EAST WEST UNIVERSITY

CSE 475 – Section 3

Machine Learning

Lab Report – 01

Title	
Mango Leaf Disease Classification using Random Forest and Decision Tree	
Submitted To	**Submitted By**
Dr Raihan Ul Islam	Md.Safinur Rahman
Associate Professor	2019-3-60-019
Department of Computer Science and Engineering	
Date Submitted: 9th November, 2024	

Introduction

Mango leaf diseases are commonly identified by the attributes and visual aspects of the leaves themselves. Hence, we can use classification algorithms like Decision Tree and Random Forest to classify what class a specific leaf belongs to.

In this report, we go through the process of understanding the Mango Leaf dataset, how the features are defined and how we use the classification models to predict the class of these leaves.

Objectives

We have a few clearly defined objectives which will guide us throughout this lab exercise. They are:

1. Use Exploratory Data Analysis techniques (e.g., Univariate Analysis, Bivariate Analysis, Histograms, Scatter Plots) to analyze and gain better understanding of the patterns and other hidden information present in the dataset.
2. Implement an ID3 or Decision Tree classification model to classify the dataset entries into their correct classes.
3. Implement a Random Forest classification model to achieve the same objective as objective (2).
4. Evaluate and compare performance of the two classification models using accuracy, precision, F-1 and recall scores.

Describing the Dataset

The dataset, titled “MangoLeafBD”, is a compilation of 4000 images of mango leaves. Each of these leaves are categorized by their overall condition (e.g., Health, Die Back, Cutting Weevil).

Below is a summary of all the specifications of this dataset.

Type of Data	240x320 images of mango leaves
Data Format	JPG
Number of Instances	4000
Classes	8 (Anthracrose, Bacterial Canker, Cutting Weevil, Die Back, Gall Midge, Healthy, Powdery Mildew, Sooty)
Distribution of Instances	500 images per class
How the data is acquired	Captured from mango trees through the mobile phone camera.

Dataset Preprocessing

Computers only understand numbers, so we can’t really feed our dataset, which consists of images, into any model. We have to find someway to tell the computer which set of pixels correspond to what class of the dataset.

Resizing

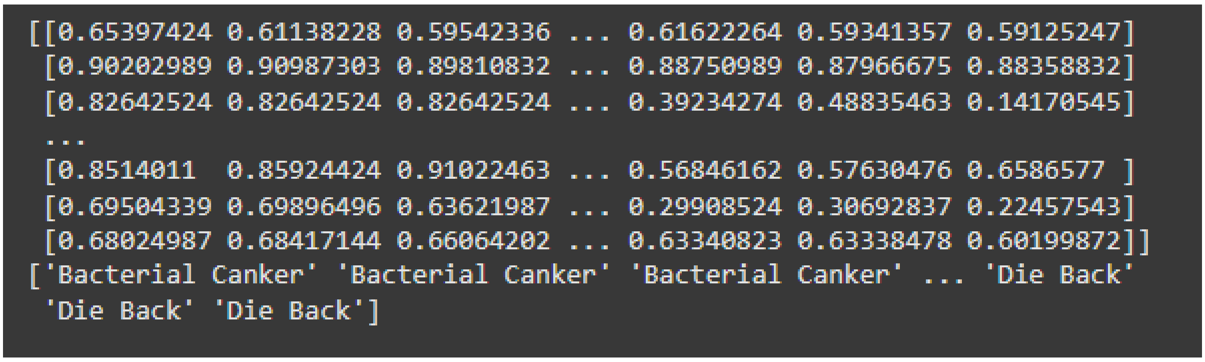
The first step we have to take to clean and prepare our dataset for our models is to resize all of the images. Currently, the images in the dataset have various resolutions and aspect ratios. What we want is all of the images to be of one uniform resolution and shape, so that we can easily standardize them later on.

For the task, I decide to resize every single image to a standard size of 64x64 pixels. This ensures that all of the images are in one uniform resolution, fit for further preprocessing.

Flattening

After we are done resizing, we must flatten each of the images, which are in a 2D (64, 64) arrays, into 1D vectors. This ensures that all of the features in the image are compressed down further for better processing.

After preprocessing all of the dataset, this is what they look like:



```
[[0.65397424 0.61138228 0.59542336 ... 0.61622264 0.59341357 0.59125247]
 [0.90202989 0.90987303 0.89810832 ... 0.88750989 0.87966675 0.88358832]
 [0.82642524 0.82642524 0.82642524 ... 0.39234274 0.48835463 0.14170545]
 ...
 [0.8514011  0.85924424 0.91022463 ... 0.56846162 0.57630476 0.6586577 ]
 [0.69504339 0.69896496 0.63621987 ... 0.29908524 0.30692837 0.22457543]
 [0.68024987 0.68417144 0.66064202 ... 0.63340823 0.63338478 0.60199872]]
['Bacterial Canker' 'Bacterial Canker' 'Bacterial Canker' ... 'Die Back'
 'Die Back' 'Die Back']
```

Exploring the Dataset

Now that we are familiar with the dataset, we can move on to exploring it! What we mean by exploring the dataset is figuring out if there are any patterns or hidden insights in it that are not apparent when looked with our naked eyes.

For this task, we can use Exploratory Data Analysis (EDA), a framework of useful techniques that will allow us to see through the data and find out its patterns and inner workings.

There are a few types of EDA that contain their own differing techniques. For this dataset, we will be utilizing some techniques from each of these types.

Univariate Analysis

This type of analysis would look at a single variable and try to understand its internal structure. For this dataset, we used the following Univariate Analysis techniques:

1. **Box Plot:** These types of plots are useful for detecting any outliers in the dataset along with understanding the spread and skewness of the data. In this case, we used it to visualize the distribution of pixel intensities of each class.

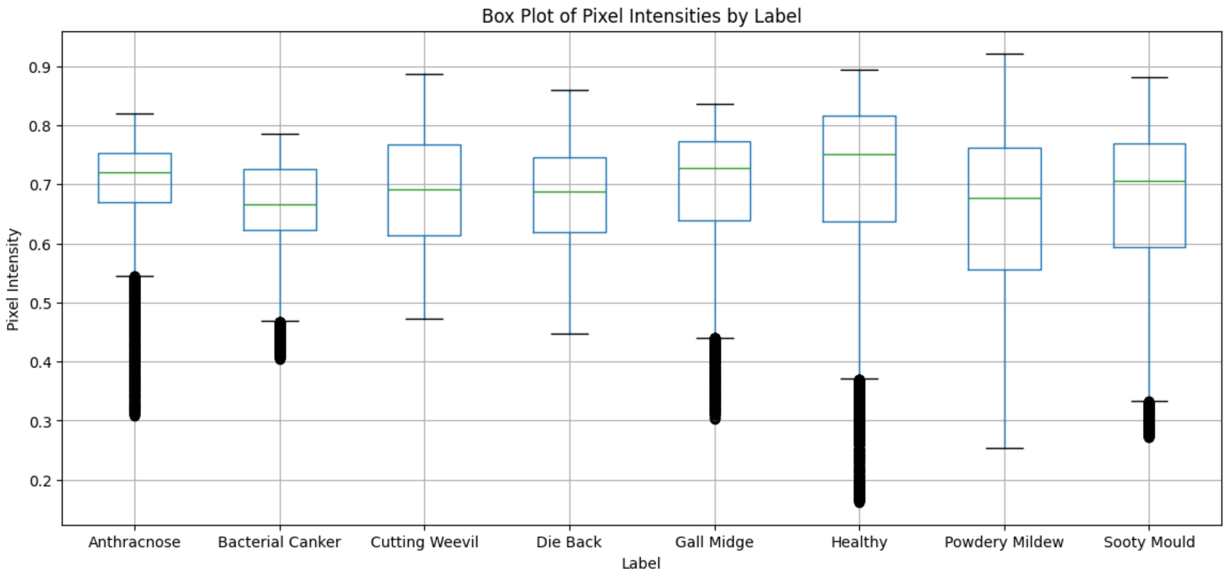


Figure: A box plot of pixel intensities by label

2. **Histogram:** Histograms are useful charts that allow us to visualize the distribution of a single variable. For this dataset, we used it to visualize the distribution of brightness levels in terms of pixel intensities.

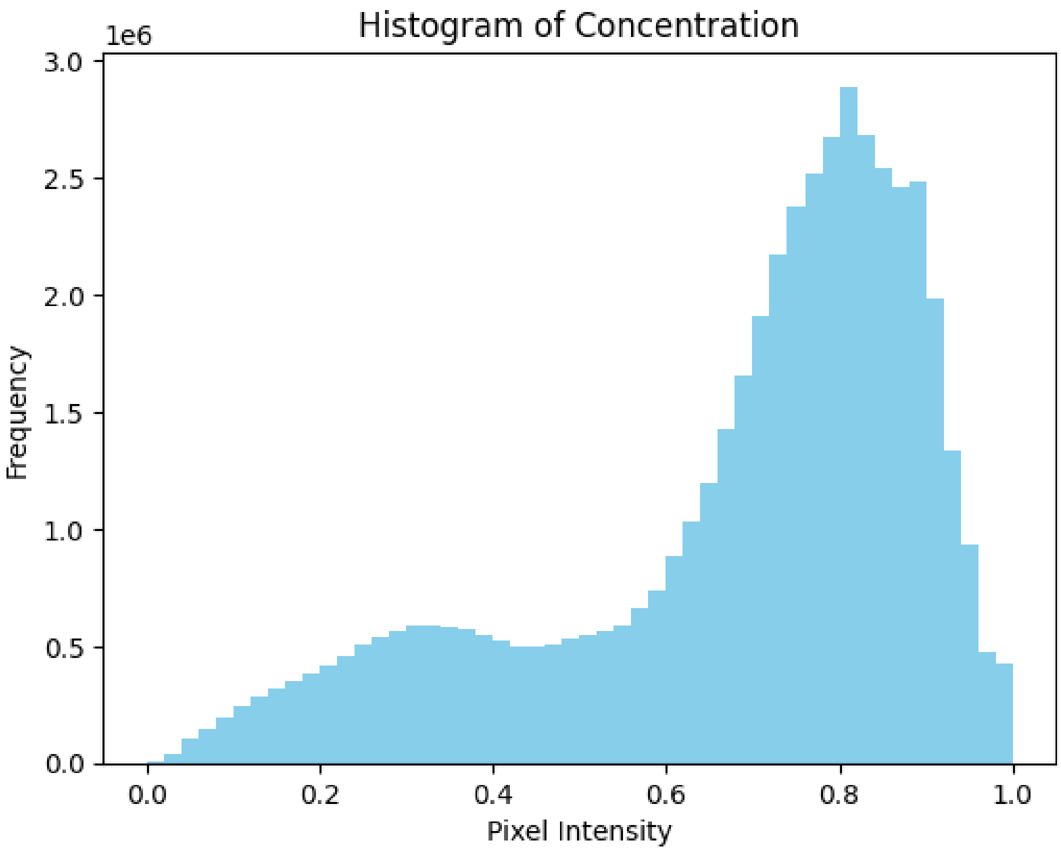


Figure: A histogram of pixel intensities

Bivariate Analysis

This type of analysis involves exploring the connection between variables. It allows us to find associations, correlations, and dependencies between pairs of variables. For this dataset, we used the following Bivariate techniques.

1. **Scatter Plot:** One of the staples of Bivariate Analysis, it allows us to visualize the relationship between two continuous variables. In our approach, we provided a visualization of a scatter plot in terms of pixel intensity vs labels, describing the distribution of data and its intensity.

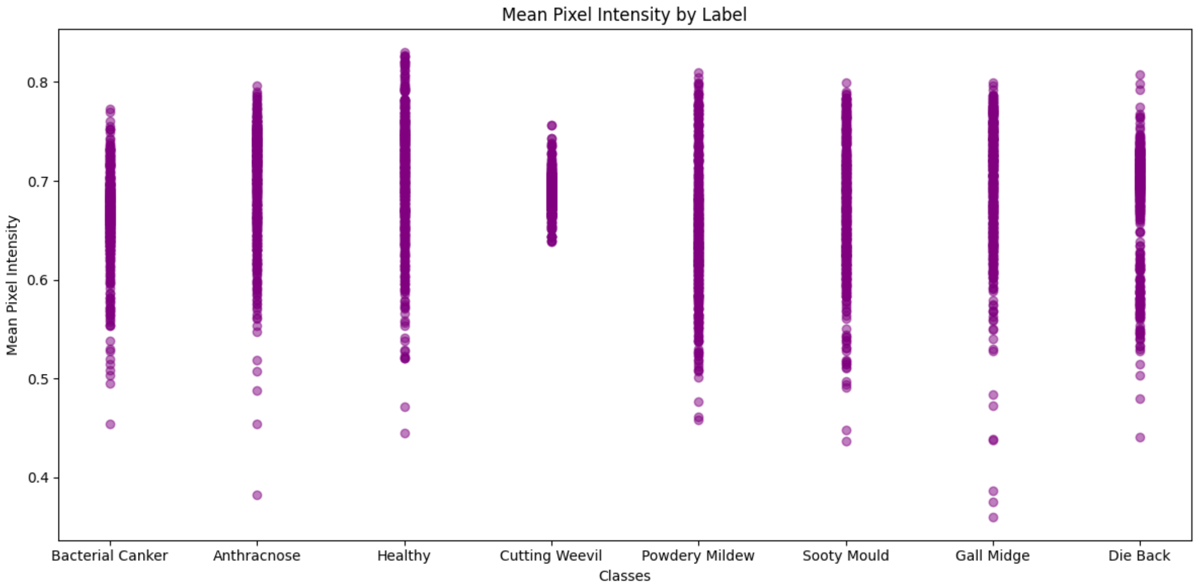


Figure: A Scatter Plot of Pixel Intensity means vs Labels

2. **Correlation Heatmap:** Here, we have combined two concepts, the Pearson's correlation Coefficient for linear relationships and a heatmap, to quantify the degree to which two variables are related. In short, we can identify what the average color of the leaf looks like for a class.

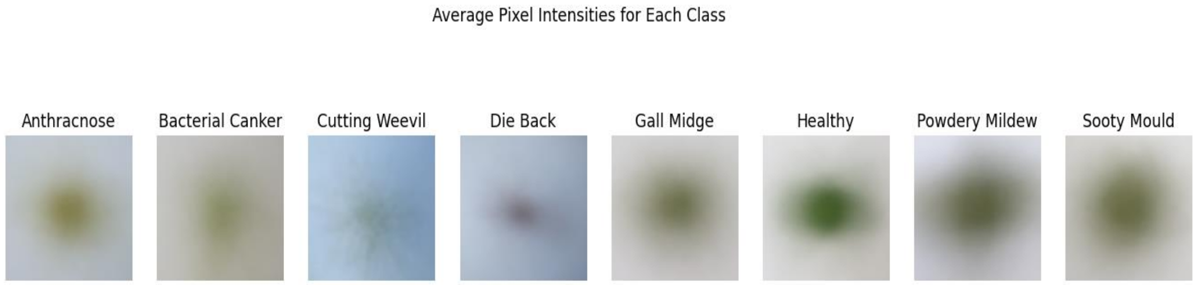


Figure: A heatmap of average pixel intensities for each class

Multivariate Analysis

Finally, we get to multivariate analysis, where the relationships between two or more variables are examined. For our dataset, we used Principal Component Analysis (PCA) to reduce the dimensions of our dataset, while preserving as much variance as possible.

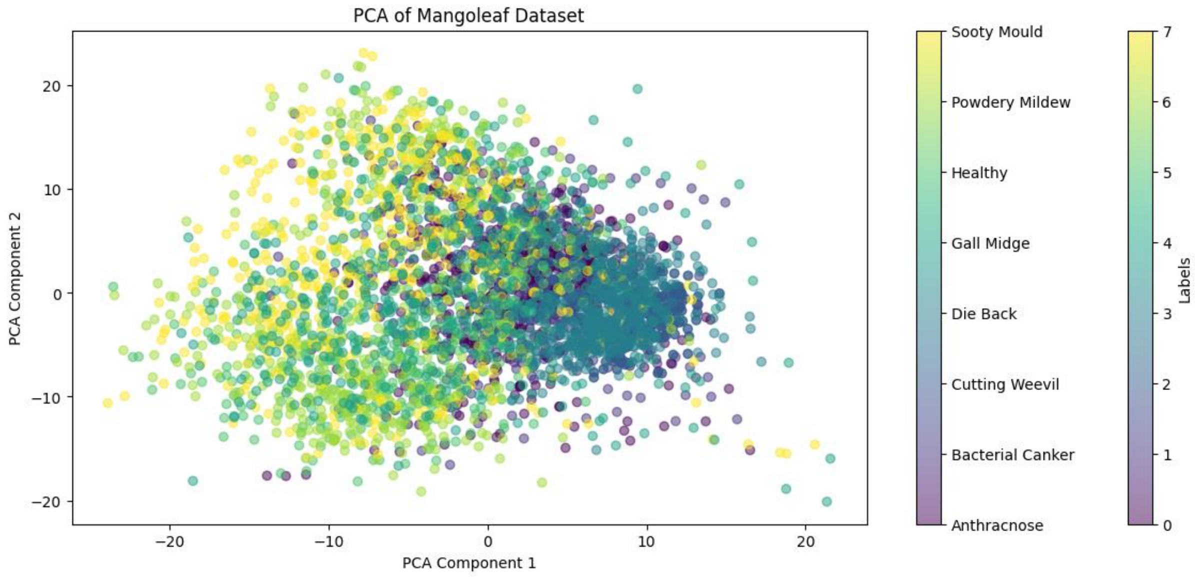


Figure: What our dataset looks like after dimensionality reduction

Classification

At last, we reach the stage of classifying the dataset. Who belongs to what class? How can we predict them? For this task, we will be using two popular classification techniques.

Decision Tree

A decision tree is essentially a partitioning structure that utilizes a tree-like model of the decisions and the possible outcomes. In machine learning, we might recognize as the ID3 algorithm, but there are many different variants of it as well.

For our purposes, we will be using the ID3 algorithm to classify our dataset. Using technologies like Python and scikit-learn, we can easily implement a decision tree algorithm in no time!

Results

We can find out the results of the model and how it is performing using the help of four classical ways of judging model performance.

1. Accuracy
2. Precision
3. F-1 Score
4. Recall

In the case of the decision tree, these were the results I could obtain.

	Precision	Recall	F-1 Score	Support
Anthracnose	0.56	0.58	0.57	86
Bacterial Canker	0.68	0.8	0.74	123
Cutting Weevil	0.88	0.83	0.86	101
Die Back	0.74	0.82	0.78	93
Gall Midge	0.44	0.36	0.40	91
Healthy	0.69	0.48	0.57	112
Powdery Mildew	0.54	0.55	0.55	104
Sooty Mould	0.43	0.52	0.47	90

Accuracy	-	-	0.62	800
Macro Average	0.62	0.62	0.62	800
Weighted Average	0.63	0.62	0.62	800

Random Forest

Random Forest is a variation of the Decision Tree algorithm where a bunch of decision trees work collaboratively to produce a single output. Every single tree is constructed using a random subset of the data set to measure a random subset of features in each partition.

Results

Again, like ID3, we can quickly implement it by using Python and the scikit-learn library. After implementing it, these were the results I could procure.

	Precision	Recall	F-1 Score	Support
Anthracnose	0.87	0.91	0.89	86
Bacterial Canker	0.92	0.88	0.90	123
Cutting Weevil	0.98	1.00	0.99	101
Die Back	0.94	0.96	0.95	93
Gall Midge	0.83	0.88	0.86	91
Healthy	0.91	0.96	0.94	112
Powdery Mildew	0.95	0.88	0.91	104
Sooty Mould	0.87	0.81	0.84	90

Accuracy	-	-	0.91	800
Macro Average	0.91	0.91	0.91	800

Weighted Average	0.91	0.91	0.91	800

Findings

We are now finished with implementing and training both a Decision Tree (ID3) model and a Random Forest model on our dataset.

What is immediately apparent from the results is that the Random Forest classifier model far outperforms the ID3 model with a staggering 135% accuracy increase.

The recall scores are also better than the Decision Tree model, as are the precision scores. Both of the models resulted in about the same amount of support, but we can clearly see that the Random Forest outshines the Decision Tree model in all aspects.

Conclusion

With our task finished, we can conclude on the following objectives:

1. We successfully analyzed the dataset using Exploratory Data Analysis methods like Univariate and Bivariate techniques.
2. We successfully trained Decision Tree and Random Forest classifiers on our dataset.
3. We evaluated both of the models implemented and came out with the conclusion that the Random Forest classifier performs better than the Decision Tree classifier with an almost 1.5x times performance increase.

References

1. What is Exploratory Data Analysis,What is Exploratory Data Analysis? - GeeksforGeeks, 16 May, 2024

Appendix

Github Repository URL	CSE475/lab01 at main · shafin-r/CSE475