

# Statistical Machine Learning for Medicinal Plant leaves Classification

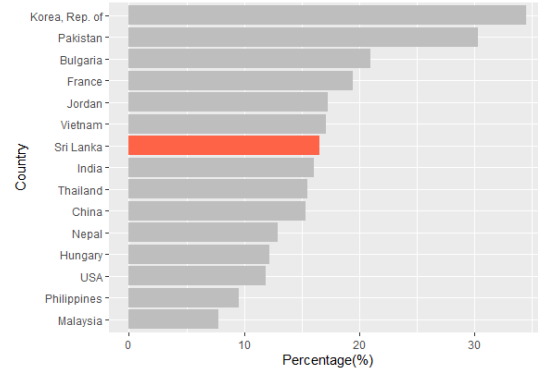
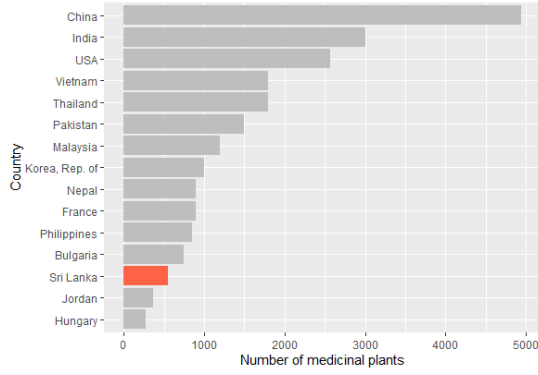
## Abstract

Medicinal plants are usually identified by practitioners based on years of experience through sensory or olfactory senses. The other method of recognizing these plants involves laboratory-based testing, which requires trained skills, data interpretation which is costly and time-intensive. Automatic ways to identify medicinal plants are useful especially those that are lacking experience in herbal recognition. There is no standard mechanism in identification of medicinal plants. Therefore, we introduce an automatic approach based on statistical machine learning to identify medicinal plants. The main objective is to develop an automatic algorithm to classify medicinal plants using medicinal plant leaves. We refer to our medicinal plant classification algorithm as MEDIPI which is divided into offline phase and online phase. The classification algorithm is trained in the offline phase. In the online phase, the pre-trained classification model is used to real-time leaf image classification for general users. Our classification algorithm operates on the features extracted from the image leaves. The offline phase of the algorithm contains four main steps: i) Image processing, ii) Feature extraction, iii) Label images, and iv) Trained an algorithm. The purpose of image processing is to improve the leaf image by removing undesired distortion. The main image processing steps are i) Convert original image to RGB image, ii) Gray scaling, iii) Gaussian smoothing, iv) Binary thresholding, v) Remove stalk, vi) Closing holes, and vii) Resize image. The second stage is to extract features from plant leaf images. We introduced 52 computationally efficient interpretable features to classify plant species. These features are mainly classified in to four groups as (i) shape, (ii) color, (iii) texture, and (iv) scagnostics. Length, width, area, mean of red values, texture correlation, and monotonicity are some of them. Next, we trained our algorithm using random forest, gradient boosting, and extreme gradient boosting. The model trained with random forest algorithm provides the highest accuracy. Our algorithm works as a hierarchical classification system which contains 3 levels. The first level classifies images according to the shape. The second level classifies according to the edge types. The bottom level classifies the plant species. Furthermore, we used high dimensional visualization approaches to visualize what is happening inside the trained algorithm and provides transparency to our black-box model. We compare the accuracy of our proposed algorithm against several benchmarks and other commonly used algorithms for medicinal plants classification. The MEDIPI algorithm yields accurate results to the state-of-the-existing techniques in the field which is developed based on Python. MedLEA is an open-source repository R software established by us. In addition to this we use three benchmark open-source datasets as flavia, Swedish, and kaggle to evaluate our algorithm.

## 1 Introduction

Located in the tropics, Sri Lanka has a collection of plant species with various medicinal properties that have been consumed by generations as herbal treatments for control of diseases and to cure various medical issues. Traditional medicine system which has more than 3000 years of tested and proven efficacy, is still in use (Waisundara and Watawana 2014). It consists of Ayurveda, Unani, and Deshiya Chikitsa (Gunawardana and Jayasuriya 2019). Some of the diseases with complicated etiologies such as diabetes, arthritis, and cancer (for which a permanent cure is not in sight at present) (Waisundara and Watawana 2014) have been known to be completely controlled or cured using the traditional medicinal treatments alone (Devalaraja, Jain, and Yadav 2011). Various plant origins are used to treat disease conditions (Gunawardana and Jayasuriya 2019) in the traditional medicine system (Goyal, Kapil, and Kumar 2018).

According to the IUCN (International Union for Conservation of Nature) and the World Wildlife Fund, there are 550 medicinal plants in Sri Lanka. Furthermore, the distribution of medicinal plants is not uniform across the world and Sri Lanka is in the top 15 (see Figure 1a).

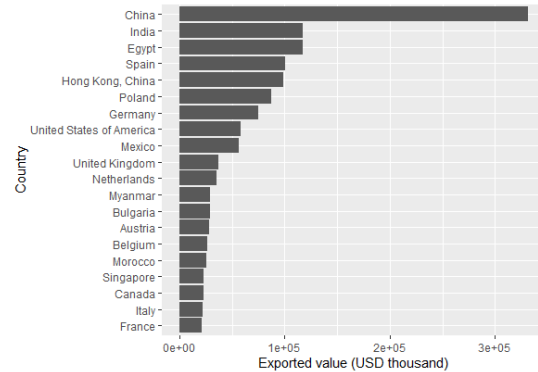
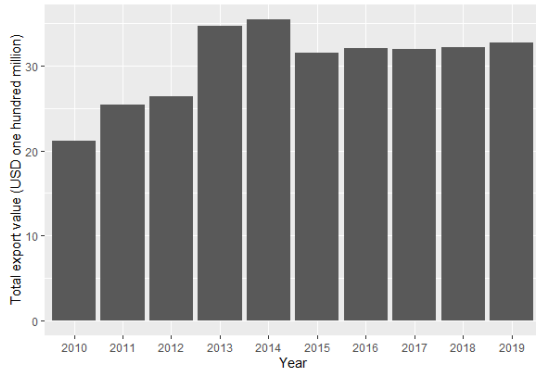


(a) Top 15 countries in the world by distribution of medicinal plants (Source: [inbook]) (b) Top 15 countries in the world by percentage of medicinal plants (Source: [inbook])

Figure 1

As shown in Figure 1a, Asian countries like China, India, Nepal, Philippines, Malaysia, Thailand and North American countries like United States (USA) have a large collection of medicinal plants when compare with Sri Lanka. Even so, the percentages of medicinal plants of China, India, Nepal, Philippines, Malaysia, Thailand and United States (USA) are lower than Sri Lanka (see Figure 1b). Furthermore, Sri Lanka is in the top 7 (see Figure 1b).

In past 10 years, Sri Lanka had a high demand for exporting medicinal plants and the value around 32 USD one hundred million (see Figure 2a) which is an evidence that Sri Lanka has a good market in exporting medicinal plants around the world.



(a) Distribution of export value of medicinal plants in Sri Lanka on last 10 years (Source: 2020, Trade Map Lanka on 2020 (Source: 2020, Trade Map - Trade statistics for international business development, statistics for International Business Development, <https://www.trademap.org/Index.aspx>) (b) Top 20 exporters of medicinal plants in Sri Lanka on 2020 (Source: 2020, Trade Map - Trade statistics for international business development, statistics for International Business Development, <https://www.trademap.org/Index.aspx>)

Figure 2

Furthermore, not only Asian countries but also European and North American countries have an interest of buying medicinal plants in Sri Lanka (See Figure 2b). This is a proof that how much valuable and popular of Sri Lankan medicinal plants.

Even though medicinal plants have a high demand around the world there is no standard mechanism in identifying medicinal plants.

Most algorithms use images as inputs to train the model. Hence, the quality of the image has a direct impact on its performance. Therefore researchers use built-in cameras of a mobile device (Waldchen and

Mader 2018) or a special camera or a scanner to take photographs. Also most of the researchers use secondary datasets such as Flavia, Swedish etc (Goyal, Kapil, and Kumar 2018; Waldchen and Mader 2018). Due to less availability (Goyal, Kapil, and Kumar 2018) of adequate databases and the datasets contain few number of plant species (Goyal, Kapil, and Kumar 2018; Waldchen and Mader 2018), researchers tend to collect their own image datasets. There are restrictions while capturing the plant images. Single leaf, light illumination, shadow effect, and line of sight angle are few of them (Goyal, Kapil, and Kumar 2018).

Images of various parts as leaf, flower, bark, and fruit (Waldchen and Mader 2018) of the plant species use to train the model. Since leaf contains significant features, most of the researchers use to identify and classify the plant species in developing. Furthermore most of the researchers were focused on shape features (Goyal, Kapil, and Kumar 2018). But it is not sufficient to train reliable model properly. Therefore researchers more concern to find what are the most important features to classify plant species.

Furthermore, the existing algorithms mostly developed based on CNN, ANN, PNN, KNN, etc. These models require a large number of memories and become computationally prohibitive and hence, its usefulness can be limited. In addition to that while these methods can deliver good predictions their interpretability and transparency of the model is limited. We address these research gaps by proposing a image feature-based statistical machine learning algorithm.

Normally medicinal plants are grown in the backyards of houses and very little nurturing effort is required for their growth. They also have high growth rates. Therefore sometimes medicinal plants are considered as weeds (Waisundara and Watawana 2014). Most Sri Lankans are familiar with the traditional medicinal system and are even able to identify or administer the medicinal plants growing within their area of residence. Therefore, the locals can be observed consuming these medicinal plants to control a disease without the advice of a traditional medicinal practitioner, as they are familiar with the usage of these herbs because of the traditional knowledge, which has been passed down by their ancestors (Saslis-Lagoudakis et al. 2014) substantial botanical expertise is required by the manual identification process and it is also costly and time-consuming. This identification process is a very challenging task for the general public. There is also no standard mechanism in identification of medicinal plants.

Therefore by addressing the issues above, our main objective is to develop an automatic algorithm to classify medicinal plants by using statistical machine learning approach. To accomplish this main objective, we seek to achieve some other objectives.

A database of leaf images of medicinal plants in Sri Lanka is not yet available. Hence through this research, we establish a repository of medicinal plant images. Researchers usually struggle and spend a lot of time establishing a database by gathering many leaf samples as raw data. By sharing our database we produce a training/test database to other researchers to evaluate their algorithm. Leaf images are considered as they contain large number of diverse set of features such as shape, veins, edge features, apices, etc. Therefore through this research we identify features that are useful in classifying medicinal plants based on leaves images. Another objective is to develop an algorithm to extract and quantify leaf features. Furthermore, we used high dimensional visualization approaches to visualize what is happening inside the trained algorithm. We develop the proposed algorithm through an open-source software to identify medicinal plants in Sri Lanka by using leaf images.

The significance of this research is to avoid misidentifying medicinal plants in Sri Lanka. This is beneficial in conservation and ecological efforts. Researchers define that endangered medicinal plants as the plants which are facing a high risk of becoming extinct because they are either few in numbers, or threatened by changing environmental parameters (Nilaweera 2010). The International Union for Conservation of Nature (IUCN) has defined Threatened Herbal plants in three schemes as Critically Endangered, Endangered, and vulnerable. In the world, nearly 15,000 species of medicinal plants are now threatened. In Sri Lanka 280 plant species are threatened. According to the recent surveys (Nilaweera 2010), there are 1432 medicinal plant species in Sri Lanka, and out of the 100-200 species are threatened. Abarema begimena, Ashoka tree(Saraca Asoka), Beautiful Leaf(Calopyllum trapezifolium), Aglaia apiocarpa are few of them.

The algorithm developed by us is based on the leaf images. Since leaves are relatively easy to obtain without damaging the plants, there is no harm for the plants because of the development of algorithm. Our algorithm works as a hierarchical classification system. Therefore even though we don't know the exact

species name, we can follow the first 2 levels. As the result of that misidentification rate and computation time will be decreased.

Outline should be written.

## 2 Methodology

### 2.1 Overview of the Algorithm

The aim of this chapter is to provide a general overview of the methodology used to develop our classification algorithm. The classification algorithm we introduce contains two main phases (i) The offline phase, and (ii) The online phase.

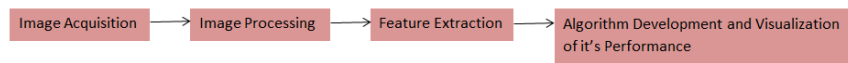


Figure 3: Workflow of the offline phase of the algorithm

As shown in Figure 3, the workflow of the offline phase of the algorithm contains main 4 steps as:

1. Image Acquisition
2. Image Processing
3. Feature Extraction
4. Algorithm Development and Visualization of Algorithm Performance

Figure 4 shows the overview of the methodology that we followed. Online phase of the study is colored by orange and offline phase of the study is colored by blue. Firstly we acquire the images of leaves from existing datasets and the leaf image dataset that was collected by ourselves. Then each leaf image data set is divided as training and test images. Training image dataset was contained 80% of the images and test image dataset is contained 20% of the images from each leaf image dataset. We use four datasets to built and evaluate our algorithm. A brief summary of the datasets are given in the Table 1.

Dataset	Image format	Total number of leaf images
Actual Leaf Image Dataset (MedLEA)	Color	1099
Flavia dataset	Color	1907
Swedish Leaf Image Dataset	Color	975
Kaggle Leaf Image Dataset	Binary	1584

Table 1: Summary of datasets used in the algorithm

Next step is image processing. As shown in Figure 4, the main image processing steps are Convert to RGB image, Gray scaling, Gaussian smoothing, Binary thresholding, Remove stalk, Closing holes and Resize image. Since Kaggle leaf image dataset contains only binary images, resizing step is enough as an image processing technique. We can follow remove stalk and closing holes technique only if the dataset contains leaf images with stalk and with holes (eg:- diseased leaves). After applying image processing steps, the images are ready to extract features. There are four classes of features: (i) Shape features, (ii) Color features, (iii) Texture features, and (iv) Scagnostics features of Cartesian and polar coordinates. In our research we also introduce some new features: Correlation of Cartesian contour, x and y coordinates of the contour, Number of minimum and maximum points, Number of convex points. Now the dataset contained all the features with

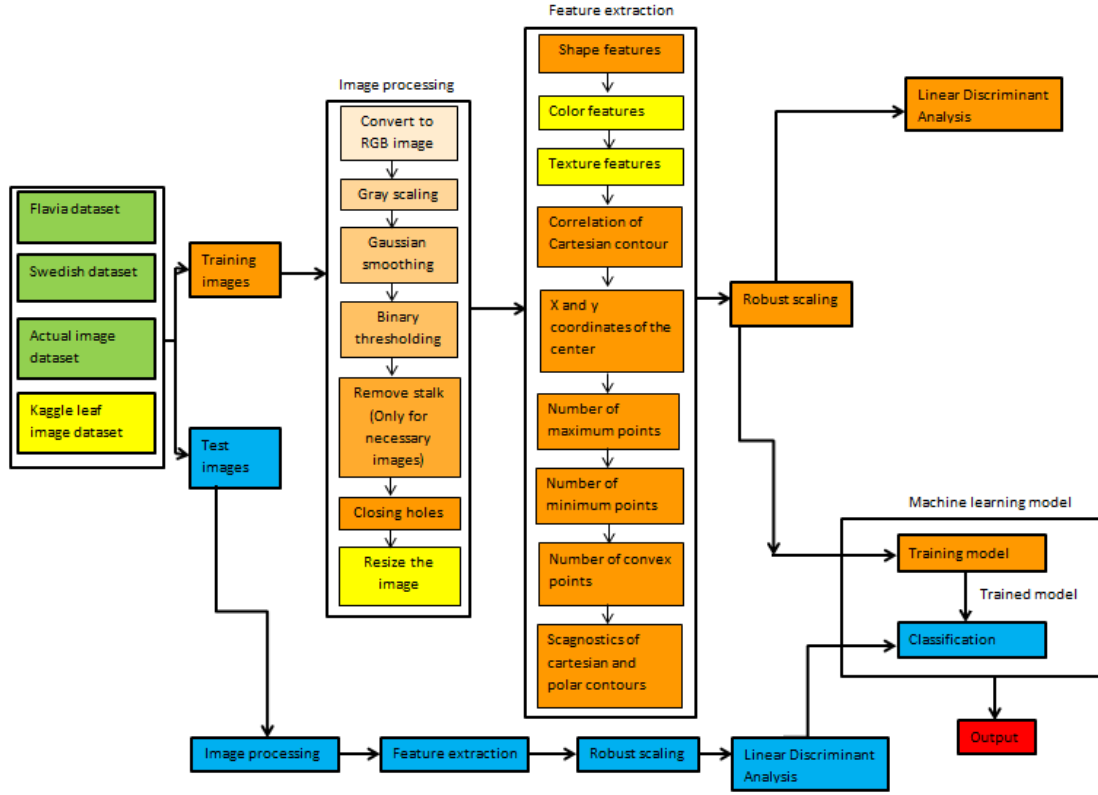


Figure 4: Methodology Diagram

the leaf image id. But Kaggle leaf image dataset doesn't have Color and Texture features. Robust scaling is applied to scale the data. To visualize the feature dataset with labels, Linear Discriminant Analysis is used. Our algorithm operates according to a hierarchical classification system. First the leaves are classified according to the shape such as; (i) diamond, (ii) simple round, (iii) round, (iv) needle, and (v) heart shape. The second level classifies according to the edge types. The bottom level classifies the plant species. Before training the model we labeled all leaves according to shape type, edge type, leaf arrangement, apex type, base type etc. These labels are identified by exploring "Ayurvedic Medicinal Plants of Sri Lanka", medicinal leaf repository maintained by, Barbery Ayurveda resort and University of Ruhuna. The information we gathered by exploring the "Ayurvedic Medicinal Plants of Sri Lanka", medicinal leaf repository are made available through our R package MedLEA. Next step is to train the model for the training dataset by using machine learning techniques. This is a multi-class supervised learning classification problem. The trained model is used to predict labels in the the test dataset.

- MEDUPI

Our medicinal plant classification algorithm is defined as MEDUPI: **MED**icinal **P**lant **I**dentification. The MEDUPI is divided into offline phase and online phase. The classification algorithm is trained in the offline phase. In the online phase the pre-trained classification model is used to real-time leaf image classification for general users.

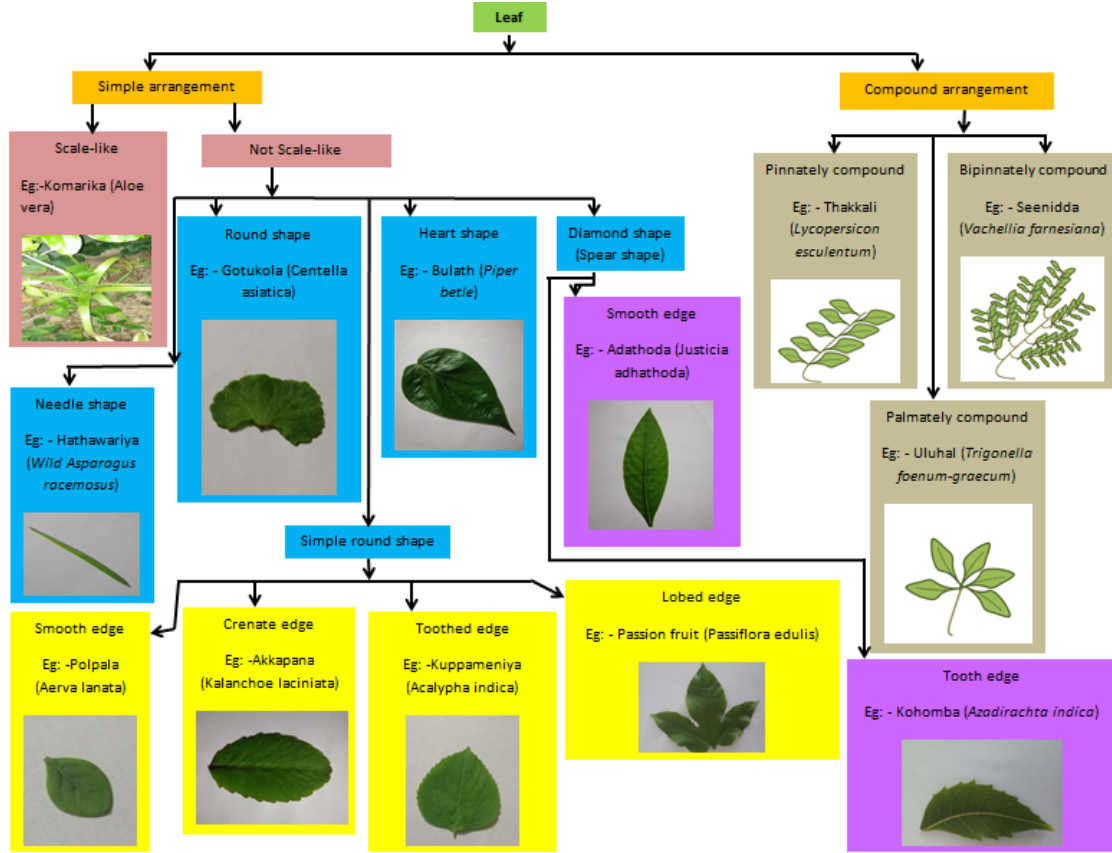


Figure 5: Classification hierarchy

## 2.2 Data

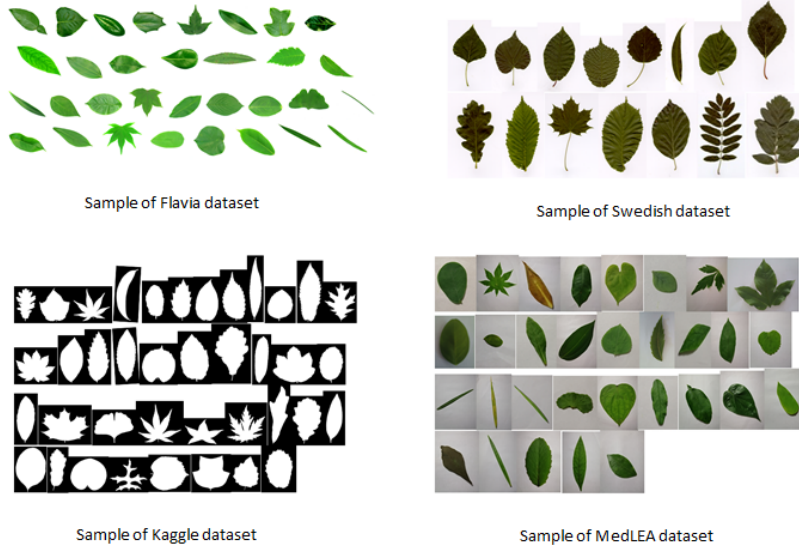
We use four datasets. There is one primary dataset and three secondary datasets. The primary dataset is named as MedLEA. The secondary datasets are Flavia, Swedish, and Kaggle.

### 2.2.1 Secondary Data

A brief summary of the secondary datasets are given in the Table 2.

Dataset	Image format	Number of Species	Number of images from one species	Number of leaf images	Collected country	Remarks
Flavia	Color	32	50-77	1907	China	Scanners and digital cameras are used to acquire the leaf images on plain background. The isolated leaf images contain blades only, without petiole. The images of isolated leaf scans on a plain background
Swedish	Color	15	75	1125	Swedish	
Kaggle	Binary	99	16	1584	United Kingdom	

Table 2: Summary of secondary datasets



### 2.2.2 Primary Data

Image collection process contains 5 main steps as shown in figure 6. This approach is very simple, easy, and can be followed without any expertise knowledge.

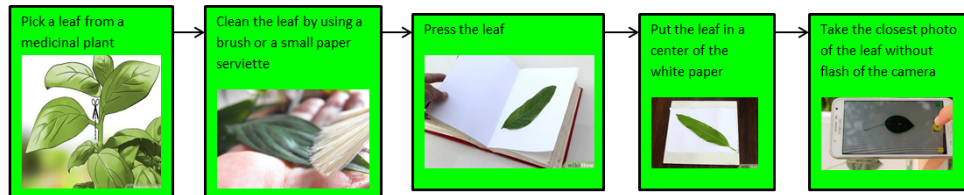


Figure 6: Image collection process of medicinal plants in Sri Lanka

Firstly we have to select a plant that we are going to use for this classification. Then have to find a leaf and pick it. In this step, have to be more careful about selecting the leaf. Our algorithm considers only

the leaf images without any diseases. When picking the leaf, use a scissor to pick the leaf without petiole. Because the algorithm considers only the leaf without petiole. Make sure that the leaf has to pick in the morning time. Because the leaf looks fresh in the morning time.

After picking the leaf, have to clean it by using a small brush or a piece of paper serviette. Because there are small water bubbles, soil seeds and mud patches.

In some cases, the leaf looks like rounding from the apex or base or margin of the leaf can't put on a flat surface. Therefore will be problematic when putting it to the algorithm. Because the algorithm is difficult to capture the shape of the leaf correctly. To avoid these problems, press the leaf approximately 1 or 2 days (In some cases less than 1 day is enough), before taking the photos.

Then keep the pressed leaf in a white paper. In this step, we have to consider about where we have to keep it. Make sure to keep the leaf in the centre of the white paper. The reason is that the converting to binary image work well when the leaf is in the centre of the white paper.

Finally when taking the photo, have to take the closest photo without the flash of the camera (see figure 7). Closest photo because algorithm is difficult to extract the contour of a very small leaf (see figure 7), decrease the amount of computational load that is exerted upon the graphic processing unit, and reduce the unnecessary foreground region (Goyal, Kapil, and Kumar 2018). When converting to the binary images, to capture the shape of the leaf correctly have to remove the shadow of the leaf much as can. Therefore by using the camera without flash, can remove the shadow (see figure 7). Make sure the photo is taken in the daylight to ignore the effect of light illumination.

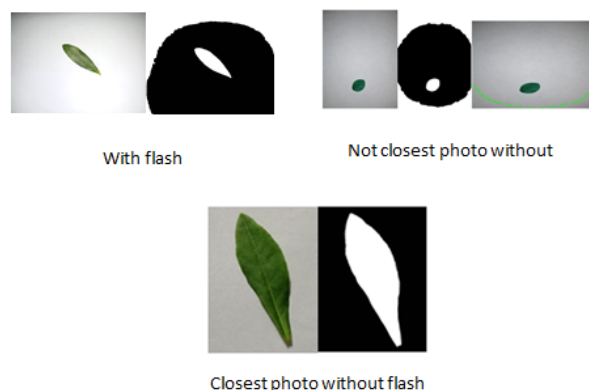


Figure 7

- MedLEA

Through this research, we establish a repository of medicinal plant images in Sri Lanka. This repository is made available to the public through an open-source R software MedLEA, available at <https://CRAN.R-project.org/package=MedLEA> for research reproducibility.

There are 1099 images of leaf images of 31 species and 29-45 images per species of medicinal plants in Sri Lanka. These leaves have simple arrangement. A single leaf that is never divided into smaller leaflet units is know as a leaf with simple arrangement. That leaf is always attached to a twing by its stem or the petiole. The margins, or edges, of the leaf can be smooth, lobed, or toothed. The photos were taken from the device, Huawei nova 3i. The closest photos are captured on a white background.

## 2.3 Image Processing

Image processing plays a vital role in leaf image identification. Image processing is applied to reduce noise, background subtraction and content enhancement in the identification process (Goyal, Kapil, and



Kumar 2018). The workflow we use to process images in this paper is shown in Figure 8. This includes seven main steps. They are: i) converting BGR (Blue-Green-Red) image to RGB (Red-Green-Blue), ii) gray scaling, iii) Gaussian filtering, iv) binary thresholding, v) remove stalk, vi) close holes, and vii) image resizing. Some of these steps applicable only for specific images. For example, apply remove stalk is applicable only to leaf images which has stalk.

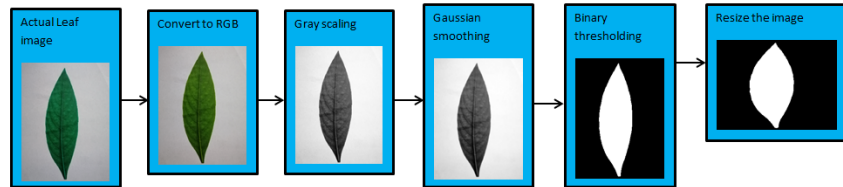


Figure 8: Image processing

Feeding RGB images with gray scaling, optimize the contrast and intensity of images by reducing dimensions and complexity. Smoothing techniques are applied to remove noise and make the image less clear or distinct. Furthermore, as the result of binary thresholding is used to separate foreground from its background. Removal of stalk and closing holes in foreground is important when capturing the shape of the leaf.



Figure 9

Figure 9 shows that binary image after removing the stalk and closing holes according to the order.

More details about the image processing steps are discussed in the Computer-aided Interpretable Features for Leaf Image Classification paper.

## 2.4 Feature extraction

Most crucial part is to extract distinctive leaf features from the images. Therefore most of the time research more focused on neural network models like CNN (Wu et al. 2007; Azlah et al. 2019; Herdiyeni and Wahyuni 2012) which are complicated and hard to understand what happening inside the algorithm. We introduced pre-calculate features which can be easy to interpret and generalize. They are also computational efficient. Mainly we focused on four types of features of leaf images as shape features, texture features, color features and scagnostics features. We identified altogether 52 features. More details about the features of the leaf are discussed in the Computationally Efficient Features paper. The following table shows the summary of all features.











Image type	Feature category	Feature	Feature name	Figure	Formula	Range	Software	Software package
Binary	Shape	$F_1$	Diameter		$F_1 = \max(\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}); \forall i, j, i \neq j$	$[0, \infty]$		combinations, numpy
		$F_2$	Physiological length		$F_2 = \text{Length of the rectangle}$	$[0, \infty]$		OpenCV
		$F_3$	Physiological width		$F_3 = \text{Width of the rectangle}$	$[0, \infty]$	Python	OpenCV
		$F_4$	Area		$F_4 = \text{Number of zero pixels covered by the contour}$	$[0, \infty]$		OpenCV
		$F_5$	Perimeter		$F_5 = \sum_{i=0}^n d_i$ ; where $n$ is the number of distances around the contour	$[0, \infty]$		OpenCV
		$F_6$	Eccentricity		$F_6 = \sqrt{1 - \frac{b^2}{a^2}}$ ; where $a$ is semi major axis and $b$ is semi minor axis	$[0, 1]$		OpenCV
		$F_7, F_8$	x and y coordinate of center					scipy.ndimage
		$F_9$	Aspect ratio		$F_9 = \frac{F_2}{F_3}$	$[0, \infty]$		
		$F_{10}$	Roundness/Circularity		$F_{10} = \frac{4\pi F_4}{F_5^2}$	$[0, \infty]$		numpy
		$F_{11}$	Compactness		$F_{11} = \frac{F_5^2}{F_4}$	$[0, \infty]$		
		$F_{12}$	Rectangularity		$F_{12} = \frac{F_5^2}{F_4}$	$[0, \infty]$		
		$F_{13}$	Narrow factor		$F_{13} = \frac{F_1}{F_2}$	$[0, \infty]$		

Table 3 continued from previous page









Image type	Feature category	Feature	Feature name	Figure	Formula	Range	Software	Software package
Gray scale	Texture	$F_{14}$	Perimeter ratio of diameter		$F_{14} = \frac{F_5}{F_1}$	$[0, \infty]$		
		$F_{15}$	Perimeter ratio of physiological length		$F_{15} = \frac{F_5}{F_2}$	$[0, \infty]$		
		$F_{16}$	Perimeter ratio of physiological length and width		$F_{16} = \frac{F_5}{F_2 * F_3}$	$[0, \infty]$		
		$F_{17}$	Perimeter convexity		$F_{17} = \frac{\text{Perimeter of convex hull}}{F_5}$	$[0, \infty]$		OpenCV
		$F_{18}$	Area convexity		$F_{18} = \frac{(\text{Area of convex hull} - F_4)}{F_4}$	$[0, \infty]$		OpenCV
		$F_{19}$	Area ratio of convexity		$F_{19} = \frac{F_4}{\text{Area of convex hull}}$	$[0, \infty]$		OpenCV
		$F_{20}$	Equivalent diameter		$F_{20} = \sqrt{\frac{4 * F_4}{\pi}}$	$[0, \infty]$		numpy
		$F_{21}$	Number of convex points		$F_{21} = \text{Number of vetices of the convexHull}$	$[0, \infty]$		OpenCV
		$F_{22}$	Contrast		$\frac{\sum_{a=1}^{columns} \sum_{b=1}^{rows} (a-b)^2 h(a,b)}{\text{Number of gray levels}-1}$	$[0, \infty]$		
		$F_{23}$	Entropy		$-\sum_{a=1}^{columns} \sum_{b=1}^{rows} h(a,b) \log_2(h(a,b))$	$[-\infty, 0]$	Python	mahotas
		$F_{24}$	Correlation		$\frac{\sum_{a=1}^{columns} \sum_{b=1}^{rows} (ab)h(a,b) - \mu_x \mu_y}{\sigma_x \sigma_y}$	$[-1, 1]$		
		$F_{25}$	Inverse difference moments		$\sum_{a=1}^{columns} \sum_{b=1}^{rows} \frac{h(a,b)}{(a-b)^2}$	$[0, \infty]$		
		$F_{26}$	Mean red intensity value		$F_{26} = \frac{\text{Total intensity value of red channel of the image pixels}}{\text{Total intensity value of the image}}$	$[0, \infty]$		
		$F_{27}$	Mean blue intensity value		$F_{27} = \frac{\text{Total intensity value of blue channel of the image pixels}}{\text{Total intensity value of the image}}$	$[0, \infty]$	Python	numpy

Table 3 continued from previous page

Image type	Feature category	Feature	Feature name	Figure	Formula	Range	Software	Software package
Binary	Scagnostics	$F_{28}$	Mean green intensity value		$F_{28} = \frac{\text{Total intensity value of green channel of the image pixels}}{\text{Total intensity value of the image}}$	$[0, \infty]$	R	binostics
		$F_{29}$	Standard deviation of red intensity value		$F_{29} = \frac{\sqrt{\sum_{j=0}^h \left( \frac{\text{Red channel} - \text{mean intensity value}}{\text{Total intensity value of the image}} \right)^2}}{\text{Total intensity value of the image}};$ where $h$ is number of pixels	$[0, \infty]$		
		$F_{30}$	Standard deviation of blue intensity value		$F_{30} = \frac{\sqrt{\sum_{j=0}^h \left( \frac{\text{Blue channel} - \text{mean intensity value}}{\text{Total intensity value of the image}} \right)^2}}{\text{Total intensity value of the image}};$ where $h$ is number of pixels	$[0, \infty]$		
		$F_{31}$	Standard deviation of green intensity value		$F_{31} = \frac{\sqrt{\sum_{j=0}^h \left( \frac{\text{Green channel} - \text{mean intensity value}}{\text{Total intensity value of the image}} \right)^2}}{\text{Total intensity value of the image}};$ where $h$ is number of pixels	$[0, \infty]$		
		$F_{sc1}$	Outlying		$F_{sc1} = \frac{\text{Total length of edges adjacent to outlying points}}{\text{Total edge length of minimum spanning tree}} =$	$[0, 1]$		
		$F_{sc2}$	Skewed		$F_{sc2} = 1 - \text{weight} * (1 - \text{qu skew})$ $F_{sc3} = \text{weight} * 90 \text{ th percentile}$	$[0, 1]$		
		$F_{sc3}$	Sparse		of the distribution of edge lengths in the minimum spanning tree where $\text{weight} = 0.7 + \frac{0.3}{1 + \text{Number of vertex}^2}$	$[0, 1]$		
		$F_{sc4}$	Clumpy		$F_{sc4} = \max_j \left[ 1 - \frac{\max_k [\text{length}(e_k)]}{\text{length}(e_j)} \right]$	$[0, 1]$		
		$F_{sc5}$	Striated		$F_{sc5} = \frac{1}{ Ve } \sum_{\nu \in Ve^{(2)}} I(\cos \theta_{e(\nu,a)e(\nu,b)} < -0.75)$ where $Ve^{(2)} \subseteq Ve$ and $I()$ be an indicator function	$[0, 1]$		
		$F_{sc6}$	Convex		$F_{sc6} = \text{weight} * \frac{\text{Area of alpha hull}}{\text{Area of convex hull}}$ where $\text{weight} = 0.7 + \frac{0.3}{1 + \text{Number of vertex}^2}$	$[0, 1]$		
		$F_{sc7}$	Skinny		$F_{sc7} = 1 - \frac{\sqrt{4 * \pi * \text{Area of alpha hull}}}{\text{Perimeter of alpha hull}}$	$[0, 1]$		
		$F_{sc8}$	Stringy		$F_{sc8} = \frac{ Ve^{(2)} }{ Ve  -  Ve^{(1)} }$ $Ve$ is the number of vertices	$[0, 1]$		
		$F_{sc9}$	Monotonic		$F_{sc9} = r_{Spearman}^2$	$[0, 1]$		
		$F_{32}$	Number of minimum points		$F_{32} = \text{Number of global minimum points}$	$[0, \infty]$		
		$F_{33}$	Number of maximum points		$F_{33} = \text{Number of global maximum points}$	$[0, \infty]$		
		$F_{34}$	Correlation of cartesian contour		$F_{34} = \frac{\sum_{i=0}^m (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=0}^m (x_i - \bar{x})^2 (y_i - \bar{y})^2}}$	$[-1, 1]$		

Table 3: Definitions of features

## 2.5 Classification framework

Our algorithm works as a hierarchical classification system. The hierarchy contains 3 levels. The first level classifies images according to the shape. The second level classifies according to the edge types. The bottom level classifies the plant species. If we use species names rather than hierarchical classification, it is difficult to manage and occur class imbalance problem due to large number of class labels. In order to develop hierarchical system we explore actual plant image repository: “Ayurvedic Medicinal Plants of Sri Lanka” (<http://www.instituteofayurveda.org/plants/>) which describes about most commonly used medicinal plants for practice of Ayurveda in Sri Lanka. This website was created as the result of the project implemented by Barbelyn Ayurveda resort and University of Ruhuna. The website was updated on 11th May 2017. Our pilot study was based on this database.

We investigate 471 medicinal leaves in this repository. By investigating leaf images of each medicinal plant, we recorded the physical appearances (morphological characteristics) of the leaf. Rather than adding variables that regarded to physical appearances, we recorded sinhala name (local name), family name, and scientific name as well. There were 22 variables with the primary key (unique) as “Id”. There were 18 variables that described about physical appearances of the medicinal leaf images such as leaf arrangement, shape, edge type, apex, base etc.

### 2.5.1 Exploratory Data Analysis of Ruhuna Dataset

In this section, we present Exploratory Data Analysis to get an idea about the common morphological features of leaves.

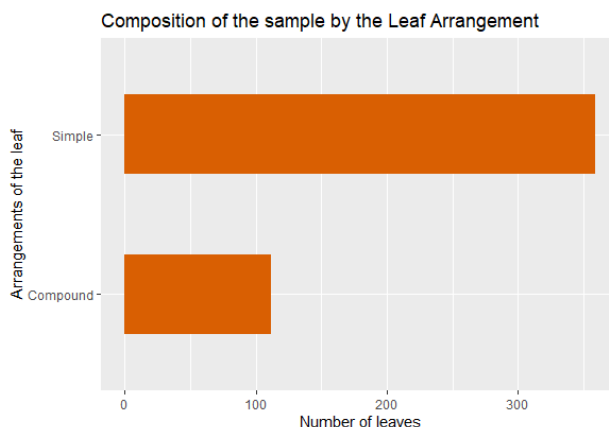


Figure 10: Composition of sample of ruhuna dataset by arrangement of leaves

According to the Figure 10, most of the leaves are arranged in Simple arrangement. Therefore further analysis, we selected the leaves that have simple arrangement.

According to Table 4, most common shapes of the leaves are (i) Diamond, (ii) Simple round, (iii) Heart, (iv) Needle, and (v) Round. Therefore we used these common 5 leaf shapes as the first level of the hierarchy.

According to Table 4, most of the leaves have smooth edges. Based on the results of the study we identify 4 common edge types as (i) Smooth, (ii) Toothed, (iii) Lobed, and (iv) Crenate.

As seen in Table 4, most of diamond, heart, round, needle, and simple round shaped leaves have Smooth edges. All of the needle shaped leaves have smooth edges. There are no crenate edged heart shaped leaves. Furthermore, diamond, round, and simple round shaped leaves have smooth, toothed, lobed, and crenate edges. We used edge types to go to the bottom level of the hierarchy.

By conducting the study, we identified that what are the common shapes and edge types of the medicinal leaves in Sri Lanka. We proceed with 5 main shapes as diamond, heart, round, needle, and simple round. By

Shape label	Edge type				Row total (%)
	Smooth	Tooth	Lobed	Crenate	
Diamond	167	27	22	1	217 (60.4)
Row %	77.0	12.4	10.1	0.5	
Column %	66.0	44.3	55.0	20.0	
Total %	46.5	7.5	6.1	0.3	
Heart	26	14	6	0	46 (12.8)
Row %	56.5	30.4	13.0	0.0	
Column %	10.3	23.0	15.0	0.0	
Total %	7.2	3.9	1.0	0.0	
Total %	7.2	3.9	1.0	0.0	
Needle	21	0	0	0	21 (5.8)
Row %	100	0.0	0.0	0.0	
Column %	8.3	0.0	0.0	0.0	
Total %	5.8	0.0	0.0	0.0	
Round	5	3	1	1	10 (2.8)
Row %	50	30	10	10	
Column %	2.0	4.9	2.5	20	
Total %	1.4	0.8	0.3	0.3	
Simple round	31	14	11	3	59 (16.4)
Row %	52.5	23.7	18.6	5.1	
Column %	12.3	23.0	27.5	60	
Total %	8.6	3.9	3.1	0.8	
Scale-like shaped	3	3	0	0	6 (1.7)
Row %	50	50	0.0	0.0	
Column %	1.2	4.9	0.0	0.0	
Total %	0.8	0.8	0.0	0.0	
Column total	253	61	40	5	359
(%)	(70.5)	(17.0)	(11.1)	(1.4)	(100)

Table 4: Table of ruhuna dataset

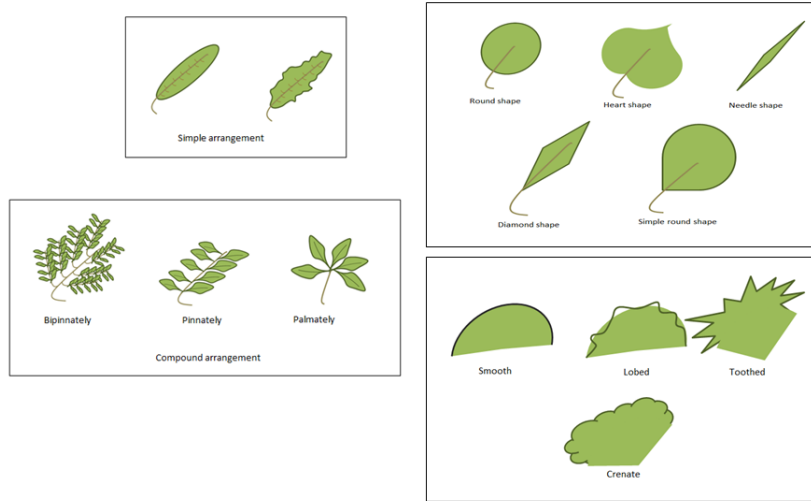


Figure 11: Common arrangements, shape, and edge types

using main 4 edge types as smooth, toothed, lobed, and crenate, we go to the bottom level of the hierarchy. Based on this dataset hierarchy (see Figure 5) is created. Based on these information we develop a hierarchical structure to classify the leaf images by identifying main features. The general hierarchical structure is shown in Figure 12.

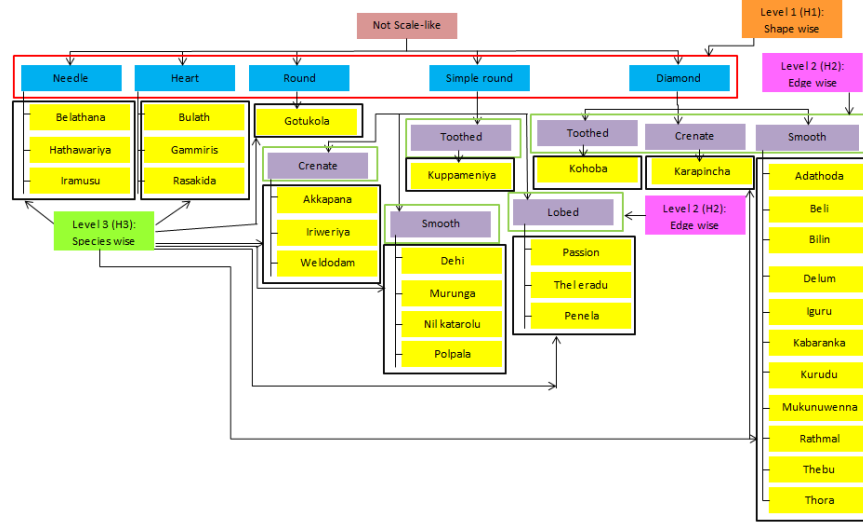


Figure 12: General hierarchical structure

## 3 Results

### 3.1 Experiments

In our study, we conducted two experiments as shown in Figure 22a and Figure 22b. Furthermore, most of studies in literature considered shape features only, to explore if adding new features help in improving accuracy we arrange the experiments in two ways as; (i) Using all the feature categories (shape, color, texture, and scagnostic features), and (ii) Using only with shape features.

#### 3.1.1 Experiment 1

As shown in Figure 22a, experiment 1 was designed by using 80% of training and 20% of test set of the same dataset separately. Experiment 1 was conducted in two ways. First the experiment 1 was conducted by considering all the features (shape, color, texture, and scagnostic features) as inputs to the algorithms. Second the experiment 1 is conducted by considering only with the shape features as input to the algorithms. Our algorithm works as a hierarchical classification system. The hierarchy contains 3 levels. The first level (H1) classifies images according to the shape. The second level (H2) classifies according to the edge types. The bottom level (H3) classifies the plant species. Flavia, Swedish and Kaggle datasets used to evaluate only the first level of the hierarchy i.e: classification is based only with the shapes of the leaves. Whereas Actual dataset is analyzed from top to bottom level of the hierarchy and recorded classification accuracy of each level.

The following Table 5 shows that the shape-wise (H1) and overall classification accuracy in experiment 1 which means that out of sample accuracy in H1 (see Figure 13), with different algorithms.

According to the results shown in Table 5, Random Forest algorithm works extremely well in classifying simple round, needle, round, and heart shape leaves. The out-of-sample accuracy is 100%. However, Random Forest and Gradient Boosting have same overall classification accuracy as 97%.

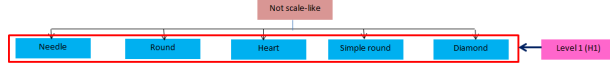


Figure 13: Level 1 of the hierarchy (H1)

Dataset	Diamond	Simple Round	Needle	Heart Shape	Round	Overall Classification Accuracy
<b>Actual</b>						
Random Forest	0.94	<b>1.00</b>	<b>1.00</b>	<b>1.00</b>	<b>1.00</b>	<b>0.97</b>
Extreme Gradient Boosting	0.94	0.95	<b>1.00</b>	0.96	<b>1.00</b>	0.95
Gradient Boosting	<b>0.96</b>	0.96	<b>1.00</b>	<b>1.00</b>	<b>1.00</b>	<b>0.97</b>
<b>Flavia</b>						
Random Forest	0.98	<b>0.97</b>	0.96	0.97	<b>1.00</b>	<b>0.98</b>
Extreme Gradient Boosting	<b>0.99</b>	0.95	0.93	0.98	<b>1.00</b>	0.97
Gradient Boosting	0.98	0.96	<b>1.00</b>	<b>1.00</b>	<b>1.00</b>	<b>0.98</b>
<b>Swedish</b>						
Random Forest	0.99	<b>1.00</b>	<b>1.00</b>	<b>0.95</b>	<b>1.00</b>	<b>0.98</b>
Extreme Gradient Boosting	<b>1.00</b>	<b>1.00</b>	<b>1.00</b>	0.88	0.92	0.97
Gradient Boosting	0.99	0.98	0.96	0.86	<b>1.00</b>	0.96
<b>Kaggle</b>						
Random Forest	<b>0.76</b>	<b>0.69</b>	-	0.82	<b>0.77</b>	<b>0.74</b>
Extreme Gradient Boosting	0.75	0.68	—	0.70	0.70	0.71
Gradient Boosting	0.74	0.68	—	<b>0.92</b>	0.66	0.71

Table 5: Shape-wise (H1) and overall classification accuracy (Training and test sets are from the same dataset)

When consider the shape-wise (H1) accuracy of Flavia dataset, Gradient Boosting algorithm is the best. Because feeding the Flavia dataset to Gradient Boosting, the algorithm needle, heart, and round shape leaves are correctly classified with 100% accuracy. But Random Forest and Gradient Boosting have same overall classification accuracy as 98%.

When consider the shape-wise (H1) accuracy of Swedish dataset, Random Forest algorithm is the best. Because feeding the Swedish dataset to Random Forest, the algorithm simple round, needle, heart, and round shape leaves are correctly classified with higher level of accuracy than other two algorithms. Random Forest also have the highest overall classification accuracy as 98%.

When consider the shape-wise (H1) accuracy of Kaggle dataset, Random Forest algorithm is the best. Because feeding the Kaggle dataset to Random Forest, the algorithm diamond, simple round, heart, and round shape leaves correctly classified with higher level of accuracy than other two algorithms. Random Forest also have the highest overall classification accuracy as 74%.

With all the features (with all feature categories), out of sample overall classification accuracy in H1 is higher for Random Forest than Extreme Gradient Boosting and Gradient Boosting of all the datasets.

As identified in the first phase of experiment 1, we used Random Forest to go to the bottom level of the hierarchy. Because Random Forest perform well in the first phase of experiment 1 with high value of overall classification accuracy.

The following Table 7 shows that the edge-wise (H2) and overall classification accuracy in experiment 1 which means that out of sample accuracy in H2 with Random Forest algorithm.

When consider the edge-wise (H2) accuracy of actual dataset for diamond and simple round leaves, Random Forest has the overall classification accuracy of 97% for each. There is 100% edge wise accuracy in smooth, and toothed edged diamond shaped leaves. But there is 100% edge wise accuracy in toothed, lobed,



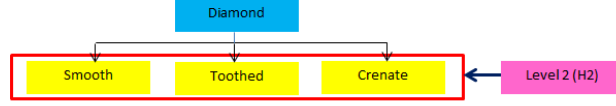


Figure 14: Level 2 hierarchy (H2) of diamond shaped leaves

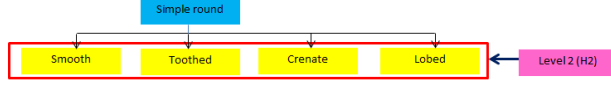


Figure 15: Level 2 hierarchy (H2) of simple round shaped leaves

Dataset	Diamond			Overall Classification Accuracy	Simple Round				Overall Classification Accuracy
	Smooth	Toothed	Crenate		Smooth	Toothed	Lobed	Crenate	
<b>Actual</b>									
Random Forest	1.00	1.00	0.4	0.97	0.94	1.00	1.00	1.00	0.97

Table 6: Edge-wise (H2) and overall classification accuracy (Training and test sets are from the Actual dataset)

and crenate edged simple round shaped leaves.

The following Table 7 shows that the species-wise (H3) and overall classification accuracy of heart and needle shaped leaves in experiment 1 which means that out of sample accuracy in H3 with Random Forest algorithm of heart and needle shaped leaves.



Figure 16: Level 3 hierarchy (H3) of needle shaped leaves



Figure 17: Level 3 hierarchy (H3) of heart shaped leaves

Dataset	Needle			Overall Classification Accuracy	Heart Shape			Overall Classification Accuracy
	Belathana	Hathawariya	Iramusu		Bulath	Gammiris	Rasakida	
<b>Actual</b>								
Random Forest	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00

Table 7: Species-wise (H3) and overall classification accuracy given the shape (Training and test sets are from the Actual dataset)

When consider the species-wise (H3) accuracy of actual dataset for heart and needle shaped leaves, Random Forest has the overall classification accuracy of 100% for each. There is 100% species wise accuracy in all the species of needle and heart shaped leaves.

The following Table 8 shows that the species-wise (H3) and overall classification accuracy in experiment 1 which means that out of sample accuracy in H3 with Random Forest algorithm.



Figure 18: Level 3 hierarchy (H3) of diamond shaped smooth edged leaves

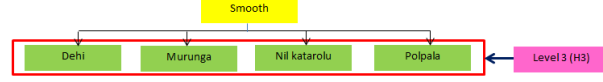


Figure 19: Level 3 hierarchy (H3) of simple round shaped smooth edged leaves

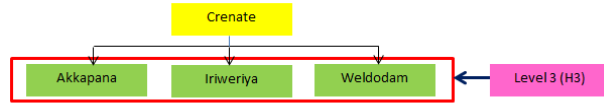


Figure 20: Level 3 hierarchy (H3) of simple round shaped crenate edged leaves



Figure 21: Level 3 hierarchy (H3) of simple round shaped lobed edged leaves

When consider the species-wise (H3) accuracy of actual dataset for all edge types in diamond and simple round leaves, Random Forest has the overall classification accuracy of 100% for each. There is 100% species-wise accuracy in all the species of diamond and simple round shaped leaves.

Dataset	Diamond												Simple Round												Crenate	
	Smooth												Smooth												Crenate	
	Adathoda	Beli	Bilin	Delum	Iguru	Kabaranga	Kurulu	Mukunuwenna	Rathmal	Thelbu	Thora	Overall Classification Accuracy	Dehi	Murunga	Nil katarolu	Polpala	Overall Classification Accuracy	Passion	Penela	Thel eradu	Overall Classification Accuracy	Akkapana	Iriweriya	Weldodam	Overall Classification Accuracy	
Actual																										
Random Forest	1.00	1.00	1.00	0.5	1.00	0.43	1.00	0.88	0.50	1.00	0.88	0.87	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	

Table 8: Species-wise (H3) and overall classification accuracy given the edge (Training and test sets are from the Actual dataset)

The following Table 9 shows that the shape-wise (H1) and overall classification accuracy only with shape features in experiment 1 which means that out of sample accuracy in H1 only with shape features.

When consider the shape-wise (H1) accuracy of actual dataset, Random Forest algorithm is best. Because feeding the actual dataset to the algorithm needle, and heart shape leaves are correctly classified with 100% accuracy. But Extreme Gradient Boosting has the highest overall classification accuracy as 92%.

When consider the shape-wise (H1) accuracy of Flavia dataset, Gradient Boosting algorithm is best. Because feeding the Flavia dataset to Gradient Boosting, the algorithm needle, heart, and round shape leaves are correctly classified with 100% accuracy. But all of the algorithms have overall classification accuracy as 96%.

When consider the shape-wise (H1) accuracy of Swedish dataset, Extreme Gradient Boosting algorithm is best. Because feeding the Swedish dataset to Gradient Boosting algorithm diamond, simple round, needle, and round shape leaves are correctly classified with 100% accuracy. Extreme Gradient Boosting also have the highest overall classification accuracy as 97%.

Dataset							Overall Classification Accuracy
Training Set	Test Set	Diamond	Simple Round	Needle	Heart shape	Round	
<b>Actual</b>	<b>Actual</b>						
	Random Forest	0.89	0.86	<b>1.00</b>	<b>1.00</b>	0.86	0.90
	Extreme Gradient Boosting	0.92	<b>0.89</b>	0.97	0.97	<b>1.00</b>	<b>0.92</b>
<b>Flavia</b>	Gradient Boosting	<b>0.93</b>	0.83	<b>1.00</b>	0.92	0.71	0.90
	<b>Flavia</b>						
	Random Forest	0.93	<b>1.00</b>	0.91	0.97	<b>1.00</b>	<b>0.96</b>
<b>Swedish</b>	Extreme Gradient Boosting	<b>0.96</b>	0.96	<b>1.00</b>	0.96	0.98	<b>0.96</b>
	Gradient Boosting	0.94	0.98	0.95	<b>1.00</b>	<b>1.00</b>	<b>0.96</b>
	<b>Swedish</b>						
	Random Forest	0.96	<b>1.00</b>	<b>1.00</b>	0.85	0.94	0.95
	Extreme Gradient Boosting	<b>0.99</b>	<b>1.00</b>	<b>1.00</b>	0.82	<b>1.00</b>	<b>0.97</b>
	Gradient Boosting	0.95	0.98	<b>1.00</b>	<b>1.00</b>	0.92	0.96

Table 9: Shape-wise (H1) and overall classification accuracy (Training and test sets are from the same datasets: Only with shape features)

Only with the shape features, out of sample overall classification accuracy in H1 is higher for Extreme Gradient Boosting than Random Forest and Gradient Boosting.

### 3.1.2 Experiment 2

As shown in Figure 22b, experiment 2 was designed by using 80% of training and 20% of test set of the different datasets. In this experiment we evaluate the generalizability of our algorithm (out of sample accuracy) by cross combining the different datasets. For example: New dataset 1 is created based on 80% data from Flavia and 20% data from Kaggle. Experiment 2 was conducted in two ways. First the experiment is conducted by considering all the features as inputs. Second the experiment is conducted by considering only the shape features as inputs. All of datasets used to evaluate only the first level of the hierarchy i.e: classification of leaves according to their shape.

There were 6 new datasets that created by existing datasets as shown in Figure 22b.

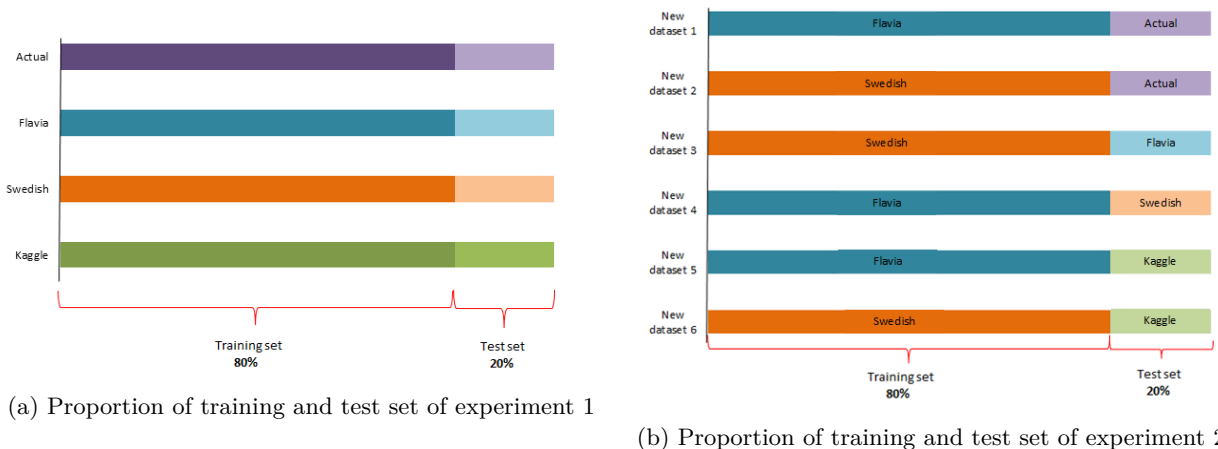


Figure 22

The following Table 10 shows that the shape-wise (H1) and overall classification accuracy in experiment 2 which means that out of sample accuracy in H1 with different algorithms.

Dataset							Overall Classification Accuracy
Training Set	Test Set	Diamond	Simple Round	Needle	Heart shape	Round	
<b>Flavia</b>	<b>Actual</b>						
	Random Forest	<b>0.70</b>	0.28	0.23	<b>0.04</b>	0.10	0.41
	Extreme Gradient Boosting	0.65	0.28	<b>0.56</b>	0.03	<b>0.29</b>	<b>0.43</b>
	Gradient Boosting	0.63	<b>0.31</b>	0.45	0.03	0.26	0.42
<b>Swedish</b>	<b>Actual</b>						
	Random Forest	0.38	0.22	0.06	<b>0.12</b>	0.39	0.26
	Extreme Gradient Boosting	0.40	0.23	<b>0.18</b>	0.04	0.23	0.25
	Gradient Boosting	<b>0.50</b>	<b>0.33</b>	0.11	<b>0.12</b>	<b>0.55</b>	<b>0.36</b>
<b>Swedish</b>	<b>Flavia</b>						
	Random Forest	0.29	<b>0.43</b>	0.46	0.0	0.0	0.27
	Extreme Gradient Boosting	<b>0.42</b>	0.34	0.40	0.01	0.01	0.31
	Gradient Boosting	0.40	0.37	<b>0.53</b>	<b>0.12</b>	<b>0.27</b>	<b>0.37</b>
<b>Flavia</b>	<b>Swedish</b>						
	Random Forest	0.57	0.72	0.03	0.0	0.01	0.39
	Extreme Gradient Boosting	<b>0.58</b>	0.67	<b>0.13</b>	0.0	0.15	<b>0.41</b>
	Gradient Boosting	0.51	<b>0.79</b>	0.12	0.0	<b>0.18</b>	<b>0.41</b>
<b>Flavia</b>	<b>Kaggle</b>						
	Random Forest	<b>0.66</b>	0.39	0.0	0.0	0.17	<b>0.44</b>
	Extreme Gradient Boosting	0.56	<b>0.48</b>	0.0	0.0	0.21	0.43
	Gradient Boosting	0.51	0.47	0.0	0.0	<b>0.25</b>	0.42
<b>Swedish</b>	<b>Kaggle</b>						
	Random Forest	0.42	0.12	0.0	0.07	0.05	0.24
	Extreme Gradient Boosting	0.35	0.12	0.0	0.10	<b>0.14</b>	0.23
	Gradient Boosting	<b>0.44</b>	<b>0.14</b>	—	<b>0.14</b>	0.13	<b>0.27</b>

Table 10: Shape-wise (H1) and overall classification accuracy (Training and test sets are from the different datasets)

When consider the shape-wise (H1) accuracy of new dataset 1, Extreme Gradient Boosting algorithm and Random Forest are best. Because feeding the new dataset 1 to the Random Forest and Extreme Gradient Boosting algorithms diamond, needle, heart, and round shape leaves are correctly classified with higher accuracy. But Extreme Gradient Boosting have same overall classification accuracy of 43%.

When consider the shape-wise (H1) accuracy of new dataset 2, Gradient Boosting algorithm is best. Because feeding the new dataset 2 to Gradient Boosting, the algorithm diamond, simple round, heart, and round shape leaves are correctly classified with higher level of accuracy than other two algorithms. Gradient Boosting has the highest overall classification accuracy as 36%.

When consider the shape-wise (H1) accuracy of new dataset 3, Gradient Boosting algorithm is best. Because feeding the new dataset 3 to Gradient Boosting algorithm needle, heart, and round shape leaves are correctly classified with higher level of accuracy than other two algorithms. Gradient Boosting also has the highest overall classification accuracy as 37%.

When consider the shape-wise (H1) accuracy of new dataset 4, Gradient Boosting algorithm is best.

Because feeding the new dataset 4 to the Gradient Boosting algorithm simple round and round shape leaves correctly classified with higher level of accuracy than other two algorithms. Extreme Gradient Boosting and Gradient Boosting have same overall classification accuracy as 41%.

When consider the shape-wise (H1) accuracy of new dataset 5, Random Forest algorithm is best. Because feeding the new dataset 5 to Random Forest algorithm diamond shape leaves are correctly classified with higher level of accuracy than other two algorithms. Random Forest also has the highest overall classification accuracy as 44%.

When consider the shape-wise accuracy (H1) of new dataset 6, Gradient Boosting algorithm is best. Because feeding the new dataset 6 to Gradient Boosting algorithm diamond, simple round, and heart shape leaves are correctly classified with higher level of accuracy than other two algorithms. Gradient Boosting also has the highest overall classification accuracy as 27%.

With all the features, out of sample overall classification accuracy in H1 is higher for Random Forest than Extreme Gradient Boosting and Gradient Boosting in experiment 2.

The following Table 11 shows that the shape-wise (H1) and overall classification accuracy in experiment 2 which means that out of sample accuracy in H1 only with shape features.

Training Set	Dataset						Overall Classification Accuracy
	Test Set	Diamond	Simple Round	Needle	Heart shape	Round	
<b>Flavia</b>	<b>Actual</b>						
	Random Forest	0.57	0.29	0.81	0.04	0.13	0.42
	Extreme Gradient Boosting	<b>0.65</b>	<b>0.37</b>	0.85	0.03	0.48	<b>0.50</b>
	Gradient Boosting	0.55	0.36	<b>0.95</b>	<b>0.05</b>	<b>0.58</b>	0.49
<b>Swedish</b>	<b>Actual</b>						
	Random Forest	0.37	0.31	<b>0.09</b>	<b>0.10</b>	0.14	0.24
	Extreme Gradient Boosting	0.38	<b>0.35</b>	0.0	0.05	0.08	0.23
	Gradient Boosting	<b>0.53</b>	0.27	0.03	0.08	<b>0.39</b>	<b>0.31</b>
<b>Swedish</b>	<b>Flavia</b>						
	Random Forest	0.29	<b>0.45</b>	0.38	0.0	0.0	0.27
	Extreme Gradient Boosting	0.46	0.32	<b>0.41</b>	0.0	0.06	0.33
	Gradient Boosting	<b>0.52</b>	0.36	0.27	<b>0.13</b>	<b>0.20</b>	<b>0.38</b>
<b>Flavia</b>	<b>Swedish</b>						
	Random Forest	0.46	<b>0.88</b>	0.15	0.0	<b>0.01</b>	<b>0.39</b>
	Extreme Gradient Boosting	<b>0.47</b>	0.72	<b>0.16</b>	0.0	<b>0.01</b>	0.36
	Gradient Boosting	0.39	0.84	<b>0.16</b>	0.0	<b>0.01</b>	0.36

Table 11: Shape-wise (H1) and overall classification accuracy (Training and test sets are from the different datasets: Only with Shape Features)

When consider the shape-wise (H1) accuracy of new dataset 1, Gradient Boosting algorithm is best. Because feeding the new dataset 1 to the Gradient Boosting algorithms needle, heart, and round shape leaves are correctly classified with higher accuracy. But Extreme Gradient Boosting has the highest overall classification accuracy of 50%.

When consider the shape-wise (H1) accuracy of new dataset 2, Gradient Boosting and Random Forest algorithms are best. Because feeding the new dataset 2 to Gradient Boosting and Random Forest algorithms diamond, needle, heart, and round shape leaves are correctly classified with higher level of accuracy than other two algorithms. Gradient Boosting has the highest overall classification accuracy as 31%.

When consider the shape-wise (H1) accuracy of new dataset 3, Gradient Boosting algorithm is best. Because feeding the new dataset 3 to Gradient Boosting algorithm diamond, heart, and round shape leaves are correctly classified with higher level of accuracy than other two algorithms. Gradient Boosting also has the highest overall classification accuracy as 38%.

When consider the shape-wise (H1) accuracy of new dataset 4, Random Forest is best. Because feeding the new dataset 4 to the Random Forest algorithm simple round shape leaves correctly classified with higher level of accuracy than other two algorithms. Random Forest has the highest overall classification accuracy as 39%.

Only with shape features, out of sample overall classification accuracy in H1 is higher for Extreme Gradient Boosting than Random Forest and Gradient Boosting in experiment 2.

With all the features, out of sample overall classification accuracy in H1 is higher for Random Forest than Extreme Gradient Boosting and Gradient Boosting in experiment 1 and 2 when consider all the datasets.

Only with shape features, out of sample overall classification accuracy in H1 is higher for Extreme Gradient Boosting than Random Forest and Gradient Boosting in experiment 1 and 2 when consider all the datasets.

With all features and only with shape features, out of sample overall classification accuracy in H1 of experiment 1 is higher than experiment 2. i.e: the experiment with training and test set from same dataset gives the best results to classify according to the first level of the hierarchy (H1).

### 3.2 Algorithm Performance & Comparison with Benchmarks

In this section, we discuss the performance of algorithms under hierarchical and non-hierarchical classification approaches that is followed by actual image dataset. Random Forest, Extreme Gradient Boosting, and Gradient Boosting algorithms are compared in both hierarchical and non-hierarchical approaches. Other than that Linear Discriminant Analysis (LDA), K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Probabilistic Neural Network (PNN), and Artificial Neural Network (ANN) algorithms are compared in non-hierarchical approach. To compare the results overall classification accuracy, macro average accuracy, and weighted average accuracy are considered. By calculating the average of ranks, the best algorithm can be find from all 3 of the accuracy types.

Case	Algorithm	Overall classification		Macro average		Weighted average		Average of ranks
		Accuracy	Rank	Accuracy	Rank	Accuracy	Rank	
Using hierarchical level	Random Forest	<b>1.00</b>	1	<b>0.99</b>	1	<b>1.00</b>	1	<b>1</b>
	Extreme gradient boosting	0.99	2	0.99	1	0.99	2	1.67
	Gradient Boosting	0.99	2	0.99	1	0.99	2	1.67
	Radom forest	0.99	2	0.99	1	0.99	2	1.67
	Extreme gradient boosting	0.98	5	0.98	5	0.98	5	5
Without using hierarchical level	Gradient boosting	0.98	5	0.98	5	0.98	5	5
	LDA	0.98	5	0.98	5	0.98	5	5
	KNN	0.84	9	0.81	9	0.84	9	9
	SVM	0.47	10	0.40	11	0.42	11	10.67
	PNN	0.47	10	0.44	10	0.45	10	10
	ANN	0.98	5	0.98	5	0.98	5	5

Table 12: Comparison of overall classification accuracy, weighted, and micro average accuracy and their ranks

As shown in the above Table 12, hierarchical approach with Random Forest has the highest overall classification accuracy. Furthermore, hierarchical approach with Random Forest obtained rank 1 across all categories.

### 3.3 Visualization of Algorithm Performance

In this section, we use Linear Discriminant Analysis (LDA) as high dimensional visualization approaches to visualize what is happening inside the trained algorithm and provides transparency to our black-box model.

#### 3.3.1 Training and Test both are in Actual Leaf Image Dataset

##### LDA with Random Forest

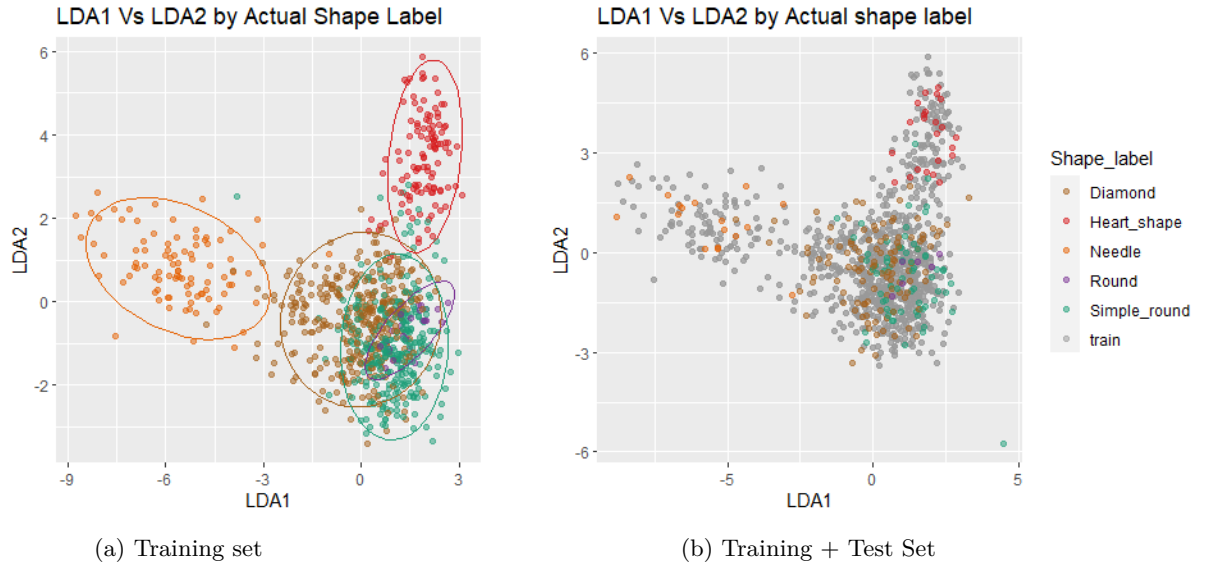


Figure 23: LDA1 Vs LDA2 of Actual leaf image dataset with actual shape labels

According to Figure 23a, needle and heart shaped Leaves are correctly classified. But there is an overlap of diamond, round, and simple round shaped leaves. In the test set, the leaves are also projected to the same space as in the training set according to Figure 23b. But there is a leaf that is not in the training space. The actual label of this leaf is simple round.

According to Figure 24a, needle and heart shaped Leaves are correctly classified. But there is an overlap of diamond, round, and simple round shaped leaves. In the test set, the leaves are also projected to the same space as in the training set according to Figure 24b. But there is a leaf that is not in the training space. The predicted label of this leaf is simple round.

Therefore as in Figure 23b and 24b, there is a leaf that is not in the training space. Both actual label and predicted label of this leaf is simple round.

According to Figure 25a, needle shaped leaves are correctly classified. But there is an overlap of diamond, round, heart, and simple round shaped leaves. In the test set, the leaves are also projected to the same space as in the training set according to Figure 25b. But there are three leaves that are not in the training space. The actual label of these leaves are simple round and diamond. Out of these leaves two are simple round and the other one is diamond shaped.

According to Figure 26a, needle shaped leaves are correctly classified. But there is an overlap of diamond, round, heart, and simple round shaped leaves. In the test set, the leaves are also projected to the same space as in the training set according to Figure 26b. But there are three leaves that are not in the training space. The predicted shape label of these leaves are simple round and needle. Out of these leaves two are simple round and the other one is needle shaped.

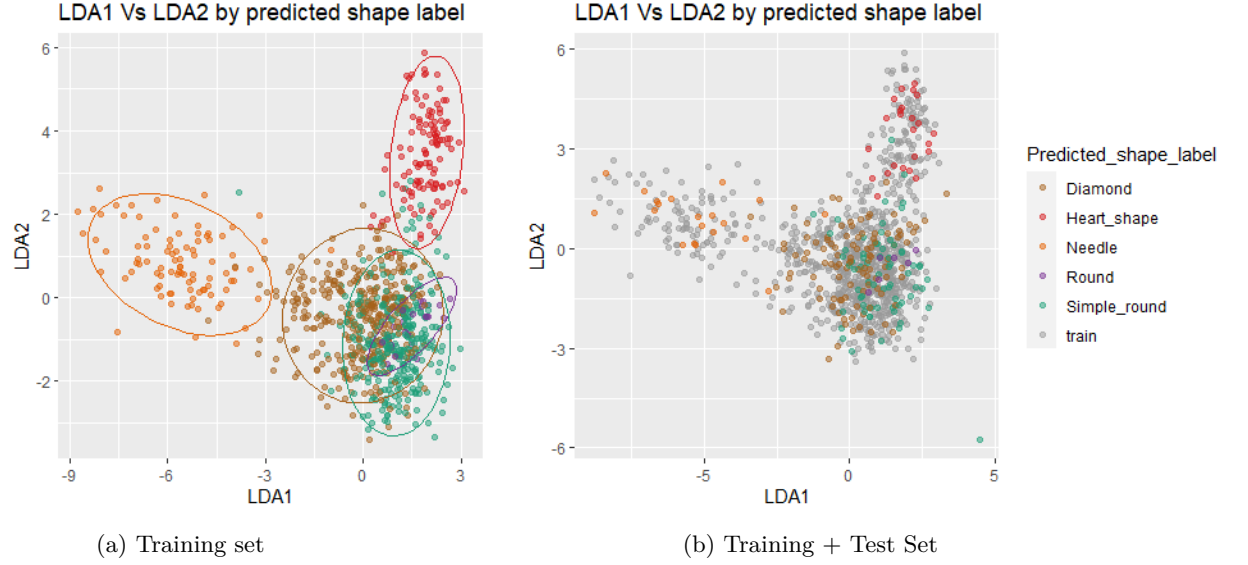


Figure 24: LDA1 Vs LDA2 of Actual leaf image dataset with predicted shape labels

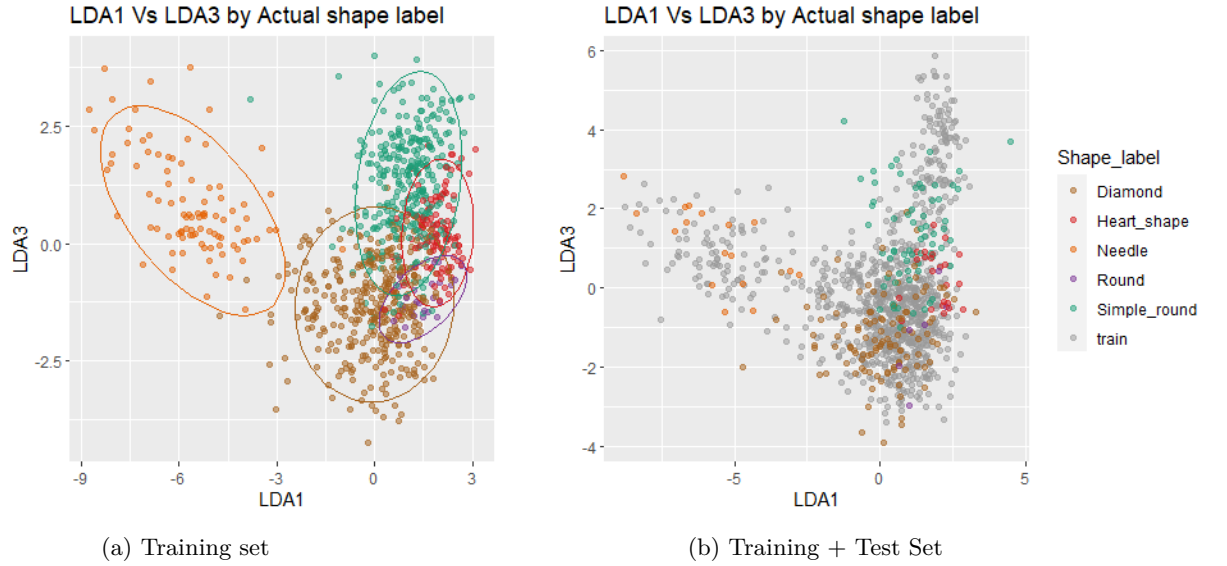


Figure 25: LDA1 Vs LDA3 of Actual leaf image dataset with actual shape labels

Therefore as in Figure 25b and 26b, there are three leaves that are not in the training space. Among three, two of them have the actual label simple round. The predicted label of these two are also simple round. But the remaining leaf has actual label diamond and the predicted label of that leaf is needle.

According to Figure 27a, there is an overlap of all shaped leaves. In the test set, the leaves are also projected to the same space as in the training set according to Figure 27b. But there are three leaves that are not in the training space.

According to Figure 28a, there is an overlap of all shaped leaves. In the test set, the leaves are also projected to the same space as in the training set according to Figure 28b.

As shown in Figure 29, needle shaped leaves are correctly classified as shown in Figure A and B. Heart shaped leaves are correctly classified as shown in Figure A. There is overlap of diamond, round, and simple round shaped leaves as shown in Figure A, B, and C.



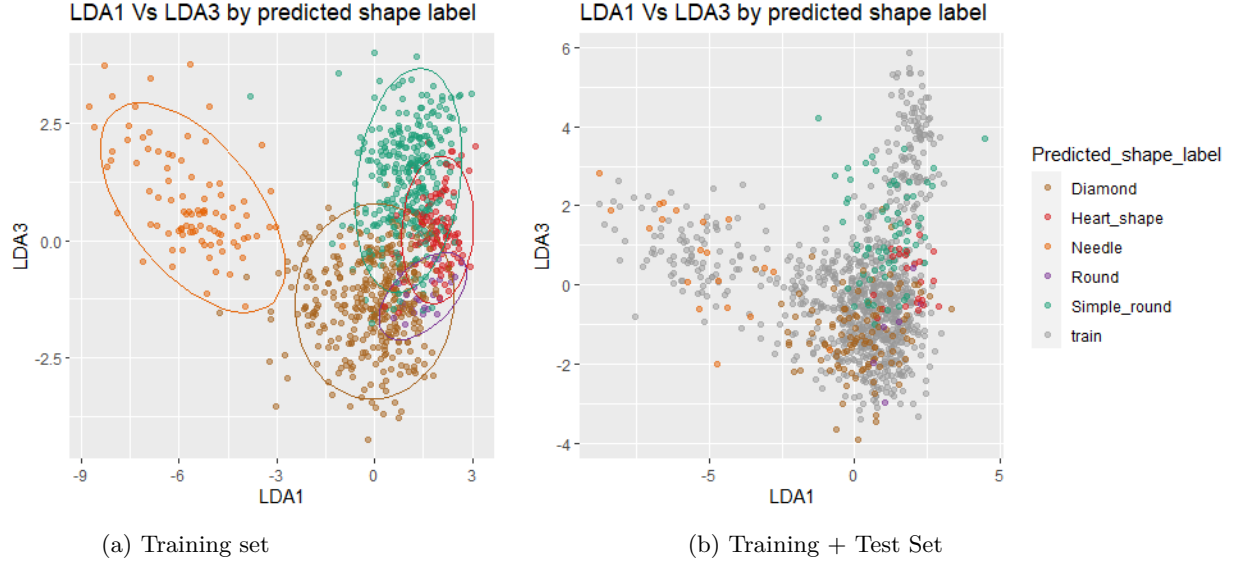


Figure 26: LDA1 Vs LDA3 of Actual leaf image dataset with predicted shape labels

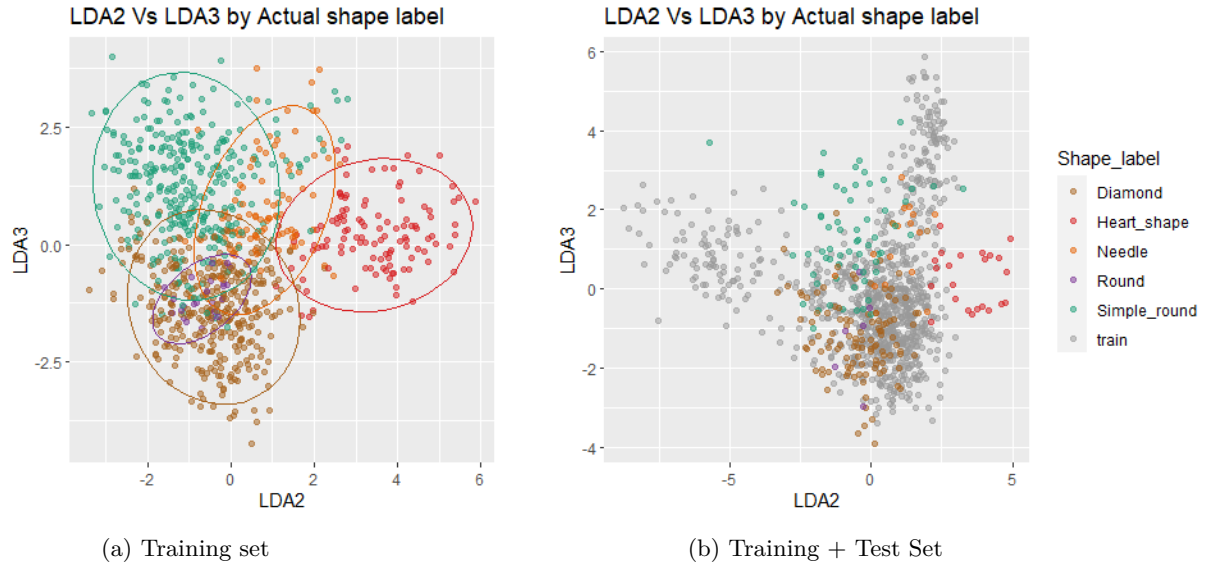


Figure 27: LDA2 Vs LDA3 of Actual leaf image dataset with actual shape labels

According to Figure 30, needle shaped leaves are correctly classified as shown in Figure A and B. Heart shaped leaves are correctly classified as shown in Figure A. There is overlap of diamond, round, and simple round shaped leaves as shown in Figure A, B, and C.

According to Figure 31, there are 7 misclassified leaves. They are predicted as diamond, but actually three of them are needle shaped, one is heart shaped and the remaining three are simple round shaped.

According to Figure 32, there are 7 misclassified leaves. They are predicted as diamond, but actually three of them are needle shaped, one is heart shaped and the remaining three are simple round shaped.

According to Figure 33, there are 7 misclassified leaves. They are predicted as diamond, but actually three of them are needle shaped, one is heart shaped and the remaining three are simple round shaped.

The above Figure 34 shows a good classification by using the Out of Bag (OOB) probabilities that

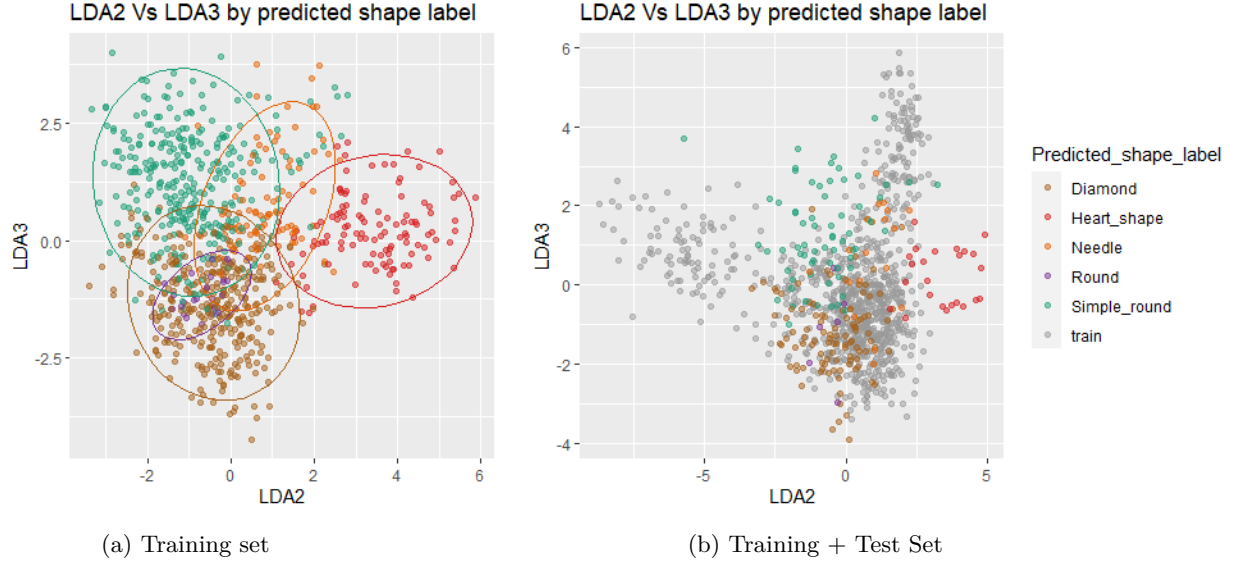


Figure 28: LDA2 Vs LDA3 of Actual leaf image dataset with predicted shape labels

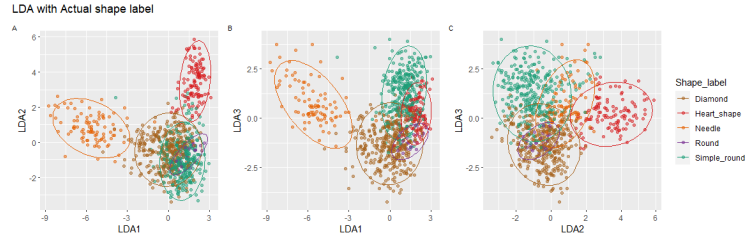


Figure 29: LDA of Actual leaf image dataset with actual shape labels

computed from Random Forest. The Out of Bag (OOB) probability is 1 for each shape label as in the training space shown in Figure F.

According to Figure 35, eccentricity values of needle shaped leaves are higher than all other shaped leaves. Therefore eccentricity can be used to classify needle shaped leaves. Area ratio convexity and perimeter convexity values are high for all shaped labels. Circularity, area, equivalent diameter, and physiological length values of needle shaped leaves are very low when compared with other shaped leaves. But compactness, narrow factor, and perimeter ratio length values of needle shaped leaves are higher than other shaped leaves. Therefore circularity, equivalent diameter, physiological length, compactness, narrow factor, and perimeter ratio length values are used to classify needle shaped leaves. Number of convex points of heart shaped leaves are higher than other shaped leaves. Therefore number of convex points can be used to classify heart shaped leaves. Area convexity, compactness, correlation, perimeter, perimeter ratio diameter and perimeter ratio length and width values of round shaped leaves are higher than other shaped leaves. Area ratio convexity, number of convex points, perimeter convexity, and rectangularity values are very low than other shaped leaves. Therefore area convexity, compactness, perimeter, perimeter ratio diameter, perimeter ratio length and width, area ratio convexity, number of convex points, perimeter convexity, and rectangularity values are

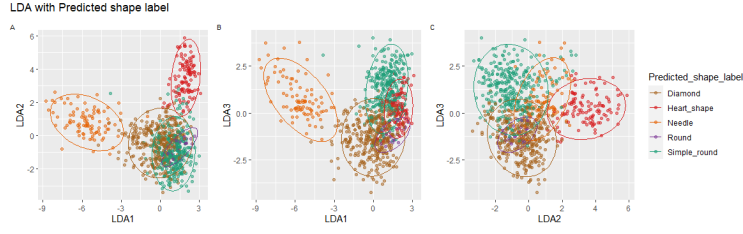


Figure 30: LDA of Actual leaf image dataset with predicted shape labels

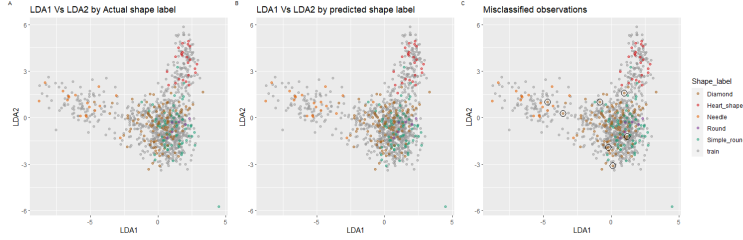


Figure 31: LDA1 Vs LDA2 of Actual leaf image dataset with actual, predicted shape labels and misclassified observations

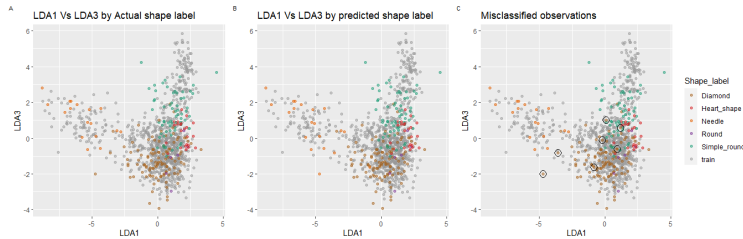


Figure 32: LDA1 Vs LDA3 of Actual leaf image dataset with actual, predicted shape labels and misclassified observations

used to classify round shaped leaves.

According to Figure 36, contrast values of heart shaped leaves are higher than other shaped leaves. Correlation texture values of heart, round, and simple round shaped leaves are higher than correlation texture

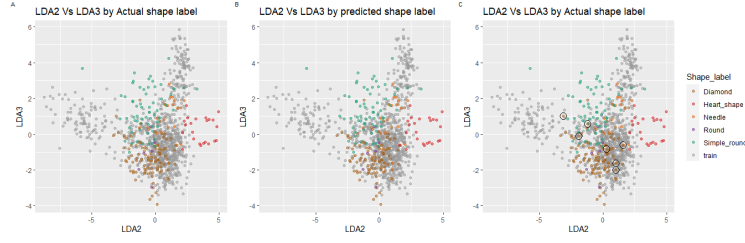


Figure 33: LDA2 Vs LDA3 of Actual leaf image dataset with actual, predicted shape labels and misclassified observations

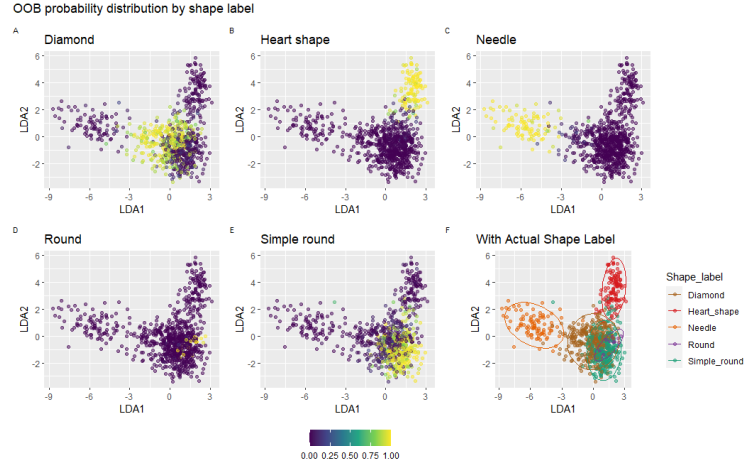


Figure 34: LDA1 Vs LDA2 of OOB probabilities by shape label

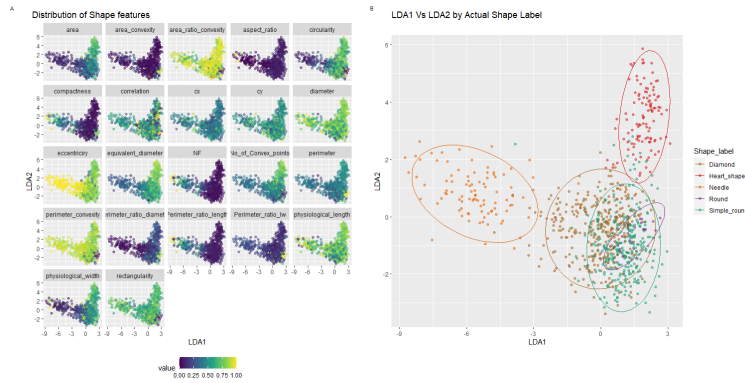


Figure 35: LDA1 Vs LDA2 of shape features

values of needle and diamond shaped leaves. Therefore correlation texture can be used to classified heart, round, and simple round shaped leaves from needle and diamond shaped leaves. Entropy values of heart shaped leaves are higher than other shaped leaves. Therefore entropy can be used to classify heart shape

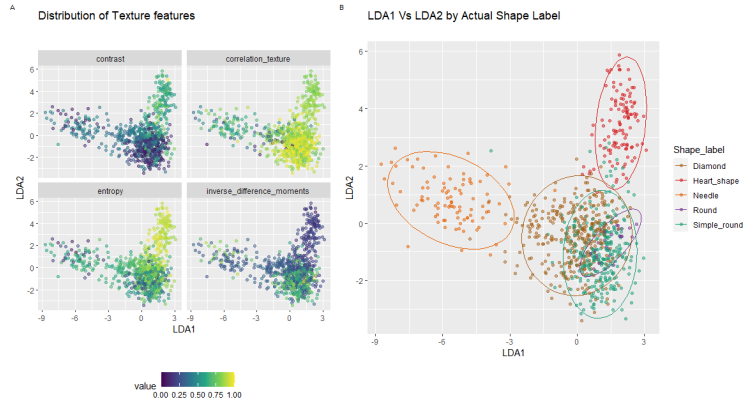


Figure 36: LDA1 Vs LDA2 of texture features

leaves. Inverse difference moments values of simple round shaped leaves are higher than other shaped leaves.

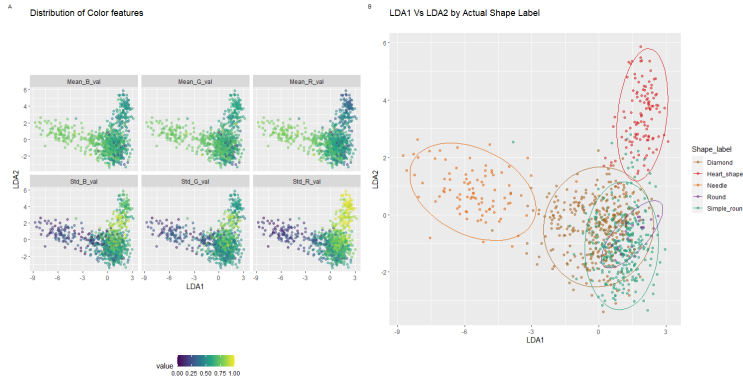


Figure 37: LDA1 Vs LDA2 of color features

According to Figure 37, mean blue, green, and red values of needle, diamond, round, and simple round shaped leaves are higher than mean blue, green, and red values of heart shaped leaves. Therefore mean blue, green, and red values can be used to classify heart shaped leaves. standard deviation of blue and green values of heart, round, and simple round shaped leaves are higher than standard deviation of blue and green values of needle and diamond shaped leaves. Standard deviation of red value of heart shaped leaves are higher than other shaped leaves. Therefore by using standard deviation of red value heart shaped leaves can be classified. Standard deviation of red value of needle shaped leaves are lower than other shaped leaves. Therefore standard deviation of red value can be used to needle shaped leaves.

According to Figure 38, monotonic contour, monotonic polar values of round shaped leaves are higher than other shaped leaves. Therefore monotonic contour and monotonic polar can be used to classify round shaped leaves. Striated contour, striated polar, stringy contour, and stringy polar values are high in all shaped leaves. Clumpy contour, clumpy polar, convex contour, outlying polar, sparse contour, sparse polar, and stringy polar values are low in all shaped leaves. Outlying contour has medium values for all shaped leaves.

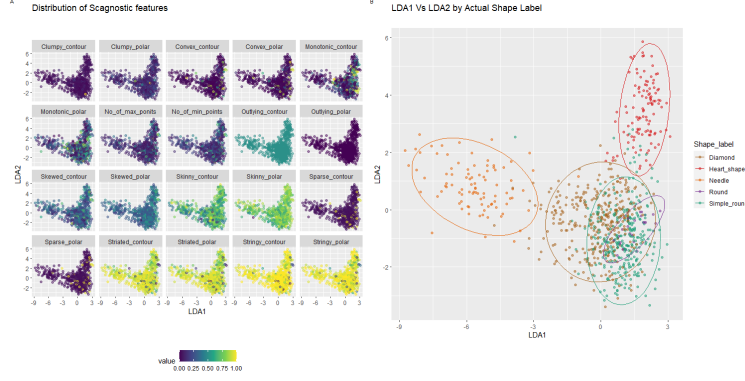


Figure 38: LDA1 Vs LDA2 of scagnostic features

### 3.3.2 Visualisation of Actual, Flavia and Swedish Leaf Datasets

Actual, Flavia and Swedish leaf image datasets contain color images. Therefore shape, texture, color, and scagnostic features are extracted. All together there are 52 features without the shape label. According to Figure 39a Actual, Flavia, and Swedish datasets are projected in three different spaces.

### 3.3.3 Visualisation of Actual, Flavia, Swedish, and Kaggle Leaf Datasets

Kaggle leaf image dataset contains only binary images. Therefore only shape and scagnostic features are extracted. When extracting shape and scagnostic features of all datasets, Kaggle leaf dataset is projected in a different space without overlapping with other three. According to Figure 39c Actual, Flavia, and Swedish datasets are projected with overlapping.

When extracting shape features of all datasets, Kaggle leaf dataset is projected in a different space without overlapping with other three. According to Figure 39b Actual, Flavia, and Swedish datasets are projected with overlapping.

## 4 Feature Importance

In this section, we discuss about the important features in each levels of the hierarchy.

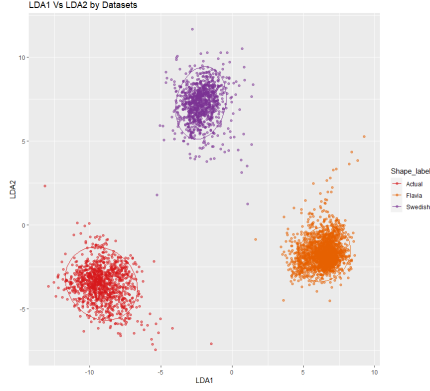
As shown in Figure 40 by exploring the variable importance measures of Random Forest, Gradient boosting, and Extreme gradient boosting algorithms on Actual leaf image dataset, we identified the most important features contributing to the shape-wise classification of the machine learning algorithms.

As shown in Figure 41 by exploring the variable importance measures of Random Forest on heart and needle shaped leaves in Actual leaf image dataset, we identified the most important features contributing to the species-wise classification of the machine learning algorithms.

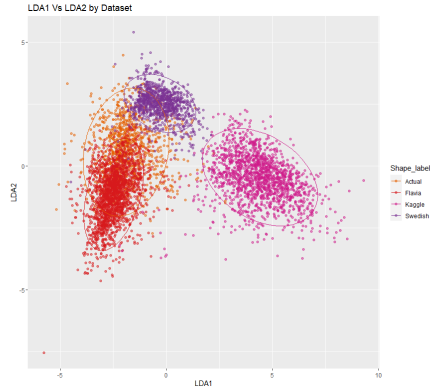
As shown in Figure 43 by exploring the variable importance measures of Random Forest on diamond and simple round shaped leaves in Actual leaf image dataset, we identified the most important features contributing to the edge-wise classification of the machine learning algorithms.

As shown in Figure 43 by exploring the variable importance measures of Random Forest on diamond shaped smooth edged leaves in Actual leaf image dataset, we identified the most important features contributing to the species-wise classification of the machine learning algorithms.

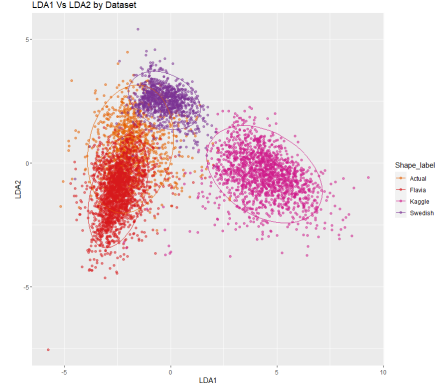
As shown in Figure 43 by exploring the variable importance measures of Random Forest on simple round shaped leaves in Actual leaf image dataset, we identified the most important features contributing to the species-wise classification of the machine learning algorithms.



(a) LDA1 Vs LDA2 of Actual, Flavia and Swedish leaf datasets with all feature categories



(b) LDA1 Vs LDA2 of Actual, Flavia, Swedish, and Kaggle Leaf Datasets only with shape features



(c) LDA1 Vs LDA2 of Actual, Flavia, Swedish, and Kaggle leaf datasets only with shape features

Figure 39

## 5 Discussion and Conclusions

Automatic medicinal plant species identification using leaf images is a popular research field with several critical applications. Through this research, we introduce an automatic algorithm to classify medicinal plants using medicinal plant leaves. Leaf images are considered as they contain large number of diverse set of features such as shape, veins, edge features, apices, etc that are useful in identifying medicinal plants.

In order to identify medicinal plant species using leaf images, we first do a preliminary study to get an idea about the morphological characteristics like shape, edge type, apex, base, arrangement etc. We identify five main shapes as: (i) Diamond, (ii) Simple round, (iii) Heart shaped, (iv) Needle, and (v) Round and four main edge types as: (i) Smooth, (ii) Toothed, (iii) Lobed, and (iv) Crenate by observing images in medicinal plant repository maintained by Barbeyrn Ayurveda resort and University of Ruhuna available at <http://www.instituteofayurveda.org/plants/>. Our observed results are converted into an open source R software package called MedLEA: **M**edicinal **LEA**f (<https://CRAN.R-project.org/package=MedLEA>). Furthermore, most of the researches are based on the existing databases like Flavia, Swedish etc. These existing databases contain few plant species and it is not sufficient to train a reliable model properly. In addition to that, a database of leaf images of medicinal plants in Sri Lanka is not yet available. Hence through this research, we establish a repository of medicinal plant images which is available at MedLEA. We collect the leaf images by following simplest and reliable approach which can be followed without expertise knowledge. The images were taken on a white background, positioning center of the white paper and the images are obtained from a normal smartphone without flash light to remove the shadow.

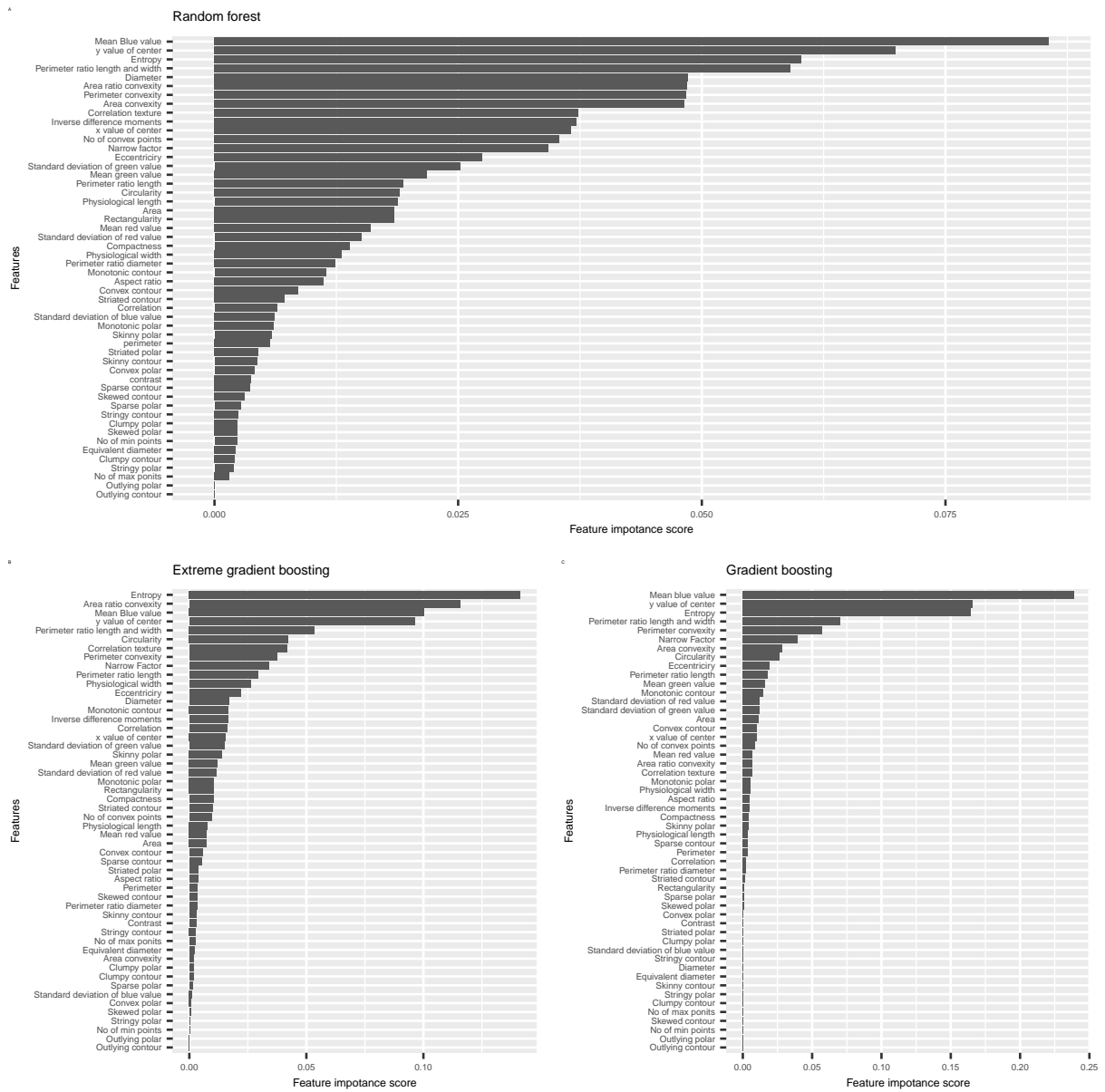


Figure 40: Importance of features in Actual Image dataset



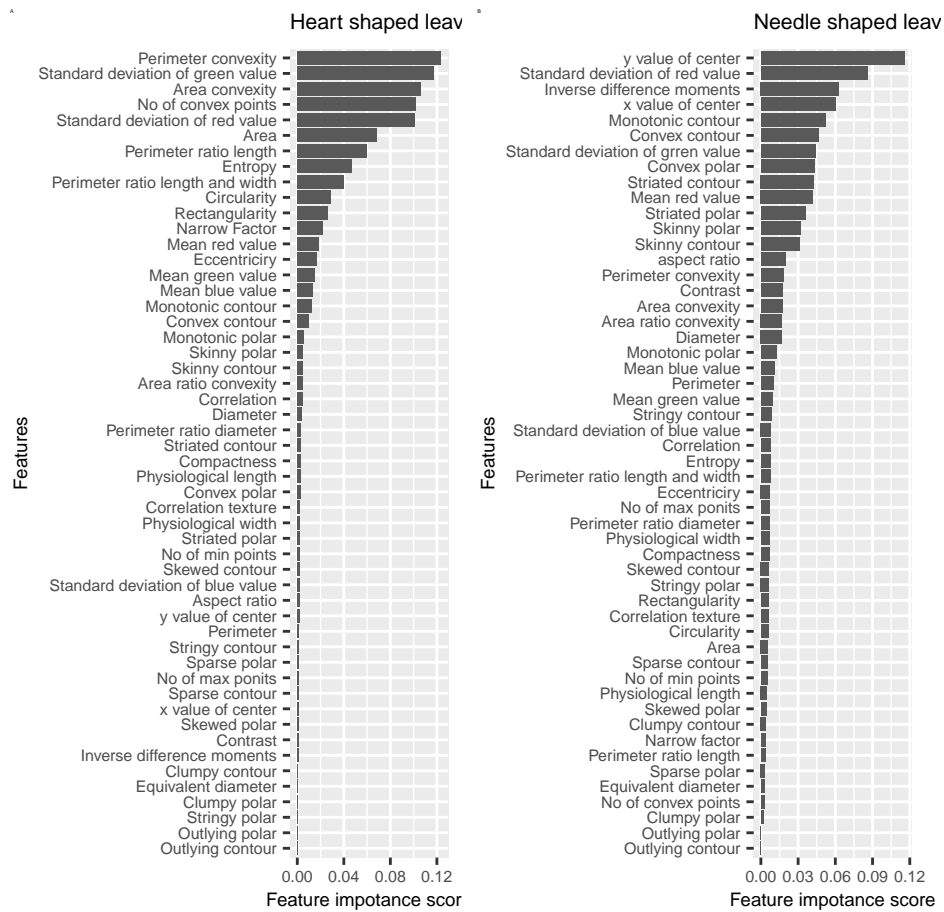


Figure 41: Importance of features in needle and heart shaped leaves of Actual Image dataset

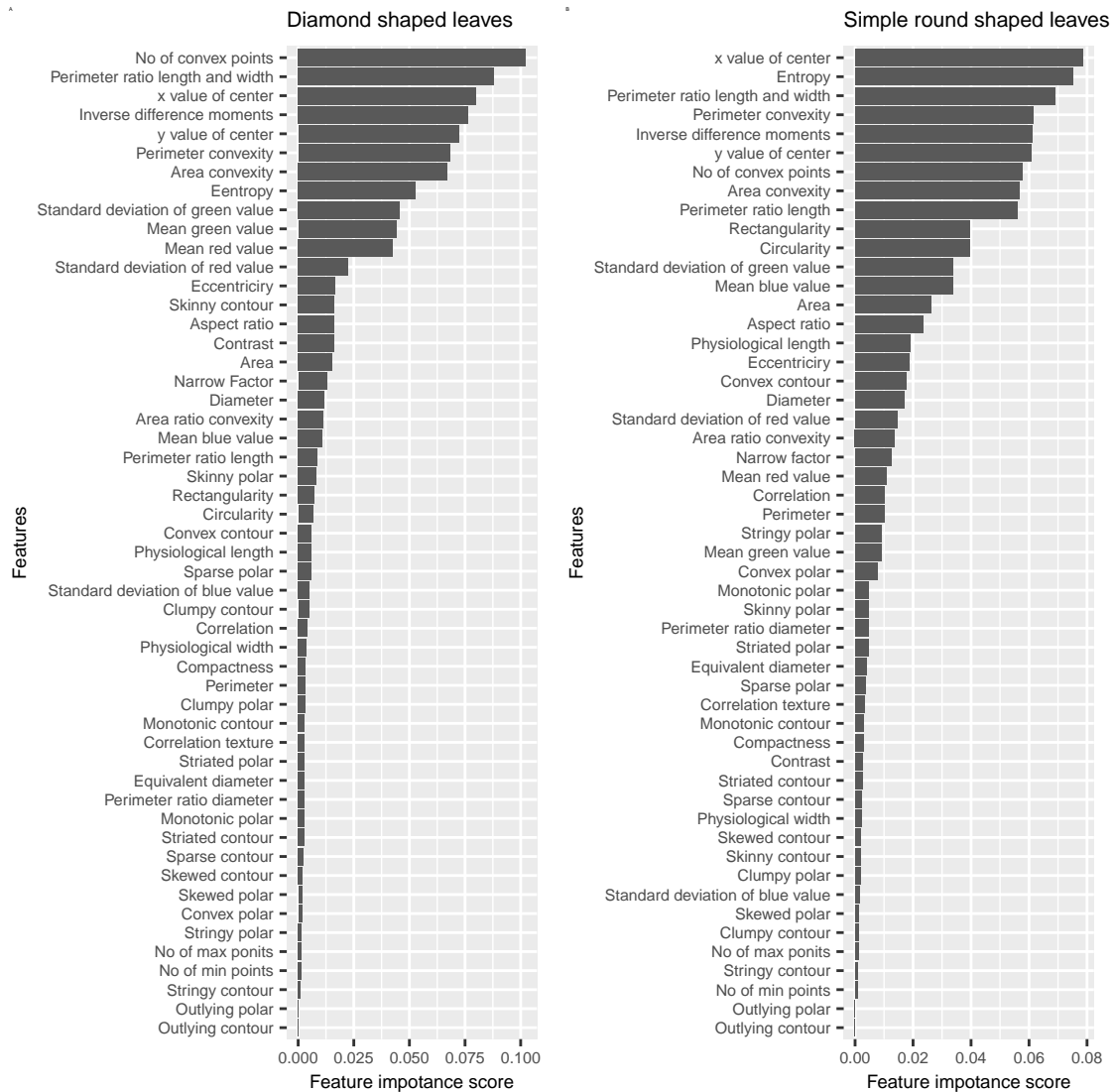


Figure 42: Importance of features in edge-wise classification of diamond and simple round shaped leaves of Actual Image dataset

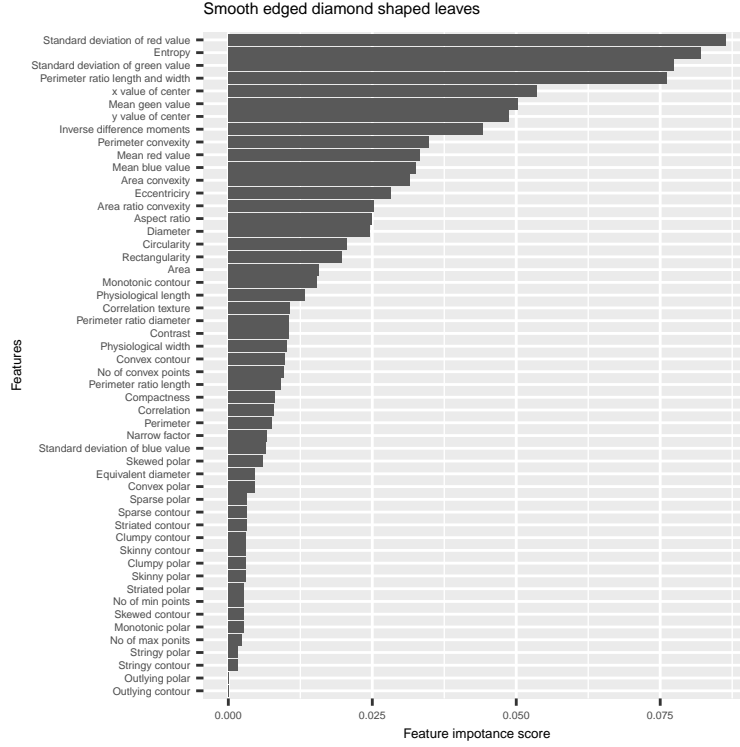


Figure 43: Importance of features in species-wise classification of diamond shaped smooth edged leaves of Actual Image dataset

Furthermore, we introduce our medicinal plant classification algorithm as MEDIPI : **MEDI**icinal **P**lant **I**dentification. The MEDIPI is divided into offline phase and online phase. The classification algorithm is trained in the offline phase. In the online phase, the pre-trained classification model is used to real-time leaf image classification for general users. Our classification algorithm operates on the features extracted from the image leaves. Through this research, we introduce 52 computer aided, interpretable features for leaf image recognition. There are four main categories of features that are used to classify leaf images. Many researches are based on shape, texture, and color features. In this research, we introduce new feature category called scagnostics for leaf image classification. Other than that correlation of cartesian coordinate, number of convex points, number of minimum and maximum points are introduced as new shape features. We explore the ability of features to discriminate the classes of interest under supervised learning and unsupervised learning settings using principal component analysis and linear discriminant analysis. Under both experimental settings clear separation of classes are visible in their projection spaces.

In addition to that, the offline phase of the algorithm contains four main steps: (i) Image processing, (ii) Feature extraction, (iii) Label images, and (iv) Trained a algorithm. The purpose of image processing is to improve the leaf image by removing undesired distortion. The main image processing steps are (i) Convert original image to RGB image, (ii) Gray scaling, (iii) Gaussian smoothing, (iv) Binary thresholding, (v) Remove stalk, (vi) Closing holes, and (vii) Resize image.

Furthermore, we train our algorithm using random forest, gradient boosting, and extreme gradient boosting. The model trained with random forest algorithm provides the highest accuracy. Our algorithm works as a hierarchical classification system. The hierarchy contains 3 levels. The first level classifies images according to the shape. The second level classifies according to the edge types. The bottom level classifies the plant species. We observe that shape features like (i) x value of Center (cx), (ii) y value of Center (cy), (iii) Entropy, (iv) Perimeter ratio of length and width, (v) Diameter, (vi) Area convexity, (vii) Perimeter convexity, (viii) Narrow Factor, (ix) Area ratio convexity, (x) Physiological length, (xi) Physiological width,

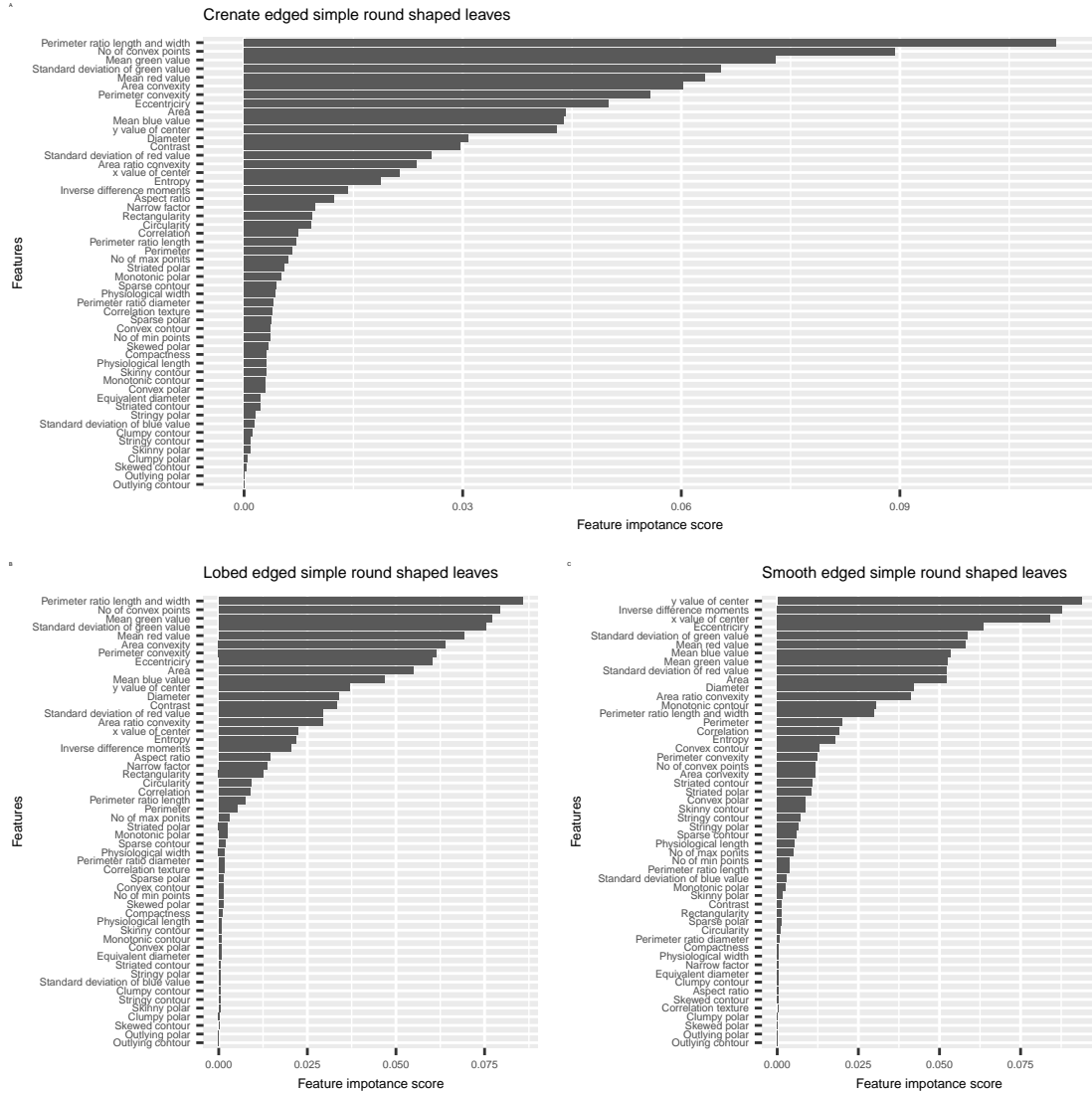


Figure 44: Importance of features in species-wise classification of simple round shaped leaves of Actual Image dataset

(xii) Rectangularity, and (xiii) Eccentricity are more important when classify the leaf images in the first level of the hierarchy. Scagnostic features like (i) Monotonic contour, (ii) Convex polar, (iii) Convex contour, (iv) Striated polar, (v) Striated contour, (vi) Skinny contour, and (vii) Skinny contour are more important in identifying leaf species in the bottom level of the hierarchy.

In addition to that, we use high dimensional visualization approaches as Linear Discriminant Analysis (LDA) to visualize what is happening inside the trained algorithm and provides transparency to our black-box model. We compare the accuracy of our proposed algorithm against several benchmarks and other commonly used algorithms for medicinal plants classification. The MEDIPI algorithm yields accurate results to the state-of-the-existing techniques in the field. We have to use training/test from same dataset to get accurate results. Most of the literatures are based on shape feature. By train the algorithms (i) Only with shape features, and (ii) With all feature categories (Shape, color, texture, scagnostic), we observe that shape feature is not sufficient to classify leaf images.

## Reference

- Azlah, Muhammad, Lee Suan Chua, Fakhrul Rahmad, Farah Abdullah, and Sharifah Alwi. 2019. "Review on Techniques for Plant Leaf Classification and Recognition." *Computers* 8 (October): 77. <https://doi.org/10.3390/computers8040077>.
- Devalaraja, Samir, Shalini Jain, and Hariom Yadav. 2011. "Exotic Fruits as Therapeutic Complements for Diabetes, Obesity and Metabolic Syndrome." *Food Research International (OTTAWA, ONT.)* 44 (August): 1856–65. <https://doi.org/10.1016/j.foodres.2011.04.008>.
- Goyal, N., Kapil, and N. Kumar. 2018. "Plant Species Identification Using Leaf Image Retrieval: A Study." In *2018 International Conference on Computing, Power and Communication Technologies (Guccon)*, 405–11.
- Gunawardana, Shehara, and W. J. A. Banukie Jayasuriya. 2019. "Medicinally Important Herbal Flowers in Sri Lanka." *Evidence-Based Complementary and Alternative Medicine* 2019 (May): 1–18. <https://doi.org/10.1155/2019/2321961>.
- Herdiyeni, Yeni, and Ni Wahyuni. 2012. "Mobile Application for Indonesian Medicinal Plants Identification Using Fuzzy Local Binary Pattern and Fuzzy Color Histogram." In *2012 International Conference on Advanced Computer Science and Information Systems, ICACSIS 2012 - PROCEEDINGS*.
- Nilaweera, D. D. 2010. "Journal of Machine Learning Research."
- Saslis-Lagoudakis, C. Haris, Julie Hawkins, Simon Greenhill, Colin Pendry, Mark Watson, Will Tuladhar-Douglas, Sushim Baral, and Vincent Savolainen. 2014. "The Evolution of Traditional Knowledge: Environment Shapes Medicinal Plant Use in Nepal." *Proceedings. Biological Sciences / the Royal Society* 281 (February): 20132768. <https://doi.org/10.1098/rspb.2013.2768>.
- Waisundara, Viduranga Y, and Mindani I Watawana. 2014. "The Classification of Sri Lankan Medicinal Herbs: An Extensive Comparison of the Antioxidant Activities." *Journal of Traditional and Complementary Medicine* 4 (3): 196–202. <https://doi.org/10.4103/2225-4110.126175>.
- Waldchen, Jana, and Patrick Mader. 2018. "Plant Species Identification Using Computer Vision Techniques: A Systematic Literature Review." *Archives of Computational Methods in Engineering* 25 (April): 507–43. <https://doi.org/10.1007/s11831-016-9206-z>.
- Wu, S. G., F. S. Bao, E. Y. Xu, Y. Wang, Y. Chang, and Q. Xiang. 2007. "A Leaf Recognition Algorithm for Plant Classification Using Probabilistic Neural Network." In *2007 Ieee International Symposium on Signal Processing and Information Technology*, 11–16.