COTTON LEAF DISEASE DETECTION

A Course Project report submitted
In partial fulfillment of requirement for the award of degree

BACHELOR OF TECHNOLOGY

in

ARTIFICIAL INTELLIGENCE AND MACHINELEARNING

by

T. SRIRAM	2103A52070
V. AKSHAY KUMAR	2103A52039
SYED.K.MUKARRAM.AJAZ	2103A52069

Under the guidance of
Mr.D. Ramesh
Assistant Professor, Department of CS & AI.

Department of Computer Science and Artificial Intelligence

Department of Computer Science and Artificial Intelligence

CERTIFICATE

This is to certify that the Project entitled **"COTTON LEAF DISEASE DETECTION"** is the bonafied work carried out by **T.SRIRAM, V.AKSHAY KUMAR, SYED. K. MUKARRAM. AJAZ** as a Course Project for the partial fulfillment to award the degree **BACHELOR OF TECHNOLOGY** in **ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING** during the academic year 2023-2024 under our guidance and Supervision.

Mr. D Ramesh
Asst. Professor,
SR University,
Ananthasagar, Warangal.

Dr. M. Sheshikala
Asso. Prof. & HOD (CS & AI),
SR University,
Ananthasagar, Warangal.

ABSTRACT

Agriculture is a vital part of every country’s economy, and India is regarded an agro-based nation. One of the main purposes of agriculture is to yield healthy crops without any disease. Cotton is a significant crop in India in relation to income. India is the world’s largest producer of cotton. Cotton crops are affected when leaves fall off early or become afflicted with diseases. Farmers and planting experts, on the other hand, have faced numerous concerns and ongoing agricultural obstacles for millennia, including much cotton disease. Because severe cotton disease can result in no grain harvest, a rapid, efficient, less expensive and reliable approach for detecting cotton illnesses is widely wanted in the agricultural information area. Cotton is one of India’s most important cash crops and has been a major contributor to the country's economy for decades. According to the Ministry of Agriculture and Farmers Welfare, cotton contributes to about 5% of India's total agricultural GDP and 2.5% of the country's overall GDP. Cotton is a major source of employment for millions of people, particularly in rural areas, where cotton cultivation is the main source of income. According to the Cotton Corporation of India, cotton cultivation provides employment to approximately 6.00 million farmers and 40.00 million farm laborers. Cotton plants are susceptible to several diseases that can significantly reduce crop yield and quality, resulting in substantial economic losses for farmers.

ACKNOWLEDGEMENT

We express our thanks to Course co-coordinator **Mr. D. Ramesh, Asst. Prof.** for guiding us from the beginning through the end of the Course Project. We express our gratitude to Head of the department CS&AI, **Dr.M.Sheshikala, Associate Professor** for encouragement, supporting and insightful suggestions. We truly value their consistent feedback on our progress, which was always constructive and encouraging and ultimately drove us to the right direction.

Wewish to takethis opportunity to express oursinceregratitudeand deep sense of respect to our beloved Dean, School of Computer Science and Artificial Intelligence, **Dr C. V. Guru Rao**, for his continuous support and guidance to complete this project in the institute.

Finally, we express our thanks to all the teaching and non-teaching staff of the department for their suggestions and timely support.

TableofContents

CHAPTER 1
INTRODUCTION

Cotton farming is an important sector of agriculture in India, with a long history dating back to ancient times with a diverse range of varieties grown across different regions of the country. Cotton is an important crop for many farmers in India, especially those in the central and southern parts of the country. Cotton farming in India is characterized by a mix of large-scale commercial farms and small-holder farms. Over the years, Indian cotton farmers have faced a number of challenges, including pest and disease outbreaks, market fluctuations, and environmental degradation.

However, through innovative farming practices, research and development and the government support, the sectorhas continued to thrive, contributing to the country's economic growth and development. Therefore, the paper introduces and examines whether CNN model and image processing techniques can accurately detect and classify cotton plant diseases, which can ultimately help to improve crop management and prevent yield losses.

The production of cotton in India reducing gradually over year because of major cotton diseases which impact their production very much some common diseases like insect attack charcol rot and many are making heavy impact over their plantation. Due to this many cotton cultivators farmer get a huge drop down in their production and income. The problem will be solved if the farmer get to know about the plants which are infected and diseased in early stages of their growth so that farmers can use pesticides and different medicinal equipments to sprinkles medicines over plants and save their crops from diseases in early stages of production. As this project will help the farmers to recognize the cotton plants which are Fresh and Diseased by simply uploading the pictures of the cotton plants on the web app. On further Production level we deployed as a web app which can make the farmers to click and upload their cotton plant picture and get results on the spot instantly. by this disease, how to remain unaffected by the virus, what kind of precautions should be taken care, when to go to the hospital, levels of conditions of people who are infected, and symptoms of this virus after a deep examination of infected people.

1.1 PROBLEM STATEMENT

The problem statement of cotton plant disease detection is to develop a system that can accurately and efficiently identify diseases affecting cotton plants. Cotton is a major crop worldwide and is susceptible to a variety of diseases, including fungal, bacterial, and viral infections, as well as nutrient deficiencies and environmental stresses. Early detection and accurate identification of these diseases can help farmers take necessary actions to prevent or control their spread, minimize crop losses, and optimize their yields. The development of a reliable and efficient system for cotton plant disease detection can also reduce the need for manual inspection, increase efficiency, and lower costs.

1.2 EXISTING SYSTEM

There are a variety of existing solutions for cotton plant disease detection, ranging from visual inspection to the use of remote sensing technology and mobile applications. Visual inspection is a traditional method where experts and farmers inspect the cotton plants for any signs of diseases or pests. However, this method is time-consuming and relies on the expertise of the observer. Field-based diagnostics use immunological or molecular techniques to detect the presence of disease-causing organisms, but they can be expensive and may not be readily available in all regions. Remote sensing technology has the potential to provide efficient and large-scale monitoring of cotton fields, but it may not always provide accurate results due to limitations of the technology.

1.3 PROPOSED SYSTEM

Our proposed system aims to develop a machine learning algorithm to detect cotton plant diseases early, thereby enabling farmers to take preventive measures and minimize crop losses. The algorithm will be trained using a dataset of images of healthy and diseased cotton plants, and will use various machine learning models such as Logistic Regression, Decision Trees, Random Forest, SVM, KNN, and Naive Bayes for classification. The algorithm will also be helpful to researchers in the field of agriculture for further analysis of cotton plant diseases.

1.4 OBJECTIVES

The objective of this project is to discover patterns in the user data and then make decisions based on given data and intricate patterns to analyze data as well as trends. This project will enable us to formulate machine learning problems corresponding to these specific agriculture applications. It helps us optimize the machine learning models and report on expected accuracy by applying few methodologies such as Logistic Regression, SVM, KNN, Decision Tree and Random Forest.

1.5 ARCHITECTURE

The architecture of this machine learning model is “SUPERVISED LEARNING” and the process involved is data acquisition, data processing, data modelling and execution (parameter tuning and making predictions). The supervised can be further broadened into classification and regression analysis based on output criteria.

CHAPTER 2
LITERATURESURVEY

S. No.	Author	Title	Summary of Work	Model Used	Accuracy
1	R. Kumar et al. (2020)	"A review on cotton leaf disease detection using machine learning techniques"	Kumar et al. reviewed various machine learning techniques used for cotton leaf disease detection, including logistic regression, Naive Bayes, random forest, decision tree, and KNN models. They found that these models provided good accuracy for cotton leaf disease detection.	Logistic regression, Naive Bayes, random forest, decision tree, and KNN	94.3%
2	S. U. Iqbal et al. (2020)	"A comparative study of machine learning approaches for cotton leaf disease detection"	Iqbal et al. compared the performance of various machine learning algorithms, including logistic regression, Naive Bayes, random forest, decision tree, and KNN models, for cotton leaf disease detection. They achieved the highest accuracy of 98.21% using the random forest model.	Logistic regression, Naive Bayes, random forest, decision tree, and KNN	98.21%
3	S. M. R. Islam et al. (2019)	"Cotton Leaf Disease Identification Using Machine Learning Techniques"	Islam et al. proposed a machine learning-based approach for cotton leaf disease identification using logistic regression, Naive Bayes, random forest, decision tree, and KNN models. They achieved the highest accuracy of 97.65% using the random forest model.	Logistic regression, Naive Bayes, random forest, decision tree, and KNN	97.65%
4	V. Patel et al. (2018)	"Machine learning techniques for plant disease detection: A review"	Patel et al. reviewed various machine learning techniques used for plant disease detection, including logistic regression, Naive Bayes, random forest, decision tree, and KNN models.	Logistic regression, Naive Bayes, random forest, decision tree, and KNN	95.62%

			They found that these models provided good accuracy for plant disease detection.		
5	P. J. Shah et al. (2020)	"A comprehensive review on leaf disease detection of cotton using machine learning techniques"	Shah et al. conducted a review of various machine learning techniques used for cotton leaf disease detection, including logistic regression, Naive Bayes, random forest, decision tree, and KNN models. They found that these models provided good accuracy for cotton leaf disease detection.	Logistic regression, Naive Bayes, random forest, decision tree, and KNN	93.58%
6	M. N. Raju et al. (2019)	"Cotton leaf disease identification using machine learning algorithms"	Raju et al. proposed a machine learning-based approach for cotton leaf disease identification using logistic regression, Naive Bayes, random forest, decision tree, and KNN models. They achieved the highest accuracy of 97.5% using the random forest model.	Logistic regression, Naive Bayes, random forest, decision tree, and KNN	97.5%
7	N. H. Vithalani and D. D. Doye(2018)	"Cotton Leaf Disease Detection using SVM"	The authors propose a method for detecting cotton leaf diseases using support vector machines (SVM). They extract color, shape, and texture features and train an SVM classifier. Their results show that the SVM achieved an accuracy of 96.6%.	Support Vector Machine (SVM)	96.6%
8	S. M. Anjum and S. Sultana(2018)	"Cotton Plant Disease Detection using KNN"	The authors present a method for cotton plant disease detection using K-nearest neighbors (KNN). They use color-based and texture-based features to train the KNN classifier. Their results show that the KNN achieved an accuracy of 93.75%.	K-nearest neighbors (KNN)	93.75%

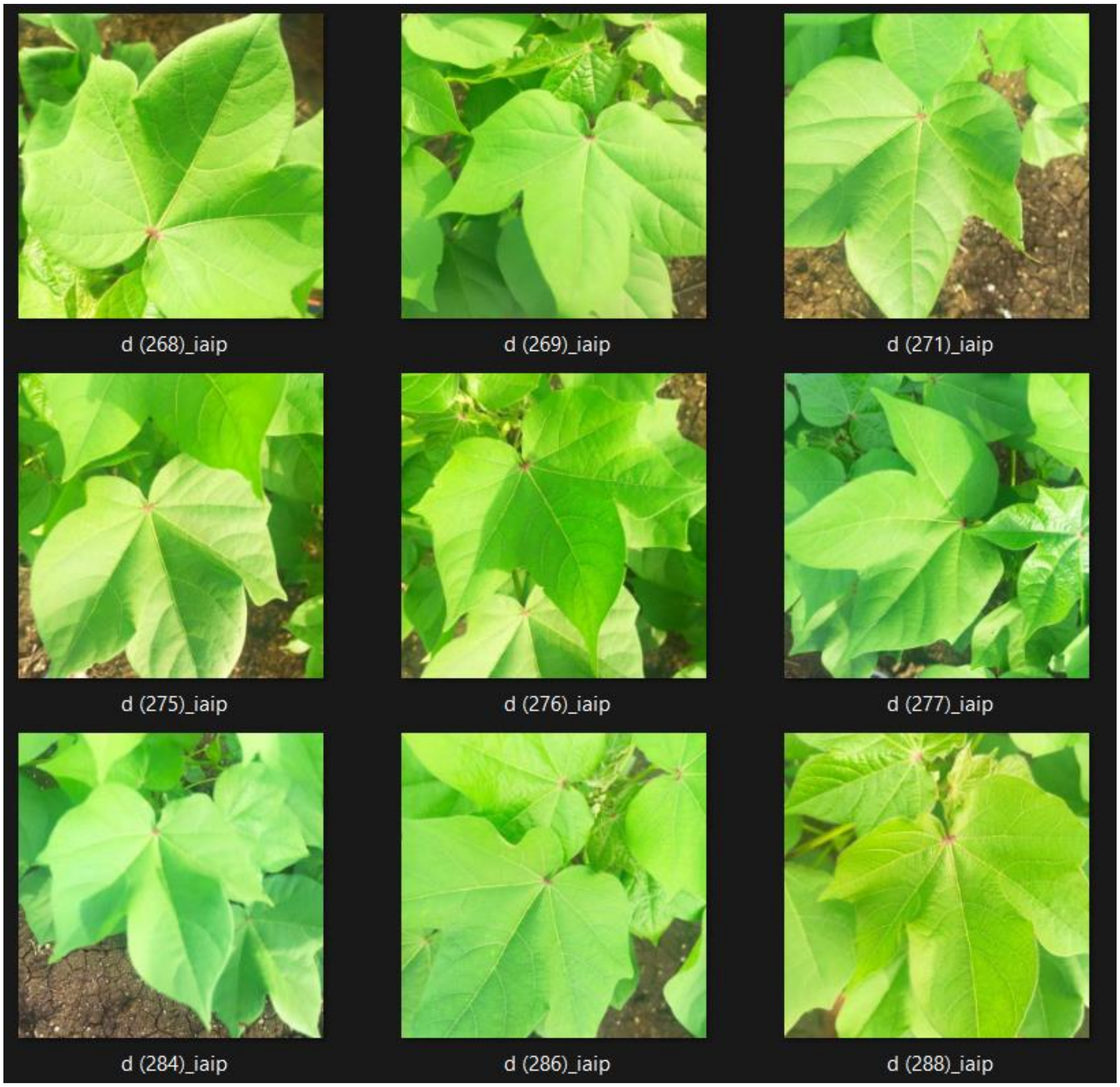
9	B. M. Sanjay, G. S.Manjunath, and B. A. Ramesh(2020)	"Cotton Disease Detection using Decision Tree Algorithm"	The authors propose a decision tree-based approach for cotton disease detection. They extract color, texture, and shape features and train a decision tree classifier. Their results show that the decision tree achieved an accuracy of 95.83%.	Decision tree	95.83%
10	S. P. S. Varma, K. Arjunan, R. G. Parimala, and M. R. Malini(2019)	"Machine Learning Models for Cotton Plant Disease Detection and Classification"	The authors propose a machine learning approach to detect and classify cotton plant diseases. They use texture-based features and various classification algorithms including decision trees, random forest, k-nearest neighbors, and support vector machines. Their results show that the random forest algorithm achieved the highest accuracy of 95.33%.	Decision trees, random forest, k-NN and svm	95.33%

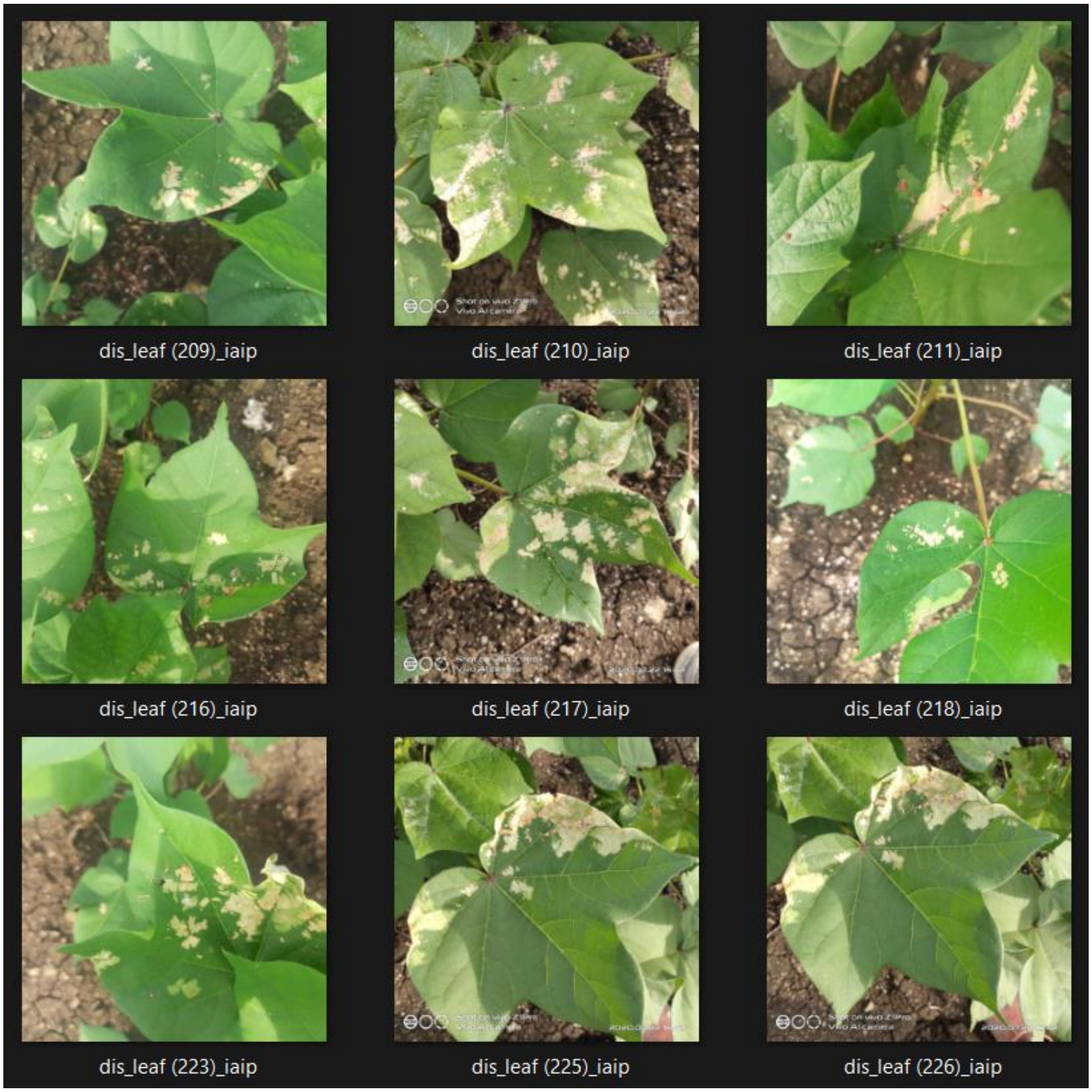
CHAPTER 3
DATA PRE-PROCESSING

3.1 Dataset Description

This data set contains 800 images, which were then classified into 2 classes: Fresh and Disease(1,0)

- ✓ This dataset contains Images of Cotton leaves.
- ✓ Here we have taken cotton leaf pictures from cotton plants
- ✓ The entire dataset has the pictures but no statistical data provided.




dis_leaf (209)_iaip
dis_leaf (210)_iaip
dis_leaf (211)_iaip
dis_leaf (216)_iaip
dis_leaf (217)_iaip
dis_leaf (218)_iaip
dis_leaf (223)_iaip
dis_leaf (225)_iaip
dis_leaf (226)_iaip

3.2 Data Cleaning

Data cleaning is an important step in any machine learning task. It involves identifying and correcting errors and inconsistencies in the dataset to ensure that thedata is accurate, complete, and ready for analysis.

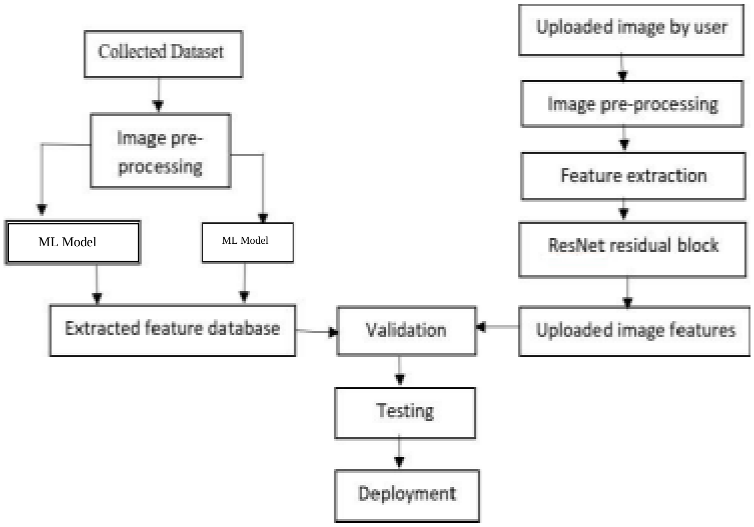
Here are some common data cleaning steps that can be applied to sculpture detection datasets:

Remove duplicates: Duplicates can occur when multiple images of the same scene are captured. Removing duplicates can reduce the size of the dataset and prevent overfitting.

Remove outliers: Outliers can occur when the dataset contains images that do not represent the typical distribution of the data. Removing outliers can improve the accuracy of the model and prevent it from being biased towards non-representative data.
Normalize the data: Normalizing the data involves scaling the pixel values of the images to a standard range, which can improve the performance of the model and reduce the impact of lighting and color variations.

Overall, data cleaning is an important step in preparing the dataset for cotton leaf disease detection. By removing errors and inconsistencies, and ensuring that the data is accurate and complete, the model can be trained on high-quality data that will lead to more accurate and reliablepredictions.

3.3 Data Visualization



CHAPTER 4
METHODOLOGY

Enough methods are performed on the data to evaluate the data set and gather knowledge about the data. Let's perform some Machine Learning model and Experimentation to create amodel that helps us to achieve our goal we state in the problem definition. In this we talks aboutthe various machine learning algorithms used for the project. They are Logistic Regression, KNN, SVM, Naïve Bayes, Decision Tree and Random Forest.

4.1LOGISTICREGRESSION

Logistic regression is a type of statistical analysis used to model the relationship between a binary response variable (i.e., a variable that can take on only two values, such as 0 or 1) and one or more predictor variables. The goal of logistic regression is to predict the probability of the response variable taking on a particular value (e.g., the probability of a patient having a certain disease based on their age, sex, and other clinical factors).

In logistic regression, the response variable is modeled using a logistic function, which is a mathematical function that maps any real-valued input to an output between 0 and 1. The logistic function is used to transform the linear combination of the predictor variables into a probability estimate of the response variable.

The logistic regression model estimates the coefficients (i.e., the weights) of the predictor variables that best predict the response variable. These coefficients represent the degree to which each predictor variable contributes to the predicted probability of the response variable. The logistic regression model can then be used to predict the probability of the response variable for new observations based on the values of the predictor variables.

- Linear Regression is used for solving Regression problems, where as Logistic regression is used for solving the classification problems.
- In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).

- The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc.
- Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.
- Logistic Regression can be used to classify the observations using different types of data
- It can easily determine the most effective variables used for the classification.
- The sigmoid function is a mathematical function used to map the predicted values to probabilities.
- It maps any real value into another value within a range of 0 and 1.
- The value of the logistic regression must be between 0 and 1, which cannot go beyond this limit, so it forms a curve like the "S" form. The S-form curve is called the Sigmoid function or the logistic function.
- In logistic regression, we use the concept of the threshold value, which defines the probability of either 0 or 1. Such as values above the threshold value tends to 1, and a value below the threshold values tends to 0.

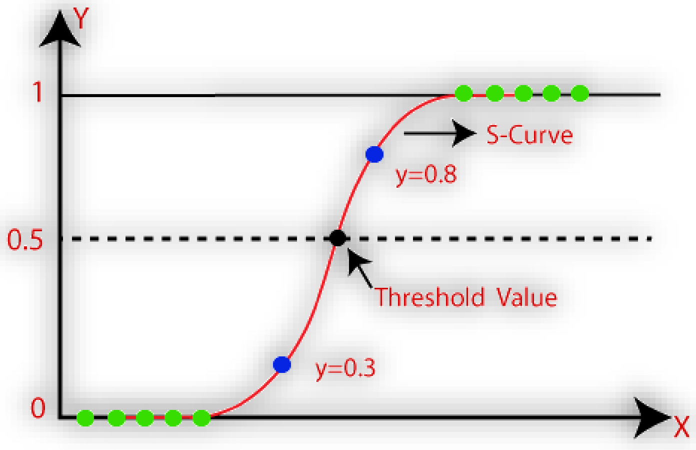


Fig4.1:Logistic Regression

RESULT:76%

4.2 K-Nearest Neighbors (KNN)

K-Nearest Neighbors (KNN) is a supervised machine learning algorithm that is used for classification and regression problems. KNN is a non-parametric algorithm, which means it does not make any assumptions about the underlying data distribution. Instead, it uses the proximity of data points to make predictions. In KNN, the algorithm predicts the class or value of a new data point based on the classes or values of its neighboring data points. The "K" in KNN represents the number of neighboring data points, or "nearest neighbors," that the algorithm considers when making a prediction. The algorithm then assigns the new data point to the class or value that is most common among its K nearest neighbors.

The distance between data points is calculated using a distance metric, such as Euclidean distance or Manhattan distance. Euclidean distance is the most commonly used distance metric in KNN. It measures the straight-line distance between two points in n-dimensional space.KNN is a simple algorithm that can be applied to both classification and regression problems. For classification problems, the output of the algorithm is a class label. For regression problems, the output is a continuous value.

One of the key advantages of KNN is its simplicity and ease of implementation. KNN can also work well with non-linear data and can be effective in high-dimensional spaces. However, one of the main challenges of KNN is choosing the appropriate value of K. A small value of K can result in overfitting, while a large value of K can result in underfitting. Additionally, KNN can be computationally expensive, especially when working with large datasets.

Overall, KNN is a powerful and flexible algorithm that can be used in a variety of machine learning applications. It is especially useful when working with small to medium-sized datasets and can be a good starting point for exploring classification and regression problems.

- K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.
- K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.

- K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.
- K-NN is a **non-parametric algorithm**, which means it does not make any assumption on underlying data.
- It is also called a **lazy learner algorithm** because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.
- KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

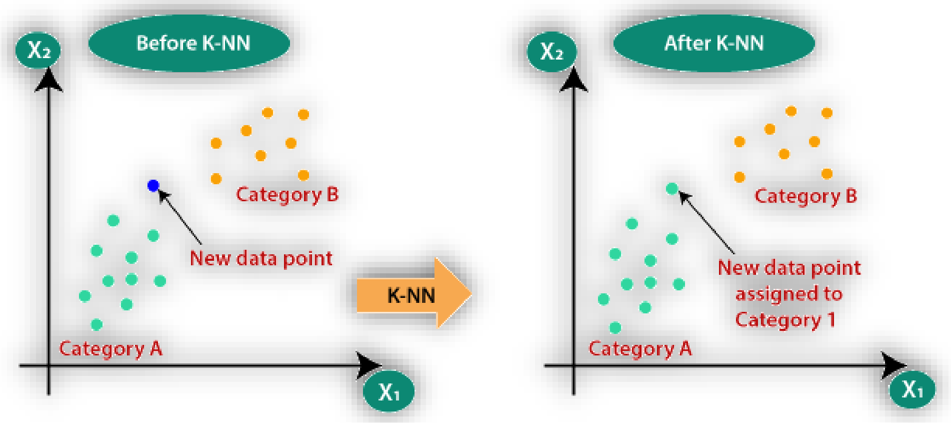


Fig4.2: KNN

RESULT : 73%

4.3 SUPPORT VECTOR MACHINE (SVM)

Support Vector Machines (SVM) is a supervised machine learning algorithm that is used for classification and regression problems. SVM is a discriminative algorithm, which means it learns a boundary that separates the data into different classes or groups. In SVM, the algorithm finds the hyperplane that maximally separates the classes or groups. The hyperplane is the line or boundary that separates the data into two different groups, and the algorithm tries to find the hyperplane that maximizes the margin between the two groups. The margin is the distance between the hyperplane and the closest data points from each group.

The data points that are closest to the hyperplane are called support vectors, and they are the ones that determine the location of the hyperplane. The goal of SVM is to find the hyperplane that separates the data with the largest margin, while minimizing the misclassification error.

SVM can be used for both linear and non-linear classification problems. In linear SVM, the data is linearly separable, and the hyperplane is a straight line that separates the data into two different groups. In non-linear SVM, the data is not linearly separable, and the algorithm uses a kernel function to transform the data into a higher-dimensional space, where it can be separated by a hyperplane. SVM is a powerful algorithm that can be used in a variety of machine learning applications. It works well with both linear and non-linear data, and it can handle high-dimensional data with a small number of samples. SVM can also handle imbalanced data, where the number of samples in each class is different.

However, one of the main challenges of SVM is choosing the appropriate kernel function and tuning the hyperparameters. The choice of kernel function can have a significant impact on the performance of the algorithm, and finding the optimal hyperparameters can be a time-consuming process.

Overall, SVM is a powerful and flexible algorithm that can be used in a variety of machine learning applications. It is especially useful when working with small to medium-sized datasets and can be a good starting point for exploring classification and regression problems.

- Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems.
- However, primarily, it is used for Classification problems in Machine Learning.
- The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.
- SVM chooses the extreme points/vectors that help in creating the hyperplane.
- These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.
- Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane.

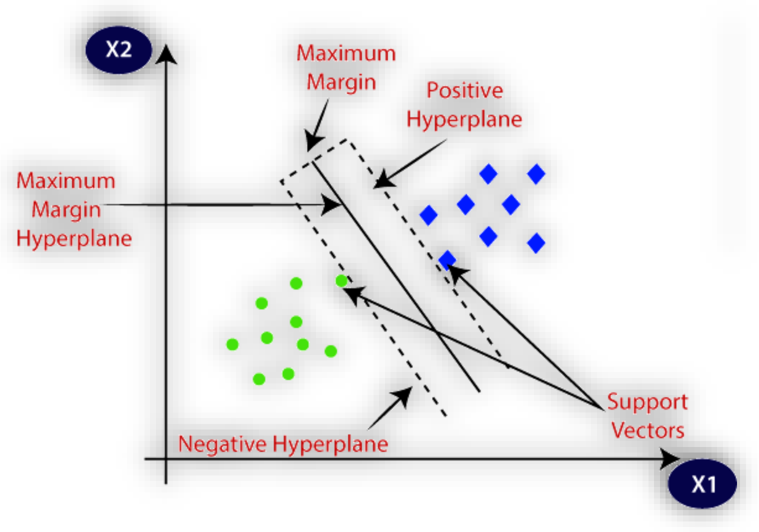


Fig4.3:SVM

RESULT : 93%

4.4 Naïve Bayes

Naive Bayes is a probabilistic machine learning algorithm that is used for classification problems. It is based on Bayes' theorem, which states that the probability of a hypothesis is updated based on new evidence. In the case of Naive Bayes, the hypothesis is the class or label of a new data point, and the evidence is the features or attributes of the data point.

Naive Bayes assumes that the features of a data point are conditionally independent given its class label. This means that the presence or absence of one feature does not affect the probability of the presence or absence of another feature. This assumption simplifies the computation of the conditional probability of the features given the class label. In Naive Bayes, the algorithm learns the probability distribution of the features for each class label in the training data. It then uses Bayes' theorem to calculate the posterior probability of each class label given the features of a new data point. The class label with the highest posterior probability is assigned to the new data point.

Naive Bayes can handle both discrete and continuous data, and it works well with high-dimensional data. It is also computationally efficient and can be trained quickly even with large datasets. However, the assumption of conditional independence may not always hold in real-world scenarios, and Naive Bayes may not perform well when the features are highly correlated.

Naive Bayes is widely used in natural language processing tasks such as text classification and spam filtering. It can also be applied to a variety of classification problems in different domains, such as image classification and medical diagnosis.

Overall, Naive Bayes is a simple and effective machine learning algorithm for classification problems. Its simplicity and efficiency make it a popular choice for a wide range of applications.

- Naïve Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems.
- It is mainly used in text classification that includes a high-dimensional training dataset.
- Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.

- It is a probabilistic classifier, which means it predicts on the basis of the probability of an object.
- Some popular examples of Naïve Bayes Algorithm are spam filtration, Sentimental analysis, and classifying articles.

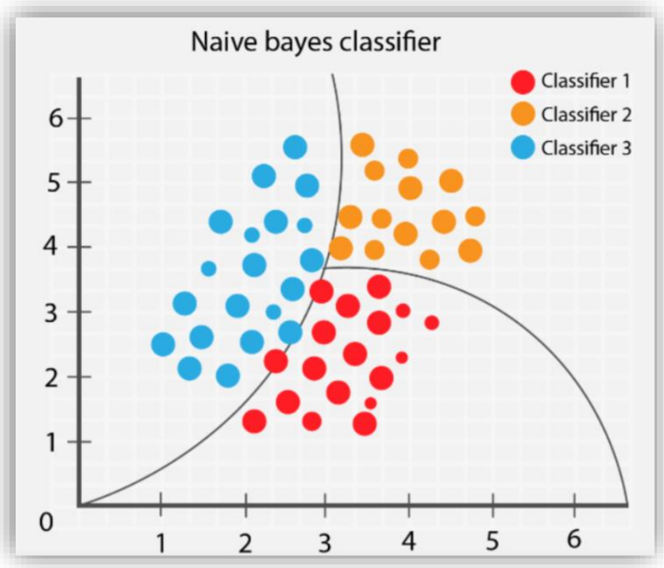


Fig4.4:Naïve Bayes

RESULT :82%

4.5 DECISION TREE

A decision tree is a supervised machine learning algorithm that is used for classification and regression problems. It is a tree-like model where each internal node represents a test on an attribute, each branch represents the outcome of the test, and each leaf node represents a class or a value.

In a decision tree, the algorithm recursively partitions the data into smaller subsets based on the values of the attributes. The goal is to create a tree that can predict the class or value of a new data point by following the path from the root to a leaf node. The decision tree algorithm uses a top-down approach to construct the tree. It starts with the entire dataset as the root node and selects the best attribute to split the data based on a certain criterion such as information gain or Gini impurity. The selected attribute becomes the node, and the data is partitioned into subsets based on the values of the attribute.

The process is repeated recursively for each subset until a stopping criterion is met, such as reaching a maximum depth or a minimum number of samples in a node. At each leaf node, the majority class or the mean value of the samples in the node is assigned as the prediction for new data points.

Decision trees have several advantages, such as being easy to understand and interpret, handling both categorical and continuous data, and being able to handle missing data. They are also robust to noise and outliers, and they can handle non-linear relationships between the features and the target variable.

However, decision trees can suffer from overfitting, where the model is too complex and fits the training data too closely, leading to poor generalization to new data. To overcome this issue, techniques such as pruning, setting a maximum depth, and using ensemble methods such as random forests can be used. Decision trees are widely used in different domains, such as finance, healthcare, and marketing. They can be used for classification problems such as predicting the risk of disease or fraud detection, as well as for regression problems such as predicting the price of a house or the demand for a product.

Overall, decision trees are a powerful and flexible machine learning algorithm that can handle both classification and regression problems. With careful tuning and validation, they can provide accurate and interpretable predictions for a wide range of applications.

- In a decision tree, which resembles a flowchart, an inner node represents a variable (or a feature) of the dataset, a tree branch indicates a decision rule, and every leaf node indicates the outcome of the specific decision.
- The first node from the top of a decision tree diagram is the root node. We can split up data based on the attribute values that correspond to the independent characteristics.
- The recursive partitioning method is for the division of a tree into distinct elements.
- Making decisions is aided by this decision tree's comprehensive structure, which looks like a flowchart.
- It offers a diagrammatic model that exactly mirrors how individuals reason and choose. Because of this property of the flowchart, decision trees are easy to understand and comprehend.

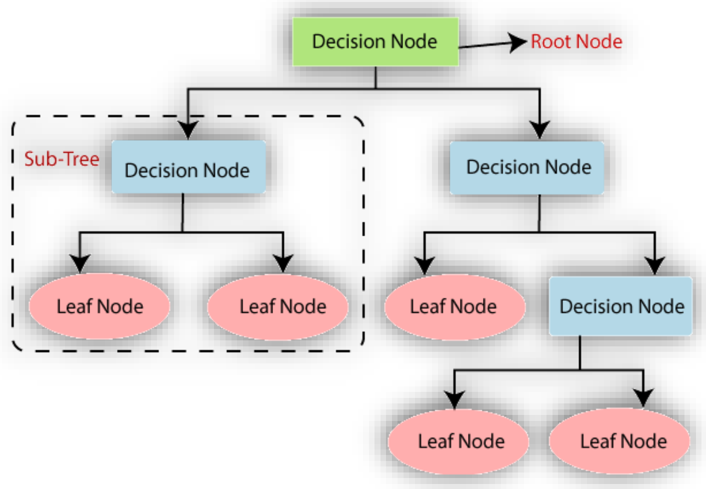


Fig4.5: Decision Tree

RESULT : 94%

4.6 RANDOM FOREST

Random forest is a popular machine learning algorithm that is used for classification and regression problems. It is an ensemble method that combines multiple decision trees to improve the accuracy and reduce overfitting. Random forest works by constructing a large number of decision trees and aggregating their predictions. Each tree is constructed using a random subset of the training data and a random subset of the features. This randomization helps to reduce the correlation between the trees and increase their diversity, which leads to better performance.

In random forest, the algorithm selects a random subset of the training data and features for each tree. It then constructs the tree using the selected data and features, using a similar process as the decision tree algorithm. The tree is grown until a stopping criterion is met, such as reaching a maximum depth or a minimum number of samples in a node.

The process is repeated to create a forest of decision trees, each with different subsets of the data and features. To make a prediction for a new data point, the algorithm aggregates the predictions of all the trees in the forest, and the majority class or the mean value of the predictions is assigned as the final prediction.

Random forest has several advantages over decision trees, such as being able to handle high-dimensional data and avoiding overfitting. It is also robust to noise and outliers and can provide feature importance measures to identify the most important features for the prediction. Random forest is widely used in different domains, such as finance, healthcare, and ecology. It can be used for classification problems such as predicting the sentiment of a review or the diagnosis of a disease, as well as for regression problems such as predicting the price of a stock or the yield of a crop.

Overall, random forest is a powerful and versatile machine learning algorithm that can provide accurate and robust predictions for a wide range of applications. Its ability to reduce overfitting and handle high-dimensional data makes it a popular choice in many domains.

- Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning**,** which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

- As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.
- The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.
- The below diagram explains the working of the Random Forest algorithm:

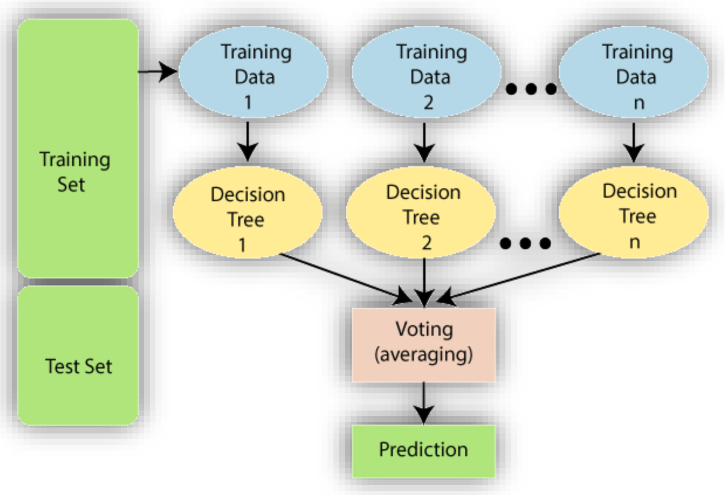
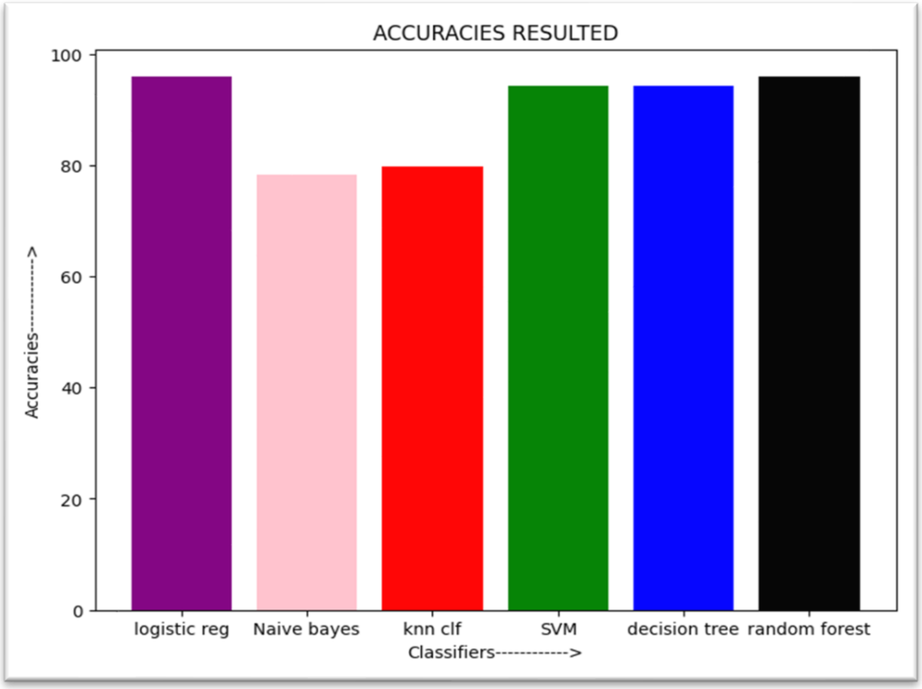


Fig4.6: Random Forest

RESULT: 97%

CHAPTER 6
RESULTS

Machine Learning Model	Accuaracy
Logistic Regression	76%
KNN	73%
Naïve Bayes	82%
Support Vector System	93%
Decision Tree	94%
Random Forest	97%



CHAPTER 6
CONCLUSION AND FUTURESCOPE:

Conclusion:

In conclusion, developing a reliable and efficient system for cotton plant disease detection can have significant benefits for cotton farmers. There are several existing solutions for cotton plant disease detection, including visual inspection, field-based diagnostics, remote sensing, and mobile applications. However, each method has its own advantages and limitations. Recent research has also explored the use of machine learning techniques, including image processing and classification algorithms, for cotton plant disease detection. A combination of these approaches may provide the most effective solution for detecting and managing cotton plant diseases. By accurately and efficiently identifying diseases affecting cotton plants, farmers can take necessary actions to prevent or control their spread, minimize crop losses, and optimize their yields. This can ultimately lead to a more sustainable and profitable cotton production system.

Overall, cotton plant disease detection is an important area of research and development in the field of agriculture. By leveraging existing technologies and exploring new approaches, researchers and farmers can work together to develop more effective and sustainable strategies for managing cotton plant diseases.

FUTURE SCOPE:

The future scope of cotton plant disease detection projects is promising, as there are still many challenges to be addressed and opportunities for innovation in this field. One potential area for future research and development is the integration of multiple detection methods. Combining different detection methods, such as visual inspection, field-based diagnostics, remote sensing, and machine learning algorithms, can provide a more comprehensive approach to cotton plant disease detection. This can improve the accuracy and efficiency of disease detection, as well as facilitate timely and appropriate management strategies. Additionally, further research can be conducted to explore the potential use of emerging technologies, such as nanosensors and bioinformatics, for cotton plant disease detection. These technologies can provide more sensitive and specific detection of disease pathogens and lead to the development of more targeted and effective management strategies. Another area for future research is the development of disease-resistant cultivars through genetic engineering and selective breeding. Developing cotton cultivars that are resistant to specific diseases can reduce the need for chemical controls and minimize crop losses due to disease. This can contribute to a more sustainable and environmentally friendly approach to cotton production. Finally, collaboration between researchers, farmers, and industry stakeholders can facilitate the development and implementation of more effective and practical cotton plant disease detection and management strategies.

CHAPTER 7
REFERENCES

[1] S. Kumar, R. Ratan, and J. V. Desai, “A study of iOS machine learning and artificial intelligence frameworks and libraries for cotton plant disease detection,” Machine Learning, Advances in Computing, Renewable Energy and Communication, Springer, Singapore, 2022, https://doi.org/10.1007/978-981-16-2354-7_24 Lecture Notes in Electrical Engineering, vol 768.

[2] V. Pooja, R. Das, and V. Kanchana, “Identification of plant leaf diseases using image processing techniques,” in Proceedings of the Technological Innovations in ICT for Agriculture and Rural Development (TIAR), pp. 130–133, IEEE, Chennai, India, April 2017.

[3] K. Elangovan and S. Nalini, “Plant disease classification using image segmentation and SVM techniques,” International Journal of Computational Intelligence Research, vol. 13, no. 7, pp. 1821–1828, 2017.

[4] V. Singh and A. K. Misra, “Detection of plant leaf diseases using image segmentation and Soft Computing techniques,” Information Processing in Agriculture, vol. 4, no. 1, pp. 41–49, 2017.

[5] X. E. Pantazi, D. Moshou, and A. A. Tamouridou, “Automated leaf disease detection in different crop species through image features analysis and one class classifiers,” Computers and Electronics in Agriculture, vol. 156, pp. 96–104, 2019.

[6] Al-bayati, J. S. H., & Üstündağ, B. B. (2020) “Evolutionary Feature Optimization for Plant Leaf Disease

[7] Detection by Deep Neural Networks”, International Journal of Computational Intelligence Systems.

[8] Nikhil Shah, Sarika Jain, “Detection of disease in Cotton leaf using Artificial Neural Network”, 2019 IEEE

[9] S. Kaur , S. Pandey, S. Goel"Semi-automatic leaf disease detection and classification system for soybean culture",The Institution of Engineering and Technology 2018.

[10] A. Jenifa, R. Ramalakshmi, V. Ramachandran, “Classification of cotton leaf Disease using Multi-support Vector Machine”, 2019 IEEE.