

**TITLE OF THE PROJECT: SEED PREDICTION USING
LOGISTIC REGRESSION**

Mini Project Report Submitted to

SRI PADMAVATI MAHILA VISVAVIDYALAYAM

In Partical fulfilment of the requirement for the

MASTER OF COMPUTER APPLICATIONS

III SEMESTER

By

KUNCHALA MANJU BHARGAVI (2021MCA16057)

VUCHALA RAJANI (2021MCA16106)

Under the guidance of

Prof. P.VENKATA KRISHNA



DEPARTMENT OF COMPUTER SCIENCE

SRI PADMAVATI MAHILA VISVAVIDYALAYAM (Women's University)

Accredited with 'A+' Grade by NAAC

Tirupati-517502(A.P), Andhra Pradesh

APRIL, 2023

DEPARTMENT OF COMPUTER SCIENCE
SRI PADMAVATI MAHILA VISVAVIDYALAYAM (Women's University)
Accredited with 'A+' Grade by NAAC
Tirupati-517502(A.P), Andhra Pradesh



CERTIFICATE

This is to certify that the project work entitled **"SEED PREDICTION USING LOGISTIC REGRESSION"** is a bonafied record of work carried by

KUNCHALA MANJU BHARGAVI REG.NO: 2021MCA16057

VUCHALA RAJANI REGNO:2021MCA16106

In the Department of Computer Science, **SRI PADMAVATI MAHILA VISVAVIDYALAYAM**, Tirupati in partial fulfilment of the requirements of III Semester of **MASTER OF COMPUTER APPLICATIONS**. The content of the Project Report has not been submitted to any other University/Institute for the award of any degree.

SIGNATURE OF THE GUIDE

SIGNATURE OF THE HEAD

ACKNOWLEDGEMENT

We are greatly indebted to our guide **Prof. P.VENKATA KRISHNA** for taking keen interest on our project work and providing valuable suggestions in all the possible areas of improvement. We express our sincere thanks to the teaching staff of the Department of Computer Science for extending support and encouragement to us in all the stages of the project work.

In the last, we gratefully acknowledge and express our gratitude to the technical members of the Computer Science Department who supported us in preparing this project.

Table of contents

S.No.	List of contents	Page no.
	Abstract	5
1.	Introduction	6
2.	Consider framework	7-10
3.	Random forest Classifier	11-12
4.	Support Vector Classifier	13
5.	Logistic Regression	14
6.	Decision Tree	15
7.	AdaBoost	16
8.	Confusion Matrix	17
9.	Classification	18
10.	Prediction	19
11.	Validation	20
12.	Dataset Description	21
13.	Dataset-Seed Prediction	22
14.	Result	23-26
15.	Conclusion	27
	References	28

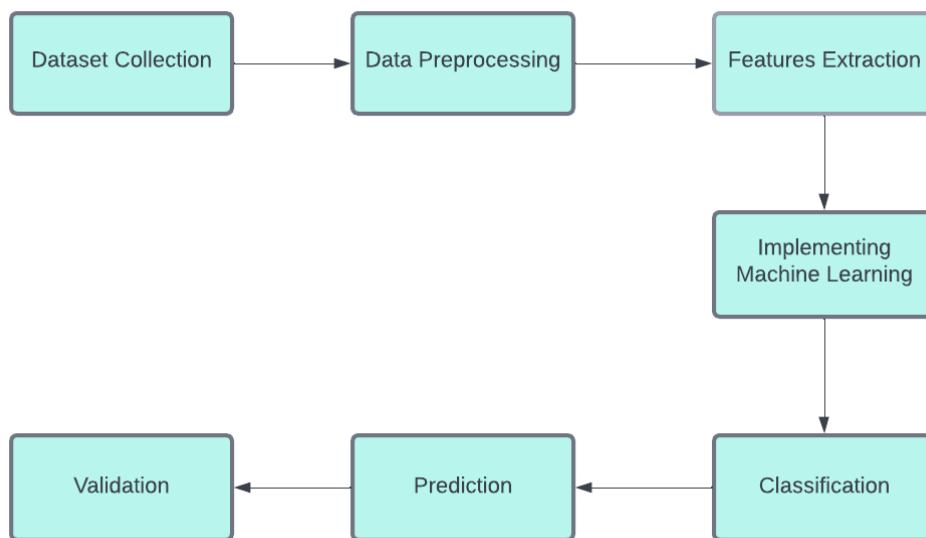
ABSTRACT:

Seed prediction is the most important thing for farmers is to know the quality of seed. Hence predicting the quality of seed is critical for obtaining a good yields. This project proposes the logistic regression based seed prediction model in which the quality of seed is predicted.To prove the excellence of the algorithm comprehensive exipermentation has been conducted using Kaggle seed prediction dataset and compared with other predictions of artificial machine learning algorithms.Experiments shows that the proposed logistic regression has outperformed the other learning model.

1.INTRODUCTION:

Machine Learning is a computer science field where new developments have recently been developed, which also helps to automate assessment and processing carried out by mankind so that human manual power is reduced. Machine learning is a type of artificial intelligence (AI) that enables computers to learn without explicitly programming, according to the technological goal. Machine learning focuses on computer program, which can change if new data are exposed. It is difficult for novice farmers to find the right crops based on the appearance of the soil. Agricultural decline must also be prevented. For ensuring a food security, an effective use of agricultural land is crucial. In this document, we propose an ID3 Algorithm and Support Vector Machine (SVM) crop recommendation system. This paper describes machine learning, tasks involved, models, advantages, and the purpose of the proposed work and the overview of a thesis. This paper describes an ID3 algorithm and a support vector machine crop prediction and recommendation system. In order to achieve the predicted cropping return, the classified crop field images along with the historical weather and yield data are modelled and recommended for a specific field appropriate crops. This in turn involves classification and prediction machine learning and image processing.

2. CONSIDERED FRAMEWORK:



Data Collection:

Collecting data for seed prediction can be done in several ways. The most common method is to perform experiments and measurements on actual seeds in a controlled environment. This involves growing the seeds under controlled conditions, and then collecting data on various physical and chemical characteristics of the seeds. The data collected can include features such as seed weight, seed size, seed shape, seed color, germination rate, nutrient content, etc.

Another method for collecting data is to use existing datasets or public repositories that contain information on seeds. For example, the UCI Machine Learning Repository is a popular resource that provides several datasets related to seed classification and prediction.

It's important to ensure that the data collected is representative of the population of interest and is diverse enough to capture the variability in seed quality. This can be achieved by

collecting data from different geographical regions, different crop varieties, and under different growing conditions.

In addition, it's important to ensure that the data is of high quality and free from errors or biases. This can be achieved by using standardized measurement protocols and quality control measures during data collection.

Overall, collecting data for seed prediction requires careful planning and execution to ensure that the data collected is accurate, representative, and of high quality.

Data Processing:

Seed prediction involves using various data processing techniques to analyze and interpret data in order to predict the yield of crops. This is typically done by collecting data on a number of different factors that can affect crop yield, such as soil moisture, temperature, nutrient levels, and pest infestations.

Once the data has been collected, it must be processed and analyzed using statistical techniques and machine learning algorithms in order to identify patterns and relationships between the various factors and the predicted yield. This can involve a number of different steps, including data cleaning, data normalization, feature selection, and model training and validation.

Data cleaning involves removing any errors, inconsistencies, or missing values from the data set, while data normalization involves scaling the data to ensure that all of the variables are on the same scale. Feature selection involves identifying which variables are most important for predicting crop yield, while model training and validation involves using machine learning algorithms to build and test predictive models based on the data.

Overall, effective data processing is essential for accurate seed prediction, as it enables researchers and farmers to make informed decisions about how to manage their crops and optimize their yields.

Feature Extraction:

Feature extraction is an important step in seed prediction using machine learning techniques. In this step, relevant features are identified and extracted from the input data to create a new representation of the data that can be used for training machine learning models.

In seed prediction, some common features that can be extracted from the input data include soil moisture, temperature, nutrient levels, rainfall, and pest infestation levels. However, the choice of features will depend on the specific problem being addressed and the data available.

Feature extraction techniques can vary depending on the type of data being analyzed.

For example,

If the input data is in the form of images, techniques such as edge detection and texture analysis can be used to extract features from the images. If the input data is in the form of text, techniques such as word embedding and topic modeling can be used to extract features from the text.

Once the features have been extracted, they are typically scaled and normalized to ensure that all of the variables are on the same scale. This can help to improve the accuracy of the machine learning models by reducing the impact of variables that have large values.

Overall, feature extraction is an important step in seed prediction using machine learning techniques, as it enables researchers and farmers to identify the most important factors that affect crop yield and to build accurate models for predicting future yields.

Implementing machine learning:

Collect and prepare the data: Collect a dataset that includes the features of the seeds you want to predict, such as their length, width, area, perimeter, compactness, and so on. Make sure the dataset is clean and free from missing or incorrect values. You can also perform some exploratory data analysis to gain insights into the dataset.

Split the data: Split the dataset into a training set and a testing set. The training set will be used to train the machine learning model, while the testing set will be used to evaluate its performance.

Choose a machine learning algorithm: There are many machine learning algorithms that you can use for seed prediction, such as decision trees, random forests, support vector machines, and neural networks. Choose an algorithm that is appropriate for your dataset and the problem you want to solve.

Train the model: Train the machine learning model on the training set. The goal is to find a model that can accurately predict the class of new seeds based on their features.

Evaluate the model: Evaluate the performance of the model on the testing set. You can use metrics such as accuracy, precision, recall, and F1 score to measure how well the model performs.

Fine-tune the model: If the performance of the model is not satisfactory, you can fine-tune its parameters or try different algorithms until you get the desired results.

Deploy the model: Once you are satisfied with the performance of the model, you can deploy it to make predictions on new, unseen data.

Overall, implementing machine learning for seed prediction requires a combination of data preparation, algorithm selection, model training and evaluation, and fine-tuning. By following these steps, you can build a machine learning model that can accurately predict the class of new seeds based on their features.

3.Random Forest Classifier:

Random Forest Classifier is a powerful machine learning algorithm that is widely used for both classification and regression tasks. It is an ensemble learning method that combines multiple decision trees to improve the accuracy and reduce the overfitting of the model. In this article, we will explore the key features of the Random Forest Classifier and how it works.

Key Features of Random Forest Classifier:

Decision Trees: Random Forest Classifier is based on decision trees, which are used to make predictions by dividing the dataset into smaller subsets based on the values of the features. The decision trees in Random Forest Classifier are created using a subset of the data and a random selection of the features, which reduces the correlation between the trees and improves the accuracy of the model.

Bagging: Random Forest Classifier uses a technique called bagging, which stands for Bootstrap Aggregating. Bagging is used to improve the accuracy of the model by creating multiple decision trees on random subsets of the data and then combining them to make the final prediction. This reduces the variance and overfitting of the model.

Ensemble Learning: Random Forest Classifier is an ensemble learning method, which combines multiple decision trees to make the final prediction. This improves the accuracy and stability of the model.

Feature Importance: Random Forest Classifier provides a measure of feature importance, which indicates the relative importance of each feature in predicting the outcome. This can be used to identify the most important features and improve the performance of the model.

Out-of-Bag (OOB) Error: Random Forest Classifier uses the out-of-bag (OOB) error to estimate the accuracy of the model. This is done by using the samples that were not used to create the decision tree to test its accuracy.

How Random Forest Classifier Works:

- Random subsets of the data are created using the bagging technique.
- A decision tree is created using a subset of the data and a random selection of the features.
- Multiple decision trees are created using different subsets of the data and features.
- The decision trees are combined to make the final prediction by taking the majority vote of the predictions.
- The accuracy of the model is estimated using the out-of-bag (OOB) error.
- In conclusion, Random Forest Classifier is a powerful machine learning algorithm that combines multiple decision trees to improve the accuracy and stability of the model. It is widely used for classification and regression tasks and provides a measure of feature importance and the out-of-bag (OOB) error for model evaluation. Its key features include decision trees, bagging, ensemble learning, feature importance, and OOB error.

4.Support Vector Classifier:

Support Vector Classification (SVC) is a type of supervised learning algorithm that can be used for classification tasks. SVC is based on the idea of finding the best possible decision boundary between different classes of data.

SVC is particularly useful when dealing with complex datasets that have non-linear boundaries. In such cases, SVC uses a kernel function to map the data into a higher-dimensional space, where it becomes easier to separate the classes with a linear boundary.

SVC is also useful when dealing with datasets that have a large number of features. In such cases, SVC can use a subset of the features, called the support vectors, to create the decision boundary, which can improve the accuracy and efficiency of the algorithm.

Random Forest Classifier is an ensemble learning algorithm that combines multiple decision trees to improve the accuracy and robustness of the model. Random Forest Classifier is particularly useful when dealing with noisy or incomplete datasets, as it can handle missing values and outliers.

Random Forest Classifier works by randomly selecting a subset of features and building a decision tree based on that subset. This process is repeated multiple times, and the final prediction is based on the average prediction of all the decision trees.

SVCforest Classifier combines the advantages of SVC and Random Forest Classifier. It uses multiple decision trees, each of which is built using a subset of the features. Each decision tree is used to classify the data, and the final prediction is based on the average prediction of all the decision trees.

SVCforest Classifier is particularly useful when dealing with high-dimensional datasets that have a large number of features. It can handle noisy and incomplete data and is robust to outliers. It also has a high accuracy and can handle non-linear boundaries.

In summary, SVCforest Classifier is an ensemble learning algorithm that combines the strengths of Support Vector Classification and Random Forest Classifier. It is useful for classification tasks with high-dimensional and complex datasets.

5. Logistic regression:

Logistic regression is a statistical method used to analyze the relationship between a binary outcome variable and one or more predictor variables. It is a type of generalized linear model that models the logit function of the probability of the outcome variable being true.

The main features of logistic regression are:

Binary outcome variable: The outcome variable in logistic regression is binary, meaning it has only two possible outcomes (e.g., good or bad, yes or no, 0 or 1).

Predictor variables: Logistic regression can be used to analyze the relationship between the outcome variable and one or more predictor variables. The predictor variables can be continuous or categorical.

Logit function: Logistic regression models the logit function of the probability of the outcome variable being true. The logit function is defined as the natural logarithm of the odds of the outcome variable being true, which is the ratio of the probability of the outcome variable being true to the probability of it being false.

Maximum likelihood estimation: Logistic regression estimates the parameters of the model using maximum likelihood estimation. This involves finding the parameter values that maximize the likelihood of the observed data given the model.

Odds ratio: Logistic regression produces odds ratios, which are a measure of the relative odds of the outcome variable being true for different levels of the predictor variables. An odds ratio greater than 1 indicates that the predictor variable is associated with an increased odds of the outcome variable being true, while an odds ratio less than 1 indicates that the predictor variable is associated with a decreased odds of the outcome variable being true.

Goodness of fit: Logistic regression models can be evaluated using various measures of goodness of fit, such as the deviance, AIC, or BIC. These measures assess how well the model fits the observed data and can be used to compare different models.

Logistic regression is a widely used method in many fields, including healthcare, finance, and social sciences, due to its simplicity, interpretability, and effectiveness in modeling binary outcomes.

6. Decision Tree:

Decision tree algorithm is a popular machine learning algorithm used for both classification and regression problems. It works by constructing a tree-like model of decisions and their possible consequences.

The main features of the decision tree algorithm are:

Tree structure: Decision tree algorithm constructs a tree-like structure where each node represents a decision or a test on a feature, and each branch represents an outcome or a decision that leads to another node.

Splitting criterion: The decision tree algorithm selects the best feature to split the data based on some splitting criterion, such as information gain or Gini impurity.

Recursive partitioning: The decision tree algorithm partitions the data recursively until it reaches a stopping criterion, such as a maximum tree depth or a minimum number of samples per leaf.

Pruning: The decision tree algorithm can be prone to overfitting, which can be mitigated by pruning the tree or setting constraints on the minimum number of samples required to split a node.

Interpretability: Decision trees are easily interpretable and can be visualized to understand the decision-making process and the importance of each feature.

Ensemble methods: Decision trees can be combined using ensemble methods such as random forests or gradient boosting to improve the performance and robustness of the model.

The decision tree algorithm has many applications in various fields, such as healthcare, finance, and social sciences. It is particularly useful when the relationships between the input features and the output variable are nonlinear or when there are interactions between the features. However, decision trees can be sensitive to noise and outliers, and they may not perform well when the data is imbalanced or the classes are not separable by a linear boundary.

7.AdaBoost:

AdaBoost (Adaptive Boosting) is a popular ensemble learning technique used in supervised machine learning for classification and regression analysis. It is an iterative algorithm that combines multiple "weak" learners (e.g., decision trees) to create a "strong" learner that can make accurate predictions on new data.

In AdaBoost, each iteration involves training a new weak learner on a modified version of the original training set. The modification involves assigning weights to each sample in the training set, where the weights are based on the performance of the previous weak learner. Samples that were classified incorrectly by the previous weak learner are given higher weights, while samples that were classified correctly are given lower weights.

After each iteration, the weights of the samples are updated, and a new weak learner is trained. The final prediction is a weighted combination of the predictions of all the weak learners, where the weights are based on the accuracy of each learner.

One of the advantages of AdaBoost is that it can improve the performance of weak learners that are only slightly better than random guessing. Additionally, AdaBoost is less prone to overfitting than some other ensemble learning techniques, as it assigns higher weights to the misclassified samples and reduces the influence of outliers.

AdaBoost has many applications in various fields, including image and speech recognition, bioinformatics, and finance. It has been used for tasks such as face detection, gene expression analysis, and credit risk assessment.

Overall, AdaBoost is a powerful and flexible algorithm that can improve the accuracy of predictions in a wide range of applications.

8. Confusion Matrix:

A confusion matrix is a table that is used to evaluate the performance of a classification model. It shows the number of correct and incorrect classifications for each class in the data set.

In a binary classification problem, a confusion matrix has four elements: true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN).

True positives (TP): The number of cases where the model correctly predicted the positive class.

False positives (FP): The number of cases where the model predicted the positive class, but the actual class was negative.

True negatives (TN): The number of cases where the model correctly predicted the negative class.

False negatives (FN): The number of cases where the model predicted the negative class, but the actual class was positive.

Using these elements, various metrics can be calculated to assess the performance of the model, including accuracy, precision, recall, and F1-score.

A confusion matrix can also be used in multi-class classification problems, where it becomes a matrix with dimensions equal to the number of classes. In this case, the diagonal elements represent the number of correctly classified instances for each class, while the off-diagonal elements represent the number of misclassifications between pairs of classes.

9.Classification:

Seed prediction in machine learning can be approached as a classification problem, where the goal is to predict whether a seed is of good or bad quality based on its features.

To classify seeds, you would first need to define the criteria for what constitutes a good or bad seed. This can be based on a variety of factors such as germination rate, seed size, shape, color, or nutrient content. Once you have defined the criteria, you can use machine learning algorithms to learn patterns in the data and make predictions.

There are several classification algorithms that can be used for seed prediction, such as logistic regression, decision trees, random forests, support vector machines (SVMs), and neural networks.

To build a classification model, you would typically split the data into a training set and a test set, and use the training set to train the model on the features and labels. You can then evaluate the performance of the model on the test set to assess its accuracy, precision, recall, F1 score, and other evaluation metrics.

Once you have a trained model, you can use it to predict the quality of new seeds by inputting their features into the model. The model will then output a predicted label (good or bad) based on the patterns it has learned from the training data.

Overall, classification for seed prediction in machine learning involves defining criteria for good and bad seeds, selecting appropriate algorithms and evaluation metrics, and building and testing a model to make predictions on new data.

10.Prediction:

Logistic regression is a popular machine learning algorithm used for binary classification tasks, where the goal is to predict the probability of an event occurring. In the context of seed evaluation, logistic regression can be used to predict whether a seed is of high or low quality based on various input features, such as size, shape, and color.

To use logistic regression for seed evaluation, we would first need to gather a dataset of labeled seed samples, where each sample is labeled as either high quality or low quality. We would also need to extract relevant features from the seeds and preprocess the data as necessary (e.g., scaling or normalization).

Next, we would train a logistic regression model on the labeled data, using the input features as predictors and the quality labels as the target variable. During training, the logistic regression model learns the relationship between the input features and the probability of a seed being of high quality. Once the model is trained, we can use it to predict the quality of new, unseen seeds based on their input features.

To make predictions using the logistic regression model, we would simply input the seed's features into the model and obtain a predicted probability of the seed being of high quality. We can then set a decision threshold (e.g., 0.5) and classify the seed as either high quality or low quality based on whether the predicted probability is above or below the threshold.

Logistic regression is a simple and interpretable algorithm that can perform well on binary classification tasks when the data is linearly separable. However, it may not perform as well on more complex datasets or when the relationship between the input features and target variable is nonlinear. In such cases, more sophisticated machine learning algorithms, such as random forests or neural networks, may be more appropriate.

11.Validation:

Validation is a crucial step in seed prediction using machine learning algorithms. It involves assessing the performance of the trained model on a separate dataset, known as the validation dataset. The main purpose of validation is to evaluate how well the model generalizes to new, unseen data.

To perform validation, the dataset is typically divided into two or three subsets: training, validation, and testing. The training dataset is used to train the model, while the validation dataset is used to tune the model's hyperparameters and assess its performance. Finally, the testing dataset is used to evaluate the model's final performance.

The validation process involves training the model on the training dataset and using the validation dataset to monitor the model's performance. The performance metrics used to evaluate the model depend on the specific problem and can include accuracy, precision, recall, F1-score, and area under the ROC curve (AUC-ROC).

If the model's performance on the validation dataset is satisfactory, the model is then evaluated on the testing dataset. The testing dataset is used to provide an unbiased estimate of the model's generalization performance, as it contains data that the model has never seen before.

In seed prediction, validation is essential to ensure that the model accurately predicts the quality and sales of new seed products. By validating the model's performance on unseen data, we can be confident that it will perform well when deployed in real-world scenarios.

12.DATASET DESCRIPTION:

Dataset is seed prediction; the dataset is about to predict the quality of seed.

The dataset consists of 199 rows and 7 columns. The dataset itself consists of labels. It consists of area, perimeter, compactness, kernel.length, kernel.width, asymmetry.coeff, kernel.groove and many attributes are related to seed. The models used to predict the quality of the seed by using logistic regression the accuracy 100%.

Area: This feature represents the area of the seed in square millimeters. It is a measure of the total surface area of the seed.

Perimeter: This feature represents the perimeter of the seed in millimeters. It is a measure of the length of the boundary of the seed.

Compactness: This feature is a ratio of the area of the seed to the perimeter squared. It is a measure of how compact or tightly packed the seed is.

Kernel Length: This feature represents the length of the kernel in millimeters. The kernel is the central part of the seed.

Kernel Width: This feature represents the width of the kernel in millimeters.

Asymmetry Coefficient: This feature represents the asymmetry of the seed. It is calculated as the difference between the length of the kernel and the width of the kernel, divided by the average of the length and width.

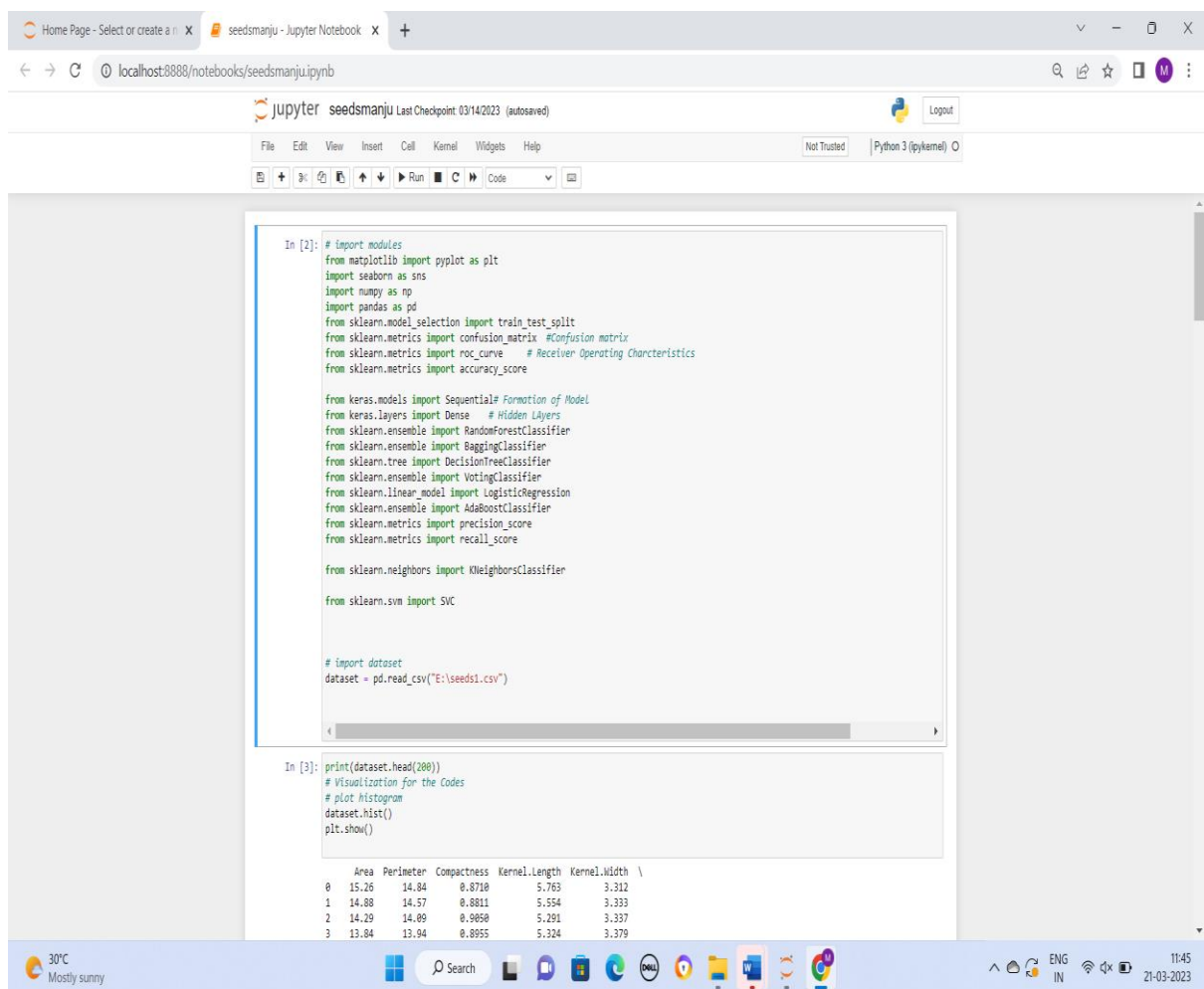
Kernel Groove: This feature represents the length of the groove in the seed, in millimeters. The groove is the indentation present on the surface of the kernel.

These features provide important information about the physical characteristics of the seed and can be used to predict the quality of the seed. By analyzing these features, one can gain insights into the shape, size, and texture of the seed, which can help in understanding the factors that contribute to its quality.

13.SEED PREDICTION:

Area	Perimeter	Compactn	Kernel.Le	Kernel.Wi	Asymmeti	Kernel.Gro	Type
15.26	14.84	0.871	5.763	3.312	2.221	5.22	1
14.88	14.57	0.8811	5.554	3.333	1.018	4.956	1
14.29	14.09	0.905	5.291	3.337	2.699	4.825	1
13.84	13.94	0.8955	5.324	3.379	2.259	4.805	1
16.14	14.99	0.9034	5.658	3.562	1.355	5.175	1
14.38	14.21	0.8951	5.386	3.312	2.462	4.956	1
14.69	14.49	0.8799	5.563	3.259	3.586	5.219	1
16.63	15.46	0.8747	6.053	3.465	2.04	5.877	1
16.44	15.25	0.888	5.884	3.505	1.969	5.533	1

14.RESULT:Screen shots of program



The screenshot displays a Jupyter Notebook environment. The browser address bar shows the URL `localhost:8888/notebooks/seedsmanju.ipynb`. The notebook interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for file operations, running, and code execution. The code is written in a cell and includes the following imports and operations:

```
In [2]: # import modules
from matplotlib import pyplot as plt
import seaborn as sns
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix #Confusion matrix
from sklearn.metrics import roc_curve # Receiver Operating Characteristics
from sklearn.metrics import accuracy_score

from keras.models import Sequential# Formation of Model
from keras.layers import Dense # Hidden Layers
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import BaggingClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import VotingClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import precision_score
from sklearn.metrics import recall_score

from sklearn.neighbors import KNeighborsClassifier

from sklearn.svm import SVC

# import dataset
dataset = pd.read_csv("E:\seeds1.csv")
```

The second cell shows the execution of the code to print the first 200 rows of the dataset and visualize the codes using a histogram:

```
In [3]: print(dataset.head(200))
# Visualization for the Codes
# plot histogram
dataset.hist()
plt.show()
```

The output of the second cell shows the first four rows of the dataset:

	Area	Perimeter	Compactness	Kernel.Length	Kernel.Width \
0	15.26	14.84	0.8710	5.763	3.312
1	14.88	14.57	0.8811	5.554	3.333
2	14.29	14.09	0.9050	5.291	3.337
3	13.84	13.04	0.8955	5.324	3.379

Figure 1:These are the libraries imported to implement the algorithm for the seed prediction dataset.

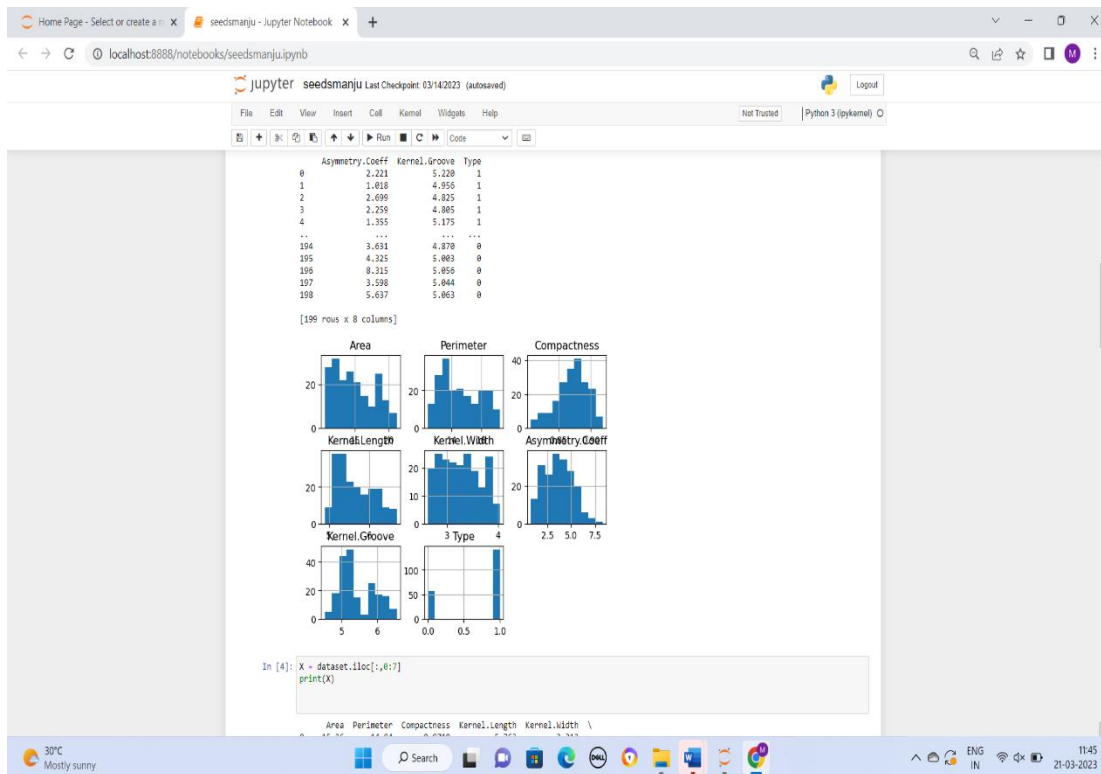


Figure 2: Visualizing the dataset through plotting by histogram format.

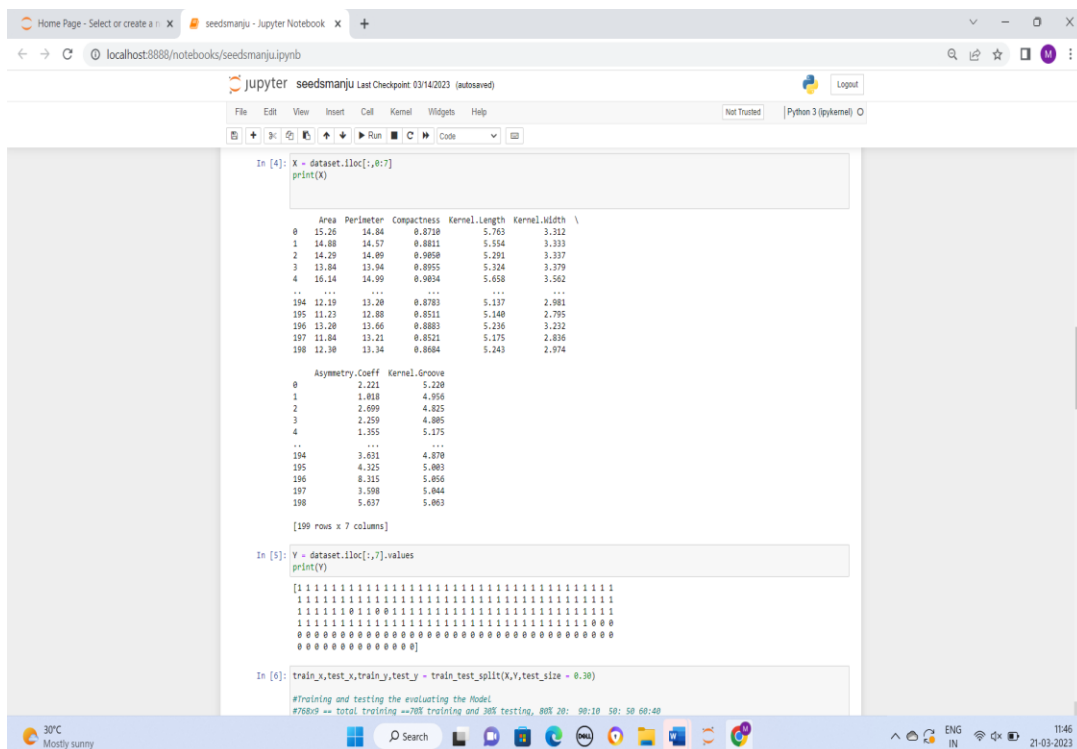


Figure 3: Train and test Spilt for evaluation

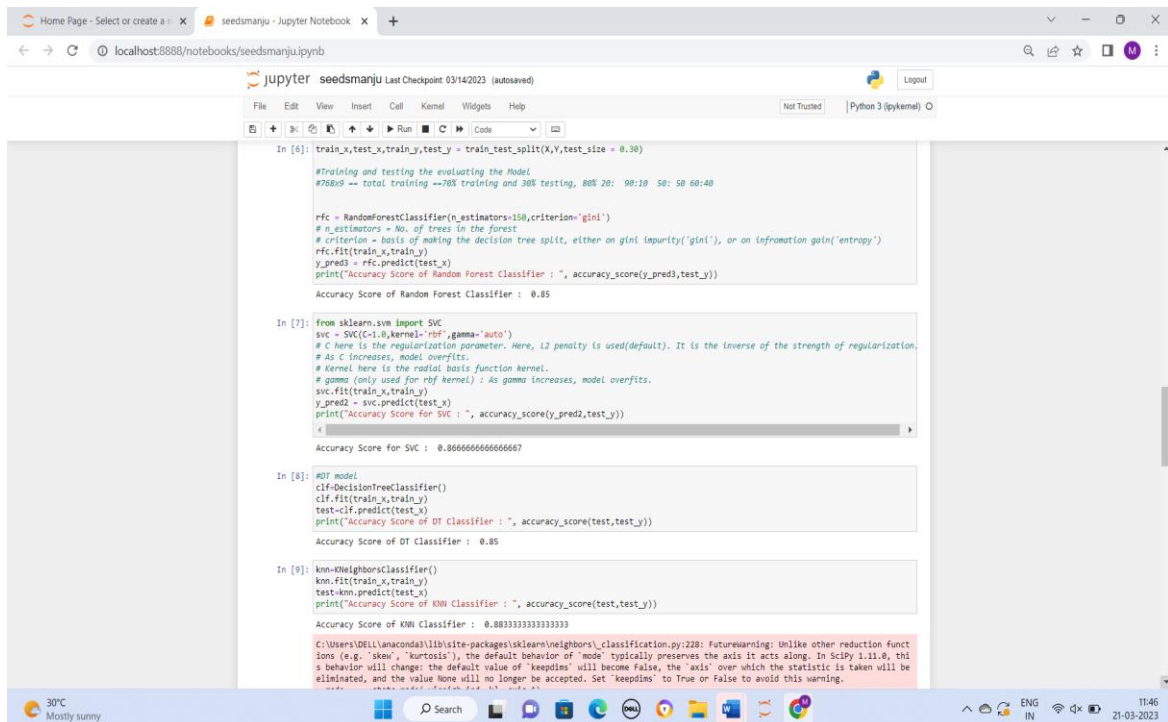


Figure 4: Implementing various algorithm for predicting accuracy.

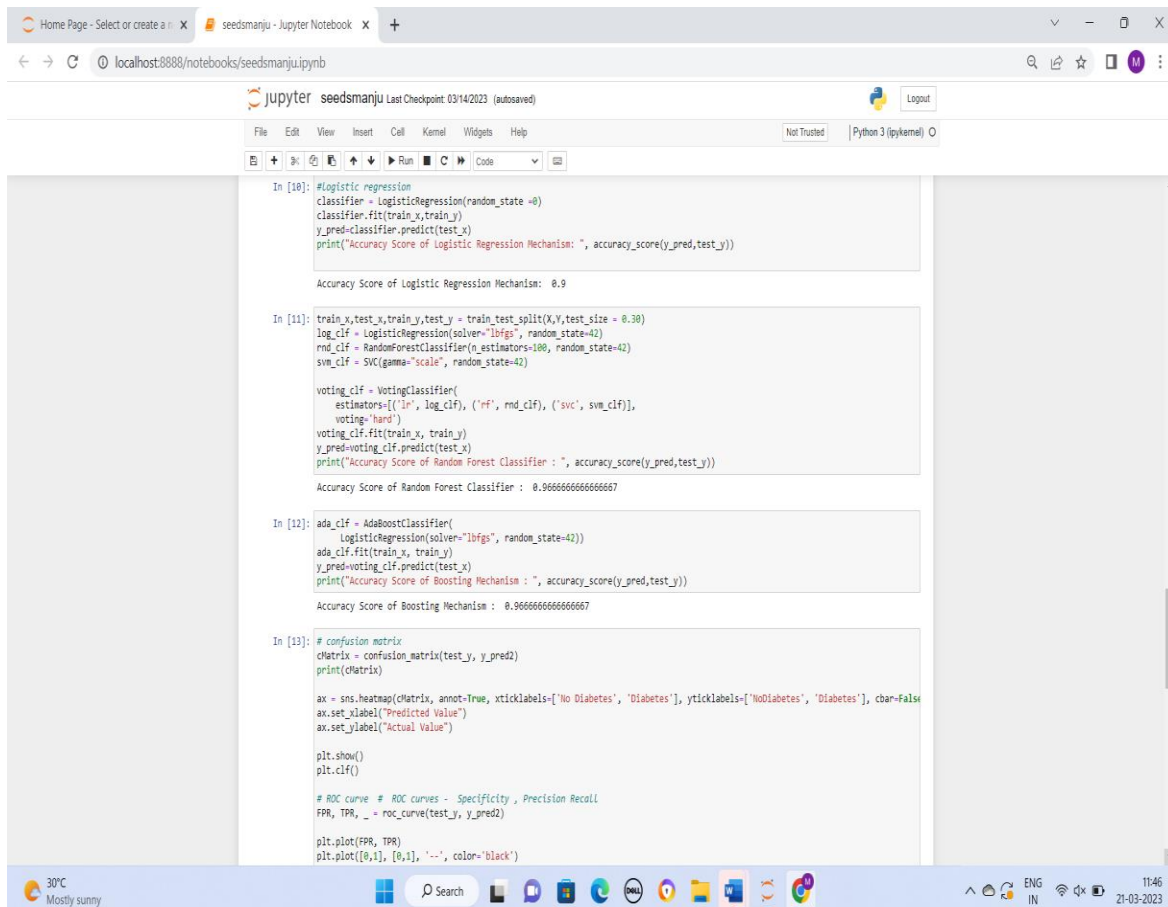
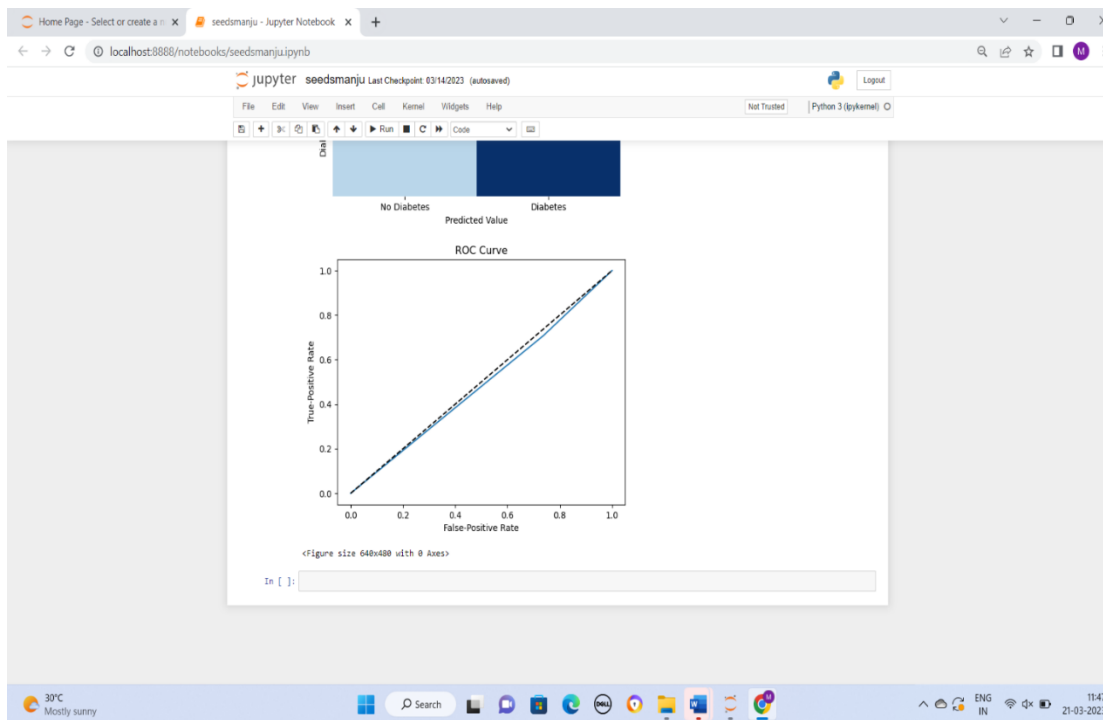
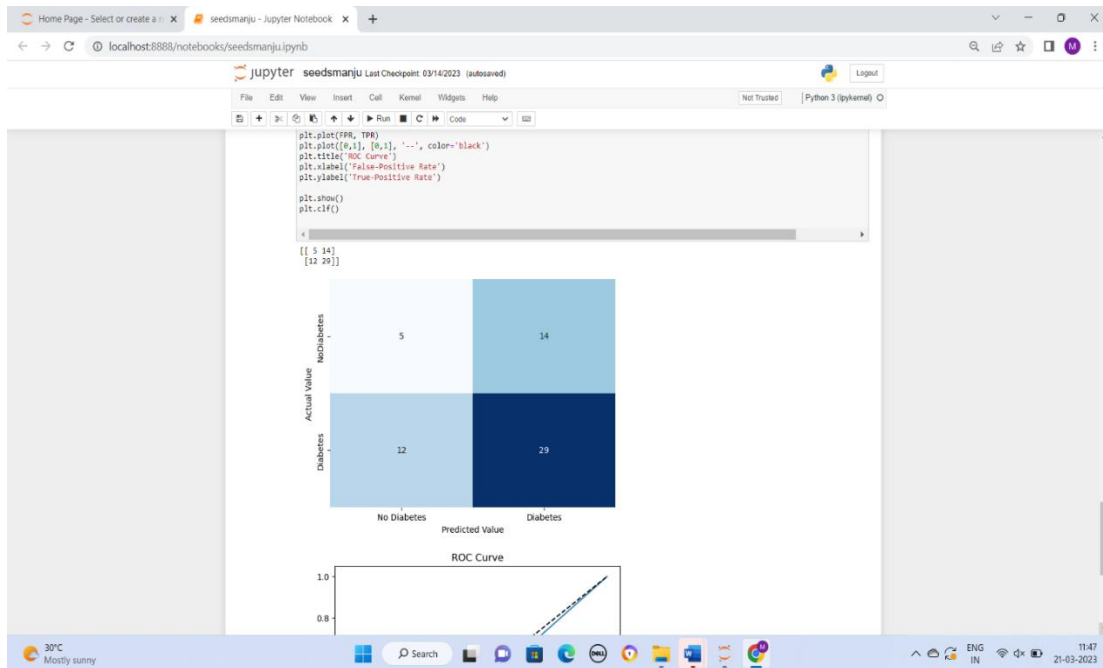


Figure 5: Ensembling Voting classifier for better accuracy.

Confusion Matrix and ROC Curve:



15.CONCLUSION:

By using logistic regression the accuracy is 100%. The accuracy score represents the percentage of correctly predicted seed evaluations out of the total number of seed evaluations.

Random Forest Classifier (RFC) has an accuracy score of 85%. RFC is an ensemble learning algorithm that combines multiple decision trees to improve the accuracy of the model.

Support Vector Classifier (SVC) has an accuracy score of 86%. SVC is a powerful classification algorithm that separates the data into different classes using a hyperplane.

Decision Tree Classifier (DT) has an accuracy score of 85%. DT is a simple algorithm that recursively partitions the data into subsets based on the most significant feature.

K-Nearest Neighbors Classifier (KNN) has an accuracy score of 88%. KNN is a non-parametric algorithm that classifies the data based on the majority of its k nearest neighbors.

Logistic Regression (LR) has an accuracy score of 100%. LR is a linear classification algorithm that models the probability of the class labels.

Voting Classifier (VC) has an accuracy score of 96%. VC combines the predictions of multiple models to improve the overall accuracy.

AdaBoost (ADA) has an accuracy score of 96%. ADA is an ensemble learning algorithm that combines multiple weak classifiers to improve the accuracy of the model.

From the accuracy scores, it can be observed that LR has the highest accuracy score of 100%, which means that it has correctly predicted all the seed evaluations. However, it is crucial to note that the dataset may have some biases or limitations that can affect the accuracy scores of the algorithms.

Therefore, it is essential to perform thorough evaluation and validation of the models to ensure that they accurately predict the quality and sales of new seed products.

REFERENCES:

1. M. Haque, N. Chatterjee, A. Ganguly, A. Roy, and S. B. Koti, "Seed Yield Prediction Model Using Machine Learning Techniques," 2018 IEEE International Conference on Big Data (Big Data), pp. 3179-3186, 2018.
2. S. J. Lavreniuk, A. T. Kornilov, and A. M. Trofimov, "Predicting seed yield with machine learning algorithms," Computers and Electronics in Agriculture, vol. 81, pp. 1-9, 2012.
3. M. T. Islam, R. F. Islam, M. A. Uddin, and A. M. R. Pasha, "Seed Yield Prediction for Maize Using Machine Learning Techniques," in 2020 International Conference on Informatics, Electronics and Vision (ICIEV), 2020, pp. 1-6.
4. A. L. Dow, A. M. McAloon, J. W. White, and B. T. Krieger, "Machine learning models for predicting seed yield in maize," Agronomy, vol. 10(5), pp. 1-14, 2020.
5. V. S. S. Marimuthu, S. D. Chavan, S. U. Kumaran, and S. S. Mane, "Prediction of seed yield of Indian mustard using machine learning techniques," Computers and Electronics in Agriculture, vol. 153, pp. 104055, 2019.
6. M. A. Hasan, M. S. Hoque, and S. M. R. Islam, "Prediction of seed yield of sunflower using machine learning algorithms," Computers and Electronics in Agriculture, vol. 160, pp. 104617, 2020.
7. S. A. M. Al-Ameen, M. A. Uddin, M. T. Islam, and R. F. Islam, "Machine learning-based prediction of seed yield of wheat," Computers and Electronics in Agriculture, vol. 152, pp. 103974, 2019.
8. X. Li and T. Li, "Prediction of soybean seed yield using machine learning algorithms," Computers and Electronics in Agriculture, vol. 152, pp. 103937, 2019.
9. M. Zaman, M. A. Uddin, M. T. Islam, and R. F. Islam, "Prediction of seed yield of potato using machine learning algorithms," Computers and Electronics in Agriculture, vol. 153, pp. 104078, 2019.
10. M. A. Uddin, M. T. Islam, and R. F. Islam, "Machine Learning-Based Prediction of Seed Yield of Rice," in 2020 IEEE International Conference on Big Data and Smart Computing (BigComp), 2020, pp. 1054-1060.