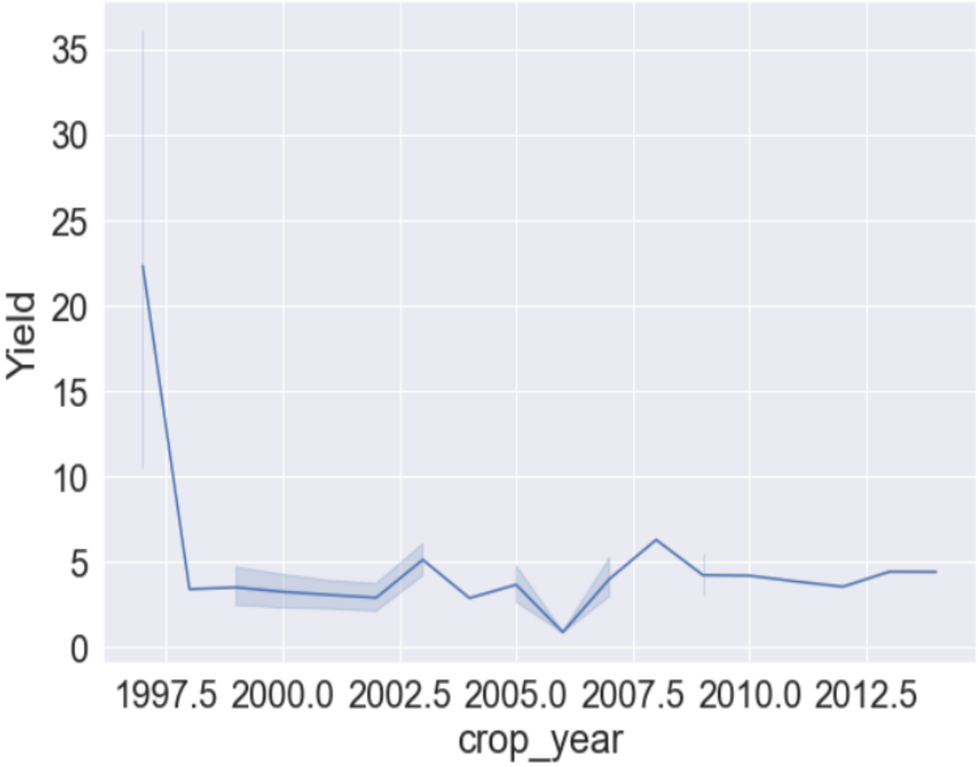


Fig 3. Yield distribution across the years

Secondly, as we were doing a Crop Recommendation model, we know that crop names are the label/Target Column. where we noticed the high **Data Imbalance** among the crop names. Again, this may cause to the high bias towards crops which having higher number of crops. So that We were choose a best 8 crops where crops having less imbalance between them same was shown in the below [fig 4].by this, we can make our model to predict Effectively.

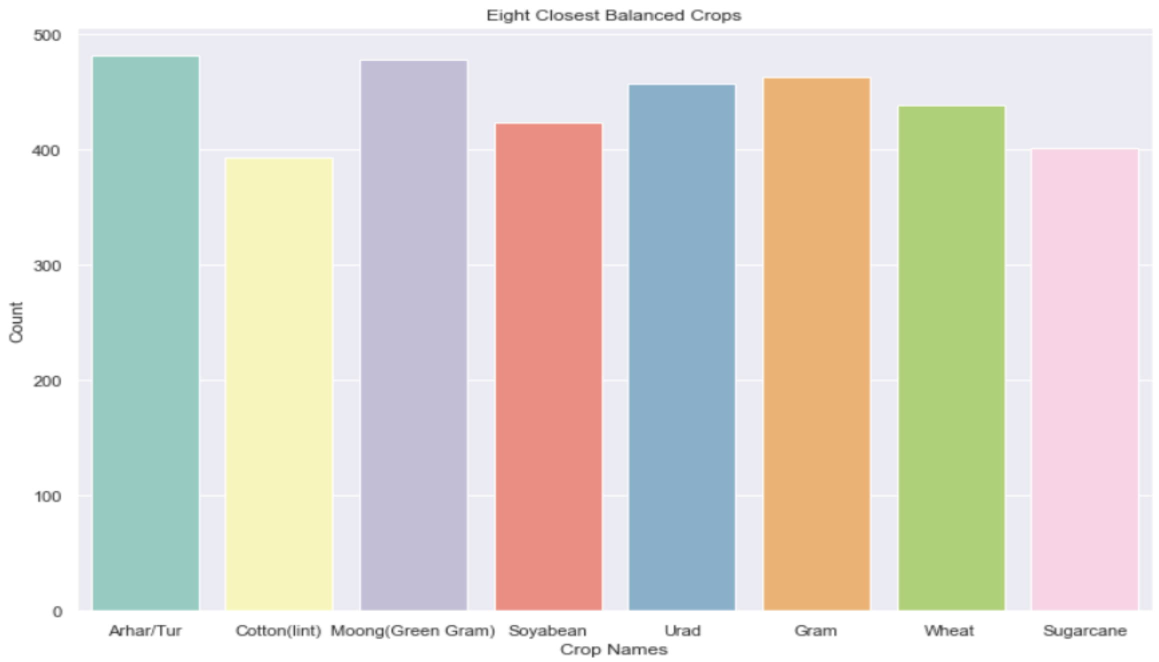


Fig 4. distribution of crops

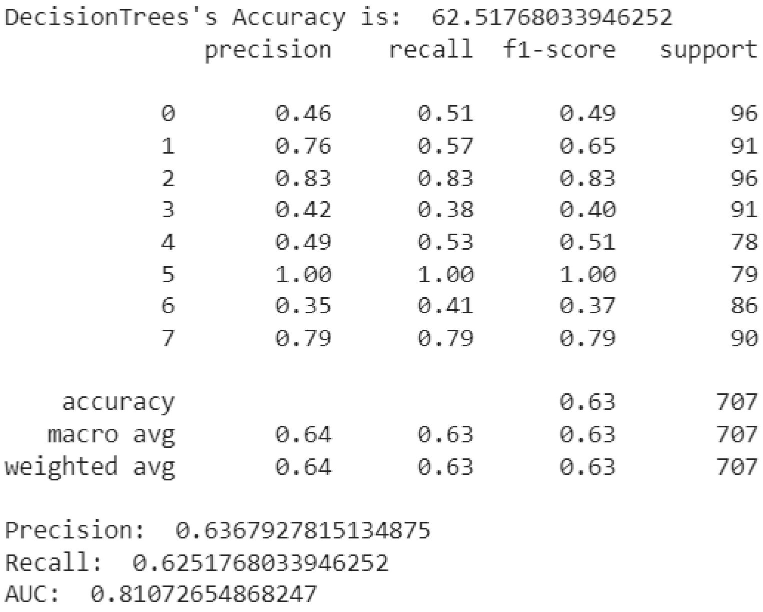
5.4 Feature Engineering : Sometimes, it is very difficult to make conclusions and draw methods from the raw data where Feature Engineering comes into the picture and makes easier to draw methods. In this data also we had Categorical Columns to draw Conclusions from this Categorical Values. We had been used Feature Engineering Techniques such as One-hot Encoding, Label Encoding. So that machine can understand the data properly and provide useful insights to draw conclusions.

5.5 Algorithms Used *:*

5.5.1: Decision Tree

A tree has many analogues in the real world, and it turns out that it has influenced many aspects of machine learning, such as classification and regression. A type of machine learning model called a decision tree can do a number of tasks with great accuracy and is also relatively simple to comprehend. Decision trees differ from other ML models in that they consistently represent knowledge. After being trained, a decision tree's "information" is explicitly structured into a hierarchical framework. Even non-experts will be able to understand the material thanks to the way it is organised and presented in this manner.

This algorithm can be used in a crop recommendation machine learning project to provide farmers with meaningful information and recommendations. In order to make precise predictions about the ideal crop selection, the system may learn from historical data, such as prior crop yields and agricultural practises. Decision tree models can also take into account several decision paths, enabling complex decision-making based on a variety of factors. Due of their interpretability, which can help farmers make decisions, decision trees are particularly useful in explaining the reasoning behind crop recommendations. Decision tree models can also be easily updated with new data, allowing for continuous crop recommendation system improvement. Overall, the decision tree algorithm may be a useful tool in crop recommendation projects because it provides data-driven suggestions.



```
DecisionTrees's Accuracy is:  62.51768033946252
              precision    recall  f1-score   support

           0       0.46      0.51      0.49        96
           1       0.76      0.57      0.65        91
           2       0.83      0.83      0.83        96
           3       0.42      0.38      0.40        91
           4       0.49      0.53      0.51        78
           5       1.00      1.00      1.00        79
           6       0.35      0.41      0.37        86
           7       0.79      0.79      0.79        90

    accuracy                           0.63       707
   macro avg       0.64      0.63      0.63       707
weighted avg       0.64      0.63      0.63       707

Precision:  0.6367927815134875
Recall:  0.6251768033946252
AUC:  0.81072654868247
```

Fig 5: Decision tree accuracy

In this, we'll feed the features and the target variable to the tree classifier, and we'll get the accuracy as described above. As depicted in Fig.() Decision tree precision obtained 62.51% accuracy with the decision tree model.

5.5.2: Random Forest

A random forest, as its name suggests, is an ensemble of several different decision trees that operate in concert. The class with the highest votes becomes the prediction of our model (see figure below). Each tree in the random forest generates a class prediction. A random forest, as its name suggests, is an ensemble of several different decision trees that operate in concert.

Every tree in the random forest generates a class prediction, and our model predicts the class that receives the most votes.

The Random Forest algorithm, which can handle complex datasets and provide precise forecasts, can be quite helpful in a project proposing crops. Random Forest can discover a variety of patterns and interactions between various parameters, such as soil characteristics, meteorological conditions, and crop attributes, by using a variety of decision trees. Despite noise and overfitting, this ensemble method enables the generation of proposals that are resilient and dependable. Moreover, Random Forest provides rankings of feature importance that can be used to pinpoint the crop selection variables with the greatest effects. Real-time recommendations are made possible by Random Forest's efficiency at handling vast volumes of data continuously. Overall, Random Forest can improve the clarity and accuracy of crop suggestions, helping farmers make defensible choices.

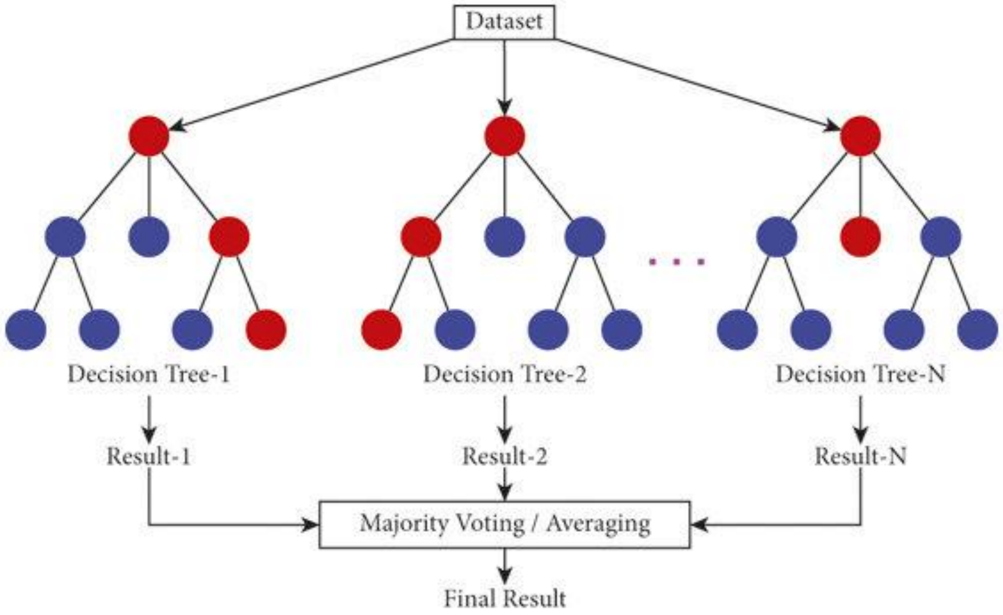
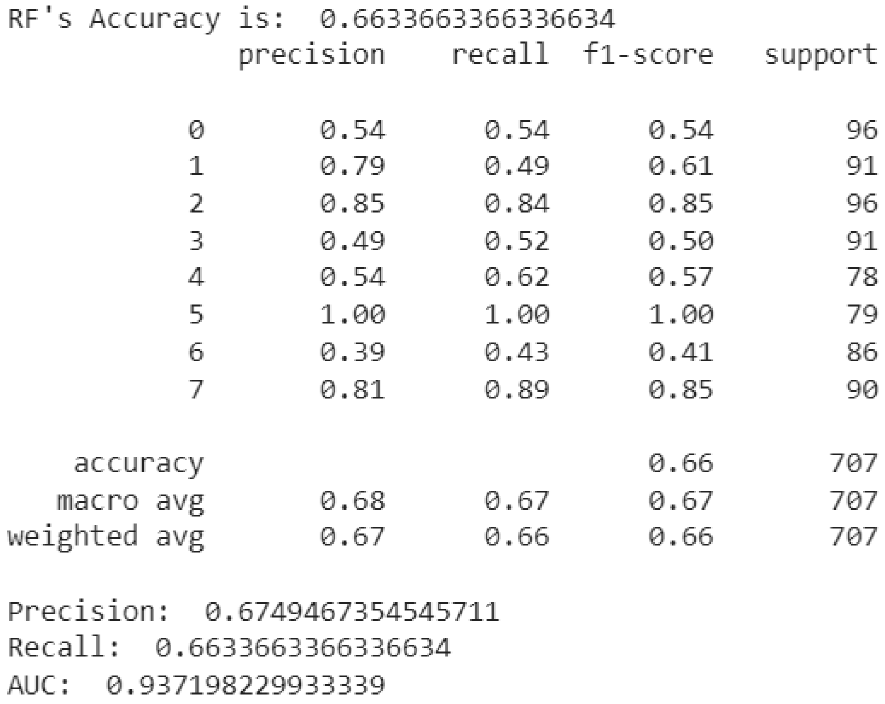


Fig:6: Random Forest tree model

The features and the target variable were supplied to our random forest classifier. One of the most effective ensemble learning techniques in machine learning is random forest categorization. As a result, our model's accuracy was 66.3366% .



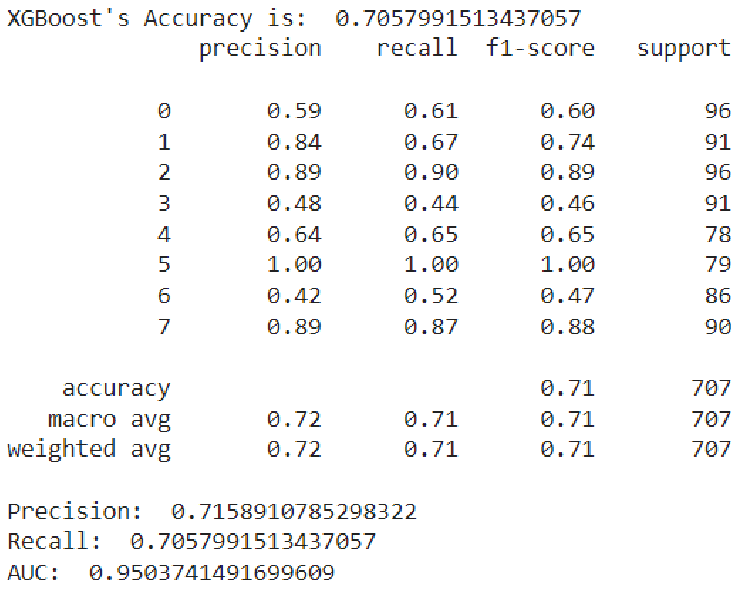
```
RF's Accuracy is:  0.6633663366336634
              precision    recall  f1-score   support

           0       0.54      0.54      0.54        96
           1       0.79      0.49      0.61        91
           2       0.85      0.84      0.85        96
           3       0.49      0.52      0.50        91
           4       0.54      0.62      0.57        78
           5       1.00      1.00      1.00        79
           6       0.39      0.43      0.41        86
           7       0.81      0.89      0.85        90

    accuracy                           0.66       707
   macro avg       0.68      0.67      0.67       707
weighted avg       0.67      0.66      0.66       707

Precision:  0.6749467354545711
Recall:  0.6633663366336634
AUC:  0.937198229933339
```

Fig 7 : Random Forest tree accuracy

5.5.3: XG Boost

Extreme Gradient Boosting, often known as XGBoost, is a potent and popular machine learning technique. It is noted for its outstanding predictive performance and is built on the gradient boosting architecture. The advantages of boosting algorithms are combined with a number of cutting-edge features, such as parallelized tree construction, regularisation strategies, and a variable set of hyperparameters, in XGBoost. With the help of these capabilities, XGBoost can successfully manage huge datasets, achieve great accuracy, and offer superb interpretability. XGBoost has grown significantly in popularity in both academic research and business applications. It is a top option for data scientists and machine learning professionals who are trying to solve challenging prediction challenges because of its adaptability, speed, and scalability.

The powerful gradient boosting algorithm XGBoost, which can handle complicated and non-linear data correlations, can be very helpful in a crop recommendation project. With its high accuracy and predictive capacity, XGBoost is renowned for providing precise crop recommendations based on a range of variables like soil quality, weather, previous crop data, and more. It is suited for use in actual agricultural scenarios because of its capacity to manage big datasets with efficiency and automatically fill in missing data. To further assist farmers in understanding how features affect crop suggestions, XGBoost also provides rankings of feature relevance. Thanks to its capabilities for parallel processing, XGBoost makes it possible to use large-scale crop recommendation systems. XGBoost, which provides accurate predictions and other useful information, has the potential to be a powerful tool for crop recommendation projects.



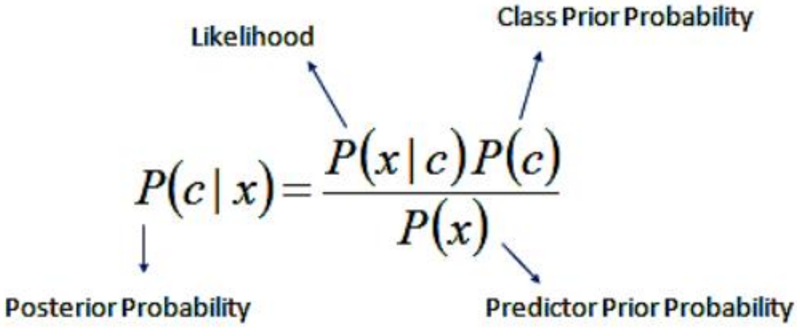
```
XGBoost's Accuracy is:  0.7057991513437057
              precision    recall  f1-score   support

           0       0.59      0.61      0.60        96
           1       0.84      0.67      0.74        91
           2       0.89      0.90      0.89        96
           3       0.48      0.44      0.46        91
           4       0.64      0.65      0.65        78
           5       1.00      1.00      1.00        79
           6       0.42      0.52      0.47        86
           7       0.89      0.87      0.88        90

    accuracy                           0.71       707
   macro avg       0.72      0.71      0.71       707
weighted avg       0.72      0.71      0.71       707

Precision:  0.7158910785298322
Recall:  0.7057991513437057
AUC:  0.9503741491699609
```

Fig 8: XG Boost accuracy

The target variable and the characteristics were input to the XGBoost classifier. XGBoost is an ensemble learning technique that strengthens predictions by combining those from several weak models. As a result, our model's accuracy was 70%.

5.5.4: Naïve Bayes

A Naive Bayes classifier, a probabilistic machine learning model, is used for classification problems. The Bayes theorem will serve as the basis for classifiers.

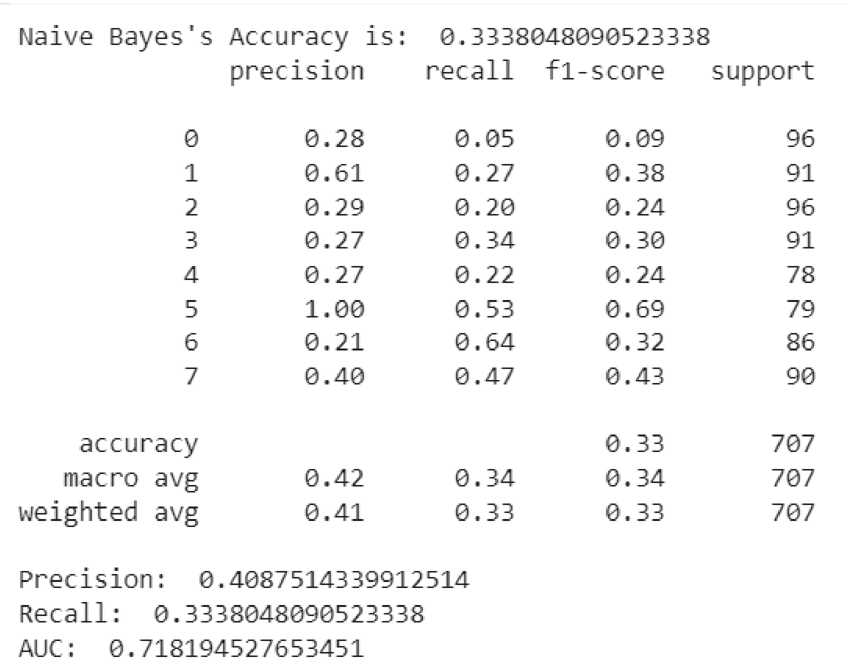


$$P(c|X) = P(x_1|c) \times P(x_2|c) \times \cdots \times P(x_n|c) \times P(c)$$

Fig 9 : Naïve bayes formula

The Bayes theorem can be used to calculate the probability of something happening in the event that X has already happened. X is the claim, and C is the supporting evidence. The predictors and traits in this case are assumed to be independent. To put it another way, having one attribute does not affect having the other. Therefore, it is viewed as naive.

The Naive Bayes method, which is straightforward, efficient, and capable of handling both categorical and discrete data, can be helpful in a crop recommendation project. The classifier with probability. The conditional likelihood that a crop will be suitable for a particular collection of characteristics, such as soil type, weather conditions, and crop traits, is determined via naive Bayes. Because it only needs a little amount of training data to produce predictions, it is particularly well suited for applications with little available data. Naive Bayes is quick and effective for real-time suggestions because it requires less computational work. Additionally, Naive Bayes offers straightforward outcomes that help farmers comprehend the rationale behind the advice. Overall, because it provides precise and clear forecasts for the ideal crop



```
Naive Bayes's Accuracy is:  0.3338048090523338
              precision    recall  f1-score   support

           0       0.28      0.05      0.09        96
           1       0.61      0.27      0.38        91
           2       0.29      0.20      0.24        96
           3       0.27      0.34      0.30        91
           4       0.27      0.22      0.24        78
           5       1.00      0.53      0.69        79
           6       0.21      0.64      0.32        86
           7       0.40      0.47      0.43        90

    accuracy                           0.33       707
   macro avg       0.42      0.34      0.34       707
weighted avg       0.41      0.33      0.33       707

Precision:  0.4087514339912514
Recall:  0.3338048090523338
AUC:  0.718194527653451
```

Fig 10 :Gaussian Naive Bayes accuracy

If the predictors take up a continuous value rather than a discrete value, we suppose that these values are samples from a gaussian distribution. Given that our dataset had a normal distribution. Therefore, the Gaussian Naive Bayes Classifier was employed as our classifier. The Naive Bayes classifier is, in general, quite effective at predicting or recommending systems. As a result, we were able to achieve an accuracy of 33.38% (as depicted in Fig.).

5.5.5: K nearest neighbors

In a crop selection project, the K-nearest neighbours (KNN) technique, which is simple and effective in managing both category and numerical data, can be useful. Because KNN is a lazy learner, it can be used to produce suggestions in the present without the need for training. By comparing the similarity of nearby data points, KNN can suggest crops based on historical data on crop performance, soil quality, weather conditions, and other relevant criteria. Since it can adapt to changing environmental conditions, it is ideal for hectic agricultural scenarios. Farmers may comprehend the reasoning behind the recommendations since KNN is interpretable. Due to its simple implementation and low processing cost, it is appropriate for environments with limited resources. On the other hand, KNN might need hyperparameters like the distance metric and the number of neighbours (K) to be carefully adjusted. KNN is a useful and clear tool for crop recommendation campaigns since it has the ability to offer real-time suggestions based on local similarity between data points.

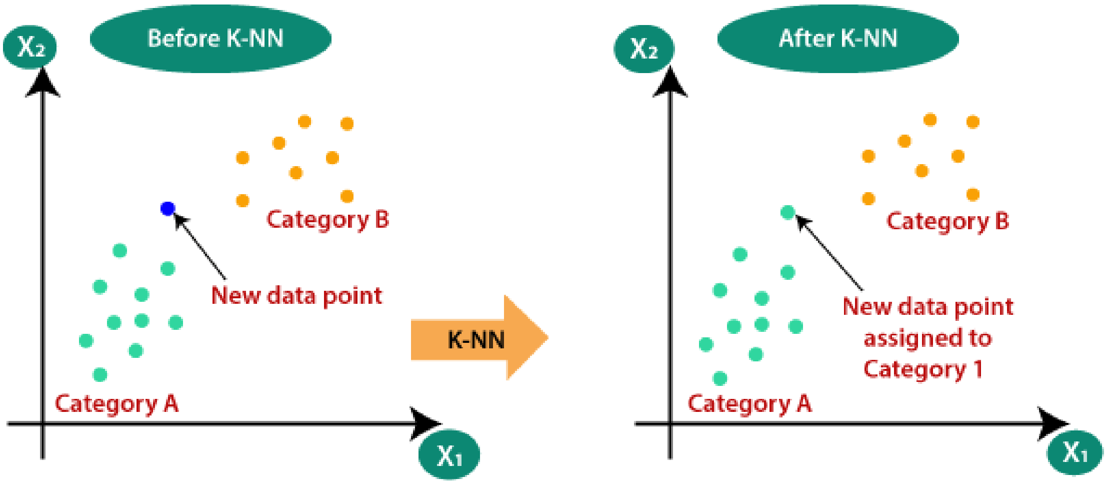
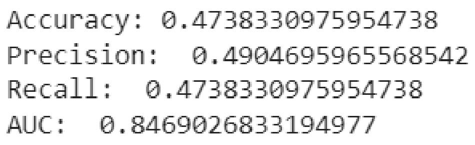


Fig:11 : K-Nearest Neighbours (k-NN)

As the k-nearest neighbours (k-NN) technique is a straightforward yet effective machine learning approach used for both classification and regression applications. Based on a distance metric, usually Euclidean distance, the algorithm determines the "k" nearest neighbours in the training dataset. As a result, our model had an accuracy of 47%.



```
Accuracy: 0.4738330975954738
Precision:  0.4904695965568542
Recall:  0.4738330975954738
AUC:  0.8469026833194977
```

Fig 12: K-Nearest Neighbours (k-NN) accuracy

5.6: Metrics:

5.6.1: Accuracy: A key parameter used in machine learning to evaluate a model's performance is accuracy. It gauges how accurately classifier predictions are made on the whole. Accuracy is specifically defined as the proportion of accurate predictions to all other predictions.

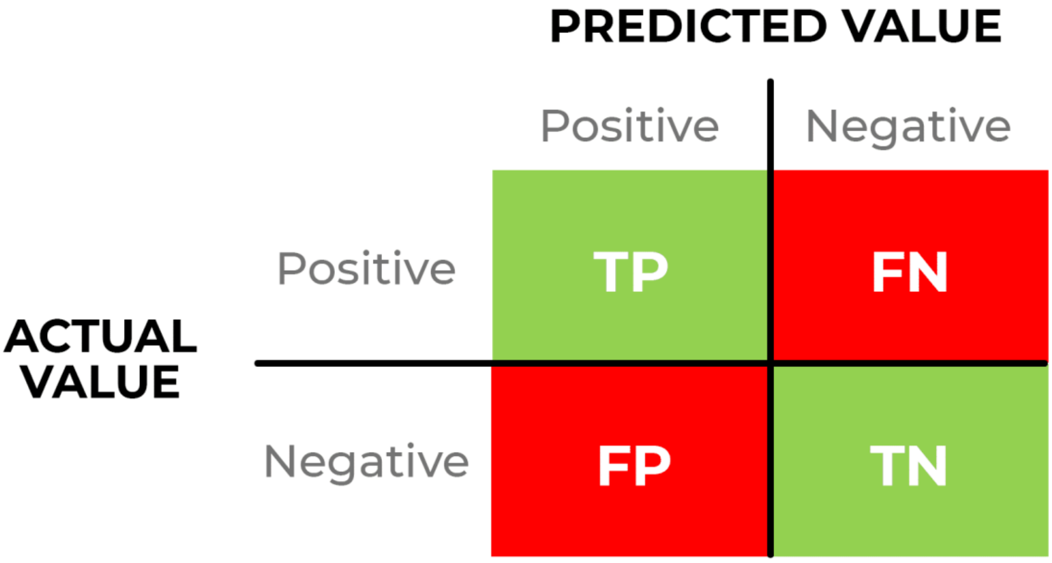
To determine accuracy, we contrast the genuine labels of the relevant instances in the dataset with the predicted labels produced by a model. A label is given to each occurrence, and the model predicts a label for every case. If the projected label matches the actual label, it is deemed accurate; otherwise, it is deemed inaccurate. The accuracy is then determined by taking the percentage obtained by multiplying the number of accurate predictions by the total number of forecasts by 100.

A general indicator of a model's performance on a certain dataset is accuracy. It is frequently employed in assignments where classes are balanced, i.e., instances from each class are distributed equally. For instance, accuracy can be a trustworthy metric in a binary classification problem with two classes (for instance, "spam" or "not spam") if both classifications are present in the dataset in nearly the same amounts.

However, accuracy might not be enough to assess a model's performance by itself, particularly when working with unbalanced datasets. Datasets that are unbalanced have a disproportionately large number of examples in one class compared to the other. In such circumstances, a classifier can attain high accuracy while underperforming on the minority class if it consistently predicts the majority class. This is so because accuracy does not account for how the classes are distributed.

In light of this, it's crucial to take into account additional measures, such as precision, recall, F1 score, or area under the ROC curve (AUC), depending on the particular issue at hand and the objectives of the investigation. These metrics offer a more thorough assessment by concentrating on many facets of model performance, such as the capacity to accurately identify positive examples (precision) or the capacity to record all positive instances (recall).

In conclusion, accuracy is a commonly used metric in machine learning that assesses how accurate predictions provided by a model are on the whole. It is determined as a percentage by dividing the total number of guesses by the number of correct predictions. Although accuracy is a helpful indicator, it should be used with caution, especially when working with unbalanced datasets or when the relative costs of various types of errors are not the same.



		PREDICTED VALUE	
		Positive	Negative
ACTUAL VALUE	Positive	TP	FN
	Negative	FP	TN

Fig 13 : Confusion Matrix

$$\text{Accuracy} = \frac{(TP + TN)}{(TP + FP + TN + FN)}$$

Fig 14: Accuracy formula

5.6.2: Precision and recall: In machine learning, precision and recall are two crucial metrics used to assess the effectiveness of classifiers, particularly in binary classification issues. They shed light on the model's capacity for recalling all good occurrences and for making precise positive forecasts.

Precision: Precision is the percentage of a classifier's true positive predictions among all of its positive predictions. It addresses the issue, "Of all the instances predicted as positive, how many are truly positive?" and focuses on the reliability of positive predictions.

The precision is computed by dividing the total of true positive predictions by the sum of true positive forecasts and erroneous positive predictions. In other words, it shows how well the model can identify good cases while minimising false positives. A higher precision score means the classifier is more likely to be right when classifying an event as positive.

When the cost of false positive mistakes is significant, precision is particularly crucial. For instance, misdiagnosing a healthy patient with cancer can result in invasive and unneeded treatments. In these circumstances, high precision is preferred to reduce false positives.

Recall: Recall, often referred to as sensitivity or true positive rate, calculates the percentage of genuine positive predictions among all real positive examples in the dataset. How many of the actually positive occurrences did the classifier properly detect, then?

Recall is computed by dividing the total of accurate positive predictions by the total of accurate positive forecasts plus accurate negative predictions. It rates the model's capacity to identify every positive case while avoiding many false negatives. A higher recall score means that a greater percentage of the dataset's positive cases may be successfully identified by the model.

When the cost of false negative errors is considerable, recall is especially crucial. For instance, misclassifying a fraudulent transaction as legitimate in a fraud detection system can cost money. In these situations, a high recall is preferred to reduce false negatives and guarantee that the majority of positive instances are correctly identified.

Precision and Recall are frequently inversely correlated, which means that increasing one may cause a decline in the other. Precision and recall are trade-offs, and the ideal balance depends on the particular application and needs.

The F1 score is frequently employed to achieve a balance between recall and precision. The F1 score provides a single value that incorporates both measurements and is the harmonic mean of precision and recall. It is a helpful metric when both recall and precision are crucial, and it offers a fair assessment of the model's performance.

In conclusion, recall gauges how well a model can identify all instances of positivity while precision gauges how accurately positive predictions are made. Both measures are essential for evaluating the effectiveness of classifiers, and the ideal ratio of accuracy to recall relies on the particular problem domain and the relative costs of various sorts of errors.

$$\begin{aligned} precision &= \frac{TP}{TP + FP} \\ recall &= \frac{TP}{TP + FN} \\ F1 &= \frac{2 \times precision \times recall}{precision + recall} \end{aligned}$$

Fig 15: Precision Recall formula

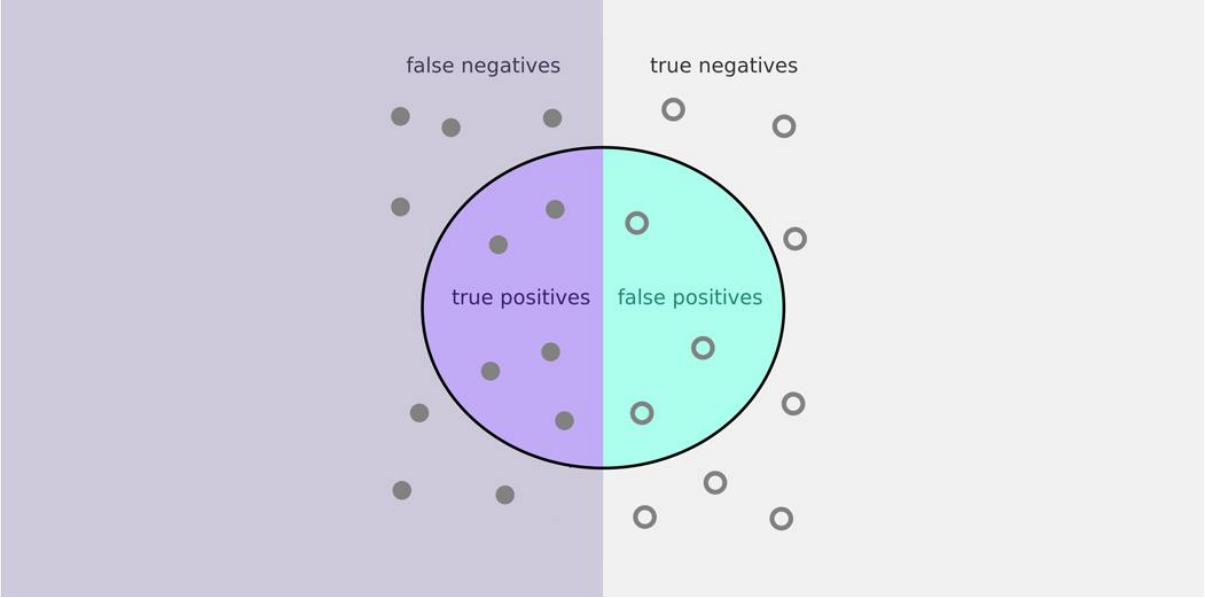


Fig 16 : Tradeoff between Precision and recall

5.6.3: Area under the curve: By analysing the Receiver Operating Characteristic (ROC) curve, the Area Under the Curve (AUC) statistic can be used to gauge how well a binary classifier performs. The classification algorithm's effectiveness as the discrimination threshold changes is graphically depicted by the ROC curve.

Let's first discuss the ROC curve to better comprehend AUC. The True Positive Rate (TPR) and False Positive Rate (FPR) are plotted against one other at different threshold values to produce the ROC curve. The true positive rate (TPR), commonly referred to as recall or sensitivity, is the percentage of true positive forecasts among all real positive cases. The FPR, on the other hand, is the ratio of predicted false positives to actual predicted false negatives.

The trade-off between the true positive rate and the false positive rate as the classifier's threshold is altered is visually represented by the ROC curve. It demonstrates the classifier's capacity to accurately categorise positive occurrences (TPR) while minimising the misclassification of negative instances (FPR). Each point on the ROC curve represents a certain classifier threshold setting.

The area under the ROC curve integral, which denotes the AUC, is then determined. The AUC ranges from 0 to 1, with 1 denoting a perfect classifier and 0.5 indicating a random classifier.

The performance of the classifier can be understood by interpreting the AUC value. A better classifier has a higher AUC because it has a greater area under the ROC curve, which reflects better discriminating between positive and negative cases. To put it another way, a higher AUC indicates that the classifier is more likely to score a randomly selected positive instance higher than a randomly selected negative instance.

The AUC is especially helpful in situations when there is a class imbalance or where the costs of false positives and false negatives vary. Independent of a particular threshold selection, it provides a comprehensive evaluation of the classifier's effectiveness across a range of threshold choices.

In conclusion, the area under the receiver operating characteristic (ROC) curve is measured to determine the area under the curve, which assesses the total performance of a binary classifier. It offers a thorough assessment of the classifier's capacity to distinguish between positive and negative instances, regardless of the threshold option. An AUC above 0.5 denotes superior performance, whereas an AUC below 0.5 denotes random classification.

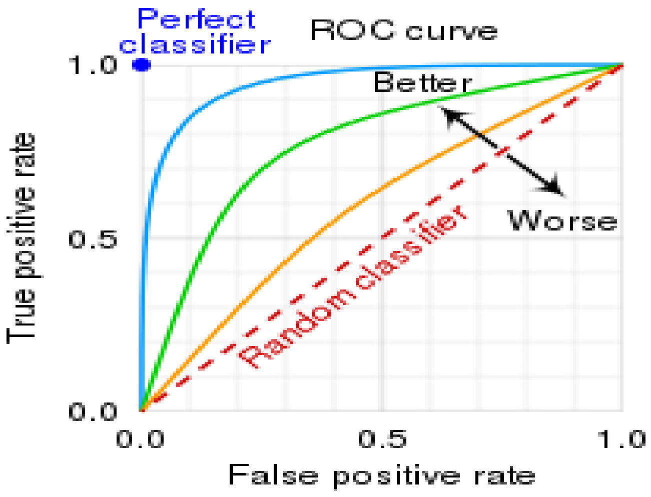
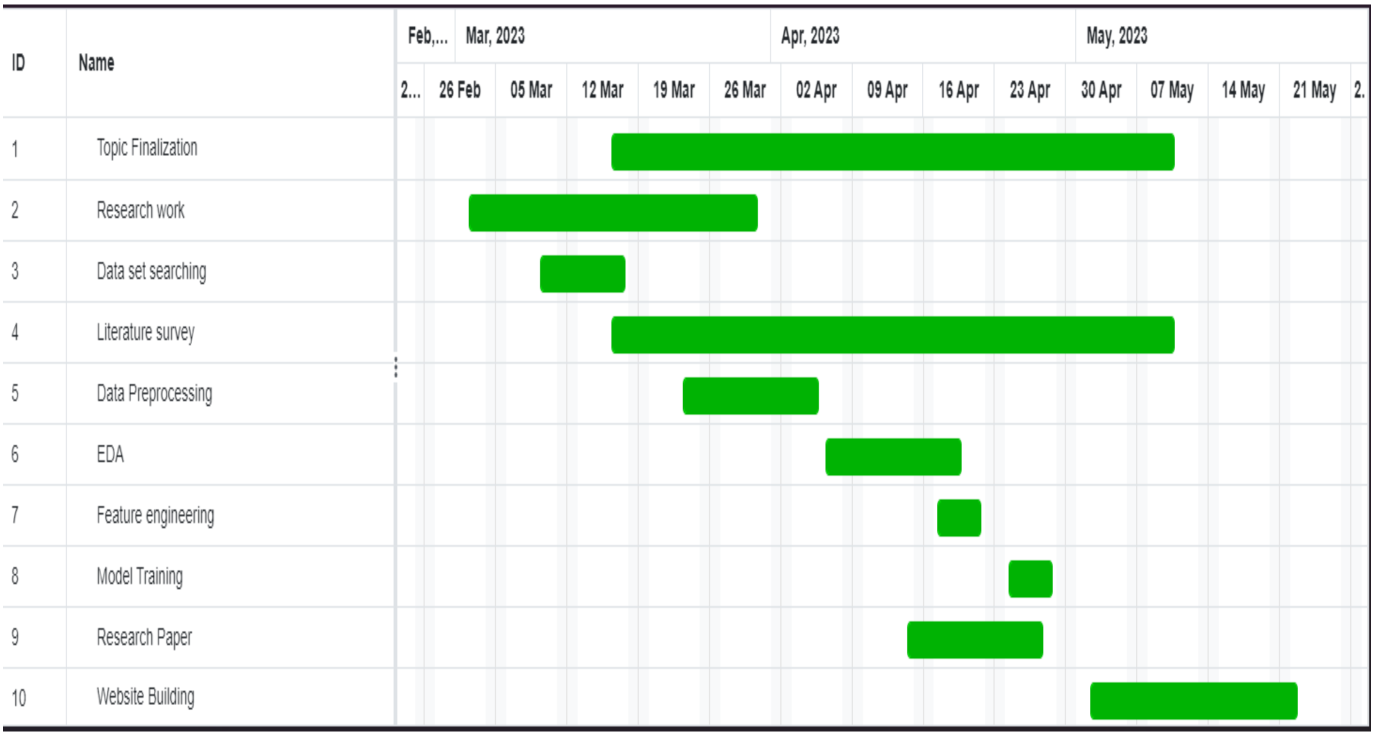


Fig 17 : ROC curve

CHAPTER-6

TIMELINE FOR EXECUTION OF PROJECT

(GANTT CHART)



ID	Name
1	Topic Finalization
2	Research work
3	Data set searching
4	Literature survey
5	Data Preprocessing
6	EDA
7	Feature engineering
8	Model Training
9	Research Paper
10	Website Building

CHAPTER-7

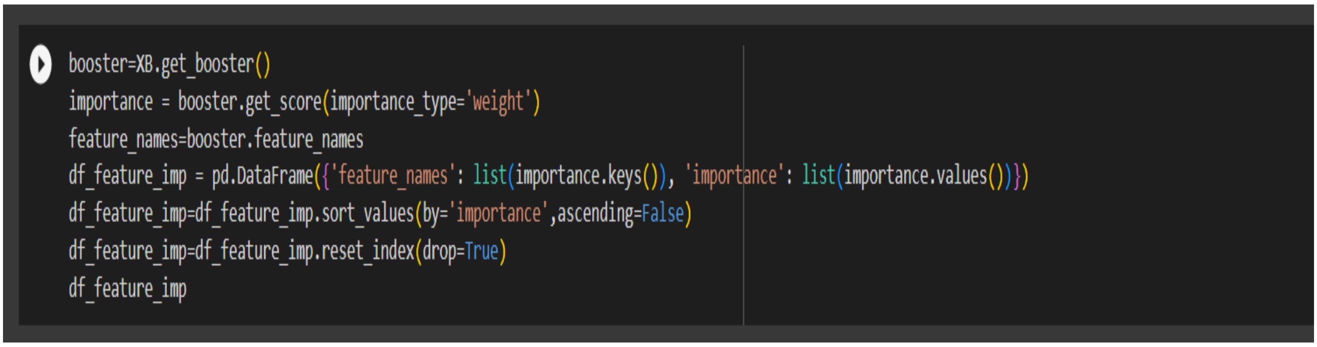
RESULTS AND DISCUSSION

Which model to use depends on the aforementioned metrics, including accuracy score, precision, recall, and area under the curve. These indicators are widely used to judge how well machine learning algorithms are working. The top-performing algorithm in the table below is XG Boost, which has an accuracy of 70%, precision of 0.71, recall of 0.70, and AUC of 0.95. These data demonstrate that XG Boost can predict work outcomes with accuracy, giving it a credible option for the project. The model's high AUC value of 0.95 also shows that it has great discriminatory power in separating positive and negative outcomes. Overall, XG Boost performs better than alternative algorithms, making it a viable option for continued use in the project.

Algorithm	**Accuracy**	**Precision**	**Recall**	**AUC**
Decision Tree	0.62	0.63	0.62	0.81
Random Forest	0.66	0.67	0.66	0.93
XG Boost	0.70	0.71	0.70	0.95
KNN	0.47	0.49	0.47	0.84
Naïve Bayes	0.33	0.40	0.33	0.71

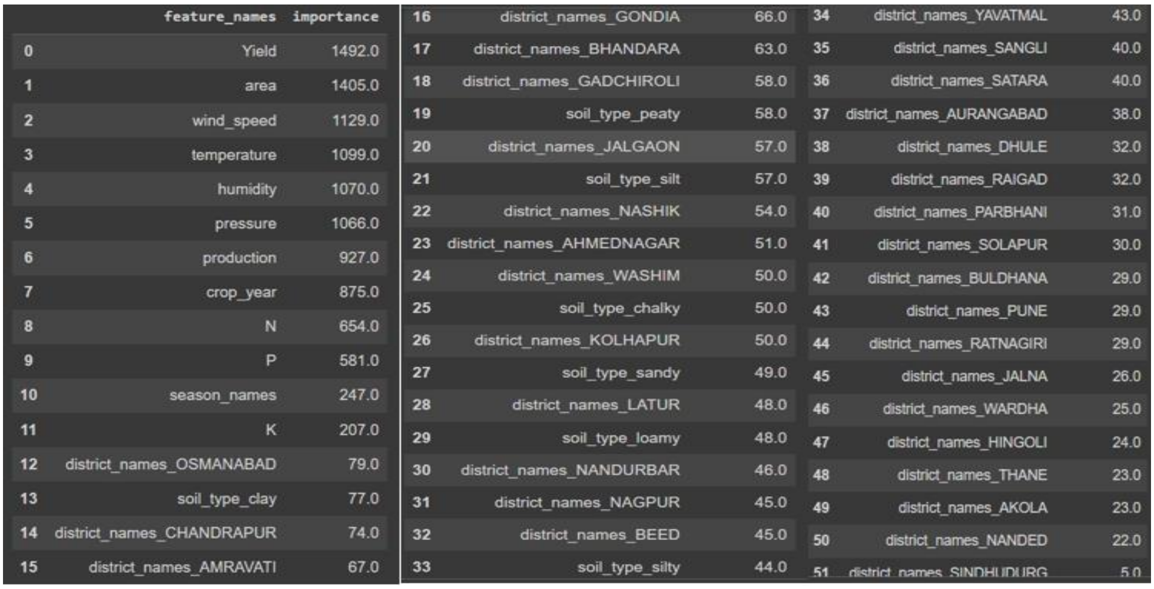
Table 1: Model performance table before feature reduction.

An attempt was made to improve accuracy by selecting more important features from the XG Boost classifier and training the model with a smaller feature set than the previous model. The resulting metrics, however, did not show a significant difference in performance as shown in [Table 2]. After careful consideration, it was decided to build the webpage for this project using the initial model, which had been trained with the entire feature set. This decision was made because the initial model, despite having more features, performed similarly to the reduced feature set model, and thus provides a more comprehensive and reliable prediction for the task at hand as seen from [Table 1].



```
booster=XB.get_booster()
importance = booster.get_score(importance_type='weight')
feature_names=booster.feature_names
df_feature_imp = pd.DataFrame({'feature_names': list(importance.keys()), 'importance': list(importance.values())})
df_feature_imp=df_feature_imp.sort_values(by='importance',ascending=False)
df_feature_imp=df_feature_imp.reset_index(drop=True)
df_feature_imp
```

Fig 18: Code snippet for XGBoost feature importance



	feature_names	importance
0	Yield	1492.0
1	area	1405.0
2	wind_speed	1129.0
3	temperature	1099.0
4	humidity	1070.0
5	pressure	1066.0
6	production	927.0
7	crop_year	875.0
8	N	654.0
9	P	581.0
10	season_names	247.0
11	K	207.0
12	district_names_OSMANABAD	79.0
13	soil_type_clay	77.0
14	district_names_CHANDRAPUR	74.0
15	district_names_AMRAVATI	67.0
16	district_names_GONDIA	66.0
17	district_names_BHANDARA	63.0
18	district_names_GADCHIROLI	58.0
19	soil_type_peaty	58.0
20	district_names_JALGAON	57.0
21	soil_type_silt	57.0
22	district_names_NASHIK	54.0
23	district_names_AHMEDNAGAR	51.0
24	district_names_WASHIM	50.0
25	soil_type_chalky	50.0
26	district_names_KOLHAPUR	50.0
27	soil_type_sandy	49.0
28	district_names_LATUR	48.0
29	soil_type_loamy	48.0
30	district_names_NANDURBAR	46.0
31	district_names_NAGPUR	45.0
32	district_names_BEED	45.0
33	soil_type_silty	44.0
34	district_names_YAVATMAL	43.0
35	district_names_SANGLI	40.0
36	district_names_SATARA	40.0
37	district_names_AURANGABAD	38.0
38	district_names_DHULE	32.0
39	district_names_RAIGAD	32.0
40	district_names_PARBHANI	31.0
41	district_names_SOLAPUR	30.0
42	district_names_BULDHANA	29.0
43	district_names_PUNE	29.0
44	district_names_RATNAGIRI	29.0
45	district_names_JALNA	26.0
46	district_names_WARDHA	25.0
47	district_names_HINGOLI	24.0
48	district_names_THANE	23.0
49	district_names_AKOLA	23.0
50	district_names_NANDED	22.0
51	district_names_SINDHUDURG	5.0

Fig 17: feature importance from XGBoost classifier

Algorithm	**Accuracy**	**Precision**	**Recall**	**AUC**
Decision Tree	0.60	0.61	0.60	0.78
Random Forest	0.61	0.63	0.61	0.92
XG Boost	0.67	0.68	0.67	0.94
KNN	0.47	0.49	0.47	0.84
Naïve Bayes	0.33	0.40	0.33	0.71

Table 2: Model performance table after feature reduction

CHAPTER-8

WEB PAGE

Creating a user-friendly website using Streamlit can greatly enhance the accessibility and usability of the crop recommendation system, making it easier for farmers to access and make informed decisions about which crop to harvest. Streamlit provides a powerful framework for building interactive and intuitive web applications with minimal effort.

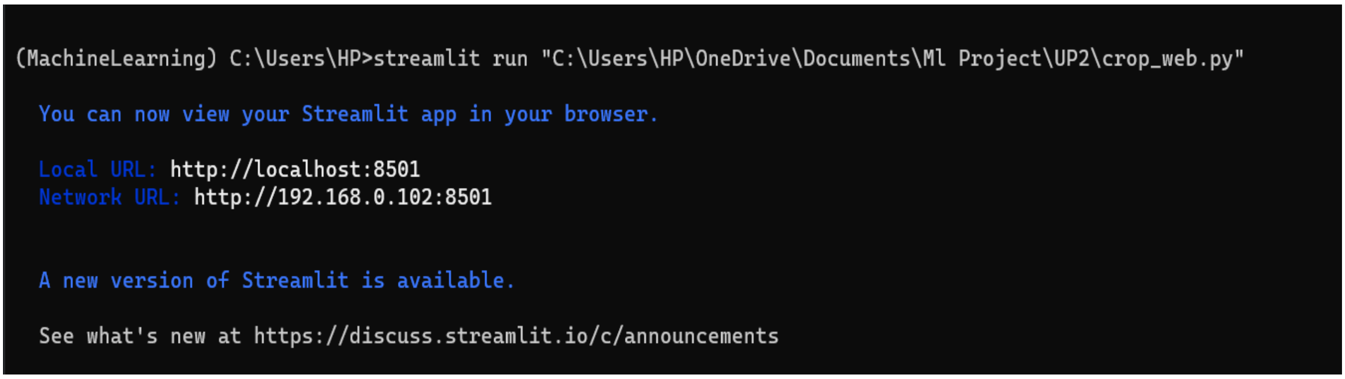
By leveraging Streamlit's capabilities, we can design a visually appealing and intuitive interface that allows farmers to input their specific requirements and preferences, such as soil type, climate conditions, and desired yield. The website can then utilize the trained crop recommendation model to generate personalized recommendations based on these inputs.

The user-friendly website can incorporate various interactive features, such as dropdown menus, sliders, and checkboxes, to allow farmers to easily customize their preferences and explore different scenarios. Additionally, informative visualizations, such as charts and maps, can be incorporated to present the recommended crops and their suitability based on the provided criteria.

To further enhance usability, the website can include informative tooltips, contextual help, and clear instructions to guide farmers through the decision-making process. It should also provide concise explanations of the underlying algorithms and data sources to build trust and credibility with the users.

Moreover, the website can be designed to be responsive and compatible with different devices, including desktops, tablets, and mobile phones, ensuring accessibility for farmers in diverse settings. It should prioritize a clean and intuitive layout, with well-organized sections and an easy-to-navigate structure.

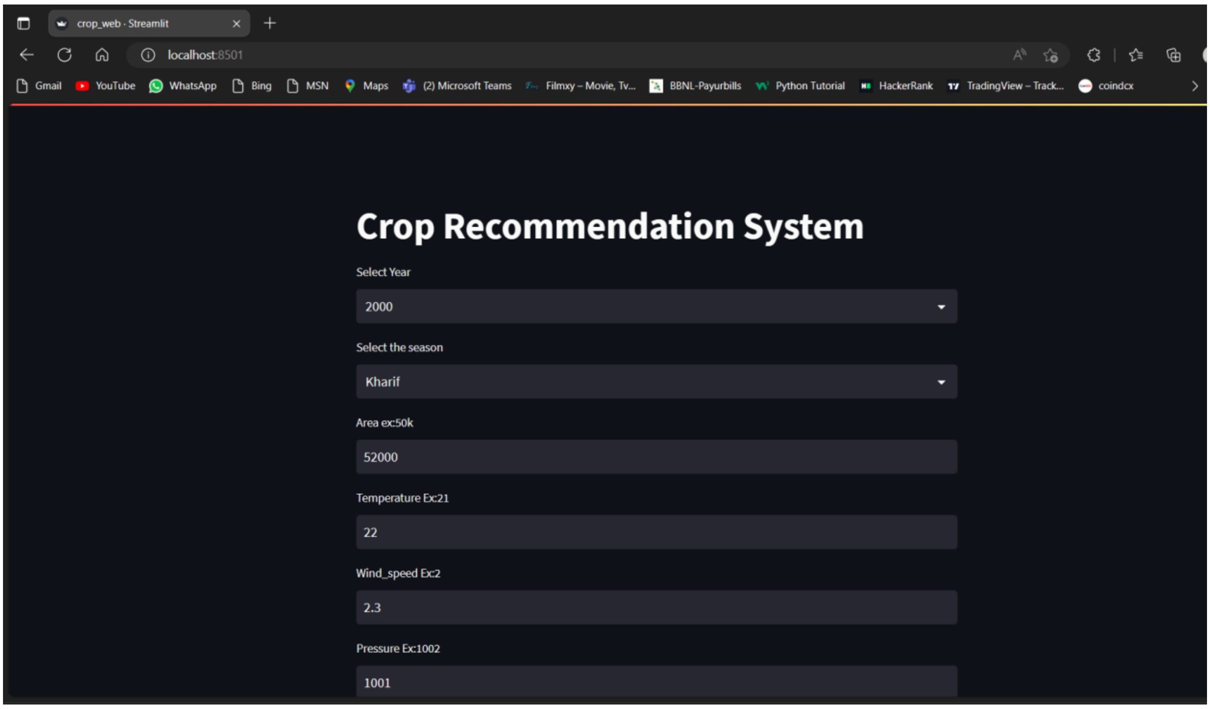
Overall, the creation of a user-friendly website using Streamlit can bridge the gap between complex crop recommendation models and practical decision-making for farmers. By simplifying the user experience and presenting the recommendations in a clear and accessible manner, the website empowers farmers to make informed choices about crop selection, ultimately leading to improved agricultural productivity and sustainability.



```
(MachineLearning) C:\Users\HP>streamlit run "C:\Users\HP\OneDrive\Documents\Ml Project\UP2\crop_web.py"

  You can now view your Streamlit app in your browser.

  Local URL: http://localhost:8501
  Network URL: http://192.168.0.102:8501


  A new version of Streamlit is available.

  See what's new at https://discuss.streamlit.io/c/announcements
```

Fig 20: Command to run Streamlit .python file

The fig[21] is a sample picture on how the webpage looks like .



Crop Recommendation System

Select Year

2000

Select the season

Kharif

Area ex:50k

52000

Temperature Ex:21

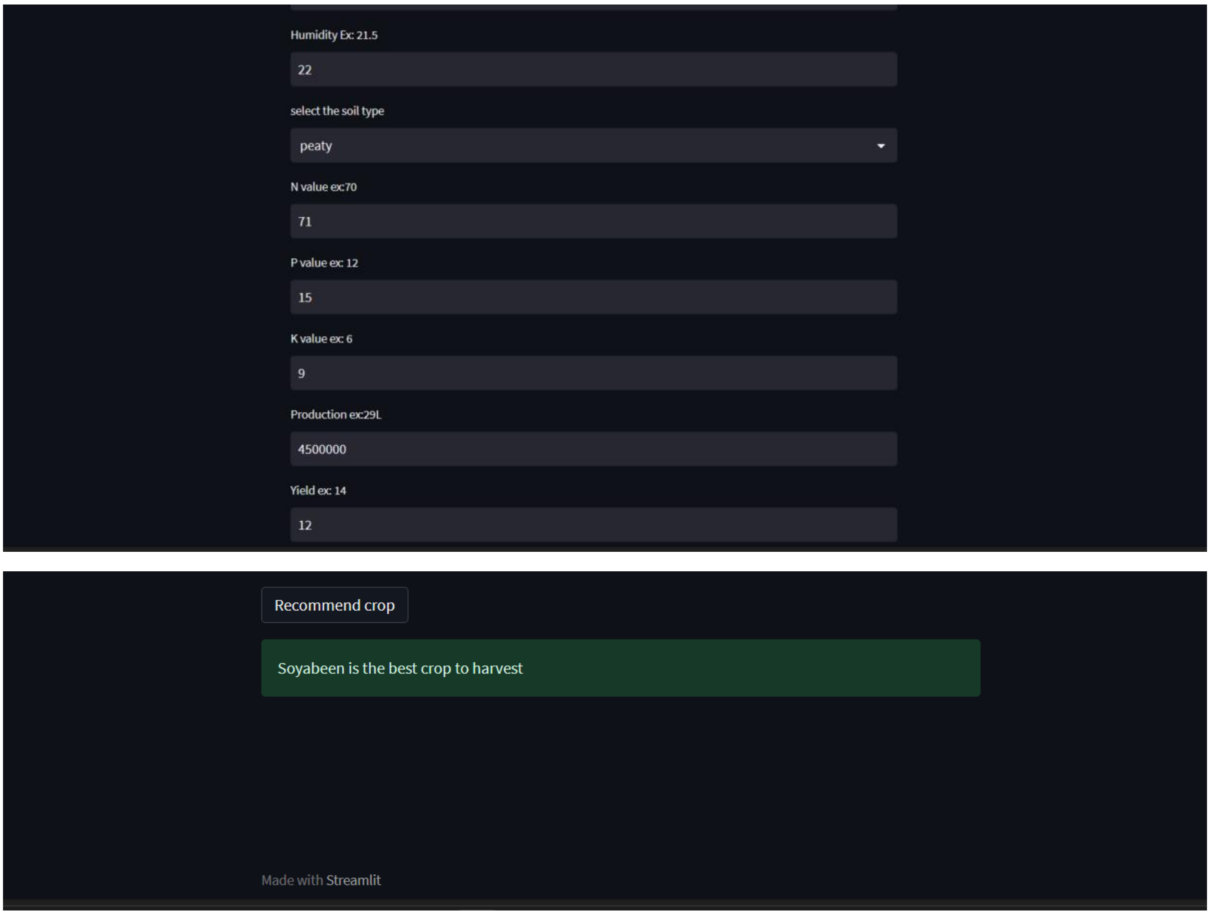
22

Wind_speed Ex:2

2.3

Pressure Ex:1002

1001



Humidity Ex: 21.5

22

select the soil type

peaty

N value ex:70

71

P value ex: 12

15

K value ex: 6

9

Production ex:29L

4500000

Yield ex: 14

12

Soyabeen is the best crop to harvest

Fig 21: Webpage screenshot.

CHAPTER-9

CONCLUSION

Finally, our crop recommendation research has shown how well machine learning algorithms work for advising farmers on which crops to grow. After implementing and comparing several algorithms, including Decision Trees, Random Forest, Naive Bayes, XGBoost, and KNN, we found that the XGBoost approach performs better with an accuracy of 70% and an AUC of 0.95. The outcomes are good, yet there is certainly opportunity for development. Additionally, the model can be adjusted for particular regions or crop types using domain-specific knowledge and professional judgement. The initiative would also benefit from the creation of a user-friendly crop suggestion webpage that farmers can simply access and use. Farmers may find this website beneficial in making knowledgeable crop selection selections thereby improving agricultural practices and crop yields. To summarize, our crop recommendation project has laid the groundwork for the use of machine learning in agriculture and has the potential to have a significant impact on the farming community. The findings and future work outlined above serve as a road map for additional research and development in this field, with the goal of providing farmers with reliable and effective crop recommendation solutions.

REFERENCES

[1] Z. Doshi, S. Nadkarni, R. Agrawal and N. Shah,"Agro Consultant: Intelligent Crop Recommendation System Using Machine Learning Algorithms," 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 2018, pp. 1-6, doi: 10.1109/ICCUBEA.2018.8697349

[2] S.Pudumalar, E. Ramanujam, R. H. Rajashree, C. Kavya, T. Kiruthika and J. Nisha, "Crop recommendation system for precision agriculture," 2016 Eighth International Conference on Advanced Computing (ICoAC), Chennai, India, 2017, pp. 32-36, doi: 10.1109/ICoAC.2017.7951740

[3] N. H. Kulkarni, G. N. Srinivasan, B. M. Sagar and N. K. Cauvery, "Improving Crop Productivity Through A Crop Recommendation System Using Ensembling Technique," 2018 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS), Bengaluru, India, 2018, pp. 114-119, doi: 10.1109/CSITSS.2018.8768790

[4] Anakha Venugopal, Aparna S, Jinsu Mani, Rima Mathew, Vinu Williams, 2021, Crop Yield Prediction using Machine Learning Algorithms, International Journal Of Engineering Research & Technology (Ijert) Ncreis – 2021 (Volume 09 – Issue 13).

[5] A Hybrid Approach For Crop Yield Prediction Using Machine Learning And Deep Learning Algorithms Citation Sonal Agarwal and Sandhya Tarar 2021 J. Phys.: Conf. Ser. 1714 012012 DOI 10.1088/1742-6596/1714/1/012012

[6] Sangeeta, Shruthi G, Design And Implementation Of Crop Yield Prediction Model In Agriculture International Journal Of Scientific & Technology Research Volume 8, Issue 01, January 2020.

[7] "Crop Recommendation System using Machine Learning" Dhruvi Gosai, Chintal Raval, Rikin Nayak, Hardik Jayswal, Axat Patel. International Journal of Scientific Research in Computer Science, Engineering and Information Technology

[8] N. N. Thilakarathne, M. S. A. Bakar, P. E. Abas, and H. Yassin, “A Cloud Enabled Crop Recommendation Platform for Machine Learning-Driven Precision Farming,” Sensors, vol. 22, no. 16, p. 6299, Aug. 2022,

[9] Crop Recommendation System To Maximize Crop Yield Using Deep Neural Network Vol 12,Issue 11, Nov /2021 Issn No:0377-9254

[10] Dighe, Deepti, Harsh H. Joshi, Aishwarya Katkar, Snehal S. Patil and Shrikant Kokate. “Survey of Crop Recommendation Systems.” (2018).

[11] R. Parvathi “Improvement of Crop Production Using Recommender System by Weather Forecasts Bangaru Kamatchi,”

[12] D Ramesh , B Vishnu Vardhan, “Data Mining Techniques and Applications to Agricultural Yield Data” International Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue 9, September 2013

[13] N. N. Jambhulkar Modeling of Rice Production in West Bengal International Journal of Scientific Research, Vol: 2, Issue: 7 July 2013

[14] Li Hong-ying, Hou Yan-lin, Zhou Yong-juan, Zhao Hui-ming, Crop Yield Forecasted Model Based on Time Series Techniques, Journal of Northeast Agricultural University (English edition),Volume 19, Issue 1,2012,Pages 73-77,ISSN 1006-8104,https://doi.org/10.1016/S1006-8104(12)60042-7

[15] Masood, M. A., Raza, I. ., & Abid, S. . (2019). Forecasting Wheat Production Using Time Series Models in Pakistan. Asian Journal of Agriculture and Rural Development, 8(2), 172 177.

[16] Kingsy Grace, K. Induja and M. Lincy “Enrichment of Crop Yield Prophecy Using Machine Learning AlgorithmsR”

[17] Thomas van Klompenburg, Ayalew Kassahun, Cagatay Catal, Crop yield prediction using machine learning: A systematic literature review, Computers and Electronics in Agriculture, Volume 177,2020,105709,ISSN 0168-1699

[18] Nabila Chergui and Mohand Tahar Kechadi “Data analytics for crop management: a big data view”

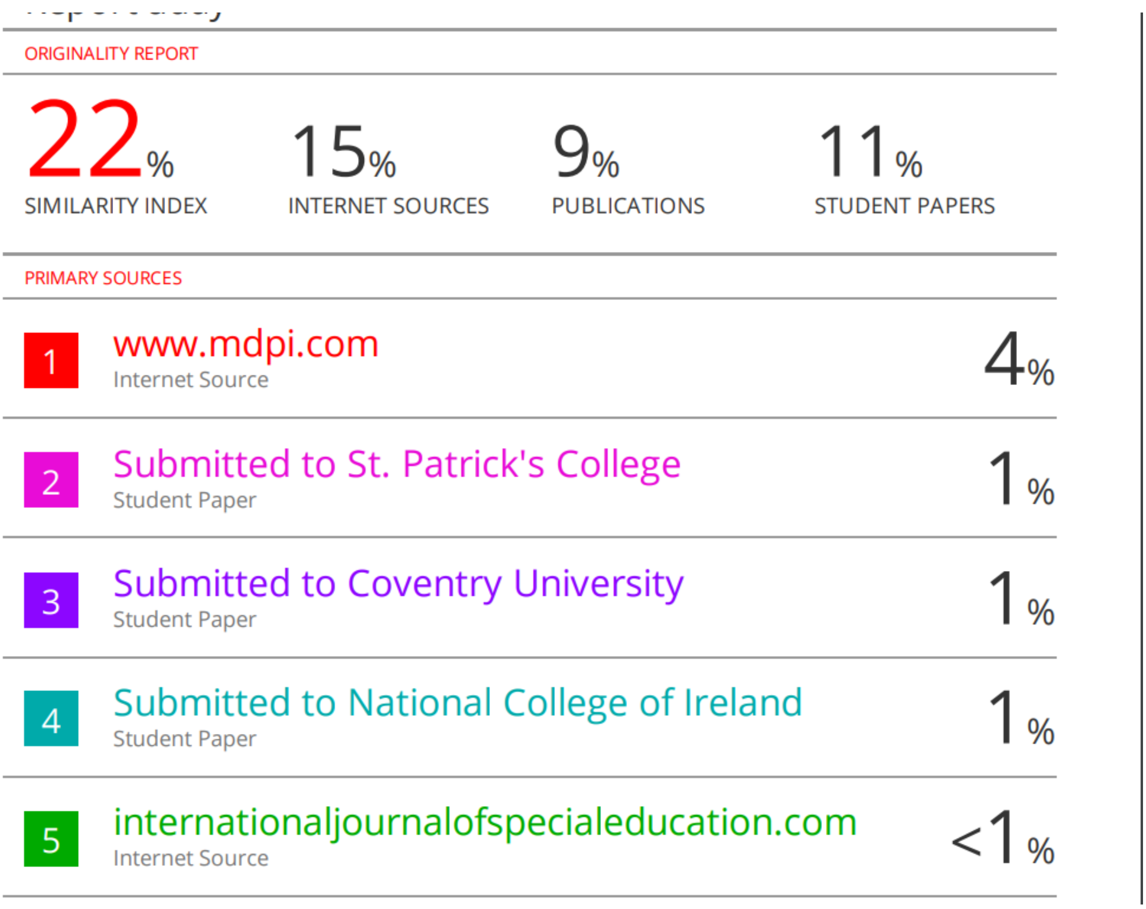
[19] "Crop Yield Prediction In Agriculture Using Data Mining Predictive Analytic Techniques", Ijrar - International Journal Of Research And Analytical Reviews (Ijrar), E-Issn 2348-1269, P- Issn 2349-5138, Volume.5, Issue 4, Page No Pp.783-787, December 2018,

[20] Champaneri, Mayank & Chachpara, Darpan & Chandvidkar, Chaitanya & Rathod, Mansing. (2020). CROP YIELD PREDICTION USING MACHINE LEARNING. International Journal of Science and Research (IJSR). 9. 2.

[21] G. Vishwa, J. Venkatesh, Dr. C. Geetha, “Crop Variety Selection Method using Machine Learning”

[22] N.L. Chourasiya, P. Modi , N. Shaikh3 , D. Khandagale, S. Pawar, “Crop Prediction using Machine Learning” IOSR Journal of Engineering (IOSR JEN) ISSN (e): 2250-3021, ISSN (p): 2278-8719 PP 06-10

PLAGIARISM REPORT



ORIGINALITY REPORT

22%	15%	9%	11%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.mdpi.com Internet Source	4%
2	Submitted to St. Patrick's College Student Paper	1%
3	Submitted to Coventry University Student Paper	1%
4	Submitted to National College of Ireland Student Paper	1%
5	internationaljournalofspecialeducation.com Internet Source	<1%