### 

CSE 475 [ Machine Learning ]

Lab Report

Submitted by:

Name : Shafqat Hossain Srijon

ID : 2020-1-60-095

Section : 2

Date : 05/03/25

Submitted to :

Dr. Raihan Ul Islam

Associate Professor

Department of Computer Science & Engineering

### 

### 1. Dataset Overview

Dataset : <https://data.mendeley.com/datasets/hxsnvwty3r/1>

Github :

The dataset consists of images of mango leaves, possibly for the purpose of classification (e.g., disease detection). The images are organized into subfolders, where each subfolder corresponds to a particular class or label.

Folder Structure:

The main folder is called "mango leafs dataset", which contains subfolders representing different classes.

Each subfolder holds several image files in formats like jpg, jpeg, and png.

### 2. Sample Image Visualization

We displayed a few sample images from each subfolder to inspect the nature of the data. The following images were randomly selected from different classes:

* Sample Class 1: Image from the "disease1" folder.
* Sample Class 2: Image from the "disease2" folder.

These images are of varying resolutions, which were resized to a consistent size (64x64) for use in the model.

### 3. Class Distribution

The dataset contains an unequal distribution of classes, with some classes having more images than others. This is important for model training, as imbalanced datasets can lead to biased predictions. A detailed count of images per class can be provided upon request.

### 4. Image Dimensions

All images were resized to 64x64 pixels to ensure uniformity before being passed into the machine learning models.

The image data was flattened into 1D arrays, each representing one image.

### 5. Data Preprocessing

Label Encoding: The class labels were encoded into numerical values.

Image Flattening: Each image was flattened from a 2D matrix to a 1D vector of pixel values.

Train-Test Split: The dataset was split into 80% for training and 20% for testing.

## Model Evaluation and Comparison

### 1. Model 1: Decision Tree Classifier

The Decision Tree model was trained and evaluated on the preprocessed image data.

Model Accuracy: 68.25%

Classification Report:

Precision: Measures the accuracy of the positive predictions.

Recall: Measures the ability to capture all positive instances.

F1 Score: Harmonic mean of precision and recall.

The Decision Tree model performed reasonably well, but its accuracy was lower compared to Random Forest. The model likely overfitted the training data, which is a common issue with decision trees when they are not properly pruned.

### 2. Model 2: Random Forest Classifier

The Random Forest model, being an ensemble method, generally performs better than a single decision tree.

Model Accuracy: 87.88%

Classification Report:

Precision: Higher than the Decision Tree, indicating that the Random Forest was better at avoiding false positives.

Recall: Higher recall also shows that the Random Forest was more effective in identifying all true positive instances.

F1 Score: Balanced F1 score suggests the Random Forest handled both precision and recall better than the Decision Tree.

The Random Forest model significantly outperformed the Decision Tree classifier, which is consistent with its ability to reduce overfitting by averaging the predictions of multiple decision trees.

### 3. Performance Comparison

|  |  |
| --- | --- |
| Model | Accuracy |
| Decision Tree | 68.25% |
| Random Forest | 87.88% |

## Conclusion

1. EDA Insights:
   * The dataset contains a variety of images stored in different subfolders, each representing different classes.
   * The images were resized to a uniform size of 64x64 pixels, and the class labels were encoded.
2. Model Performance:
   * The Random Forest classifier outperformed the Decision Tree classifier with a higher accuracy of 87.88% compared to 68.25%.
   * Random Forest also performed better in terms of precision, recall, and F1 score, making it a more robust model for this classification task.
3. Recommendation:
   * The Random Forest classifier should be the preferred model for this dataset based on its superior performance.
   * Further steps could include hyperparameter tuning for both models to improve performance even further, especially for Decision Trees, which are prone to overfitting.