

CROP YIELD PREDICTION USING MACHINE LEARNING TECHNIQUES

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Technology degree in
Information Technology

by

JEEVITHA D (38120033)

KAVIYA S (38120041)



DEPARTMENT OF INFORMATION TECHNOLOGY

SCHOOL OF COMPUTING

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

MAY - 2022



DEPARTMENT OF INFORMATION TECHNOLOGY

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of Jeevitha D(38120033) Kaviya S(38120041) who carried out the project entitled "CROP YIELD PREDICTION USING MACHINE LEARNING TECHNIQUES" under my supervision from November 2021 to May 2022.

Internal Guide

Dr. REVATHY M.E., Ph.D.

Head of the Department

Dr. R. SUBHASHINI M.E., Ph.D.

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I, Jeevitha D (38120033) Kaviya S (38120041) hereby declare that the Project Report entitled "Crop Yield Prediction Using Machine Learning Techniques" done by me under the guidance of Dr. Revathy M.E, Ph.D. is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology degree in Information Technology.

DATE:

PLACE:

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.R.Subhashini M.E., Ph.D.** Head of the Department of Information Technology 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.Revathy 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 Information Technology who were helpful in many ways for the completion of the project.

I wish to express my thanks to the Project Panel members for their valuable feedbacks during the Project Reviews which were useful in many ways for the completion of the project.

ABSTRACT

This work presents a system, in the form of a web application. The business logic in Python uses Hybrid ensemble Learning techniques in order to predict the most profitable crop in the forecasted weather and soil conditions at a specified location. The proposed system will integrate the data obtained from soil, crop repository, weather department and by applying machine learning algorithms and a prediction of most suitable crops according to current environmental conditions is made. This provides young farmers with a variety of options of crops that can be cultivated. Machine learning is concerned with the development of computer programmes that can access data and learn on their own. Crop yield prediction is one of precision agriculture's most difficult issues, and several models have been proposed and validated so far. Since crop yield is affected by a variety of factors such as climate, weather, soil, fertilizer usage, and seed variety, this issue necessitates the use of multiple datasets. This suggests that predicting crop yields is not a simple task; rather, it entails a series of complex steps. Farmers are the core of the agricultural system. Agriculture is an important part of a country's development because, as everyone knows, a country's gross domestic product must be equitable. Farmers are critical to the agricultural system's success because crops must be planted and processed for it to work. A prototype for a real-time crop recommendation algorithm in Python using Hybrid Machine Learning and Data Analytics was created to assist these farmers. Machine learning is a subset of artificial intelligence (AI) that enables computers to learn and evolve without being explicitly programmed..

CONTENTS

CHAPTER NUMBER	CHAPTERNAME	PAGE NUMBER
	ABSTRACT	V
	LIST OF ABBREVIATIONS	ix
	LIST OF FIGURES	x
	LIST OF TABLES	xi
1	INTRODUCTION	01
	1.1 Motivation	01
	1.2 Problemstatement	01
	1.3 Outline of theproject	02
	1.4 Datacollection	02
2	AIM AND SCOPE	03
	2.1 Aim	03
	2.2 Literature survey	03
	2.3 Scope	05
3	SYSTEM DESIGN	06
	3.1 Existing System	06
	3.2 RuleSet of Existing System	06
	3.3 Main Disadvantage of Existing System	08
	3.4 Proposed System	08

3.5	Exploratory DataAnalysis	09
3.5.1	<i>LinearlySeparable</i>	10
3.5.2	<i>Correlation</i>	10
3.6	SoftwareUsed	11
3.6.1	<i>Programming LanguageUsed</i>	11
3.6.2	<i>IDEUsed</i>	16
4	RESULT ANALYSIS AND DISCUSSION	17
4.1	Correlation inDataset	17
4.2	Algorithms	18
4.2.1	<i>Parameters involved inaccuracy</i>	19
4.3	Comparison Between DifferentAlgorithms	20
4.4	Comparison andConclusion	31
4.5	HybridModel	32
4.6	Hybrid EnsembleModel	32
4.7	SystemRequirements	34
4.8	Front-end	34
4.9	API	35
4.10	OutputScreenshots	35

5	CONCLUSION AND FUTURE WORK	37
	5.1 Conclusion	37
	5.2 FutureEnhancement	37
	REFERENCES	38

LIST OF ABBREVIATIONS

ABBREVIATION	EXPANSION
NPK	Nitrogen, Phosphorous, Potassium
EDA	Exploratory Data Analysis
RF	Random Forest
SVM	Support Vector Machine

LIST OF FIGURES

FIGURE NUMBER	FIGURE NAME	PAGE NUMBER
3.1	Existing System	06
3.2	Proposed System	08
3.3	Diagram for EDA	09
4.1	Data Summarization	17
4.2	Heatmap for Correlation	18
4.3	Logistic Regression Model	20
4.4	Accuracy for Logistic Regression Model	21
4.5	Decision tree model	23
4.6	Decision tree accuracy	24
4.7	Random Forest tree model	25
4.8	Accuracy for Random Forest tree model	26
4.9	SVM Model	27
4.10	SVM Model Accuracy	28
4.11	Naïve Bayes Accuracy	30
4.12	Accuracy Comparison	31
4.13	Web application Model	32
4.14	Output Screenshot – 1	33
4.15	Output Screenshot- 2	34

LIST OF TABLES

TABLE NUMBER	TABLE NAME	PAGE NUMBER
3.1	Rule baseof ExistingSystem	07
4.1	ClassPredictionTable	19
4.2	AccuracyComparisonTable	31

CHAPTER 1

INTRODUCTION

1.1 MOTIVATION:

The agriculture system is supported by farmers. Agriculture, as is well known, is an integral part of a country's development. Agriculture is very important in India's economy and job market. One of the most common problems faced by Indian farmers is that they do not choose the appropriate crop for their soil. One of the most popular issues that Indian farmers face is failing to select the right crop for their soil.

1.2 PROBLEM STATEMENT:

Seeding the wrong seed unintentionally would be a waste of time and manpower. Farmers need to improve productivity so that they can get more money from the same piece of land with less labour. Precision agriculture makes it possible. Precision farming, as the name suggests, refers to the application of specific and appropriate total of comment such as pee, fertilizers, soil, and so on to the crop at the appropriate time in order to increase its productivity and yields. Precision agriculture systems are not all created equal. However, in agriculture, it is critical that the recommendations made are accurate and precise, as errors may result in heavy material and capital losses. Since crop yield is affected by a variety of factors such as climate, weather, soil, fertilizer usage, NPK, and seed variety, this issue necessitates the use of multiple datasets. This suggests that predicting crop yield is not a simple task; rather, it entails a series of complex steps. The NPK are nothing but the primary nutrients (Nitrogen, Phosphorus, Potassium) of the soil or plants. It decides the growth of the plant on the field. Lack of monitoring this NPK content in the soil may lead to the massive loss during harvesting. There are several factors in the environment that also decide the growth and type of plants on the soil or land.

1.3 OUTLINE OF THE PROJECT:

Many studies are being conducted in order to develop an accurate and effective crop prediction model. Ensembling is one of the techniques used in these types of studies. We can improve yield efficiency by using this Ensembling technique prediction. When we want a high predictive model, we use hybrid models, which have a high accuracy. Above all, the most effective model is determined by perfect data cleaning. We used Python for both the backend and the frontend in this project. For the front-end, there's HTML, CSS, and JS, as well as the Flask module for linking the frontend and backend. We've also used hybrid ensemble classifiers to improve accuracy.

1.4 DATA COLLECTION:

IMD and dedicated data collection are provided the dataset for this project. Data on soil nutrition is gathered from historical soil testing data, which provides general crop information. The model takes into account major crops such as wheat, rice, bajra, maize, and jowar, as well as minor crops such as pulses, gramme, jute, cotton, groundnut, barley, ragi, mustard, sugarcane, sesame, and sunflower.

The important features of the dataset are of soil type Sandy, Silt, Clay and Loamy Soil, pH value of the Soil is a measure of the acidity and alkalinity in soils, NPK content of the soil which is nitrogen, phosphorus and potassium are the three nutrients used by plants, Permeability of the soil is the property to transmit water and air, the water content in the soil, average rainfall, temperature and previously harvested crop. All these parameters are really important to determine the best crop for current weather, market demand and availability of market infrastructure, expected profit and risk.

CHAPTER 2

AIM AND SCOPE

2.1 AIM:

The aim of our project is to design a web application that will predict appropriate Crop yield based on the soil and environment conditions by using hybrid ensemble technique. This web based application which is been implemented in this project aims that it can be easily available and can be used any kind of soil type in across india.

2.2 LITERATURE SURVEY:

[1] Prabakaran, G et al., has presented the “Fuzzy decision support system for improving the crop productivity and efficient use of fertilizers” states that the process of reducing the fertilizer consumption and improving the crop productivity using the fuzzy logic systems. This system comprises two parts; land report based expert knowledge to stimulate the yield potential through appropriate organic lacking minerals in soil. This research work is to improve the productivity with minimum consumption of fertilizer. This study has been carried out to assess the fertilizer consumption in both the ACZ (Agro Climatic Zone) with an exhaustive daily field measurements and lab analysis of a duration of three years to determine exact fertilizer needs for every individual land. The data was analysed in MATLAB to establish feasibility rules for decision support systems for the crop to get the targeted output. A major drawback of Fuzzy Logic control systems is that they are completely dependent on human knowledge and expertise. You have to regularly update the rules of a Fuzzy Logic control system. These systems cannot recognize the accurate yield prediction. And also it is quite difficult to write the logic for the whole Fuzzy expert system.

[2] A. Toreti et al., has presented "Using reanalysis in crop monitoring and forecasting systems" States that Weather observations are essential for crop monitoring and forecasting but they are not always available and in some cases they have limited spatial representativeness. As a result, reanalyses are an additional source of data that should be investigated. We evaluate the viability of reanalysis-based crop tracking and forecasting in this study by using the European Commission's Joint Research Centre's scheme, its gridded regular meteorological measurements, the biased-corrected reanalysis AgMERRA, and the ERA-Interim reanalysis, as well as the biased-corrected reanalysis AgMERRA. And, for the period 1980–2010, concentrate on Europe and two crops, wheat and maize, under potential and water-limited conditions. They did their analysis using data from meteorological stations all over Africa. It may not be enough for other nations

[3] Silvio Gualdi et al., has discussed "Using daily data from seasonal forecasts in dynamic crop models for yield prediction: A case study for rice in Nepal's Terai", The results from this study indicate that unlike the typical ensemble averaging approach commonly used in seasonal climate forecasting, averaging the simulated yield using individual member does not guarantee a better yield prediction. Further analysis should be made, including alternative downscaling methods as well as a similar analysis for an area where quality meteorological and agricultural data are available and where these seasonal forecasts exhibit better skill.

[4] Kuanr et al., has developed "Crop Recommender System for the Farmers using Mamdani Fuzzy Inference Model", The aim of this research is to create a collective recommenders scheme for farmers, with the goal of suggesting a crop that is appropriate for the farmer's position based on weather conditions in previous months. The Mamdani Fuzzy Inference model is used to apply the proposed framework. The findings show that it offers previous crop ideas prior to seed sowing.

[5] Nishit jain et al., has developed “Crop Selection Method Based on Various Environmental Factors Using Machine Learning”,The Proposed method is used to determinethemaximumyieldbysumminguptheanalysisofalltheaffectingparameters usingafeedbackpropagationalgorithm.

[6] Kirtan Jha et al., has reviewed the “A comprehensive review on automationin agricultureusingartificialintelligence”,Thispapertalksaboutuseofdifferentautomation practiceslikeIOT,MachinelearningandArtificialIntelligenceinagriculturesystems.

2.3 SCOPE:

- To predict the appropriate crop yield based on the soil and environment factor.
- To analyze the data to get the inference.
- To analyze and compare the cleaned dataset with various machine learning algorithm.
- To frame the hybrid ensemble model along with dataset.
- To design a web UI end for user interaction with portal.

CHAPTER 3

SYSTEM DESIGN

3.1 EXISTINGSYSTEM:

The Existing system employs a fuzzy model and expert experience to forecast the best crop for a given piece of farmland. Expert knowledge of land, weather, air, and agronomists is represented by fuzzy sets. Expert expertise aids in the creation of the final rules. The fuzzy method, on the other hand, uses multiple state variables to obtain the desired result.

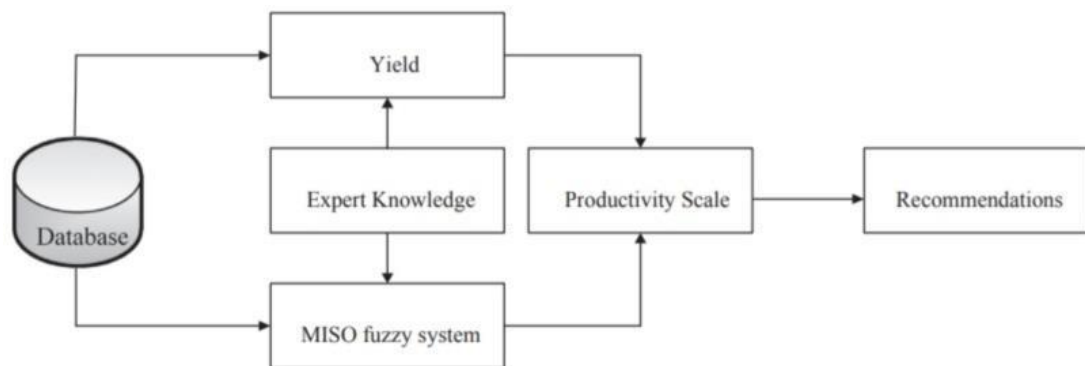


Fig.3.1: EXISTING SYSTEM

3.2 RULE SET OF EXISTINGSYSTEM:

Frequently used in the control mechanisms of many systems. Three steps had been taken to complete the fuzzy logic operation. 1. Perplexity 2. Examining the rules 3. Defuzzification. There are three types of graphical inference techniques (fuzzy controllers) that can be used to combine aggregation rules. They are 1. Mam-dani Systems 2. Sugeno Systems 3. Trukamoto Systems.

The typical multi input and single output fuzzy rule (As shown in Table No 3.1) follow as

If Input1 is A1 AND Input2 is A2, Then Output is Y1

If Input1 is B1 AND Input2 is B2, Then Output is Y2

Where A1, A2, B1, B2, Y1, Y2 are fuzzy sets and Input1, Input2 Output are fuzzy system variables. The degree to which the fuzzy operation is carried out is determined by the degree of truth in the preceding proposition.

Table No 3.1: Rule base of existing system

Rule.No.1	If Primary Nutrient Output is Poor AND Secondary Nutrient Output is Poor AND Micro Nutrient Output is Poor AND Climatic Parameter Output is Poor AND Seasonal factor Output is Poor AND Soil Properties is Satisfied AND Water Properties is Satisfied AND Pesticide Incidence is Very High Level THEN is Poor Productivity
Rule.No.2	If Primary Nutrient Output is Poor AND Secondary Nutrient Output is Moderate AND Micro Nutrient Output is Poor AND Climatic Parameter Output is Poor AND Seasonal factor Output is Poor AND Soil Properties is Satisfied AND Water Properties is Satisfied AND Pesticide Incidence is Very High Level THEN is Poor Productivity
Rule.No.3	If Primary Nutrient Output is Poor AND Secondary Nutrient Output is Good AND Micro Nutrient Output is Poor AND Climatic Parameter Output is Poor AND Seasonal factor Output is Poor AND Soil Properties is Satisfied AND Water Properties is Satisfied AND Pesticide Incidence is Very High Level THEN is Poor Productivity
	"
	"
	"
Rule.No.2500	If Primary Nutrient Output is Poor AND Secondary Nutrient Output is Excellent AND Micro Nutrient Output is Poor AND Climatic Parameter Output is Excellent AND Seasonal factor Output is Moderate AND Soil Properties is Satisfied AND Water Properties is Dissatisfied AND Pesticide Incidence is Very High Level THEN is Poor Productivity
Rule.No.16000	If Primary Nutrient Output is Excellent AND Secondary Nutrient Output is Excellent AND Micro Nutrient Output is Excellent AND Climatic Parameter Output is Moderate AND Seasonal factor is Good AND Soil Properties is Dissatisfied AND Water Properties is Dissatisfied AND Pesticide Incidence is Medium Level THEN is Good Productivity

3.3 MAIN DISADVANTAGES OF EXISTING SYSTEM:

Extensive hardware testing is needed for the validation and verification of a fuzzy knowledge-based system. It's difficult to come up with precise, fuzzy rules and membership functions.

Some people confuse fuzzy logic with probability theory and the terminology. A Fuzzy Logic control system's rules must be modified on a regular basis. These systems are unable to recognize a precise yield estimate. Furthermore, writing the logic for the whole Fuzzy expert scheme is very difficult.

3.4 PROPOSED SYSTEM:

In this proposed method we are going to use Hybrid ensemble technique to predict the better crop yields. The system uses the only NPK (Primary Nutrients) which is used only for the intimation of the soil fertility (As shown in fig:3.2). And the major factor pH, Temperature, Average rainfall, Humidity involves affecting the better growth of crop. This will suggest which type of crop will be suitable for that soil.

Feature Variable: N, P, K, pH, Temp, rainfall, Humidity

Target Variable: Crop label

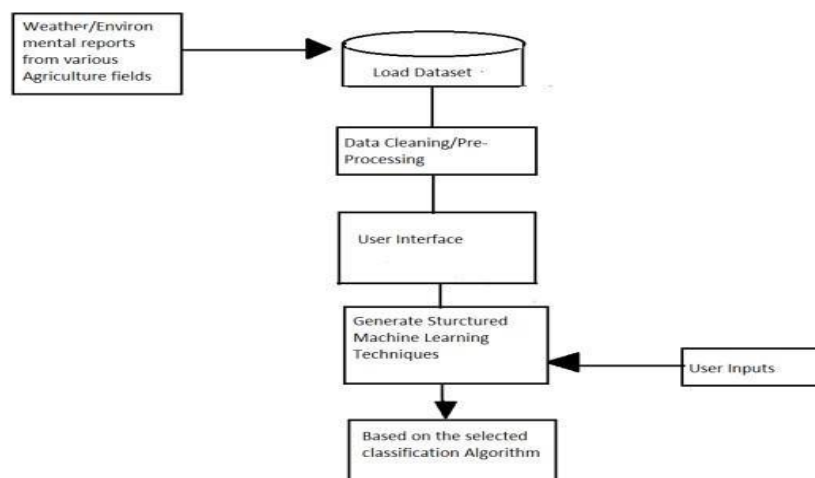


Fig: 3.2: Proposed system

In this project, I have done comparative analysis between most famous Classification algorithms to predict the best crop for the soil. The algorithm used for comparative analysis is Logistic Regression, Decision Tree, Random Forest Classification, SVM, Naive Bayes Classification.

3.5 EXPLORATORY DATA ANALYSIS:

Advanced mathematical methods are also considered to be at the heart of machine learning. However, another important aspect of any data science project is exploratory data analysis, which is often overlooked or undervalued (EDA). It's a tried-and-true method for quickly establishing a link with new data that's rarely used.

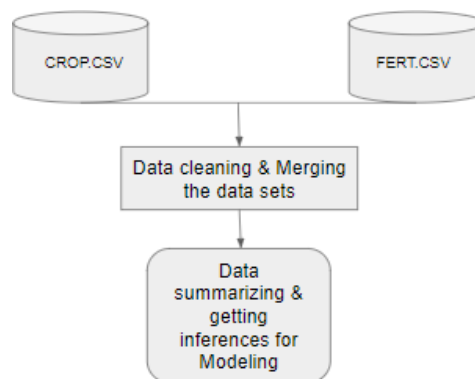


Fig: 3.3: Diagram for EDA

It's often easier to compare the effects of multiple exploratory techniques on each data collection. This phase aims to understand the dataset, recognize missing values and outliers if any, and get a sense of the story it tells using visual and quantitative methods. It indicates the project's next logical steps, concerns, or study areas..

Steps in EDA (As shown in fig:3.3) are Identification of variables and data types, Correlation Analysis, Simple Metrics Analysis, Variable Transformations, Removing Missing value

3.5.1 LINEARLY SEPARABLE:

If the data is linearly separable, let's say this translates to saying we can solve a two class classification problem perfectly, and the class label $y_i \in \{-1, 1\}$.

Once we find the plane separating them, each point will be of class -1 or 1. The problem now reduces to find 'any' plane which separates the data perfectly. Let's assume we solve the following feasibility problem, which is similar to the inequality of the Support Vector Machine (SVM) algorithm.

$$\min w$$

$$\forall i, (w'x_i + b) y_i \geq 1 \quad \text{----- 3.1}$$

Recall the equation of a plane. We say that there exists a plane such that all points with label 1 lie on 1 side of it, and the others lie on the other side. Notice we don't care about our objective (it is a constant), we only care about satisfying the separability constraint— find any plane which can separate our data. If our data is linearly separable, we will find a feasible solution to our problem. The w, b , in this case, parametrizes the hyperplane which separates our data.

This is a simple linear program which can be solved using any commercial solver. If no feasible solution exists for the optimization problem, data is not separable.

3.5.2 CORRELATION:

Correlation is a statistic that measures the degree to which two variables move in relation to each other. If the Feature and target variables have a significant correlation then, it would be fit perfectly for all the linear machine learning models for business solutions.

3.6 SOFTWARE USED:

- Python
- HTML5
- CSS3
- Bootstrap
- JavaScript

3.6.1 Programming Language used:

Python

Python is a high-level programming language for general-purpose programming that is also called an interpreted language. Python is a programming language that was invented by Guido van Rossum and first published in 1991. It has a design philosophy that emphasizes code readability and a syntax that enables programmers to express ideas in fewer lines of code. It has constructs that allow for simple programming at both small and large scales. Python has a dynamic style structure and memory management that is automated. It has a wide and extensive standard library and supports various programming paradigms, including object-oriented, imperative, functional, and procedural.

NUMPY

NumPy is the most important Python package for scientific computing. It's a Python library that includes a multidimensional array object, derived objects (such as masked arrays and matrices), and a variety of routines for performing quick array operations, such as mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation, and more.

At the core of the NumPy package, is the ndarray object. Many operations are performed in compiled code for accuracy, and this encapsulates n-dimensional arrays of homogeneous data types. Between NumPy arrays and regular Python sequences, there are a few main differences:

Unlike Python lists, NumPy arrays have a fixed size when created (which can grow dynamically). When the size of an ndarray is changed, a new array is created and the old one is deleted..

- The elements of a NumPy array must all be of the same data form and thus have the same memory usage. The exception is that arrays of (Python, including NumPy) objects can be used, allowing for arrays of various sizes.
- NumPy arrays make it easier to perform advanced mathematical and other operations on large amounts of data. Such operations are usually performed more quickly and with less code than using Python's built-in sequences.
- NumPy arrays are used by a growing number of scientific and mathematical Python-based packages; although they usually support Python -sequence input, they convert it to NumPy arrays before processing, and they often output NumPy arrays. To put it another way, learning how to use Python's built-in sequence types isn't enough to effectively use most (if not all) of today's scientific/mathematical Python-based software; you still need to know how to use NumPy arrays.

PANDAS

Pandas is a commonly used open source Python package for data science, data mining, and machine learning tasks. It is built on top of Numpy, a package that supports multi-dimensional arrays..

Pandas makes it simple to do many of the time-consuming, repetitive tasks associated with working with data, including:

- Data cleansing
- Data fill

- Data normalization
- Merges and joins
- Data visualization
- Statistical analysis
- Data inspection
- Loading and saving data
- And much more

In reality, you can do anything that makes Pandas the best data analysis and manipulation tool available, according to world-renowned data scientists.

SEABORN

Seaborn is a Python library for creating statistical graphics. It is based on matplotlib and closely integrates with pandas data structures. Seaborn assists you in exploring and comprehending the details. Its plotting functions work with dataframes and arrays containing entire datasets, performing the required semantic mapping and statistical aggregation internally to generate informative plots. Its dataset-oriented, declarative API allows you to concentrate on the meaning of your plots rather than the mechanics of drawing them.

Scikit learn

Scikit-learn offers a consistent Python framework for a variety of supervised and unsupervised learning algorithms.

Popular groups of models provided by scikit-learn include:

- **Clustering**: K-means, for example, can be used to group unlabeled data..
- **Cross Validation**: for estimating the efficiency of supervised models on data that hasn't been seen before.
- **Datasets**: for creating test datasets and datasets with particular properties to investigate model behavior.

- **Dimensionality Reduction:** Principal component analysis, for example, is used to reduce the number of attributes in data for summarization, visualization, and feature selection..
- **Ensemble methods:** for integrating several supervised models' predictions.
- **Feature extraction:** in image and text data for defining attributes.
- **Feature selection:** for determining meaningful attributes from which supervised models can be built
- **Parameter Tuning:** in order to make the most of supervised models.
- **Manifold Learning:** For summarizing and displaying multi-dimensional data that is complex.
- **Supervised Models:** generalized linear models, discriminant analysis, naive bayes, lazy methods, neural networks, support vector machines, and decision trees are only a few examples.

Flask

Flask is a popular Python web framework, meaning it is a third-party Python library used for developing web applications.

Key aspects of Flask development such as:

- Integrating Flask applications with Front-End frameworks
- How templating in Flask works
- API design and working with third-party APIs
- Databases
- Caching
- Deployment

PICKLE

For serializing and de-serializing a Python object structure, the pickle module is used. Pickling an object in Python allows it to be saved on disc. Pickle first "serializes" the object before writing it to file. Pickling is a Python method of converting an object.

HTML5

The standard markup language for documents intended to be viewed in a web browser is Hypertext Markup Language. The technologies such as Cascading Style Sheets and scripting languages like JavaScript can help in creating a web page.

CSS3

CSS (Cascading Style Sheets) is a stylesheet language for describing the presentation of an HTML or XML document (including XML dialects such as SVG, MathML or XHTML). CSS specifies how elements should appear on a computer, in print, in audio, or in other forms of media.

JAVASCRIPT

JavaScript (JS) is a first-class interpreted programming language that is lightweight, or just-in-time compiled. Although it is best known as a scripting language for Web sites, it is also used in a variety of non-browser environments, including Node.js, Apache CouchDB, and Adobe Acrobat. JS (JavaScript) is a high-level, interpreted scripting language that follows the ECMAScript specification. Curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions are all features of JavaScript.

BOOTSTRAP

Bootstrap is a free and open-source CSS platform for front-end web creation that is sensitive and mobile-first. It includes design templates for typography, shapes, buttons, navigation, and other interface elements that are based on CSS and JavaScript.

3.6.2 IDE USED:

Jupyter Notebook

Jupyter Notebook is an open-source web app that lets you build and share documents with live code, equations, visualizations, and narrative text. Data cleansing and transformation, numerical simulation, mathematical modelling, data visualization, machine learning, and many other applications are all possible.

Visual Studio Code - 2019

Visual Studio Code is a source-code editor that supports Java, JavaScript, Go, Node.js, Python, and C++, among other programming languages. It's built on the Electron platform, which is used to create Node.js Web apps that use the Blink layout engine. The same editor component (codenamed "Monaco") that is used in Azure DevOps is used in Visual Studio Code (formerly called Visual Studio Online and Visual Studio Team Services).

Rather than using a project program, it allows users to open one or more files, which can then be stored in workspaces for later use. As a result, it can be used as a language-independent code editor for any language. It supports a variety of programming languages, each with its own set of features. The settings can be used to remove unwanted files and directories from the project tree. Many aspects of Visual Studio Code are not accessible through menus or the user interface, but rather through the command palette.

CHAPTER 4

RESULT ANALYSIS AND DISCUSSION

4.1 CORRELATION IN DATASET:

Exploratory data analysis is a method of evaluating data sets in order to summarize the data's key characteristics as shown in fig:4.1.

In [40]:

df.head()

Out[40]:

	N	P	K	temperature	humidity	ph	rainfall	label
0	66	57	44	20.879744	82.002744	6.502985	202.935536	rice
1	70	60	36	21.770462	80.319644	7.038096	226.655537	rice
2	99	56	37	23.004459	82.320763	7.840207	263.964248	rice
3	100	40	36	26.491096	80.158363	6.980401	242.864034	rice
4	68	54	45	20.130175	81.604873	7.628473	262.717340	rice

In [54]:

df.describe()

Out[54]:

	N	P	K	temperature	humidity	ph	rainfall
count	1700.000000	1700.000000	1700.000000	1700.000000	1700.000000	1700.000000	1700.000000
mean	58.355294	51.496471	52.864706	25.630177	76.718347	6.394963	111.847137
std	38.694480	36.475644	55.394412	5.131419	19.007296	0.610486	55.349764
min	0.000000	5.000000	5.000000	8.825675	18.092240	4.507524	20.211267
25%	23.000000	24.000000	23.000000	22.986149	67.534062	5.947336	71.141395
50%	55.000000	45.000000	36.000000	25.404226	82.364535	6.361523	103.706242
75%	93.000000	66.000000	49.000000	28.129815	91.120006	6.802003	138.568624
max	140.000000	145.000000	205.000000	43.675493	99.981876	7.995849	298.560117

Fig: 4.1: Data Summarization

Feature variables: N, P, K, Temperature, Humidity, pH, Rainfall

Target variables: Label (Crop name)

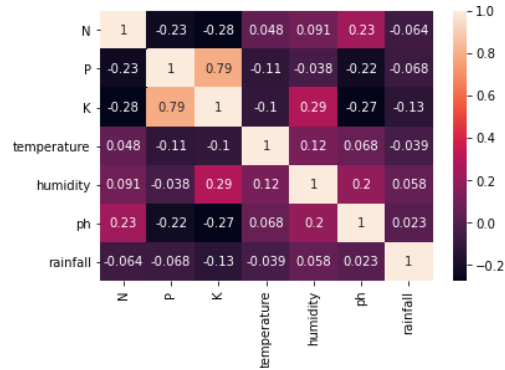


Fig: 4.2: Heatmap for Correlation

From Fig: 4.1: Heatmap for Correlation , Feature and target variables have a significant correlation. Thus, There is a correlation between feature and target variable. And the data set is linearly separable. And also it would be fit perfectly for all the linear machine learning models.

4.2 ALGORITHMS:

CLASSIFICATION VS REGRESSION

Predictive modelling problems involving classification vary from those involving regression.

- The task of classification is to predict a discrete classmark.
- The role of regression is to predict a continuous quantity.

Since we will be using one or more classes. So that it is coming under classification predictive model

4.2.1 Parameters involved in Measuring Accuracy:

Accuracy Metrics

Precision - The ratio of correctly predicted positive observations to total predicted positive observations is known as precision .

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP}) \text{ -----4.1}$$

Table No. 4.1: Class Prediction Table

Actual Class	Predicted class		
		Class = Yes	Class = No
	Class = Yes	True Positive	False Negative
	Class = No	False Positive	True Negative

Recall - The ratio of correctly expected positive observations to all observations in the actual class is called recall - yes.

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN}) \text{ ----- 4.2}$$

F1 score - The weighted average of Precision and Recall is the F1 Score.

$$\text{F1 Score} = 2 * (\text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision}) \text{ ----- 4.3}$$

4.3 COMPARISON BETWEEN DIFFERENTIAL ALGORITHMS:

Logistic Regression

A classification algorithm is logistic regression. It's a method for predicting a binary outcome from a series of independent variables. This is a linear model. In a linear model, we will be having Linear Regression as well as Logistic Regression. Basically, the linear regression was used for a regression problem which means the target variable in the dataset will be a numerical variable. But in the case of logistic regression, which is also a linear model, it is used for linear classification problems. It can be classified into one of two classes. This is true for binary logistic regression, a form of logistic regression. It can be classified into one of two classes. This is true for binary logistic regression, a form of logistic regression. It will segregate the output in the S-shaped graph. So that the prediction will be made based on that graph internally.

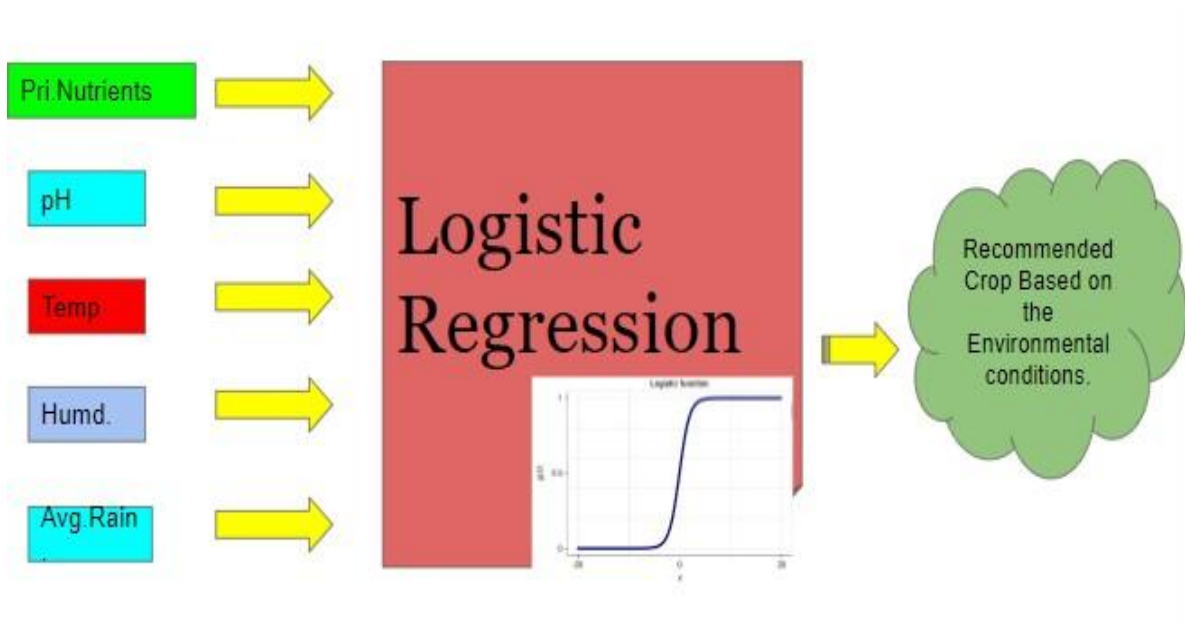


Fig:4.3: Logistic Regression Model

Justification

In our problem we will be having features as Primary nutrients, pH, Temp, Humidity, Average Rainfall and our target will be the label which is crop name. These all will be fed up into our logistic regression model and our model will get trained based on the previous dataset.

Logistic Regression's Accuracy is: 97.0 %				
	precision	recall	f1-score	support
apple	1.00	1.00	1.00	17
banana	1.00	1.00	1.00	21
black gram	0.95	0.95	0.95	19
coconut	1.00	1.00	1.00	18
coffee	1.00	1.00	1.00	21
cotton	1.00	0.95	0.98	21
grapes	1.00	1.00	1.00	25
jute	0.90	0.86	0.88	21
kidney beans	1.00	1.00	1.00	20
maize	0.89	1.00	0.94	16
mango	1.00	1.00	1.00	22
orange	1.00	1.00	1.00	23
papaya	0.95	0.90	0.93	21
pomegranate	1.00	1.00	1.00	18
rice	0.83	0.88	0.86	17
accuracy			0.97	300
macro avg	0.97	0.97	0.97	300
weighted avg	0.97	0.97	0.97	300

Fig:4.4: Accuracy for Logistic Regression Model

Since our data is normally distributed. Hence, we obtained 97.00% (As shown in fig:4.4)accuracy in the logistic regression model.

DECISION TREE MODEL

In real life, a tree has a lot of analogies, and it turns out that it has inspired a lot of machine learning, including classification and regression. Decision Trees are a form of

Machine Learning model that can achieve high accuracy in a variety of tasks while also being very easy to understand. What distinguishes decision trees from other ML models is their consistency of knowledge representation. A decision tree's "information" is explicitly formulated into a hierarchical framework after it has been educated. This structure organises and displays information in such a way that even non-experts will understand it. Induction and Pruning are the two steps used to construct Decision Tree models. Induction is the process of actually constructing the tree, that is, setting all of the hierarchical decision boundaries based on the data. Because of the nature of training decision trees they can be prone to major overfitting. Pruning is the process of removing the unnecessary structure from a decision tree, effectively reducing the complexity to combat overfitting with the added bonus of making it even easier to interpret.

Induction

On a high level, decision tree induction involves four key steps in order to build the tree:

1. Start with your training dataset, which should include some feature variables as well as classification or regression results.
2. Determine the dataset's "best feature" for data splitting; more on how we define "best feature" later.
3. Divide the data into subsets containing the best feature's possible values. This splitting essentially defines a tree node, i.e., each node is a splitting point centered on a specific feature from our data.
4. Using the subset of data generated in step 3, recursively generate new tree nodes. We keep splitting until we've reached a point where we've achieved maximum accuracy while minimizing the number of splits / nodes, in some way.

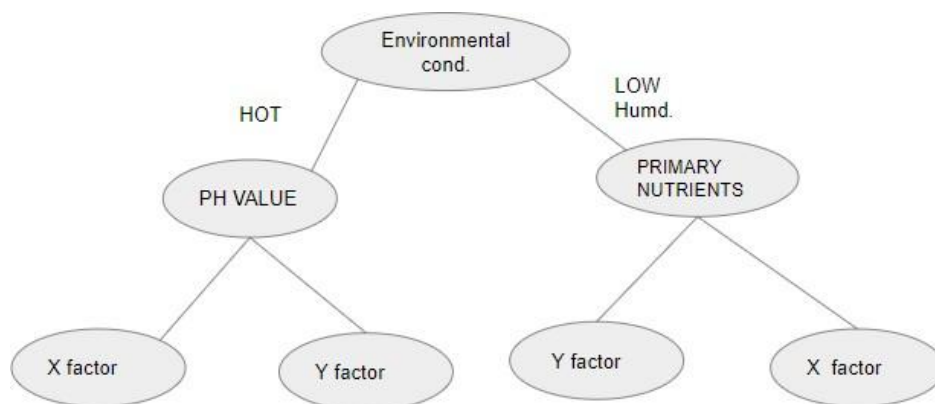
Pruning

Because of the design of decision tree preparation, they can be susceptible to significant overfitting. It can be difficult to determine the correct value for the minimum number of instances per node. Most of the time, we'll take a safe bet and set the minimum to a low value, resulting in numerous splits and a big, complex tree.

The key is that many of these splits would be redundant and unnecessary for improving our model's accuracy. Tree pruning is a technique that takes advantage of this splitting redundancy to prune out unwanted splits in our trees. Pruning, on a high level, compresses a portion of the tree from strict and rigid decision boundaries into more smooth and generalizable ones, effectively reducing tree complexity. The number of splits in a decision tree is used to determine its complexity.

Justification

Moving through each node in the tree and evaluating the effect of removing it on the cost function is a simple but highly successful pruning process. If it doesn't change much, get rid of it.



17

Fig:4.5: Decision tree model

In this, we will feed the features and target variable to the tree classifier and we obtained the accuracy as such below (As shown in Fig:4.6) Decision tree accuracy Obtained 97.33% Accuracy from Decision tree model.

DecisionTrees's Accuracy is: 97.3333333333334 %				
	precision	recall	f1-score	support
apple	1.00	1.00	1.00	17
banana	1.00	1.00	1.00	21
black gram	1.00	0.95	0.97	19
coconut	1.00	1.00	1.00	18
coffee	1.00	0.90	0.95	21
cotton	1.00	1.00	1.00	21
grapes	1.00	1.00	1.00	25
jute	0.75	1.00	0.86	21
kidney beans	1.00	1.00	1.00	20
maize	0.94	1.00	0.97	16
mango	1.00	1.00	1.00	22
orange	1.00	1.00	1.00	23
papaya	1.00	1.00	1.00	21
pomegranate	1.00	1.00	1.00	18
rice	1.00	0.71	0.83	17
accuracy			0.97	300
macro avg	0.98	0.97	0.97	300
weighted avg	0.98	0.97	0.97	300

Fig:4.6: Decision tree accuracy

Random Forest Classification

As the name suggests, a random forest is made up of a large number of individual decision trees that work together as an ensemble. Each tree in the random forest produces a class prediction, and the class with the most votes becomes the prediction of our model (see figure below). As the name suggests, a random forest is made up of a large number of individual decision trees that work together as an ensemble. Each tree in the random forest produces a class prediction, and the class with the most votes becomes the prediction of our model (see figure 4.7 below).

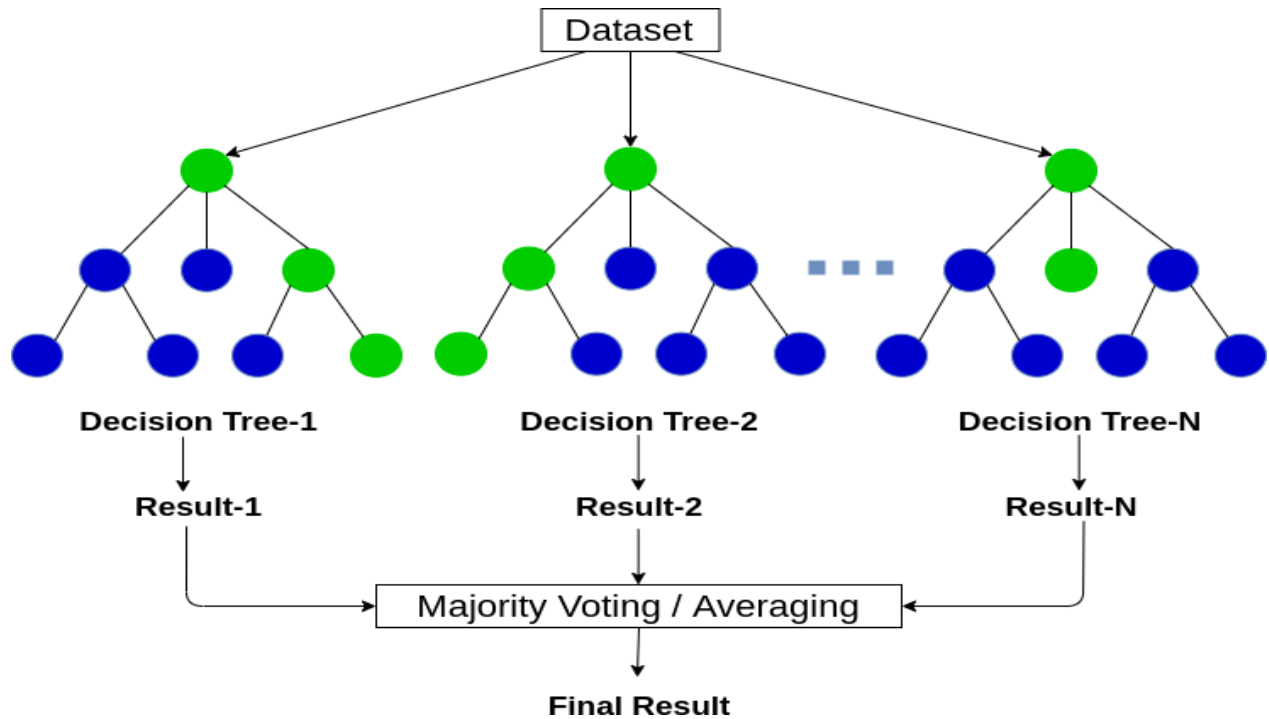


Fig:4.7: Random Forest tree model

Justification

The wisdom of crowds is the basic idea behind random forest, and it's an easy but strong one. Any of the individual constituent models will outperform a large number of relatively uncorrelated models (trees) working as a committee. The secret is the low correlation between models. Uncorrelated models can generate ensemble predictions that are more reliable than any of the individual predictions, similar to how low-correlation investments (like stocks and bonds) come together to create a portfolio that is greater than the sum of its parts. The explanation for this amazing impact is that the trees shield each other from their own mistakes (as long as they don't all make the same mistake). While some trees will be incorrect, several others will be correct, allowing the trees to move in the correct direction as a group.

As a result, the requirements for a successful random forest are as follows::

1. Our features must have some real signal in order for models designed with them to outperform random guessing.
2. The individual trees' predictions (and thus their errors) must have low correlations with one another.

Our random forest classifier was fed the target variable as well as the features. Random forest classification is one of machine learning's most efficient ensemble learning methods. Hence, we obtained 99% as an accuracy for our model.

```

RF's Accuracy is: 0.99
precision recall f1-score support
apple      1.00      1.00      1.00      17
banana     1.00      1.00      1.00      21
black gram 1.00      1.00      1.00      19
coconut    1.00      1.00      1.00      18
coffee    1.00      1.00      1.00      21
cotton     1.00      1.00      1.00      21
grapes     1.00      1.00      1.00      25
jute       0.88      1.00      0.93      21
kidney beans 1.00      1.00      1.00      20
maize      1.00      1.00      1.00      16
mango      1.00      1.00      1.00      22
orange     1.00      1.00      1.00      23
papaya     1.00      1.00      1.00      21
pomegranate 1.00      1.00      1.00      18
rice       1.00      0.82      0.90      17

accuracy   0.99
macro avg  0.99      0.99      0.99      300
weighted avg 0.99      0.99      0.99      300

```

```

[66]: score = cross_val_score(regressor, features, target, cv=5)
score
[66]: array([0.99666667, 0.99      , 0.99      , 0.99666667, 0.98666667])

```

Fig:4.8: Accuracy for Random Forest tree model

SUPPORT VECTOR MACHINE

The support vector machine algorithm's goal is to find a hyperplane in an N-dimensional space (N — the number of features) that categorizes data points clearly.

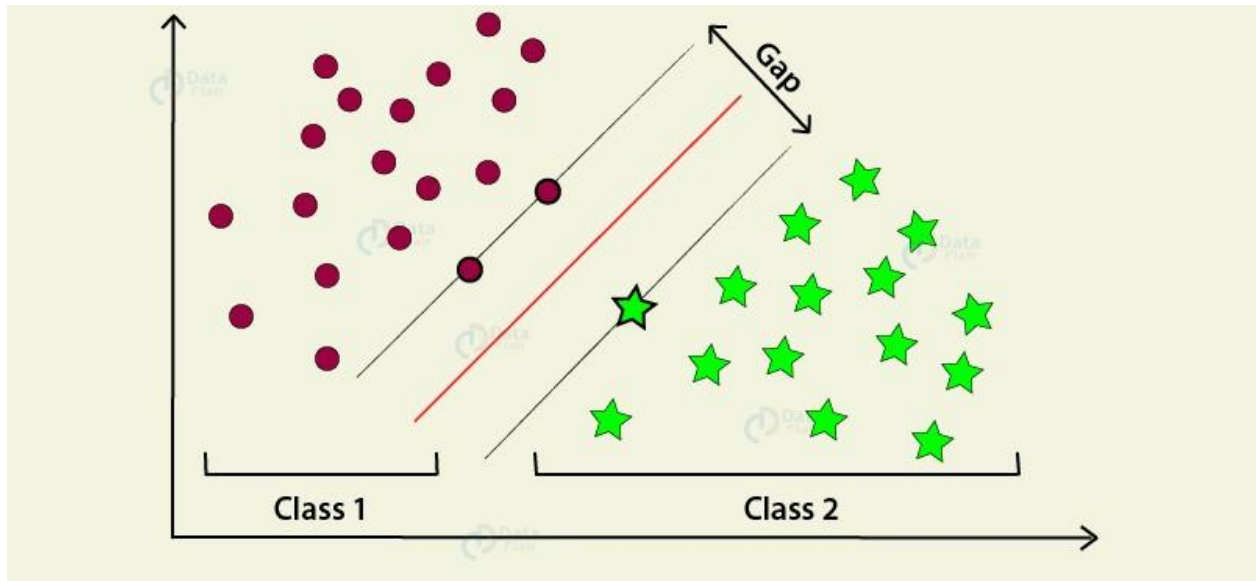


Fig:4.9: SVM MODEL

There are several different hyperplanes that could be used to distinguish the two types of data points (as seen in fig 4.9). Our aim is to find a plane with the greatest margin, or the greatest distance between data points from both groups. Maximizing the margin gap provides some reinforcement, making it easier to classify potential data points. Hyperplanes are decision boundaries that aid in data classification. Different groups may be assigned to data points on either side of the hyperplane. The hyperplane's dimension is also determined by the number of functions. If there are only two input features, the hyperplane is just a line. The hyperplane becomes a two-dimensional plane when the number of input features reaches three. When the number of features reaches three, it becomes impossible to picture.

Justification

Support vectors are data points that are closer to the hyperplane and have an impact on the hyperplane's direction and orientation. We optimize the classifier's margin by using these support vectors. The hyperplane's location would be altered if the support vectors are deleted. These are the points that will assist us in constructing our SVM. The aim of the SVM algorithm is to maximize the distance between the data points and the hyperplane. Hinge loss is a loss function that aids in margin maximization..

```
SVM's Accuracy is: 0.14
precision    recall  f1-score   support

   apple      1.00    0.53    0.69      17
  banana      1.00    0.05    0.09      21
black gram      1.00    0.16    0.27      19
  coconut      1.00    0.11    0.20      18
   coffee      1.00    0.05    0.09      21
   cotton      1.00    0.05    0.09      21
   grapes      0.00    0.00    0.00      25
    jute      1.00    0.05    0.09      21
kidney beans      0.00    0.00    0.00      20
   maize      0.06    1.00    0.11      16
   mango      1.00    0.23    0.37      22
   orange      1.00    0.04    0.08      23
  papaya      0.00    0.00    0.00      21
pomegranate      0.00    0.00    0.00      18
    rice      1.00    0.12    0.21      17

 accuracy          0.14      300
  macro avg          0.14      300
 weighted avg          0.14      300

C:\Users\ELCOT\anaconda3\lib\site-packages\sklearn\metrics\_classification.py:1221:
Warning: Some of the data points are ill-defined and being set to 0.0 in labels with no predicted samples. Use `
warnings.warn(_warn_prf(average, modifier, msg_start, len(result)))

In [62]: score = cross_val_score(SVM, features, target, cv=5)
score
Out[62]: array([0.21      , 0.24666667, 0.3      , 0.29      , 0.32      ])
```

Fig:4.10: SVM MODEL ACCURACY

When we have two group classification problems, SVM will function perfectly. For example, whether or not diseases are present in the plant, these types of issues would be ideal for SVM. If the number of rows in the dataset reaches the limits, the SVM model will not provide perfect accuracy. As a result, we were able to achieve a 14 percent accuracy for our model (as shown in fig 4.10).

Naive Bayes Classification

A probabilistic machine learning model called a Naive Bayes classifier is used for classification tasks. The Bayes theorem will be the heart of classifiers.

The diagram shows the equation $P(c | x) = \frac{P(x | c)P(c)}{P(x)}$. Arrows point from labels to the corresponding parts of the equation: 'Likelihood' points to $P(x | c)$, 'Class Prior Probability' points to $P(c)$, 'Posterior Probability' points to $P(c | x)$, and 'Predictor Prior Probability' points to $P(x)$.

$$P(c | X) = P(x_1 | c) \times P(x_2 | c) \times \dots \times P(x_n | c) \times P(c) \quad \text{-----4.4}$$

Provided that X has occurred, we can calculate the likelihood of C occurring using Bayes theorem. The proof is C , and the hypothesis is X . The predictors/features are assumed to be independent in this case. That is, the existence of one feature has no effect on the other. As a result, it is referred to as naive.

Types of Naive Bayes Classifier

Multinomial Naive Bayes

This is most often used to solve document classification issues, such as determining if a document falls in the sports, politics, or technology categories. The frequency of the terms present in the text is one of the features/predictors used by the classifier.

Bernoulli Naive Bayes

The predictors are Boolean variables, close to the multinomial naive bayes. The parameters we use to predict the class variable only accept yes or no answers, such as whether a word appears in the text or not.

Gaussian Naive Bayes

We assume these values are sampled from a gaussian distribution when the predictors take up a continuous value and are not discrete.

```
print(classification_report(Ytest,predicted_values))
```

Naive Bayes's Accuracy is: 0.99

	precision	recall	f1-score	support
apple	1.00	1.00	1.00	17
banana	1.00	1.00	1.00	21
black gram	1.00	1.00	1.00	19
coconut	1.00	1.00	1.00	18
coffee	1.00	1.00	1.00	21
cotton	1.00	1.00	1.00	21
grapes	1.00	1.00	1.00	25
jute	0.91	0.95	0.93	21
kidney beans	1.00	1.00	1.00	20
maize	1.00	1.00	1.00	16
mango	1.00	1.00	1.00	22
orange	1.00	1.00	1.00	23
papaya	1.00	1.00	1.00	21
pomegranate	1.00	1.00	1.00	18
rice	0.94	0.88	0.91	17
accuracy			0.99	300
macro avg	0.99	0.99	0.99	300
weighted avg	0.99	0.99	0.99	300

Fig:4.11: NAIVE BAYES ACCURACY

Since our dataset was normally distributed, so that we have used Gaussian Naive Bayes Classifier as our classifier. Basically, the Naive bayes classifier are well efficient in predicting or recommending systems. So that we have obtained 99%(as shown in fig 4.11) as our accuracy.

4.4 COMPARISON BETWEEN RESULTS:

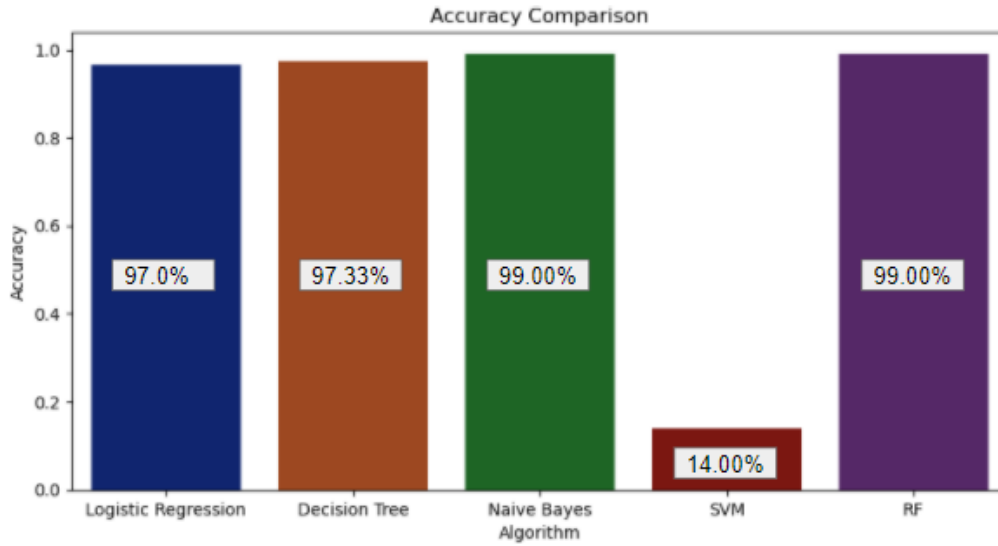


Fig:4.12: ACCURACY COMPARISON

ACCURACY TABLE:

Table No. 4.2: Accuracy Comparison Table

S.No	Name of the Algorithm	Accuracy
1.	Logistic Regression	97.00%
2.	Decision Tree	97.33%
3.	Naive Bayes Algorithm	99.00%
4.	Support Vector Machine	14.00%
5.	Random Forest	99.00%

We have obtained 99.20% (as shown in fig 4.13) accuracy for our Hybrid Ensemble model for better crop yield prediction.

```

n_iter_i = _check_optimize_result(
Accuracy of ensemble hybrid model: 99.2 %

In [18]: new_input=[[115,40,28,27.309018,55.196224,6.348316,141.483164]]

In [19]: ensemble.fit(X,Y)
new_output=ensemble.predict([[115,40,28,27.309018,55.196224,6.348316,141.483164]])
print(new_input,new_output)

C:\Users\ELCOT\anaconda3\lib\site-packages\sklearn\linear_model\_logistic.py:762: ConvergenceWarning: lbfgs failed to converge
(status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
  https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
  https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
n_iter_i = _check_optimize_result(
[[115, 40, 28, 27.309018, 55.196224, 6.348316, 141.483164]] ['coffee']
```

Fig 4.13 Accuracy for Hybrid Ensemble Model

4.5 System Requirements:

- Operating System: Windows/Mac
- RAM: 4GB
- Processor: 64x 1.0Ghz processor
- ROM: 8GB

4.6 Front-end:

In the real world scenario, we are unable to use our jupyter notebook for recommending crops to the farmers. Here the front-end comes into the picture. Front-end is nothing but the user interface which is very much helpful to allow the users to enter the input values at the web portal. The front end has connected with the API to fetch the data from our machine learning model prediction and it will be displayed on the front screen as an output.

4.7 API:

ApplicationInterface(API) which is used to connect the front end with the backend as well as with our machine learning model. Pickle package in python was used to serialize our model with pkl file and this pkl file will be embedded with our application interface. For building an API, we have used a flask python package to connect the front end and backend.

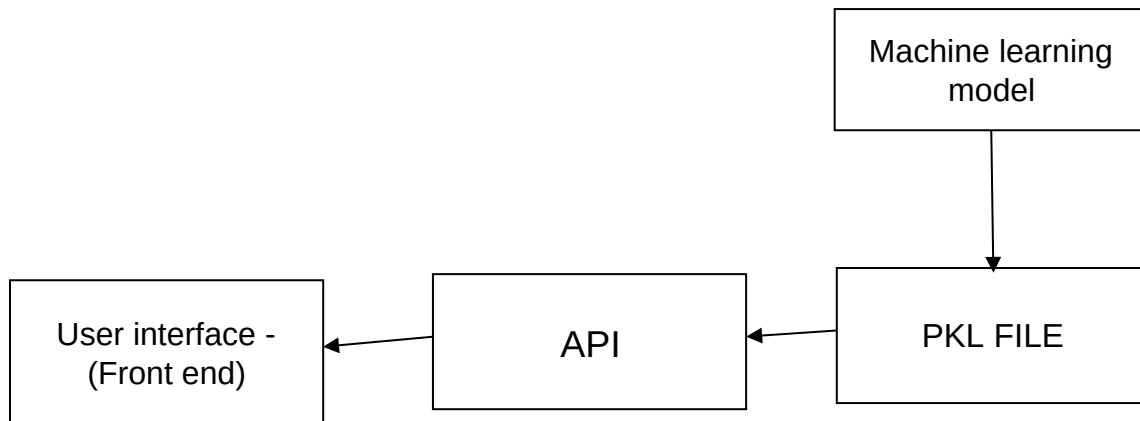


Fig 4.14 Web application Model

4.8 Output Screenshots:

The screenshot shows the web application interface for the Crop Recommendation System. The header includes the title "CROP RECOMMENDATION SYSTEM" and the Sathyabama Institute of Science and Technology logo. The main heading is "CROP RECOMMENDATION SYSTEM USING HYBRID MODEL". Below this, there are four input fields for user data:

- Nitrogen (N): 1
- Phosphorus (P): 70
- Potassium (K): 16
- Temperature: 22.64236876

Fig 4.15 Output Screenshot - 1

127.0.0.1:5000

Apps YouTube What is an Algorith... Maps Sathyabama Place... Gmail FACE Prep | Login LetsUpgrade Com... Superset Registrati... Novorésumé New Tab

Potassium (K)

16

Temperature

22.64236876

Humidity

21.59396123

Ph Value of the Soil

5.946999529

Rainfall

122.3886015

Submit

The recommended crop is KIDNEY BEANS

Fig 4.17 Output Screenshot - 2

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 CONCLUSION:

This project employs machine learning technique to assess results, with Random Forest, Logistic Regression and Decision Tree, Support Vector Machine, and Naive Bayes classification algorithms being used. Except for SVM, all of the other algorithms in our proposed model provide better yield prediction when compared. Along with the results of the classification, which demonstrate that the dataset has improved. As a result, we determined that the proposed model is more efficient than the current model for determining crop yield. The introduction of the above scheme will aid in the betterment of our country's agricultural practices. It can also be used to help farmers reduce their losses and increase crop yields in order to increase their resources in agriculture. The model can be strengthened by combining it with other departments such as horticulture, sericulture, and others to help our country's agricultural production.

5.2 FUTURE ENHANCEMENT:

There are several enhancements that could be investigated to increase the predictability and scalability of this method. We will use this method to analyse various data sets from different countries in the future since we created a generalized system. Handling multiple class labels in the prediction phase can dramatically improve crop yield efficiency, and this could be another promising research path. The dimensionality of soil, water quality, and pH level can vary widely around the world, making the identification and selection of significant attributes for future research extremely difficult. In addition, we can use an API module in conjunction with our web application to predict disease risk based on soil conditions.

REFERENCES

- [1] Prabakaran, G., D. Vaithyanathan, and Madhavi Ganesan. "Fuzzy decision support system for improving the crop productivity and efficient use of fertilizers." *Computers and electronics in agriculture* 150 (2018), pp 88-97.
- [2] Pathak, P., Phalke, R., Patil, P., Prakash, P., & Bhosale, S. (2008). Survey on Crop Suggestion Using Weather Analysis.
- [3] Kuanr, Madhusree, B. Kesari Rath, and S. Nandan Mohanty. "Crop Recommender System for the Farmers using Mamdani Fuzzy Inference Model." *International Journal of Engineering & Technology* 7, no. 4.15 (2018), pp 277-280.
- [4] Rakesh Kumar, M. P. Singh, Prabhat Kumar, J. P. Singh, "Crop Selection Method to Maximize Crop Yield using Machine Learning Technique", *International Conference on Smart Technology and Management for Computing, Communication, Controls, Energy and Materials (ICSTM)*, 2015
- [5] Pabitra Mitra, Abhijit Mukharjee, Aditi Chndra, "Machine Learning for Prediction of Crop Yield", *Case Study Of IIT Kharagpur*, 2017
- [6] Ashraf, A., Akram, M., Sarwar, M., 2014. Fuzzy decision support system for fertilizer. *Neural Comput. Appl.* 25(6), 1495–1505.
- [7] Ayoub, Ali.T., 1999. Fertilizers and the environment. *Nutr. Cycl. Agroecosyst.* 55 (2), 117–121.
- [8] Azadeh, A., Saberi, M., Gitiforouz, A., 2013. An integrated fuzzy mathematical model and principal component analysis algorithm for forecasting uncertain trends of electricity consumption. *Qual. Quant.* 47(4), 2163–2176.
- [9] Aziz, T., Maqsood, M.A., Kanwal, S., Hussain, S., Ahmad, H.R., Sabir, M., 2015. Fertilizers and environment: issues and challenges. In: *Crop Production and Global Environmental Issues*. Springer, Cham, pp. 575–598.
- [10] British Association Meeting, 1955. Trace Elements and Plant Growth. *Nature* 176, 621–623.