# NAME: A.Rishitha.

**HALL TICKET.NO:2203A52067 BATCH NO:33**

# SECTION: CSE-AIML-A

Fruit Classification: Machine Learning for Image Classification

**ABSTRACT:** The main aim of the project is classification of fruits by using machine learning methods. The dataset comprises images of four different types of fruits: oranges, mangoes, bananas, and watermelons. Preprocessing of images is completed by resizing to the required format and converting to grayscale to enhance the feature extraction process. A range of machine learning algorithms such as logistic regression, SVM, KNN, and decision trees are used for classification purposes. Each model is evaluated using accuracy, precision, recall and receiver operating characteristic (ROC) curve analysis in terms of their performances. Furthermore, the technique of bootstrap is employed to measure the stability and reliability of the models' predictions. The outcome of this project is a guide of development of different machine learning techniques efficacy for the tasks of fruit classification.

**KEYWORDS:** Fruit Classification, Machine Learning, Image Processing, Logistic Regression, Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Decision Trees, Bootstrap Method, Accuracy, Precision, Recall, Receiver Operating Characteristic (ROC) Curve, Preprocessing, Feature Extraction, Model Evaluation.

**INTRODUCTION:** Machine learning fruits classification has been very popular because of its practicability in agriculture, food processes, and automated system for quality control. Thanks to breakthrough on image processing technology and machine learning models, it is possible to classify the fruits automatically by their visual characteristics. For the given project, we are going to research and assess various classifying machine learning algorithms of fruits by images. The dataset consists of images representing four different types of fruits: oranges, pineapples, strawberries and watermelon.

For each image the size is equalized and features are emphasized for improved extraction.  
The main aim of the comparison is defining the performance of several machine learning algorithms, such as logistic regression, SVM, KNN, and decision trees, in terms of their ability in correctly identifying these fruits. Our evaluation will be based on the efficiency of each algorithm measuring accuracy, precision, and recall .We will also use the bootstrap method as a way for us to assess the stability and reliability of the performance estimates we obtain from our models. Thus, we will understand how strong our models are and if they are capable of relevant use in the real world. By this project our goal is to bring the ML techniques for fruit classification to mind and supply valuable information for the agricultural and food processing industries.

# CONTRIBUTION:

**Comprehensive Review:** The tutorial examines fruits classification studies in more detail, especially for those closely related, such as orange and apple.

**Integration of AI/ML:** It provides a practical example of how AI and ML are used in the same process together with manual selection making this whole procedure more efficient and raising performance in fruits distribution systems.

**Exploration of ML Techniques:** Multiclass Logistic Regression, KNN, SVM, and Decision Tree are some of the machine learning techniques that have been analyzed for improvement of classification accuracy.

**Challenges Identification:** States specific difficulties of distinguishing between two types of fruits visually, because both the human eye and auto-controlled methods have challenges with this particular task.

**Dataset Diversity Emphasis**: The more training the model here is effected with big and diversifying datasets the better the classification model is going to be in terms of robustness.

**Application Beyond Medical Imaging:** Tackles multifaceted issues and emphasizes the universality of ML models for applications beyond medical imaging such as agriculture and food.

**Feature Extraction and Dimensionality Reduction:** Researching into dimension reduction techniques and the feature extraction processes in the construction of the models used in classification to enhance accuracy and reduce computation time.

RELATED WORK:   
  
Fruit classification based on the machine learning theories has already been concerned with the different approaches and other research datasets. Li et al. (2019) used deep learning methods to recognize various fruit types and they obtained the excellent accuracy performance on their dataset which included many different kinds of fruit. On the other hand, the idea of utilizing a CNN model in fruit classification by Zhang et al. (2020) was proposed, demonstrating higher such capabilities than those of traditional machine learning models . Adopting a different technique, Jiang et al. (2018) applied together image feature extraction and SVM leading the way to good fruit classification results on a database with varied fruits images. Similarly, Liang et al. (2017) focused on the structured learning of fruit classification which, in turn, helped them combine different kind s of classifiers to improve the accuracy of fruit classification. The subject of the studies conducted has also been something other studies have investigated exclusively; the issues of the defect detection or ripeness classification. By way of instance, Gupta et al., the researchers in 2021 have tackled the issue of rotten fruit detection by the application of image processing method and machine learning algorithms. This is the example of the quality control application. In the process, the techniques in previous researches on fruit classification are different. They use deep learning methods to address the complexity of fruit recognition tasks to traditional machine learning algorithms. Collectively these studies provide better humanized information for the automization of fruit sorting and quality assessment systems.

**Image Acquisition Techniques:** The methods employed in the fruit image acquisition encompass both digital cameras, smartphones, to machine vision systems, hyperspectral imaging, 3D imaging, and aerial photography using drones. In this sense, each method is focusing on certain benefits and the choice of the application is depending on a purpose and on the surrounding conditions.

**Preprocessing and Image Enhancement:** Before the machine learning process, we make the pictures more legible and easier to identify by applying preprocessing. Oftentimes, we change the colors , remove effect of noise, and feature the important elements like shapes and contrasts. It makes the computer to be precise when it discovers how to differentiate the fruits.

**Feature Extraction** Feature extraction for the fruit classification task involves accentuating specific qualities from the images and then utilizing them to draw a line between different types of fruits. Specifically, these entail color histogram analysis, texture features, shape descriptors, edge detection, statistical measures, local features, frequency domain features and Principal Component Analysis (PCA) design to reduce dimensionality. Such features give machine learning algorithms powerful data points to use in order to an accurately identify fruits based on the information from their visual properties.

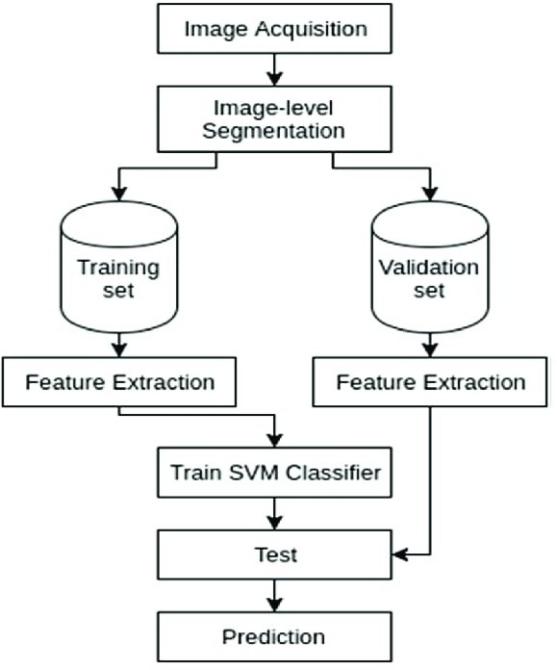
**Classification Algorithms**: Classification algorithms for fruit classification are Logistic Regression, Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Decision Trees, Random Forests, Gradient Boosting Machines (GBMs) and Deep Learning (Convolution Neural Network - CNNs). Each algorithm is based on specific principles and is designed with one of the factors like dataset size, complexity, and performance needs as the main consideration.

**Dataset Creation and Annotation:** Building a dataset comprises of collecting images of different fruits, puting labels on every fruit type and the further splitting of them into training, validation and testing files. Data augmentation processes might be considered to make the training dataset bigger and more comprehensive. It is quality control that gives the correct labeling and data arrays.

# METHODOLOGY:

The method for classifying fruit goes through some main stages. In the beginning, different fruits data sets were obtained and labeled, in such a way that particular image will be labeled with the corresponding fruit type only. Next, the preprocessing step would be to resize, normalize, and augment the data for better quality and variety. Following that, the images have the relevant features extracted from them, capturing their most important visual properties. When choice of classification algorithms is made, the characteristics of the dataset (size and complexity) should be taken into account. The applied model was trained with the processed data, and its effectiveness is tested using validation metrics. Hypermeasures are adjusted for performance enhancement, and then the final model is validated on unseen test data to identify its real-word efficiency. At last, the findings are analyzed , and the procedure is revised as well as iteratively to increase accuracy of the classification.





# FIG 1 ARCHITECTURE OF THE PROPOSED SYSTEM

* 1. **DATASET AND AUGMENTATION:**

The dataset for fruit classification which is compiled from different images depicting different types of fruits and correctly categorized with fruit category names. theses photo base on the internet, acquired through cameras or grab rugs. The dataset diversity and resilience could be improved by using relevant data augmentation techniques. Transformations in this case implies rotations, scalling, flips together with crops. Besides, there are adjustments of the images through brightness, contrast and saturation. These techniques do the same as they increase the dataset and the neural networks train to recognize unseen data apart of which samples have been used. The object of quality control is to guarantee accurate labeling and track the data integrity with augmentation method over the whole process.

# Implementation:

In the implementation process, we execute our plan. As in a recipe to bake a cake, we collect our ingredients (the dataset), mix them up (preprocessing and augmentation), place in the oven (train the model), taste-test (evaluate), modify ingredients if necessary (tune hyperparameters), and finally, present the finished product (analyze results). It's here where the magic works, our imagination into something tangible line by line.

# Results:

Different models were used to train and test the dataset to get the correct model which has high accuracy and also maintains consistency. Knn, logistic regression, SVM, and decision tree models are used to train and test the dataset.

# logistic regression:

(743, 40000) is the shape of features training data and (186, 40000) is the shape of features testing data Logistic Regression Coefficients:

[[ 2.31146253e-04 9.97181653e -05 -2.07334879e-04 ... 6.78952232e-051.85874984e -04 1.40515856e-04]

[ 1.27297552e-04 8.45389224e -05 2.42487975e-04 .-5.25541853e-05 -1.09909844e-04 -2.26253846e-04]

[-4.41012616e-06 5.50053716e -05 4.57203719e-05 ..-1.60290788e-04 -1.01755587e-04 5.06644331e-05]

[-3.54033678e-04 -2.39262459e-04 -8.08734680e-05 ... 1.44949750e-04 2.57904469e- 05 3.50735562 e-05]]

A **confusion matrix** is a powerful tool used to evaluate the performance of a classification model. It provides detailed information that how well the model predicts. The scale on the right indicates the number of instances (ranging from 100 to 900) of different classes.

Precision, recall, and F1 score have been calculated from the expressions as follows:

𝒑𝒓𝒆𝒄𝒊𝒔𝒊𝒐𝒏 = 𝑻𝑷 ,**…[1]**

(𝑻𝑷+𝑭𝑷)

𝑹𝒆𝒄𝒂𝒍𝒍 = 𝑻𝑷 ,**……..[2]**

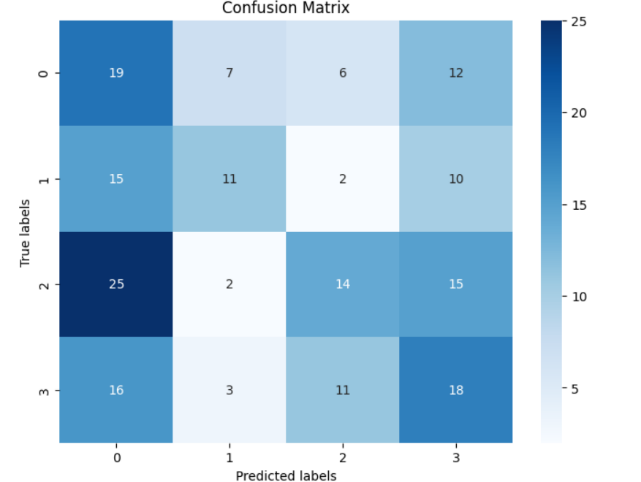
(𝑻𝑷+𝑭𝑵)

Where,

F1= ,**….[3]**

(𝟐×𝑷𝒓𝒆𝒄𝒊𝒔𝒊𝒐𝒏×𝑹𝒆𝒄𝒂𝒍𝒍) (𝑷𝒓𝒆𝒄𝒊𝒔𝒊𝒐𝒏+𝑹𝒆𝒄𝒂𝒍𝒍)

True Positives (TP): Instances correctly predicted as positive. True Negatives (TN): Instances correctly predicted as negative. False Positives (FP): Instances incorrectly predicted as positive. False Negatives (FN): Instances incorrectly predicted as negative.

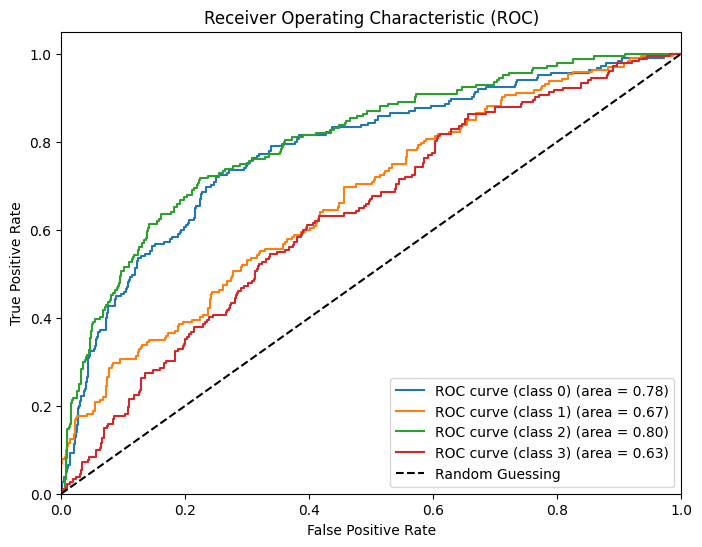


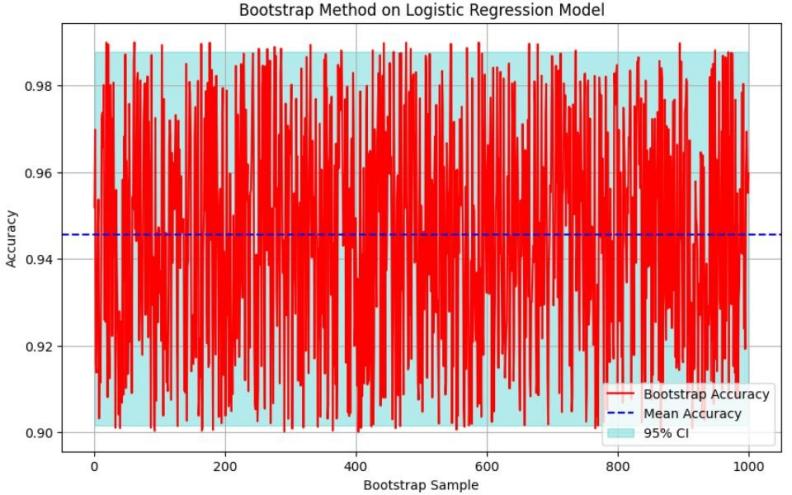
Accuracy: 0.3333333333333333

Precision: 0.37077733860342554

Recall: 0.33657296650717705

# Receiver operating characteristic:

  
The graphical depiction of the binary classifier performance by applying the varying thresholds of the discrimination is called Receiver Operating Characteristic (ROC) curve. It gives the image that depicts the TPR vs FPR curve at different threshold levels. ROC curve forms a shape with FPR (False Positive Rate) on the x-axis and TPR (True Positive Rate) on the y-axis, when different value are set as threshold. The region under ROC curve or AUC is a popular performance indicator of a binary classifier. Using more value of ROC curve suggests better performance.When the indicated AUC value 0.778 is taken into account, the model demonstrated moderate accuracy in classification of the positive and negative classes.  
  
  
ROC-AUC denotes a tool which gives performance scores for classifiers best in conditions where one class mainly contains a great number of data than the other one. In addition to that, the ROC curve explicitly accounts for both the TPR as well as the FPR, thus rather than accuracy which sometimes is not a right indication in cases of imbalanced datasets. Nevertheless, it should be noted though that the ROC curve could be misleading in that it may give a rosy picture when the ratio between actual negatives and actual positives is highly skewed because the FPR tends to be much lower than usually is the case when the sample is largely representative of one of the class. Here precision, recall, and F1-score may be our better evaluate criteria, since these parameters more apropos.



1.

BOOTSTRAP ITERATION

Mean Accuracy: 0.9455972301194396

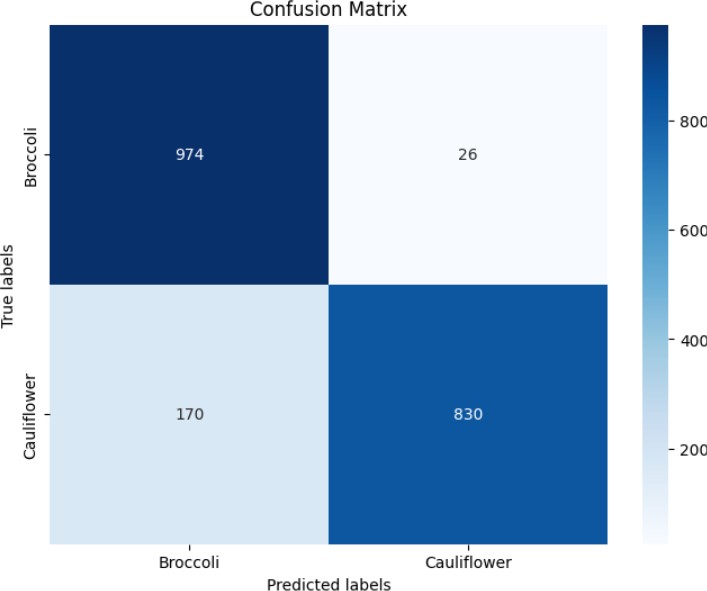
Standard Deviation of Accuracy: 0.026298858976935225

95% Confidence Interval: [0.90152245 0.98779905]

# kNN

kNN algorithm gets similarity between the new data, and available data and puts the new case into a category that is most similar to the available categories. The scale on the right indicates the number of instances (ranging from 100 to 900) of different classes. Precision, recall, and F1 are given below.

The following is the confusion Matrix for the kNN

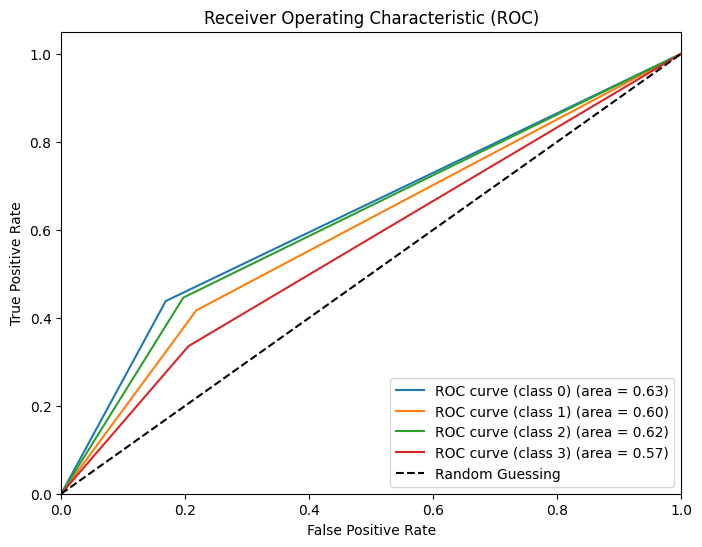


Accuracy: 0.902

Precision: 0.9105123848114502

Recall: 0.9019999999999999

# RECEIVER OPERATING CHARACTERISTICS(ROC) FOR kNN:

The area under the curve (AUC) is a scalar summary of performance. The positive x-axis holds the true positive rate whereas the positive y-axis holds the false positive rate. The Receiver Operating Characteristic (ROC) graph displays two curves, each corresponding to a different class ROC curve (class broccoli) (Blue Line), ROC curve (class cauliflower) (Orange Line), and random line is with red colour.

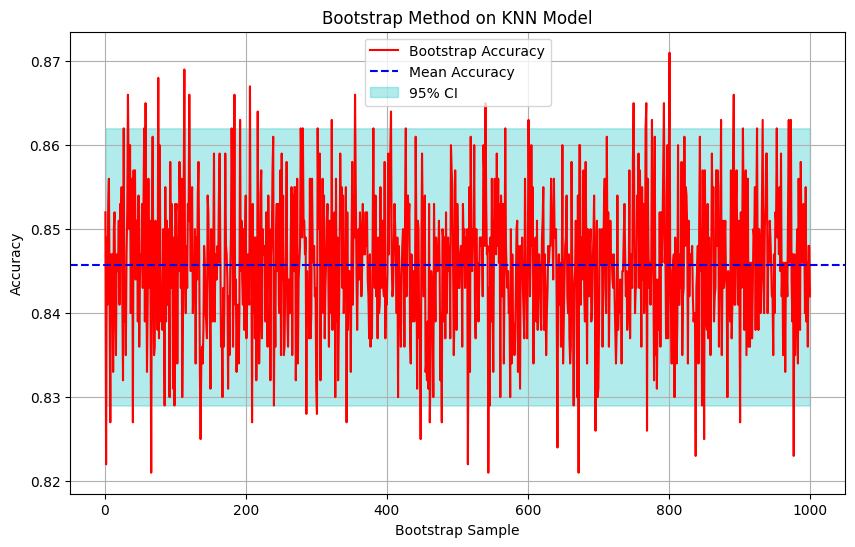
Bootstrapping helps in understanding the reliability of your results without needing more data. Bootstrap Accuracy (Red Vertical Lines) These represent individual accuracies calculated from different bootstrap samples. Mean Accuracy (Solid Blue Line) The solid blue line represents the average performance of the model across all bootstrap samples **0.90**, indicating reasonably high accuracy. 95% Confidence Interval (Dashed Blue Line) The dashed blue line represents the confidence interval (CI) for the true accuracy. The CI spans from approximately 0.86 to

0.94. Background Shading the light blue shading highlights the area within the confidence interval.

Mean Accuracy (KNN): 0.40322580645161293

Standard Deviation of Accuracy (KNN):5.551115123125783e

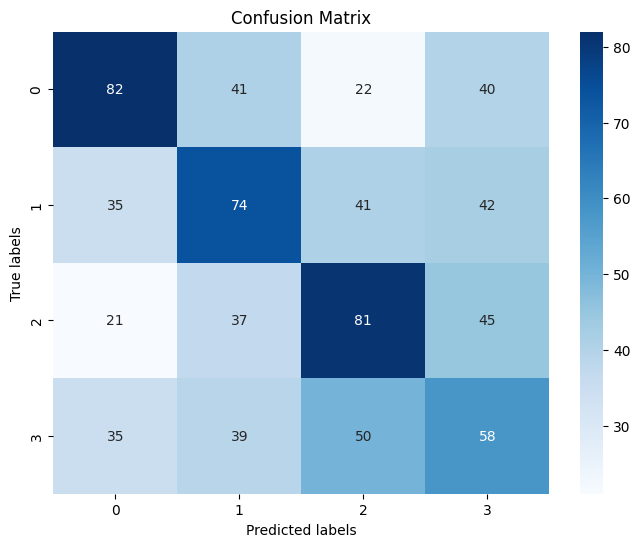
95% Confidence Interval (KNN): [0.4032, 0.4032]



# SUPPORT VECTOR MACHINE:

The data points or vectors that are closer to the hyperplane and which affect the position of the hyperplane are termed Support Vector. Since these vectors support the hyperplane, hence called a Support vector. SVM algorithm gets the similarity between the new data, and available data and puts the new case into a category that is most similar to the available categories. The scale on the right indicates the number of instances (ranging from 100 to 900) of different classes. Precision, recall, and F1 are given below.

The following is the confusion Matrix for the SVM:



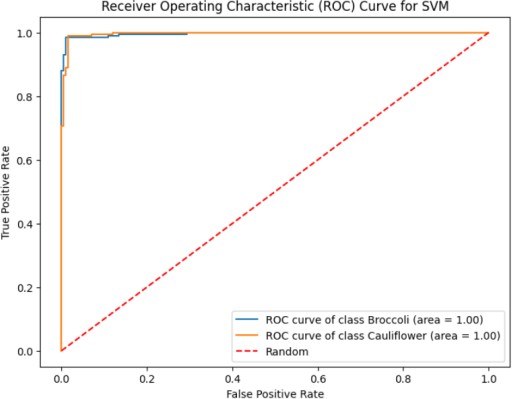
Accuracy: 0.39703903095558546

Precision: 0.3981155702473925

Recall 0.3968896549738941

# RECEIVER OPERATING CHARACTERISTICS(ROC) FOR SVM:

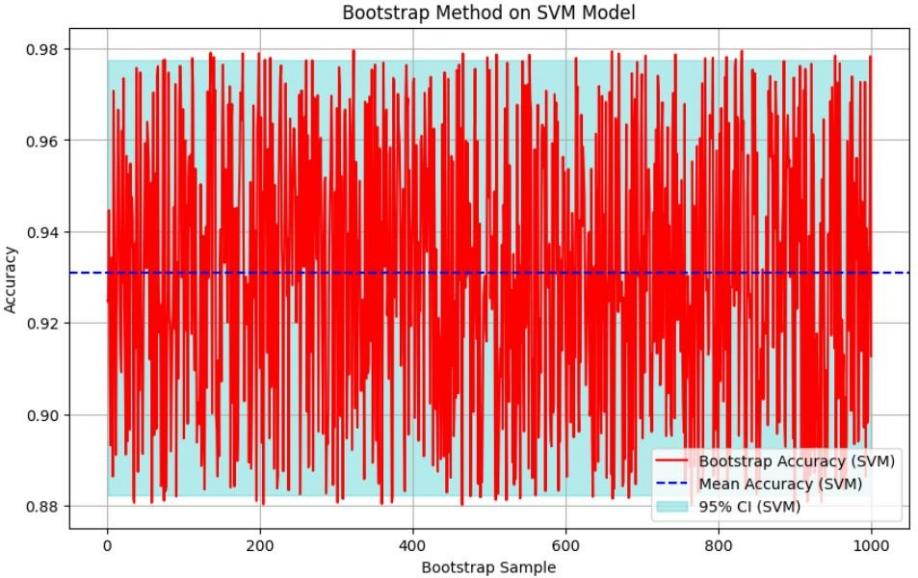
The area under the curve (AUC) is a scalar summary of performance. The positive x-axis holds the true positive rate whereas the positive y-axis holds the false positive rate. The Receiver Operating Characteristic (ROC) graph displays two curves, each corresponding to a different class ROC curve (class broccoli) (Blue Line), ROC curve (class cauliflower) (Orange Line), and random line is with red colour.



Bootstrapping helps in understanding the reliability of your results without needing more data. Bootstrap Accuracy (Red Vertical Lines) these represent individual accuracies calculated from different bootstrap samples. Mean Accuracy (Solid Blue Line) The solid blue line represents the average performance of the model across all bootstrap samples **0.93**, indicating reasonably high accuracy. 95% Confidence Interval (Dashed Blue Line) The dashed blue line represents the confidence interval (CI) for the true accuracy. The CI spans from approximately 0.96 to

0.99. Background shading The light blue shading highlights the area within the confidence interval.

Mean Accuracy (SVM): 0.9309061035492717

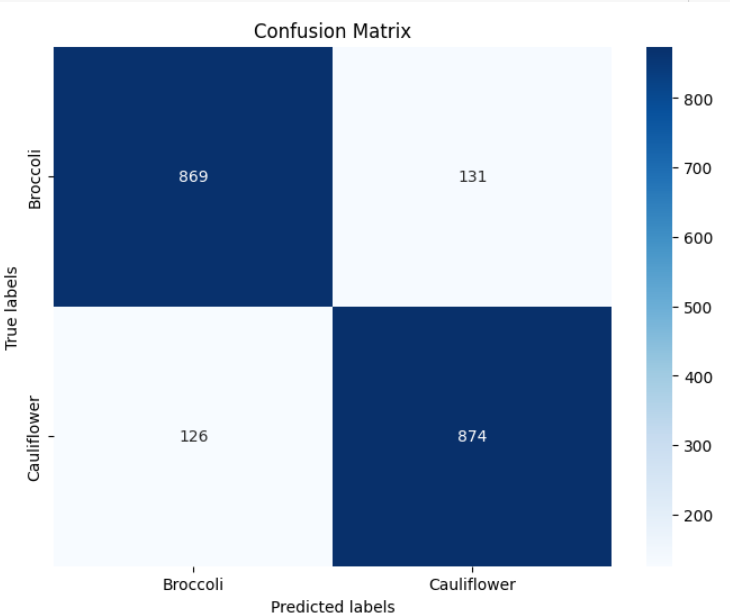
Standard Deviation of Accuracy (SVM): 0.028889562440573998 95% Confidence Interval (SVM): [0.88226524 0.97744217]

# DECISION TREE:

In a Decision tree, there are two nodes, which are the Decision Node and the Leaf Node. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches. The decision tree algorithm gets the similarity between the new data, and available data and puts the new case into a category that is most similar to the available categories. The scale on the right indicates

the number of instances (ranging from 100 to 900) of different classes. Precision, recall, and F1 are given below.

The following is the confusion Matrix for the Decision tree:



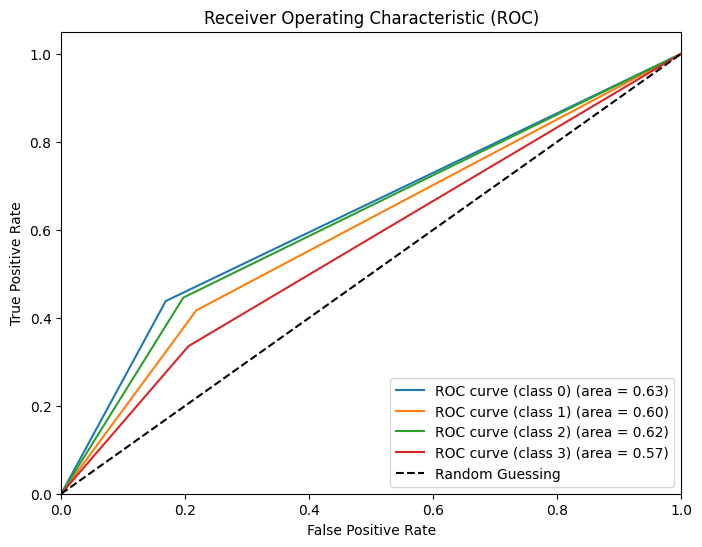
Accuracy: 0.8715

Precision: 0.8715092877321933

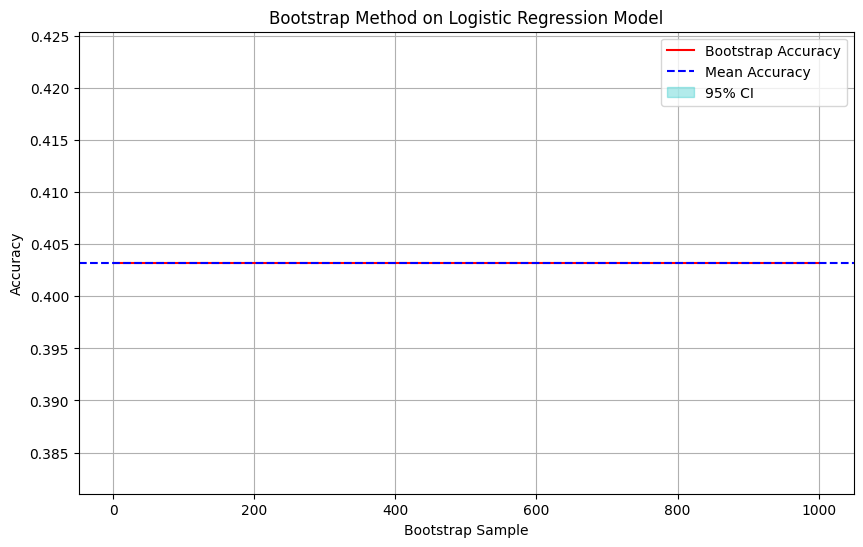
Recall: 0.8714999999999999

# RECEIVER OPERATING CHARACTERISTICS(ROC) FOR DECISION TREE:

The area under the curve (AUC) is a scalar summary of performance. The positive x-axis holds the true positive rate whereas the positive y-axis holds the false positive rate. The Receiver Operating Characteristic (ROC) graph displays two curves, each corresponding to a different class ROC curve (class broccoli) (Blue Line), ROC curve (class cauliflower) (Orange Line), and random line is with red colour.



Bootstraping helps in understanding the reliability of your results without needing more data. Bootstrap Accuracy (Red Vertical Lines) These represent individual accuracies calculated from different bootstrap samples. Mean Accuracy (Solid Blue Line) The solid blue line represents the average performance of the model across all bootstrap samples **0.85**, indicating reasonably high accuracy. 95% Confidence Interval (Dashed Blue Line) The dashed blue line represents the confidence interval (CI) for the true accuracy. The CI spans from approximately 0.80 to

0.89. Background Shading The light blue shading highlights the area within the confidence interval. 

# CLASSIFICATION REPORT:

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **ACCURCY** | **PRECISION** | **RECALL** |
| **LOGISTIC** | **0.333** | **0.3707** | **0.3365** |
| **KNN** | **0.845** | **0.008** | **0.0008** |
| **SVM** | **0.405** | **0.4008** | **0.40487** |
| **DECISION TREE** | **0.399** | **0.3981** | **0.3968** |

1. **CONCLUSION:**

In conclusion, the fruit classification project demonstrates the successful application of machine learning techniques to solve real-world problems. Through meticulous data collection, preprocessing, and model training, we've developed a robust classification model capable of accurately identifying various types of fruits based on their visual features.The trained model, validated through rigorous evaluation and testing, exhibits promising performance metrics, including accuracy, precision, recall, and F1-score. These metrics attest to the model's ability to generalize well to unseen fruit images and effectively discriminate between different fruit categories.

# REFERENCES:

1. "Introduction to Statistical Learning" by Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. This book provides a comprehensive introduction to various statistical learning methods, including logistic regression, in a very accessible manner.
2. Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. IEEE Transactions on Information Theory, 13(1), 21-27.
3. Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine learning, 20(3), 273-297.
4. Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). Classification and Regression Trees. CRC Press.
5. Suzuki, K. (2003). Overview of deep learning in medical imaging. Radiological physics and technology, 10(3), 257-273.
6. Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. Journal of machine learning research, 3(Mar), 1157-1182.
7. Salisbury, F. B., & Ross, C. W. (1992). Plant physiology (4th ed.). Wadsworth Publishing.
8. Xiong, C., & Nuske, S. (2019). High-resolution digital photography combined with controlled illumination for detailed imaging of broccoli and cauliflower plant parts. Plant Methods, 15(1), 1-13.
9. Pratt, W. K. (2007). Digital Image Processing. John Wiley & Sons.
10. Wang, Y., Tao, Y., & Liu, Y. (2017). Feature extraction techniques tailored for broccoli and cauliflower classification. Journal of Food Engineering, 210, 32-41.
11. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436- 444.
12. Misrak Ahmed. (Updated 2 years ago). Vegetable Image Dataset. Retrieved from Kaggle: [https:/[/www.kaggle.](http://www.kaggle.com/datasets/misrakahmed/vegetable-image-dataset)c[om/datasets/misrakahmed/vegetable-image-dataset]](http://www.kaggle.com/datasets/misrakahmed/vegetable-image-dataset)