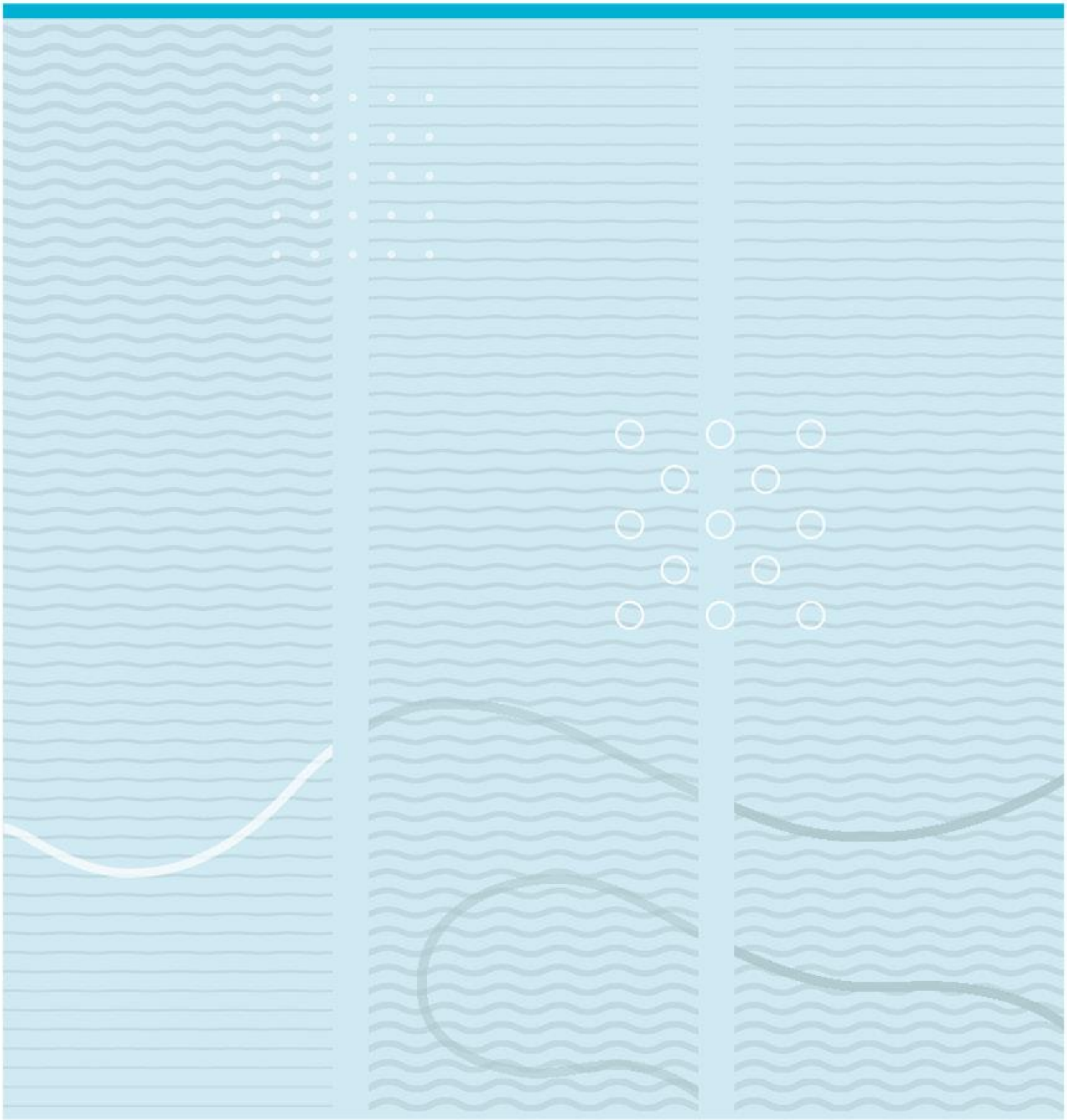
University of South-Eastern Norway
Faculty of Technology, Natural Sciences and Maritime Science
–
CS4020: Data Science
Study programme: MSCS
Spring 2023

8500
8511

Crop Prediction and Analysis using Machine Learning



University of South-Eastern Norway
Faculty of Technology, Natural Sciences and Maritime Sciences
Institute of Science and Industry Systems
PO Box 235
NO-3603 Kongsberg, Norway

http://www.usn.no

This project is worth 10 study points.

Abstract

Machine learning has the potential to have a substantial impact on the sector of agriculture. One of the key benefits of machine learning in agriculture is its ability to analyze large amounts of data quickly and accurately. Machine learning can help farmers make better decisions, maximize crop yields, reduce waste, and increase efficiency and sustainability by utilizing data and algorithms.

crop recommendation using machine learning is a promising technology for the agriculture industry. It has the potential to revolutionize the way farmers decide what crops to grow, leading to increased efficiency and sustainability in agriculture.

The aim of this report is to use soil and weather conditions to predict multiple crop outcomes. Due to the growing population and limited agricultural land, it has become crucial to determine the most suitable crops for a particular location based on prevailing weather and soil conditions. By implementing this system, farmers can efficiently select the crops that would yield optimal results based on their specific environmental conditions. This would allow them to make informed decisions and optimize their crop selection for a better harvest.

Keywords: crop prediction, weather data, soil data, Supervised learning, machine learning algorithms, Fi-score, cross validation, accuracy validation.

Preface

At the University of South Eastern Norway (USN), a campus of Kongsberg, we have been exploring crop prediction and analysis using machine learning for our project on the topic of data science (CS4020).

Having some familiarity with machine learning techniques and data analysis from past and current courses. We choose to use machine learning algorithms to anticipate many crops given the soil and weather conditions. Next, compare the algorithms to determine which one is optimal for our system.

This project offers a thorough overview of the most recent advances in machine learning-based crop prediction and analysis. We look at various machine-learning techniques and algorithms that have been created and used in the context of crop analysis and prediction. We also look at the opportunities and problems that come with applying machine learning in this area and talk about possible solutions.

Our goal is to offer a resource that will be beneficial to agricultural scholars, practitioners, and farmers. We anticipate that the data and analysis offered in this research will increase our knowledge of crop forecasting and analysis and aid in the creation of more productive and environmentally friendly farming methods.

Initially, we underestimate to process to predict more than one crop with accuracy as not all of the machine learning models support these top three accuracy predictions, and these made the process of deployment a little challenging. Despite the fact that we did not complete all of our objectives due to time constraints, we are pleased with the outcome.

List of abbrevations

ML - Machine learning

SVM - Support vector machine

KNN - K-Nearest Neighbour

N - Nitrogen

K - Potassium

P - Phosphorus

GUI - Graphical User Interface

List of Figures

List of Tables

Contents

1 Introduction

Agriculture has long been regarded as the primary source of supplies for meeting people's basic requirements. Still, now farmers are using conventional naked-eye inspection and yield healthy crops without the use of chemicals for animals and agricultural areas in order to maintain healthy diversity. But in today's world, the weather is changing quickly in opposition to natural resources, reducing the availability of food and boosting security. The agriculture sector's GDP has been steadily declining since 2005 when it was around 17.2%; it was 11.1% in 2012; 5% in 2018; and 2% in the first quarter of 2019-2020. Over 80% of farmers are located in rural areas, and if crop production revenues decline, farms on an industry level will have an impact on their way of life (Kalimuthu, Vaishnavi, & Kishore, 2020). So, now it is time to find the solution to develop a method for farmers to forecast the best crop for a given input parameter. In this agricultural productivity area, employing machine learning is one of the most effective options.

1.1 Motivation and Problem Statement

Agriculture is a critical sector that directly impacts food security and economic growth. Crop production plays a vital role in ensuring food security for the ever-growing population. Farmers should take a special interest in efficient and precise farming under certain environmental conditions; thus, this makes sense. The timely and accurate prediction of crop yield can help farmers optimize their inputs, plan their harvest, and minimize post-harvest losses. If farmers cultivate crops that are not well-suited to the local climate or soil conditions, it can result in low crop production, which can have a significant impact on the farmer's income and the waste of natural resources. Crop prediction is a complex problem that involves many dependent variables such as soil type, weather, crop variety, and management practices. Machine learning (ML) algorithms have shown great potential in predicting crops depending on climate and soil type. Because of this, creating a crop prediction model utilizing machine learning might be extremely advantageous for farmers and the agricultural sector.

Farmers' use of conventional and non-scientific methods to choose the crop best suited to their soil is a big problem. On the basis of soil characteristics, climate, and geographic location, farmers occasionally failed to select the right crops. As a result, it may have negative consequences such as decreased crop yields, increased pests and disease, increased input costs, and lower market value. On the other hand, identifying a single crop to cultivate in a particular area is insufficient. Planting a single type of crop by all farmers in a specific region with similar soil and weather conditions can result in negative impacts on both the ecology and the economy. There is a need for an accurate and timely crop prediction model using ML to help farmers make informed decisions and improve the efficiency of the agricultural supply chain and ecological system.

1.2 Project Objective

The objective of our project is to create a system that can predict the top three crops that will grow best in a specific environment based on a variety of environmental characteristics, including soil type, weather, temperature, humidity, and precipitation. In consideration of market rivalry and the highest level of demand for a particular crop variety, farmers will thus be able to select the best option for their land as well as for themselves.

A graphical user interface (GUI) will also be included in our project and our system's graphical user interface (GUI) will let farmers enter environmental data and get suggestions for the crops that would be most profitable and suitable for their specific situation. The model will also have the capacity to learn from user inputs and improve over time based on how the farmers utilize it. Additionally, this initiative will also display an image of the best match crop so that farmers who are illiterate or functionally illiterate can readily grasp it.

1.3 Assumptions and Limitations

In the dataset we use from Kaggle there are twenty-two types of crops in the label sections and from our, training model we can accurately predict the top three outputs in the user interface. We can also preserve the generated data from the output for usage in the future. The machine learning methods that are appropriate for multi-class classification and provide us with the top three outputs in the user interface have a wide range of alternatives. As a result, we need to identify the accuracy model that will produce more than one output for the given inputs.

The time scope limits what we are able to achieve, especially making our output available in the cloud and collecting datasets are really complex and resource intensive. We collect a real dataset from Kaggle that has twenty-two types of crops. As a result, our machine learning models have limitations to predict the best-matched crops from those twenty-two types of crops.

1.4 Project Contributions

A comprehensive strategy combining knowledge from machine learning algorithms, training data, data evaluation, and the user interface is required due to the complexity of the crop prediction and analysis system employing machine learning. The development of our crop prediction and analysis system has done extensive use of pair programming and training data on different machines. Collaboration, knowledge sharing, and ongoing feedback were made possible through pair programming, which led to higher-quality code with fewer errors. Machine learning models were trained on a variety of computers to assure system accuracy and reliability. This process also helped to discover and fix performance problems brought on by different hardware or operating systems, making the system more durable and portable.

2 Theoretical background

Modern agriculture cannot function without crop prediction systems because they give farmers invaluable information about crop yields and harvest dates. These systems provide precise forecasts about the anticipated crop productivity in a specific field using a range of data sources, including weather data, and soil data. Despite numerous solutions that have been recently recommended, there are still open challenges in creating a user-friendly interface with respect to predicting the best-diversified crops for farming in specific weather and soil. Crop prediction systems based on machine learning algorithms have the potential to transform agriculture by giving farmers data-driven insights regarding the best crops to cultivate and how to maximize yields. In the agriculture sector, this can lead to increased production, sustainability, profitability, and reduced waste.

2.1 Supervised Machine Learning Algorithms

A type of machine learning called supervised learning uses labeled data to teach the algorithm how to predict or categorize new, unlabeled data. In our dataset, we have labeled data and we want to predict the crop's name as an output variable based on a given set of environmental and soil conditions.

Because more than one class can be assigned to a single instance in our model, multi-class classification is the best option. A type of supervised learning called multi-class classification aims to categorize input data into one of the multiple distinct categories. So, we use supervised learning for our multi-class classification.

Because more than one class can be assigned to a single instance in our model, multi-class classification is the best option. A type of supervised learning called multi-class classification aims to categorize input data into one of the multiple distinct categories. So, we use supervised learning for our multi-class classification.

There are several multi-class classification algorithms. For our project, we use six machine learning algorithms Decision Tree, Logistic Regression, Random Forest Classifier, Support Vector Machine(SVM), Gaussian Naive Bayes, and K nearest Neighbour (KNN) to have in-built support for MLC. These algorithms operate by learning a function that associates the input features with the desired output.

2.1.1 Decision Tree

A Decision Tree is a supervised ML algorithm that is used for both classification and regression. In a decision tree, a tree-like model is created where attributes and class labels are represented using a tree. Here, the record's attribute is compared to the root attribute, and based on the result of the comparison, a new node is then reached. Until a leaf node with a predicted class value is reached, this comparison is continued. A modeled decision tree is therefore quite effective for making predictions. This procedure continues until a stopping criterion is satisfied, such as when all the data in a subset belong to the same class or when the maximum tree depth is reached (Doshi, Nadkarni, Agrawal, & Shah, 2018).

2.1.2 Logistic Regression

The Logistic Regression model is a widely used statistical model that uses a logistic function to represent a binary dependent variable in its basic form based on one or more predictor variables. In regression analysis, logistic regression, a subset of binomial regression, predict the parameters of a logistic model. These algorithms operate by learning a function that associates input data with the output variable. The calculated coefficients are used to determine the outcome's log odds, which are then converted into probabilities using the logistic function in the logistic regression model to make predictions. New observations can be categorized into one of the binary outcomes using the derived probability estimates (Gosai, Raval, Nayak, Jayswal, & Patel, 2021).

2.1.3 Random Forest Classifier

A supervised learning algorithm is Random Forest we use for a more precise and reliable forecast, random forest constructs many decision trees and combines them. This approach allows us to incorporate unpredictability into our model. Before splitting any node, Random Forest looks for the parameter that is the most crucial among all of them, and then it seeks for the best among the subset of random features. This eventually results in a model that is more accurate across a wider diversity. The splitting of a node is taken into consideration. Instead of looking for the best possible thresholds, random thresholds for the feature set can be used to increase the randomness of the trees. The generic bootstrap aggregation technique is used by the random forest training algorithm (Keerthan Kumar, Shubha, , & Sushma, 2019).

2.1.4 Support Vector Machine

A supervised machine learning technique or model called the Support Vector Machine (SVM) can be applied to classification and regression problems. However, we mainly apply it to categorization problems. SVM is typically represented as training data points in space that are divided into groups by an as-far-as-possible coherent gap. Each data item is displayed as a point in n-dimensional space using the SVM method, and each feature value corresponds to a particular coordinate. The categorization is then carried out by identifying the hyper-plane that effectively differentiates the two classes (Gosai et al., 2021).

2.1.5 Gaussian Naive Bayes

An approach for categorizing problems with binary and multiple classes is called Naive Bayes. The Naive Bayes approach is fairly simple to comprehend when input data are presented in binary or categorical form. A Naive Bayes classifier considers that the existence of one feature in a class has absolutely nothing to do with the presence of any other feature. Thus, "Naive Bayes" was defined. The Naive Bayes classifier, which is based on the Bayes hypothesis, is a valuable tool when the sources of information are highly dimensional. Naive Bayes has many uses, including real-time prediction, predicting the likelihood of various target attribute classes, spam filtering, and assisting in the development of recommendation systems when combined with collaborative filtering (Kulkarni, Srinivasan, Sagar, & Cauvery, 2018).

2.2 Tkinter

A graphical user interface (GUI) toolkit is provided by the standard Python package Tkinter for building desktop applications. The Tk GUI toolkit created for the Tcl programming language serves as its foundation. We use Tkinter to build GUIs using widgets like buttons, labels, fields, and more using Tkinter. It offers a straightforward and user-friendly interface for building frames, windows, and dialog boxes. Tkinter also enables event-driven programming, enabling programmers to create interactive and responsive programs (Lundh, 1999).

2.3 Scikit-learn

Scikit-learn commonly abbreviated as sklearn. Scikit-learn is a well-known Python machine-learning toolkit that offers a variety of tools for supervised learning, including classification, regression, and clustering algorithms. Additionally, it offers resources for choosing features, preparing data, and evaluating models.

2.4 Pandas

Pandas, a Python library of rich data structures and methods, works with structured data sets common in statistics, economics, social sciences, and many other fields. The library provides integrated, straightforward procedures for performing common data transformations and analysis on such data sets (Mckinney, 2011).

2.5 Matplotlib

The majority of scientific journals use figures to describe their findings. When it comes to creating visuals, Matplotlib is one of the best solutions. Python comes with a 2D graphics package called Matplotlib to produce images of publishing quality across user interfaces and operating systems, interactive scripting, and application development (Hunter, 2007).

3 Literature review

Numerous papers have been written about crop prediction with various technologies. Such as this paper (Rajak et al., 2017) states the precision in agriculture using data mining. The authors begin with the fundamentals of precision farming and then proceed to create a model that would support it. In this paper, lab tested soil dataset has been used for the data mining model. This paper describes a data mining model Ensemble technique which is known as the model combiner that combines more than one model to make a better prediction. For the learner's model, Support Vector Machine, NAÏVE Bayes, Random Forest, and Multi-layer Perceptron (Artificial Neural Network) have been used in this paper.

This paper (Bandara et al., 2020) presents a theoretical and conceptual framework for a recommendation system that uses integrated models to gather environmental factors using Arduino microcontrollers, machine learning methods like Naive Bayes (Multinomial), and Support Vector Machine (SVM), unsupervised machine learning algorithms like K-Means Clustering, and Natural Language Processing (Sentiment Analysis) to recommend a crop. The comprehensive objective of the model is that once the environmental factors are entered. PLU code is an attribute that is used to select the crop type to be cultivated.

This paper (Gosai et al., 2021) aims to recommend the most suitable crop based on input parameters like Nitrogen (N), Phosphorous (P), Potassium (K), PH value of soil, Humidity, Temperature, and Rainfall. This paper predicts the accuracy of the future production of eleven different crops using various supervised machine-learning approaches and recommends the most suitable crop.

In this research paper, (Doshi et al., 2018), an intelligent system called AgroConsultant was constructed. The two sub-systems that comprise the proposed system are the crop-suitability predictor and the rainfall predictor. This system's working conditions include five major crops, fifteen minor crops, and qualities including soil type, aquifer depth, soil PH, topsoil thickness, precipitation, temperature, and location parameters. Several machine learning algorithms, including Decision Tree, K Nearest Neighbor (K-NN), Random Forest, and Neural Network, have been implemented in this suggested system and multi-label classification has been carried out on it. Our proposed system achieved 71% accuracy when using a rainfall prediction model and 91% accuracy when utilizing a neural network algorithm on a crop-appropriate predictor system.

This paper (Kulkarni et al., 2018), named "Improving Crop Production Through A Crop Recommendation System Using Ensembling Method," developed a system that is used to accurately select the best crop depending on the kind and features of the soil, such as the average rainfall and surface temperature. The machine learning algorithms used by this suggested system included Linear SVM, Random Forest, and Naive Bayes. The input soil dataset was classified by this crop recommendation algorithm into the recommended crop types. Using the suggested approach produced a 99.91% accuracy rate.

Admittedly, there are some noteworthy literature reviews on this issue but most of them are focused on predicting the most suitable crop for a given environment and soil conditions. But from the aforementioned that there is not much research that focuses on more than one crop prediction with the crop cultivation accuracy rate for that specific conditions with a user-friendly interface with images. And also we want to save generated data when every time a user will use the system for future use. So, that we can train our model including the new data in our existing dataset to increase the accuracy rate and more precise detection.

4 Research Methodology

4.1 Data Collection and Data Preparation

For our project, we collect the "Crop Recommendation Dataset" from Kaggle which consists of soil and climate details from previously accessible data for India (*Crop Recommendation Dataset — kaggle.com*, n.d.). As the main focus of our project has been to deploy different models to find the model which gives us the best accuracy and stores the generated results. As a result, we can add more data to our existing dataset.
In our dataset, there are 22000 rows and 8 columns consisting of N (Nitrogen), P (Phosphorus), K (potassium), temperature, humidity, Ph, rainfall, and label. After selecting the dataset we start to prepare our dataset. At first, we find out there are 22 types of crops in our label columns. So, we can predict 22 types of crops from the other dependent variables from our dataset. We then start collecting the images for our 22 types of crops. We not only predict which top three crops are suitable but also show the best-suited crop's image. We illustrate the heatmap of our dataset in Appendix A.

We divided the dataset into training and testing sets using a 75:25 ratio in order to assess the effectiveness of our machine learning model. In further detail, 75 percent of the data were randomly chosen for the training set, while 25 percent were utilized for testing. The split was carried out to guarantee that the model was trained on a representative portion of the data and tested on unseen data to determine its potential for generalization. In order to preserve the class distribution in both the training and testing sets, we further ensured that the split was stratified. This was done to avoid bias against any one class during training and testing.

4.2 System Architecture

Figure 4.1 represents the system architecture of our project. This architecture represents all the platforms, models, and datasets that we used to complete our project and how they interact with each other. The architecture starts with a data collection process from Kaggle, an online open-source platform, and went through selecting machine learning algorithms, training models, prediction crops, model deployment, and finally finished at Data Visualisation GUI where we can show our crop prediction output.

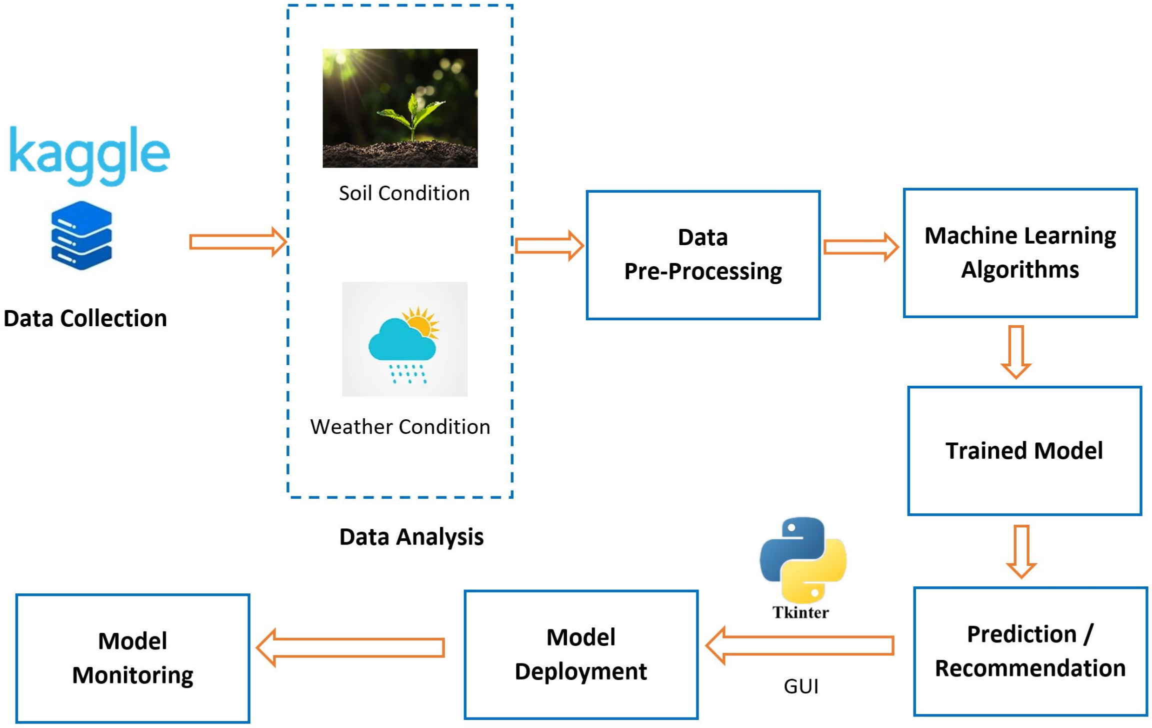


Figure 4.1: Crop Prediction and Analysis System architecture

4.3 Model Selection and Training

This section describes how we train our models and compare their accuracy to determine the one that is best suited to our project. We have conducted some research on various machine learning model types. We have chosen a handful of them to put to the test. Given that, Decision Tree is one of the most often used models, we have chosen to start with it before moving on to Logistic Regression, Random Forest Classifier, Support Vector Machine (SVM), Guassian Naive Bayes, and K-Nearest Neighbour. (KNN). Additionally, we keep track of all the models and their precision for comparison.

4.3.1 Decision Tree

To create our Decision Tree model, we use the Decision Tree Classifier from Scikit Learn. We conducted a number of tests for the random state before settling on 11 because of its better results in comparison. The model is then fitted, and predictions are made for the test instance. The Decision Tree model's accuracy is 92.7 percent. We also compare the precision, recall, f1-score, and support for each individual target label in the classification report. Additionally, we test the model using cross validation.

4.3.2 Logistic Regression

Similar to the last one, we construct our model using Scikit Learn Logistic Regression and set the random state to 11. Cross validation was performed after fitting and predicting the outcome, and the model's accuracy was 94.7 percent.

4.3.3 Random Forest Classifier

In this model, we must determine the value of the n-estimators. To accomplish this, we select several values and present a graph to determine the optimal n-estimators value. We have seen that the model performed admirably when the value was set at 8. After running the prediction, we found that this model has a 98.7 percent accuracy rate.

4.3.4 Support Vector Machine (SVM)

In SVM, the degree is a hyperparameter that is used to control the complexity of the decision boundary while balancing overfitting and underfitting. To optimize the performance of the SVM model, it is one of many hyperparameters that must be carefully adjusted. We do some tests to determine the best degree value for this model, and the value 3 produces the best results. Following the prediction, we had a 97.4 percent accuracy rate.

4.3.5 Guassian Naive Bayes

For this model, we just build it and estimate the outcome. Before storing the model, we do cross validation and achieve an accuracy of 98.5 percent.

4.3.6 K-Nearest Neighbour

To get the best performance out of this model, we must find the ideal value for k. The accuracy of the model and the complexity of the decision boundary are influenced by the choice of k in the KNN, which is crucial. A decision boundary that is smoother will have a bigger k value which might lead to underfitting. Conversely, a decision boundary that is more complex could lead to overfitting with a smaller k value. A graphical presentation is used to show that the error rate is lowest when k equals 5. We then create our model after setting the value. With this model, we achieve an accuracy of 97.4 percent.
The accuracy of all models is illustrated in Appendix C.

4.3.7 Model Comparison

To choose the ideal machine learning model for our issue, we thoroughly compared a range of models. Decision Tree, Logistic Regression, Random Forest Classifier, Support Vector Machine (SVM), Guassian Naive Bayes, K-Nearest Neighbour (KNN) were among the models that we examined. Using a variety of performance indicators, including recall, accuracy, precision, and F1-score, we analyzed these models. We discovered that Random Forest Classifier outperformed the other models in terms of accuracy and F1-score while maintaining parity in precision and recall based on our study. Therefore, we decided that Random Forest Classifier was the most appropriate model for our project. We demonstrate a graphical representation of the comparison in one frame, with Random Forest Classifier coming out on top. Furthermore, We ran additional tests to confirm the reliability and robustness of our results shown in Appendix D. We evaluated the model on a hold-out dataset and cross-validated it, which demonstrated Random Forest Classifier's superiority over other models. We can see the comparison of all models in table 4.1.

Model Name	Accuracy	Precision	Recall	F1-Score
Decision Tree	0.92	0.95	0.93	0.93
Logistic Regression	0.94	0.95	0.95	0.95
Random Forest Classifier	**0.98**	**0.99**	**0.99**	**0.99**
Support Vector Machine	0.97	0.98	0.97	0.98
Guassian Naive Bayes	0.92	0.95	0.93	0.93
K-Nearest Neighbour	0.97	0.98	0.97	0.97

Table 4.1: Model Comparison Table

4.4 Model Deployment

Using Python's Tkinter toolkit, we localized our crop prediction ML model and built a graphical user interface (GUI) for user interaction. A visual representation of our user interface has shown in Appendix E. The GUI allowed users to enter information about the instances, including the type of soil and the climate, and receive recommendations for the best crop to cultivate. The Tkinter package included built-in widgets for data entry and output display, making it simple to construct a user-friendly interface. The deployed model is easily available to farmers and landowners because it may be used on any local machine that has Python installed. Overall, the deployment of our model using Tkinter proved to be a straightforward and efficient method of making it usable by end users.

4.5 Model Monitoring

Following model deployment, we conducted various tests to evaluate its performance and accuracy. Our model has been able to accurately predict the target crops based on the inputs received. We also conducted tests to mitigate any biases that may exist in the data or model architecture.

Furthermore, our model can handle Value Exception Errors, which enhances its robustness and reliability. To ensure that our model remains up-to-date, we plan to continuously train it using new inputs from the users. This will involve monitoring for changes in data distribution and making necessary adjustments to the model architecture.

4.6 Flow diagram of the Crop Prediction and Analysis System

Figure 4.2 represents the full flow diagram of our project of the aforementioned steps.

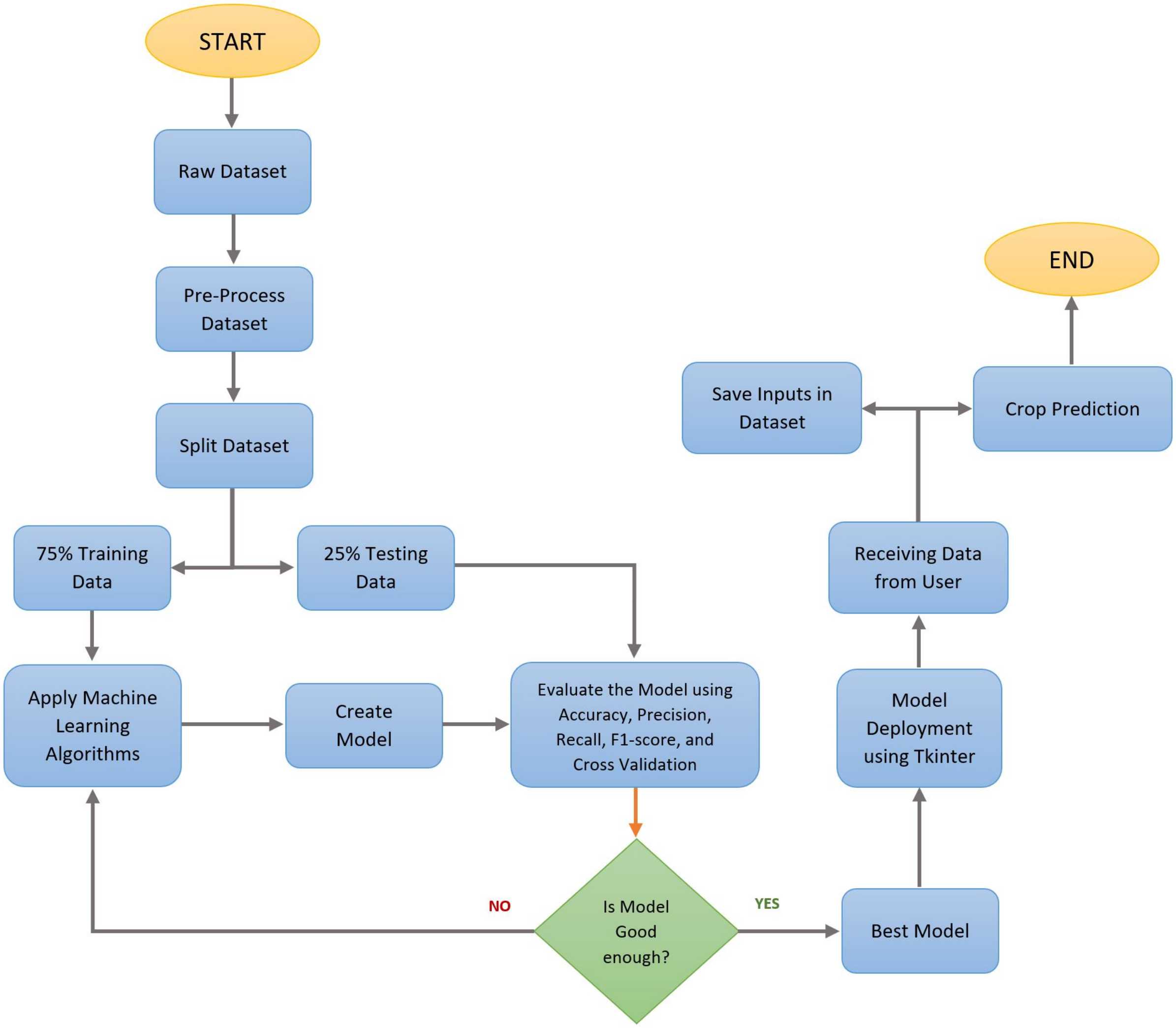


Figure 4.2: Flow Diagram of the Crop Prediction and Analysis System

5 Experiments and Results

In this project, we design a crop prediction model that can accept inputs from the user using a graphical user interface (GUI) and forecasts the best match crop to grow on the characteristics of the area. Additionally, it can display the top three crops that can be cultivated on that plot of land along with a percentage recommendation. To construct this platform, we combine Python Tkinter with the Random Forest Classifier model. Two screenshots of its usage are shown in Appendix F.

The attributes of the land, such as the ratio of phosphorus, nitrogen, and potassium, soil quality, PH level, water availability from rainfall, humidity, and temperature, are among the data utilized in this project. These properties are provided by a dataset from Kaggle and the user through GUI. We utilize the Random Forest Classifier approach to train the model, which is selected based on its capacity to handle high-dimensional data and its good performance in classification tasks. The dataset is split into training and testing sets with a 75:25 ratio.

Using the testing data, the trained model's performance has been evaluated. The algorithm has a 98.7 percent success rate in predicting which crop will do best on the given plot of land. The precision, recall, and F1-score are all outstanding in terms of our goal, and this information is summarized in table 5.1.

Evaluation Metrics	Value
Accuracy	0.98
Precision	0.99
Recall	0.99
F1-Score	0.99

Table 5.1: Example table with label and caption

6 Discussion

The project is about the development of a machine-learning model using Random Forest Classifier to predict the best crop to grow based on user inputs. The ML model comes equipped with a user-friendly graphical interface that permits users to enter details about their land. The model can handle Value Exception Errors with ease and can offer up to three crop predictions, thus demonstrating its resilience.

The performance of the model suggests that it might be a helpful tool for farmers and landowners when deciding which crops to produce on their properties. As the model can forecast the best crop with an accuracy of 98.7 percent so, this is a promising outcome. It is clear from the model's excellent precision, recall, and F1-score that it can generate reliable predictions for both positive and negative labels. Moreover, the model is designed to save user input for training purposes, allowing it to improve its accuracy over time. We plan to further enhance the model's accuracy by incorporating image processing techniques.

Furthermore, the project stands out from similar ones due to its exceptional characteristics, which include predicting the cultivation of up to three different crops, showing crop photographs on the screen, and self-learning technology for improvement. Receiving location also opens up a lot of useful opportunities for future employment.

Overall, the developed model provides a user-friendly interface for predicting crop types based on land characteristics. The combination of machine learning and image processing technologies can improve the accuracy of the model, providing more accurate and efficient crop recommendations. Even though we intended to use a self-learning method in our project to continuously update our model, due to time restrictions, we were unable to carry out the required assessments to confirm its efficacy. As a result, we view this as a potential challenge to be addressed in the future in order to strengthen the reliability and precision of our model. In addition, future work may include expanding the dataset to include more target crop labels and training the model with new data inputs to keep it up-to-date. Model monitoring and bias detection techniques may also be implemented to ensure its continued accuracy and reliability.

7 Conclusion and Future Work

7.1 Conclusion

Agricultural data is a highly valuable resource in the contemporary world, but its true value can only be realized when it is effectively utilized to enhance agricultural productivity and sustainability. The data generated by farming activities, such as weather patterns, soil characteristics, and crop yields, can be leveraged to develop insights that can inform decision-making processes related to crop selection, land management, and resource utilization.

In section 4.1, we mentioned the dataset we used to achieve our goal. We applied machine learning algorithms to predict the top three crops for farming to farmers in a given environmental parameter. From there we selected a random forest classifier to train the dataset which gives us the best accuracy. Our developed crop prediction system has the potential to enhance agricultural productivity, mitigate soil degradation in cultivated areas, and decrease monoculture farming practices. Also, our interface shows the image of the best match crop.

In order to improve the accuracy of our train model, we save the input and output data generated by the model in an Excel file. By doing so, we can merge this new data with the existing dataset used to train the model. This allows us to expand the size of our dataset, providing more examples for the model to learn from, which can ultimately improve its accuracy. Additionally, it enables us to keep track of the performance of the model over time and evaluate how well it is adapting to new data. By regularly updating and retraining our machine learning models with new data, we can ensure that they remain effective and accurate in their predictions.

By adopting the recommendations generated by our system, farmers can make more informed decisions about crop selection that are tailored to their specific location and environmental conditions. This can lead to a reduction in soil erosion and other forms of environmental damage caused by unsustainable agricultural practices. Furthermore, our approach is designed to promote the efficient use of natural resources by suggesting crops that are well-suited to the prevailing climate and soil conditions, thereby reducing the need for excessive use of water and fertilizers. By improving agricultural productivity and sustainability, our work can contribute to the growth of the agricultural economy and the welfare of farmers

7.2 Future Work

Our model has a self-learning mechanism that can adapt to user input over time. The model may continuously enhance its suggestions and deliver more precise forecasts by utilizing user feedback. To accomplish this, we plan to investigate strategies like active learning and reinforcement learning.

For prediction purposes, our dataset is currently restricted to a particular group of crops. To make our model more adaptable and suitable to a larger range of crops, we intend to expand the number of target crops in our dataset in the period ahead. In order to find the most relevant characteristics for prediction, this will entail gathering data on more crops and using feature engineering. Additionally, to enhance the performance of our model on these new crops, we intend to investigate strategies like data augmentation and transfer learning.

Implementing crop suggestions on a map-based interface, which can offer site-specific recommendations based on geographic information, is another area that could use development. This method can be helpful for large-scale farming operations and can assist farmers in selecting crops with greater knowledge. To develop this functionality, we intend to investigate the utilization of geospatial data and GIS technologies.

Moreover, we plan to incorporate land image processing techniques to improve the precision of our crop prediction model. We can extract significant features like soil characteristics, crop index, and topography of the land by combining remote sensing and aerial imagery data. These features can be used as extra inputs to our present crop prediction algorithm to increase its accuracy and improve crop yield prediction. Additionally, in order to extract more intricate features from these images, we intend to investigate deep learning methods like Convolutional Neural Networks (CNNs).

References

Bandara, P., Weerasooriya, T., Ruchirawya, T., Nanayakkara, W., Dimantha, M., & Pabasara, M. (2020). Crop recommendation system. *International Journal of Computer Applications*, *175*(22), 22–25.

Crop Recommendation Dataset — kaggle.com. (n.d.). `https://www.kaggle.com/datasets/siddharthss/crop-recommendation-dataset`. ([Accessed 24-Mar-2023])

Doshi, Z., Nadkarni, S., Agrawal, R., & Shah, N. (2018). Agroconsultant: Intelligent crop recommendation system using machine learning algorithms. In *2018 fourth international conference on computing communication control and automation (iccubea)* (p. 1-6). doi: 10.1109/ICCUBEA.2018.8697349

Fumo, J. (2017, March). *Linear regression — intro to machine learning #6 - simple AI - medium.* `https://medium.com/simple-ai/linear-regression-intro-to-machine-learning-6-6e320dbdaf06`. (Accessed: 2023-3-25)

Gosai, D., Raval, C., Nayak, R., Jayswal, H., & Patel, A. (2021). Crop recommendation system using machine learning. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 558–569.

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing in Science Engineering*, *9*(3), 90-95. doi: 10.1109/MCSE.2007.55

Ilca, L. F., & Balan, T. (2021). Purple team security assessment of firmware vulnerabilities. In M. E. Auer, K. R. Bhimavaram, & X. Yue (Eds.), *Online engineering and society 4.0 - proceedings of the 18th international conference on remote engineering and virtual instrumentation, REV 2021, hongkong, china, 24-26 february 2021* (Vol. 298, pp. 370–379). Springer. Retrieved from `https://doi.org/10.1007/978-3-030-82529-4_36` doi: 10.1007/978-3-030-82529-4_36

Kalimuthu, M., Vaishnavi, P., & Kishore, M. (2020). Crop prediction using machine learning. In *2020 third international conference on smart systems and inventive technology (icssit)* (p. 926-932). doi: 10.1109/ICSSIT48917.2020.9214190

Keerthan Kumar, T., Shubha, C., , & Sushma, S. (2019). Random forest algorithm for soil fertility prediction and grading using machine learning. *Int J Innov Technol Explor Eng*, *9*(1), 1301–

1304.

Kulkarni, N. H., Srinivasan, G. N., Sagar, B. M., & Cauvery, N. K. (2018). Improving crop productivity through a crop recommendation system using ensembling technique. In *2018 3rd international conference on computational systems and information technology for sustainable solutions (csitss)* (p. 114-119). doi: 10.1109/CSITSS.2018.8768790

Lundh, F. (1999). An introduction to tkinter. *URL: www. pythonware. com/library/tkinter/introduction/index. htm*.

Mckinney, W. (2011, 01). pandas: a foundational python library for data analysis and statistics. *Python High Performance Science Computer*.

Oshima, A., Hogue, A., et al. (2007). *Introduction to academic writing*. Pearson/Longman.

Rajak, R. K., Pawar, A., Pendke, M., Shinde, P., Rathod, S., & Devare, A. (2017). Crop recommendation system to maximize crop yield using machine learning technique. *International Research Journal of Engineering and Technology*, *4*(12), 950–953.

Appendices

Appendix A Heatmap

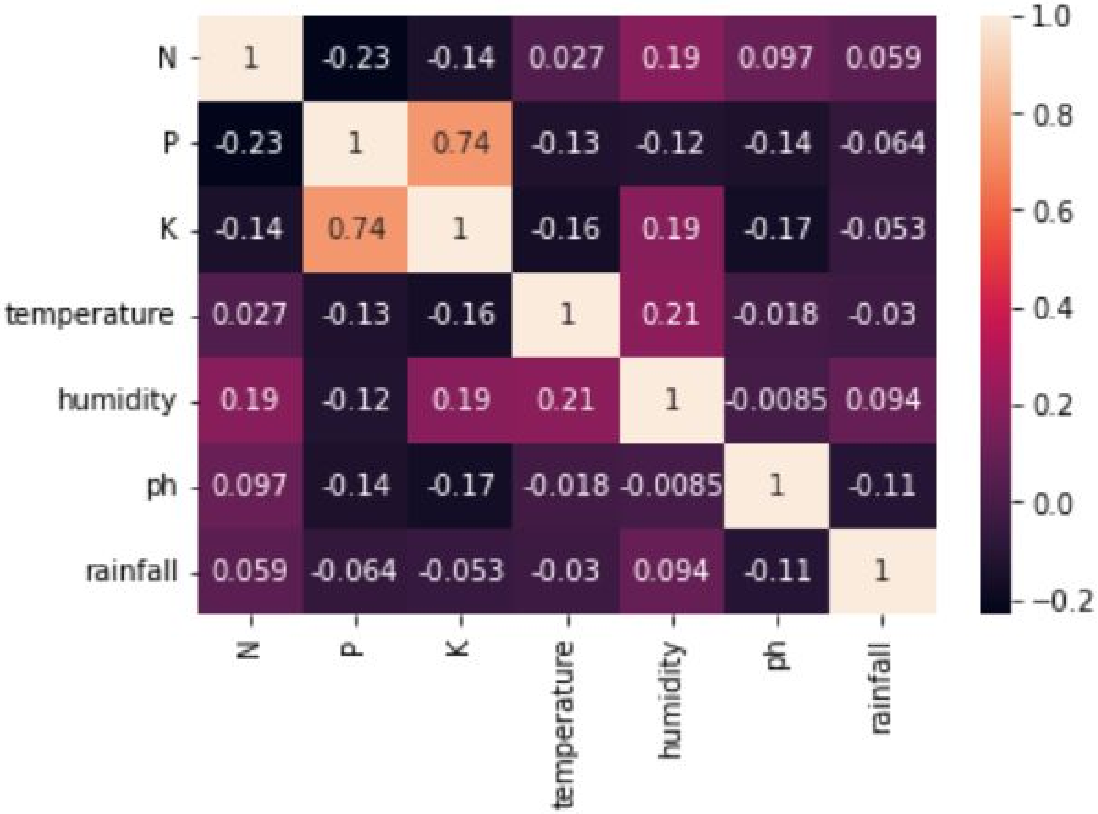


Figure A.1: Heatmap of crop prediction dataset

Appendix B Training

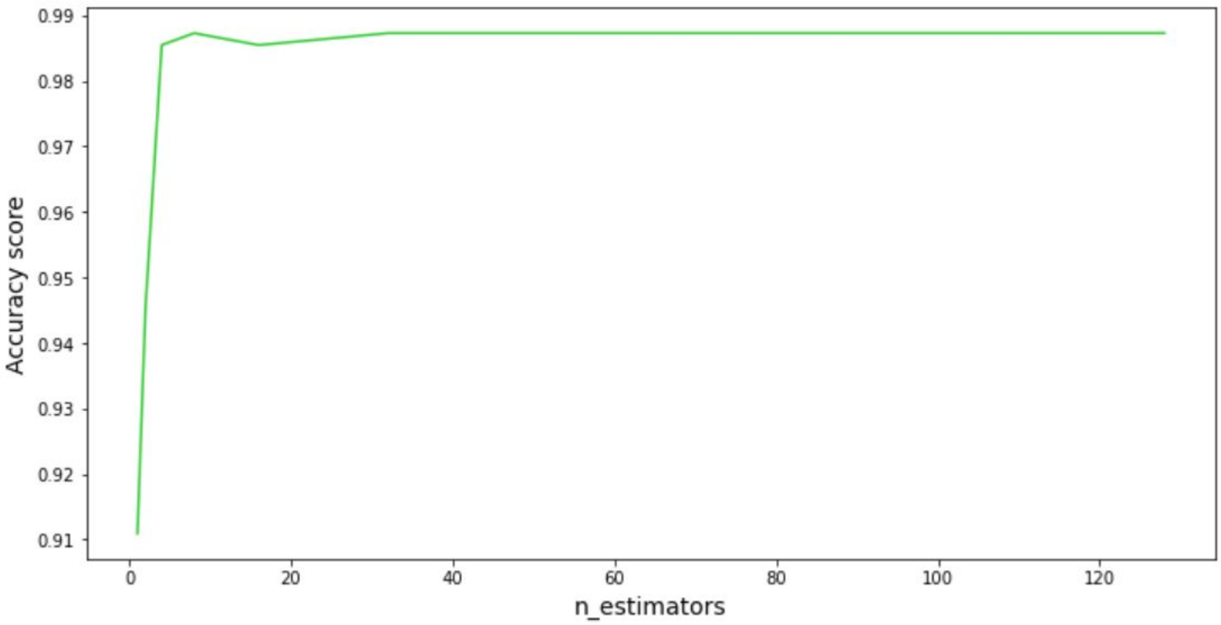


Figure B.1: Random Forest n estimator graph

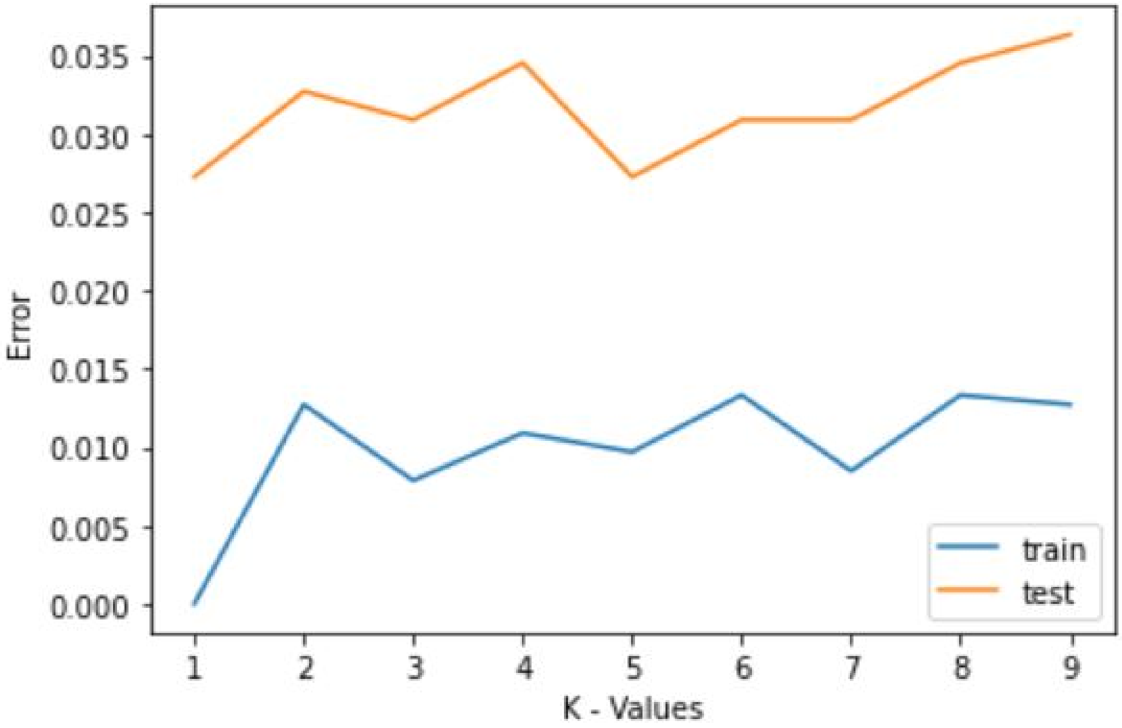
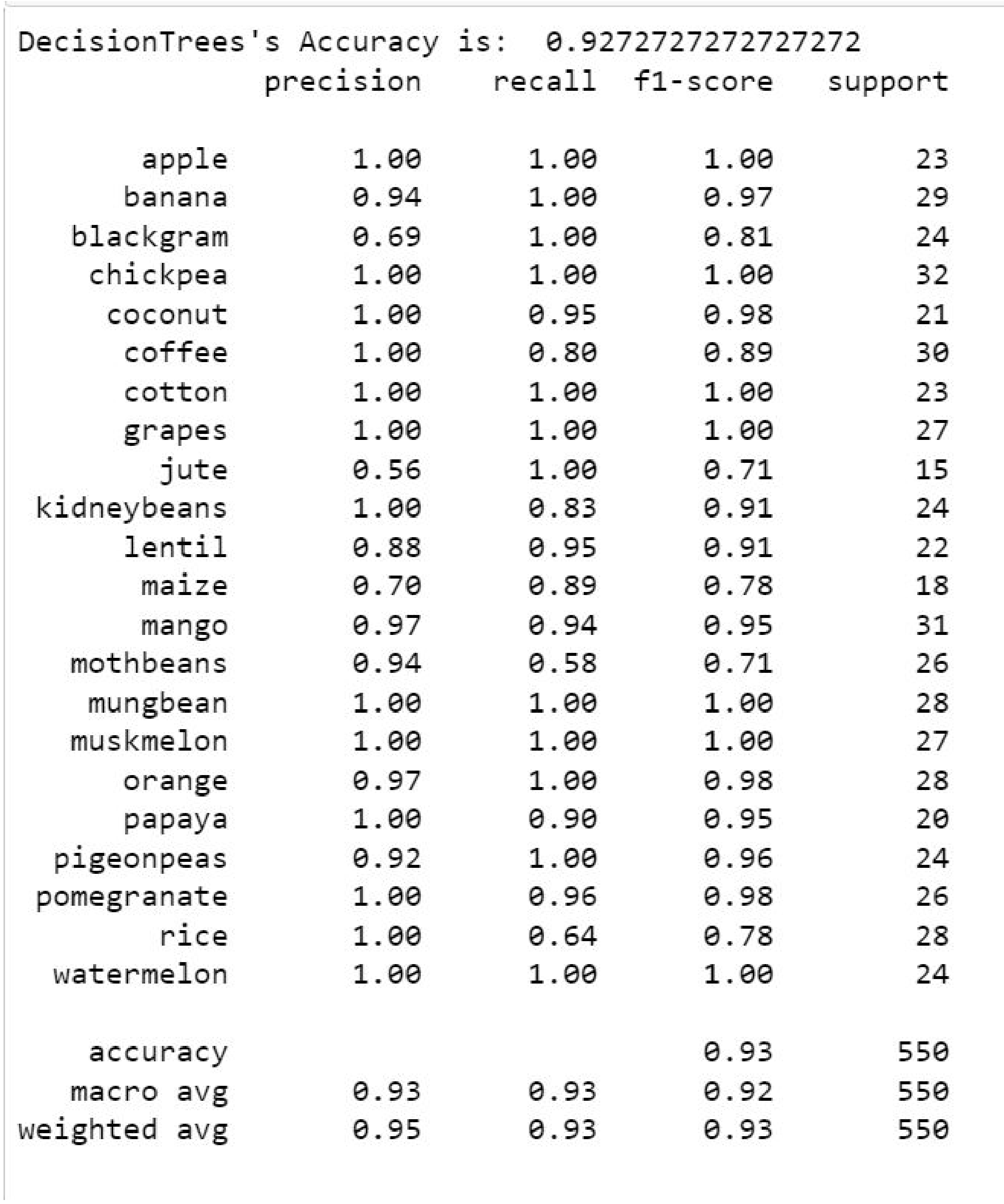


Figure B.2: KNN k values

Appendix C Accuracy of all ML models

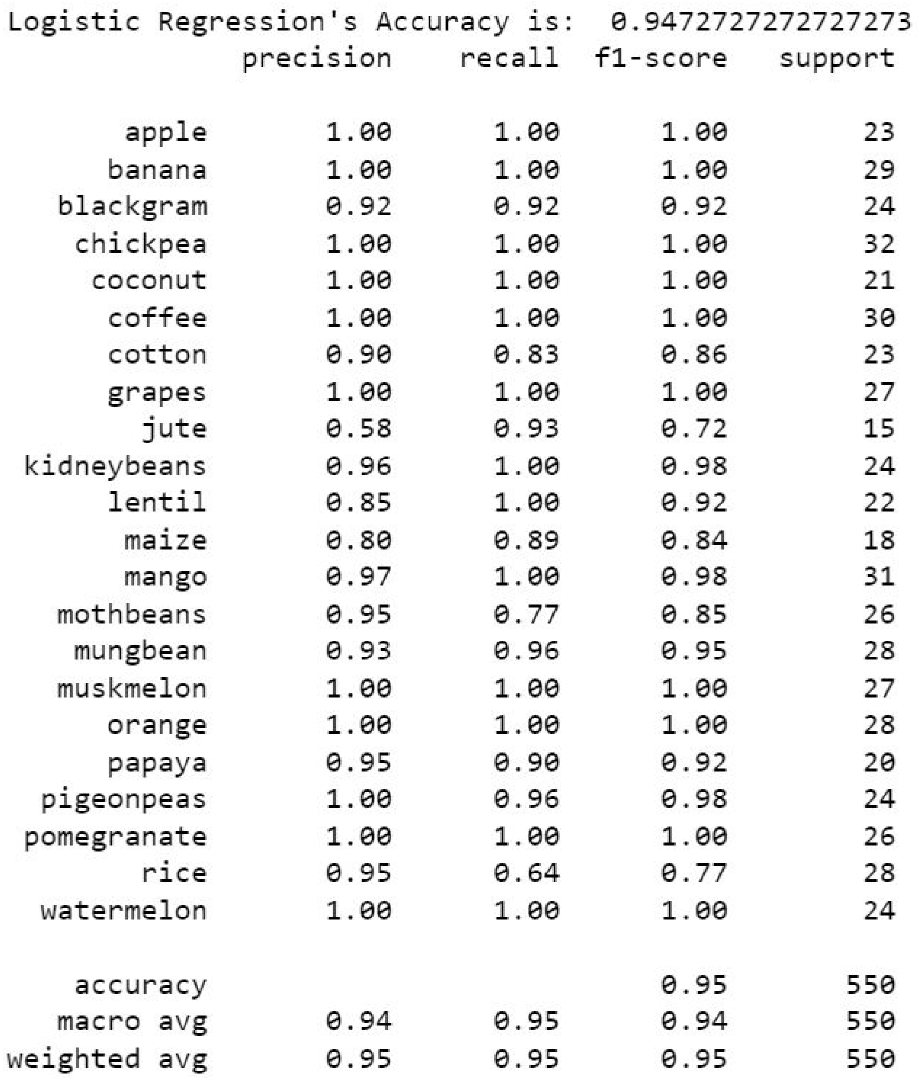


```
DecisionTrees's Accuracy is:  0.9272727272727272
              precision    recall  f1-score   support

       apple       1.00      1.00      1.00        23
      banana       0.94      1.00      0.97        29
   blackgram       0.69      1.00      0.81        24
    chickpea       1.00      1.00      1.00        32
     coconut       1.00      0.95      0.98        21
      coffee       1.00      0.80      0.89        30
      cotton       1.00      1.00      1.00        23
      grapes       1.00      1.00      1.00        27
        jute       0.56      1.00      0.71        15
 kidneybeans       1.00      0.83      0.91        24
      lentil       0.88      0.95      0.91        22
       maize       0.70      0.89      0.78        18
       mango       0.97      0.94      0.95        31
   mothbeans       0.94      0.58      0.71        26
    mungbean       1.00      1.00      1.00        28
   muskmelon       1.00      1.00      1.00        27
      orange       0.97      1.00      0.98        28
      papaya       1.00      0.90      0.95        20
  pigeonpeas       0.92      1.00      0.96        24
 pomegranate       1.00      0.96      0.98        26
        rice       1.00      0.64      0.78        28
  watermelon       1.00      1.00      1.00        24

    accuracy                           0.93       550
   macro avg       0.93      0.93      0.92       550
weighted avg       0.95      0.93      0.93       550
```

Figure C.1: Accuracy of Decision Tree

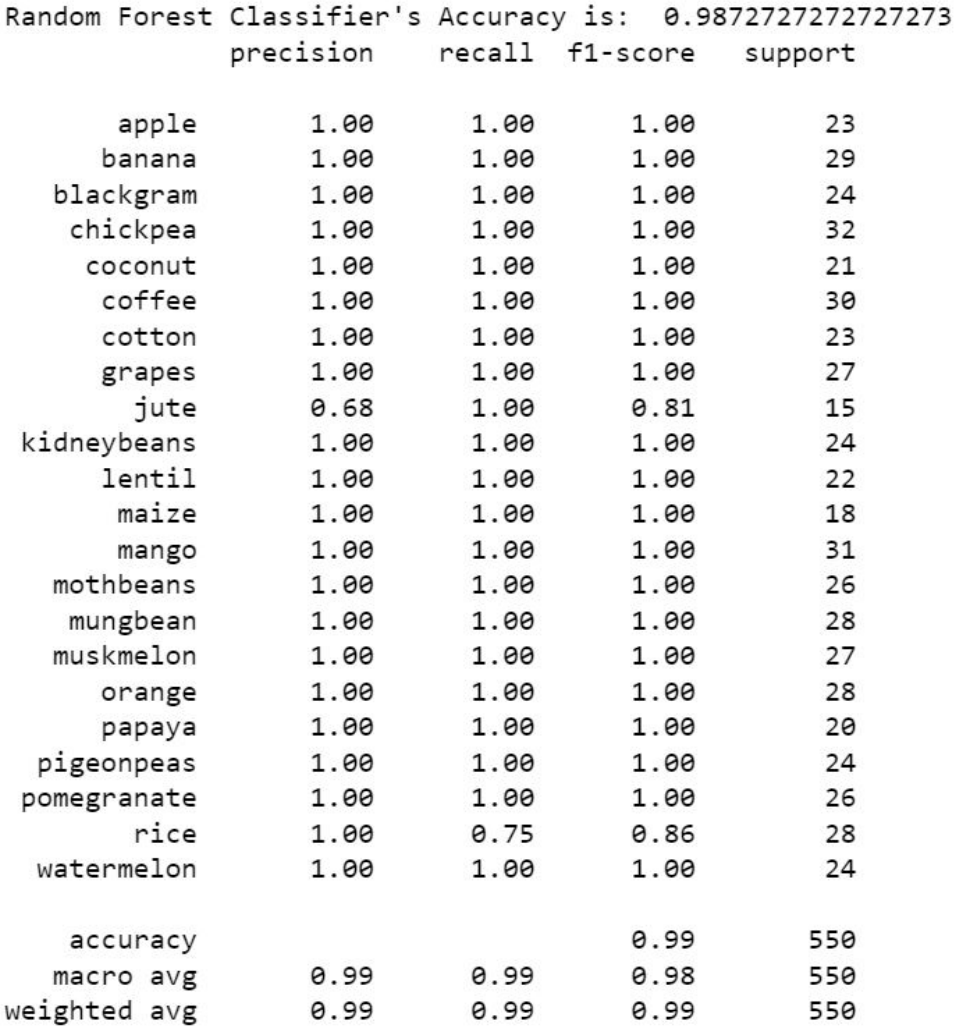


```
Logistic Regression's Accuracy is:  0.9472727272727273
              precision    recall  f1-score   support

       apple       1.00      1.00      1.00        23
      banana       1.00      1.00      1.00        29
   blackgram       0.92      0.92      0.92        24
    chickpea       1.00      1.00      1.00        32
     coconut       1.00      1.00      1.00        21
      coffee       1.00      1.00      1.00        30
      cotton       0.90      0.83      0.86        23
      grapes       1.00      1.00      1.00        27
        jute       0.58      0.93      0.72        15
 kidneybeans       0.96      1.00      0.98        24
      lentil       0.85      1.00      0.92        22
       maize       0.80      0.89      0.84        18
       mango       0.97      1.00      0.98        31
   mothbeans       0.95      0.77      0.85        26
    mungbean       0.93      0.96      0.95        28
   muskmelon       1.00      1.00      1.00        27
      orange       1.00      1.00      1.00        28
      papaya       0.95      0.90      0.92        20
  pigeonpeas       1.00      0.96      0.98        24
 pomegranate       1.00      1.00      1.00        26
        rice       0.95      0.64      0.77        28
  watermelon       1.00      1.00      1.00        24

    accuracy                           0.95       550
   macro avg       0.94      0.95      0.94       550
weighted avg       0.95      0.95      0.95       550
```

Figure C.2: Accuracy of Logistic Regression

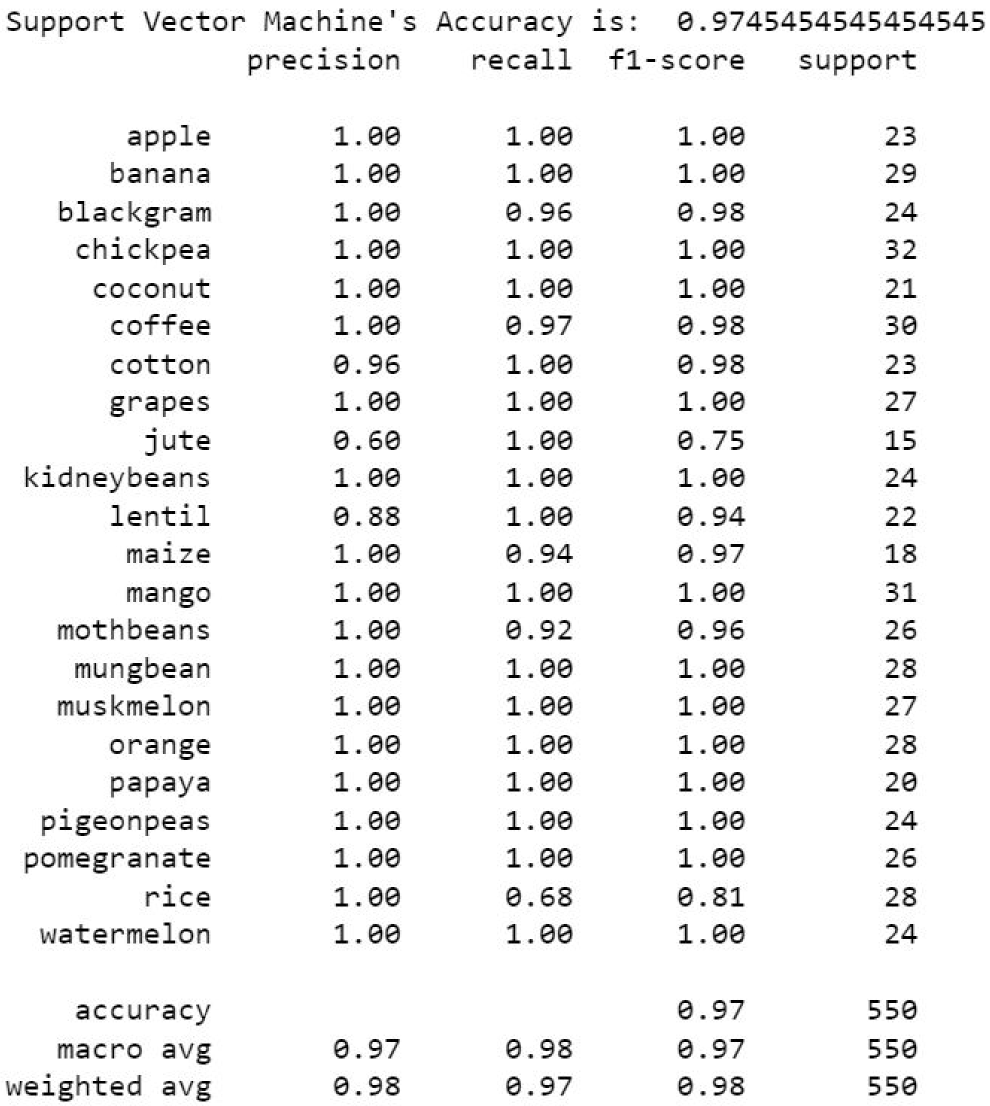


```
Random Forest Classifier's Accuracy is:  0.9872727272727273
              precision    recall  f1-score   support

       apple       1.00      1.00      1.00        23
      banana       1.00      1.00      1.00        29
   blackgram       1.00      1.00      1.00        24
    chickpea       1.00      1.00      1.00        32
     coconut       1.00      1.00      1.00        21
      coffee       1.00      1.00      1.00        30
      cotton       1.00      1.00      1.00        23
      grapes       1.00      1.00      1.00        27
        jute       0.68      1.00      0.81        15
 kidneybeans       1.00      1.00      1.00        24
      lentil       1.00      1.00      1.00        22
       maize       1.00      1.00      1.00        18
       mango       1.00      1.00      1.00        31
   mothbeans       1.00      1.00      1.00        26
    mungbean       1.00      1.00      1.00        28
   muskmelon       1.00      1.00      1.00        27
      orange       1.00      1.00      1.00        28
      papaya       1.00      1.00      1.00        20
  pigeonpeas       1.00      1.00      1.00        24
 pomegranate       1.00      1.00      1.00        26
        rice       1.00      0.75      0.86        28
  watermelon       1.00      1.00      1.00        24

    accuracy                           0.99       550
   macro avg       0.99      0.99      0.98       550
weighted avg       0.99      0.99      0.99       550
```

Figure C.3: Accuracy of Random Forest

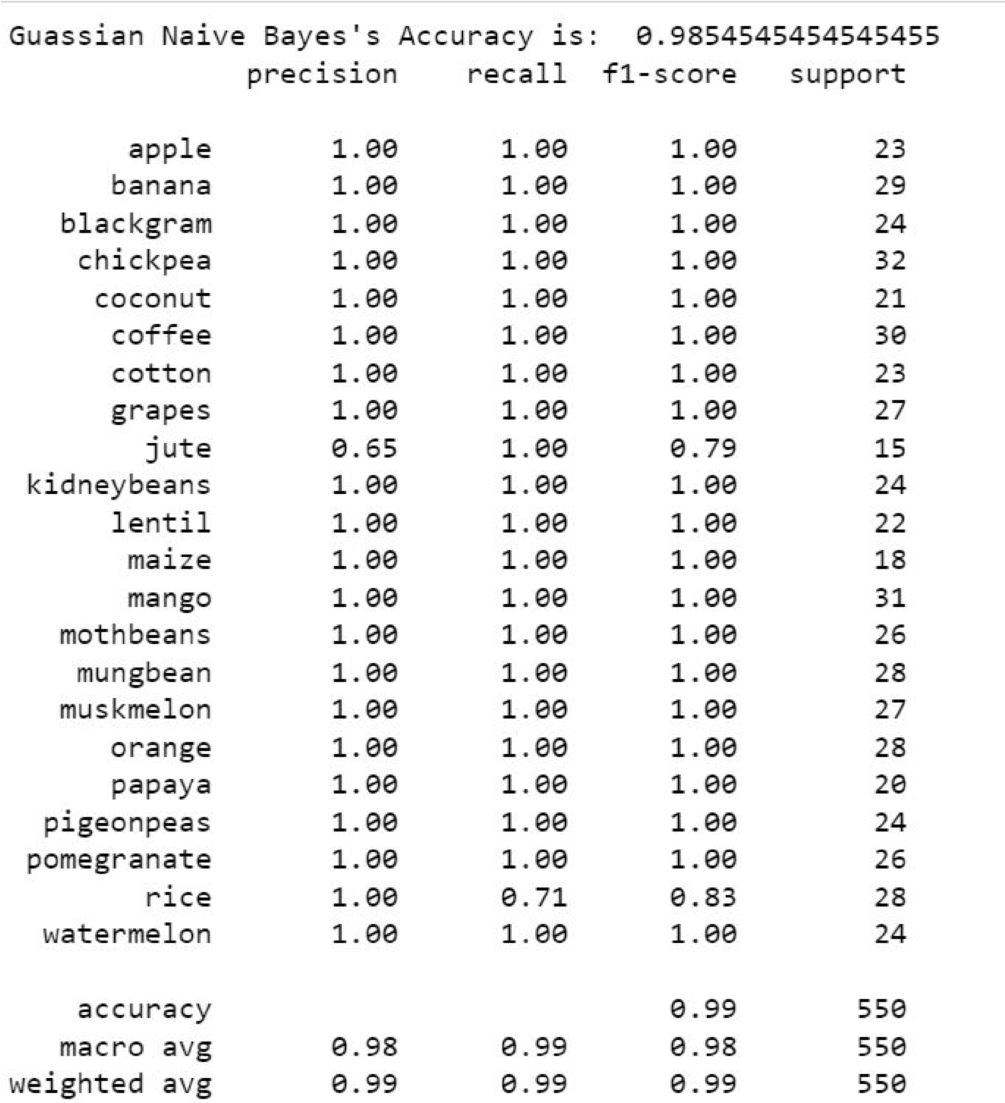


```
Support Vector Machine's Accuracy is:  0.9745454545454545
              precision    recall  f1-score   support

       apple       1.00      1.00      1.00        23
      banana       1.00      1.00      1.00        29
   blackgram       1.00      0.96      0.98        24
    chickpea       1.00      1.00      1.00        32
     coconut       1.00      1.00      1.00        21
      coffee       1.00      0.97      0.98        30
      cotton       0.96      1.00      0.98        23
      grapes       1.00      1.00      1.00        27
        jute       0.60      1.00      0.75        15
 kidneybeans       1.00      1.00      1.00        24
      lentil       0.88      1.00      0.94        22
       maize       1.00      0.94      0.97        18
       mango       1.00      1.00      1.00        31
   mothbeans       1.00      0.92      0.96        26
    mungbean       1.00      1.00      1.00        28
   muskmelon       1.00      1.00      1.00        27
      orange       1.00      1.00      1.00        28
      papaya       1.00      1.00      1.00        20
  pigeonpeas       1.00      1.00      1.00        24
 pomegranate       1.00      1.00      1.00        26
        rice       1.00      0.68      0.81        28
  watermelon       1.00      1.00      1.00        24

    accuracy                           0.97       550
   macro avg       0.97      0.98      0.97       550
weighted avg       0.98      0.97      0.98       550
```

Figure C.4: Accuracy of Support Vector Machine

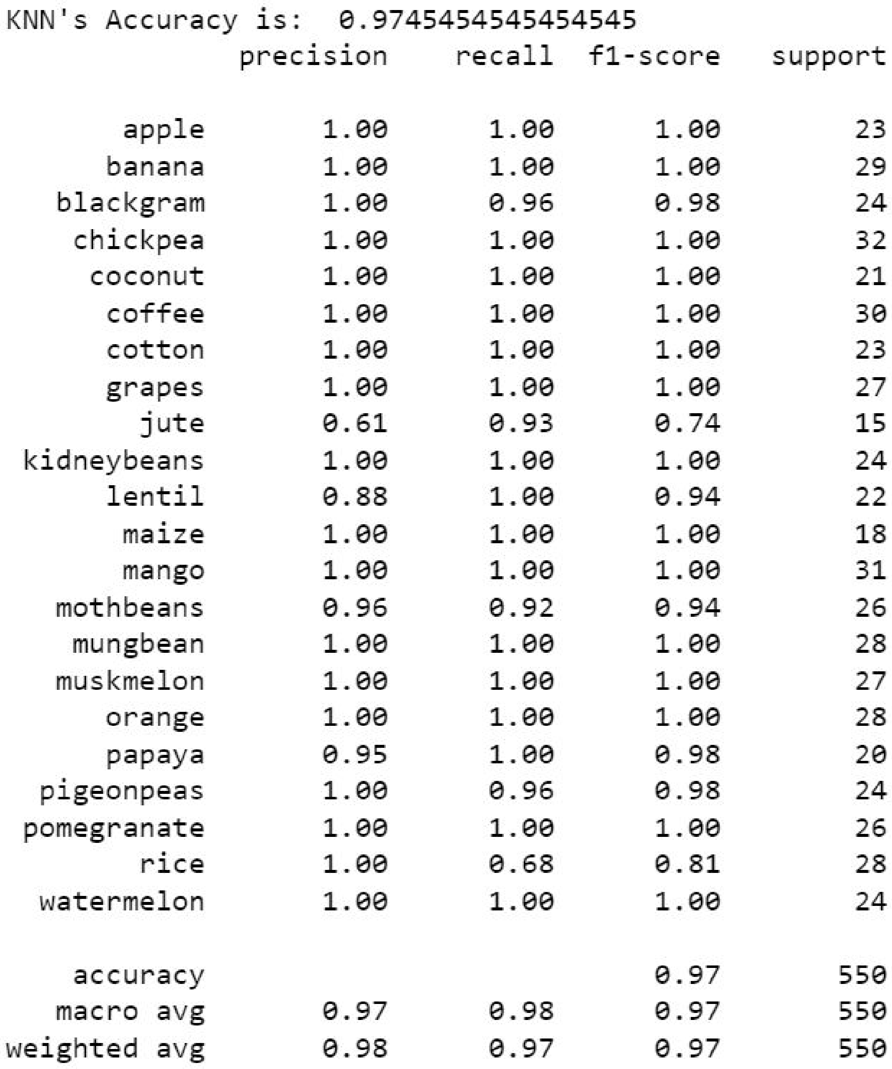


```
Guassian Naive Bayes's Accuracy is:  0.9854545454545455
              precision    recall  f1-score   support

       apple       1.00      1.00      1.00        23
      banana       1.00      1.00      1.00        29
   blackgram       1.00      1.00      1.00        24
    chickpea       1.00      1.00      1.00        32
     coconut       1.00      1.00      1.00        21
      coffee       1.00      1.00      1.00        30
      cotton       1.00      1.00      1.00        23
      grapes       1.00      1.00      1.00        27
        jute       0.65      1.00      0.79        15
 kidneybeans       1.00      1.00      1.00        24
      lentil       1.00      1.00      1.00        22
       maize       1.00      1.00      1.00        18
       mango       1.00      1.00      1.00        31
   mothbeans       1.00      1.00      1.00        26
    mungbean       1.00      1.00      1.00        28
   muskmelon       1.00      1.00      1.00        27
      orange       1.00      1.00      1.00        28
      papaya       1.00      1.00      1.00        20
  pigeonpeas       1.00      1.00      1.00        24
 pomegranate       1.00      1.00      1.00        26
        rice       1.00      0.71      0.83        28
  watermelon       1.00      1.00      1.00        24

    accuracy                           0.99       550
   macro avg       0.98      0.99      0.98       550
weighted avg       0.99      0.99      0.99       550
```

Figure C.5: Accuracy of Guassian Naive Bayes



```
KNN's Accuracy is:  0.9745454545454545
              precision    recall  f1-score   support

       apple       1.00      1.00      1.00        23
      banana       1.00      1.00      1.00        29
   blackgram       1.00      0.96      0.98        24
    chickpea       1.00      1.00      1.00        32
     coconut       1.00      1.00      1.00        21
      coffee       1.00      1.00      1.00        30
      cotton       1.00      1.00      1.00        23
      grapes       1.00      1.00      1.00        27
        jute       0.61      0.93      0.74        15
 kidneybeans       1.00      1.00      1.00        24
      lentil       0.88      1.00      0.94        22
       maize       1.00      1.00      1.00        18
       mango       1.00      1.00      1.00        31
   mothbeans       0.96      0.92      0.94        26
    mungbean       1.00      1.00      1.00        28
   muskmelon       1.00      1.00      1.00        27
      orange       1.00      1.00      1.00        28
      papaya       0.95      1.00      0.98        20
  pigeonpeas       1.00      0.96      0.98        24
 pomegranate       1.00      1.00      1.00        26
        rice       1.00      0.68      0.81        28
  watermelon       1.00      1.00      1.00        24

    accuracy                           0.97       550
   macro avg       0.97      0.98      0.97       550
weighted avg       0.98      0.97      0.97       550
```

Figure C.6: Accuracy of K-Nearest Neighbour

Appendix D Comparing Models Accuracy

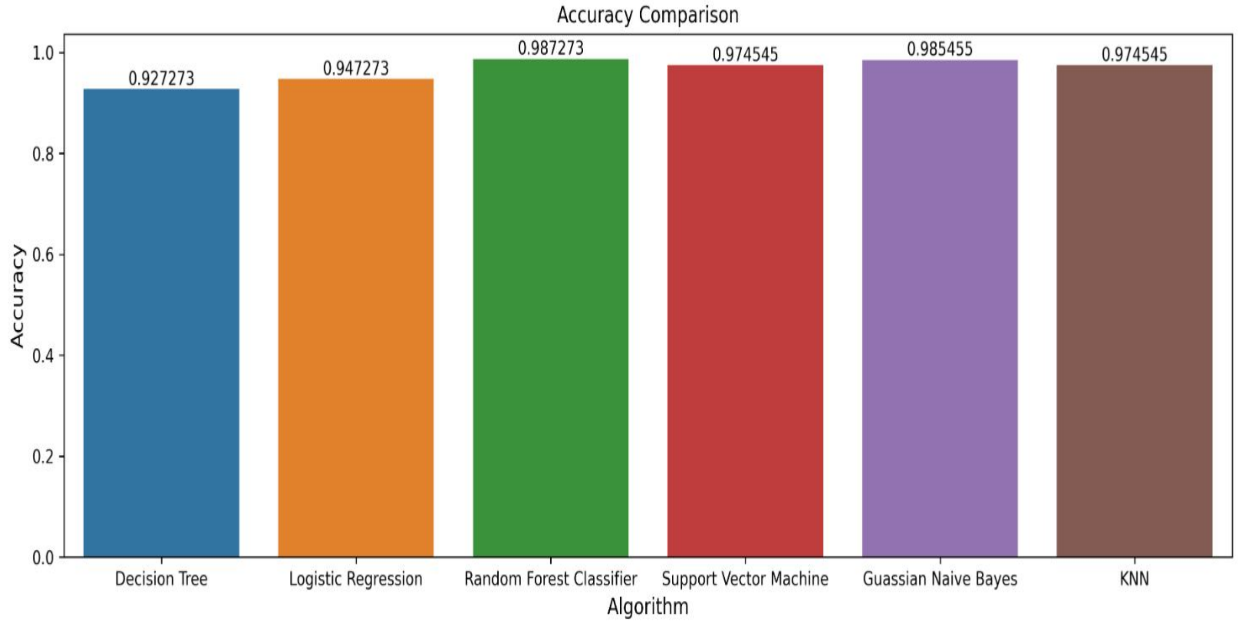
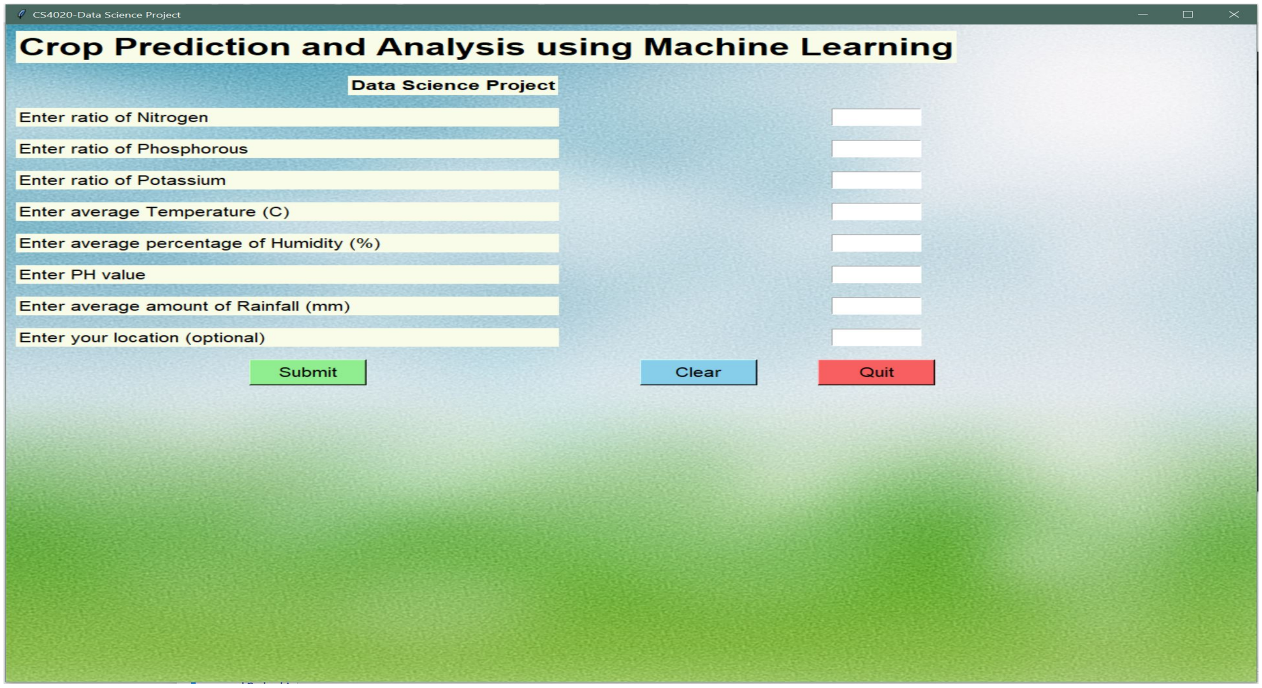


Figure D.1: Comparison graph of models

Appendix E GUI



CS4020-Data Science Project

Crop Prediction and Analysis using Machine Learning

Data Science Project

Enter ratio of Nitrogen

Enter ratio of Phosphorous

Enter ratio of Potassium

Enter average Temperature (C)

Enter average percentage of Humidity (%)

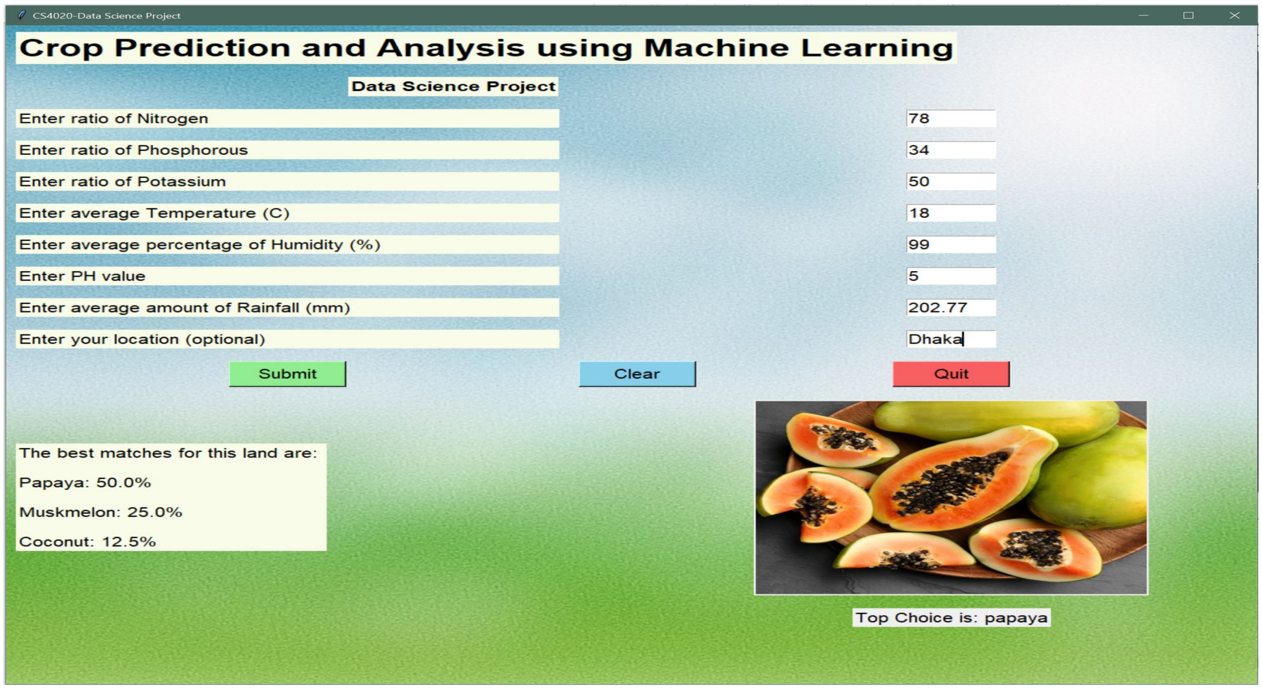
Enter PH value

Enter average amount of Rainfall (mm)

Enter your location (optional)

Figure E.1: User Interface of Crop Prediction and Analysis System

Appendix F   Result



Crop Prediction and Analysis using Machine Learning

Data Science Project

Enter ratio of Nitrogen	78
Enter ratio of Phosphorous	34
Enter ratio of Potassium	50
Enter average Temperature (C)	18
Enter average percentage of Humidity (%)	99
Enter PH value	5
Enter average amount of Rainfall (mm)	202.77
Enter your location (optional)	Dhaka

The best matches for this land are:

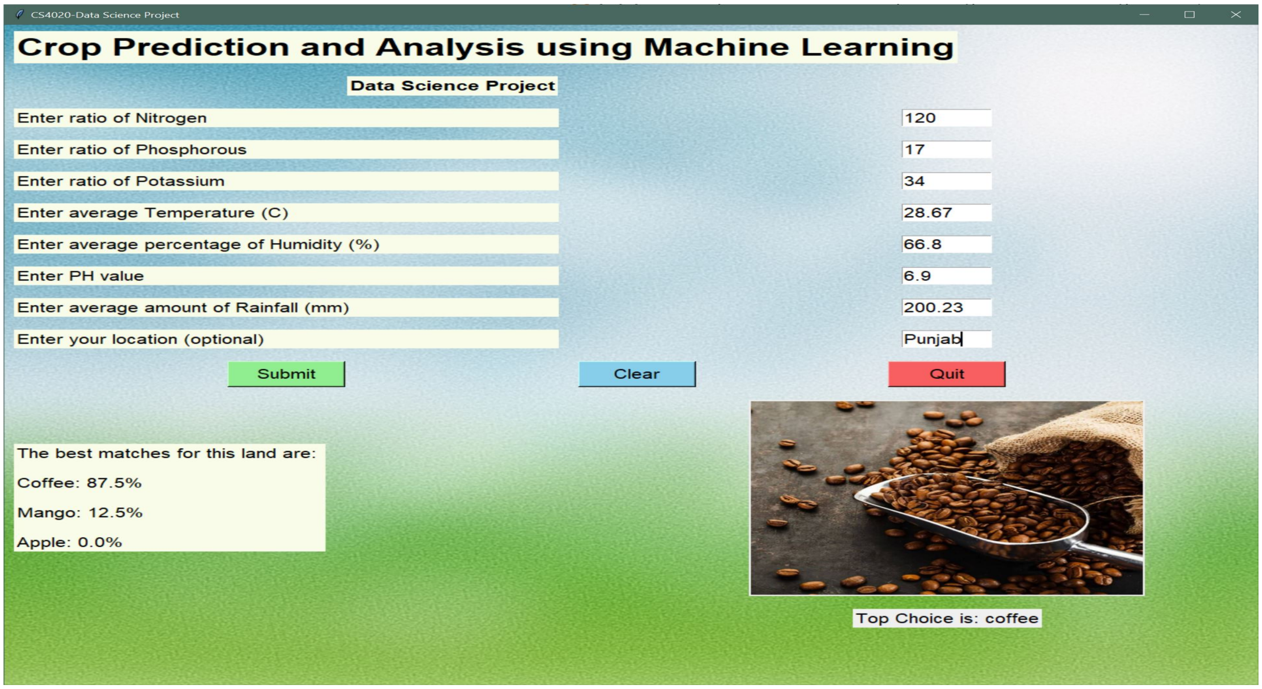
Papaya: 50.0%

Muskmelon: 25.0%

Coconut: 12.5%

Top Choice is: papaya

Figure F.1: Crop Prediction and Analysis System’s result example- 1



Crop Prediction and Analysis using Machine Learning

Data Science Project

Enter ratio of Nitrogen	120
Enter ratio of Phosphorous	17
Enter ratio of Potassium	34
Enter average Temperature (C)	28.67
Enter average percentage of Humidity (%)	66.8
Enter PH value	6.9
Enter average amount of Rainfall (mm)	200.23
Enter your location (optional)	Punjab

The best matches for this land are:

Coffee: 87.5%

Mango: 12.5%

Apple: 0.0%

Top Choice is: coffee

Figure F.2: Crop Prediction and Analysis System's result example - 2