

Computer Vision System for Automated Medicinal Plant Identification

TECH TALK

Jayani Lakshika

Associate Data Analyst

September 30, 2021

My SUPERVISOR



Dr. Thiyanga S. Talagala

Senior Lecturer in the Department of Statistics, Faculty of Applied Sciences at the University of Sri Jayewardenepura

PhD in Statistics, Monash University, Australia

<https://thiyanga.netlify.app/>



Department of Statistics
University of Sri Jayewardenepura, Sri Lanka



Department of Mathematics
University of Sri Jayewardenepura, Sri Lanka



Department of Computer Science
University of Sri Jayewardenepura, Sri Lanka



VIZUAMATIX
powered by innovation

Main objective

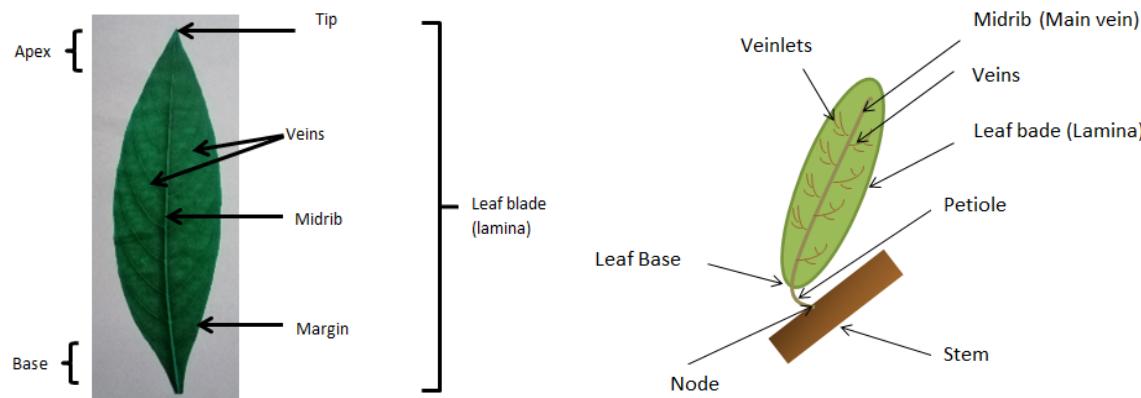
- To develop an automatic algorithm to classify medicinal plants by using statistical machine learning approach



Limitations

Why medicinal plant leaves?

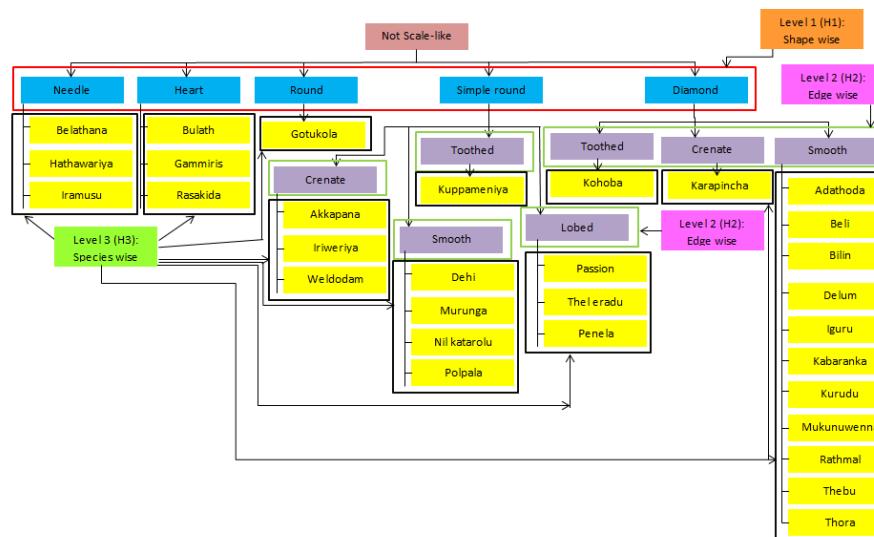
- Leaf images are considered as they contain large number of diverse set of features such as **shape, veins, edge features, apices, etc.**



- We used **non-diseased leaves with simple arrangement**.
- We used the leaves **without a petiole**.

Significance of the study

- To avoid misidentifying medicinal plants in Sri Lanka
- 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**.



Methodology



Workflow

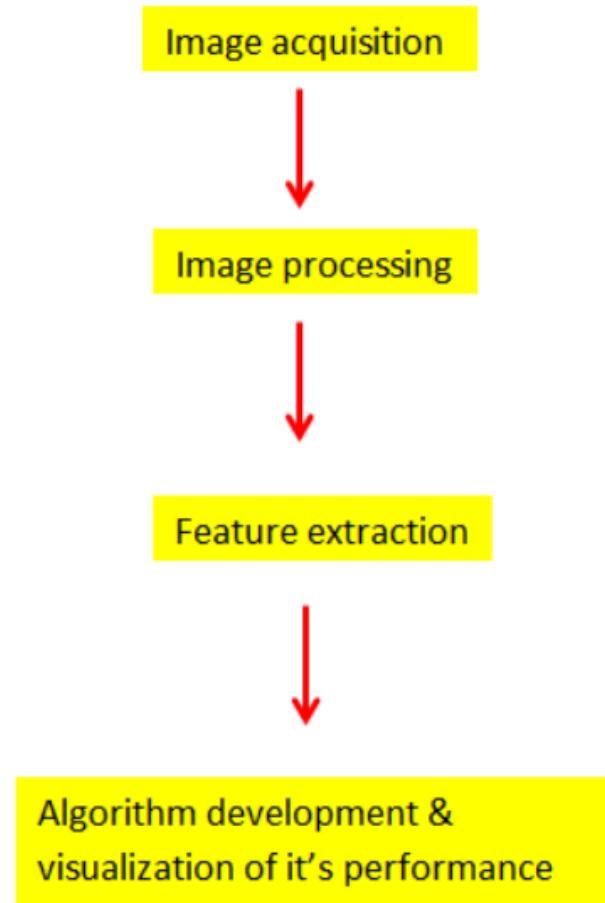
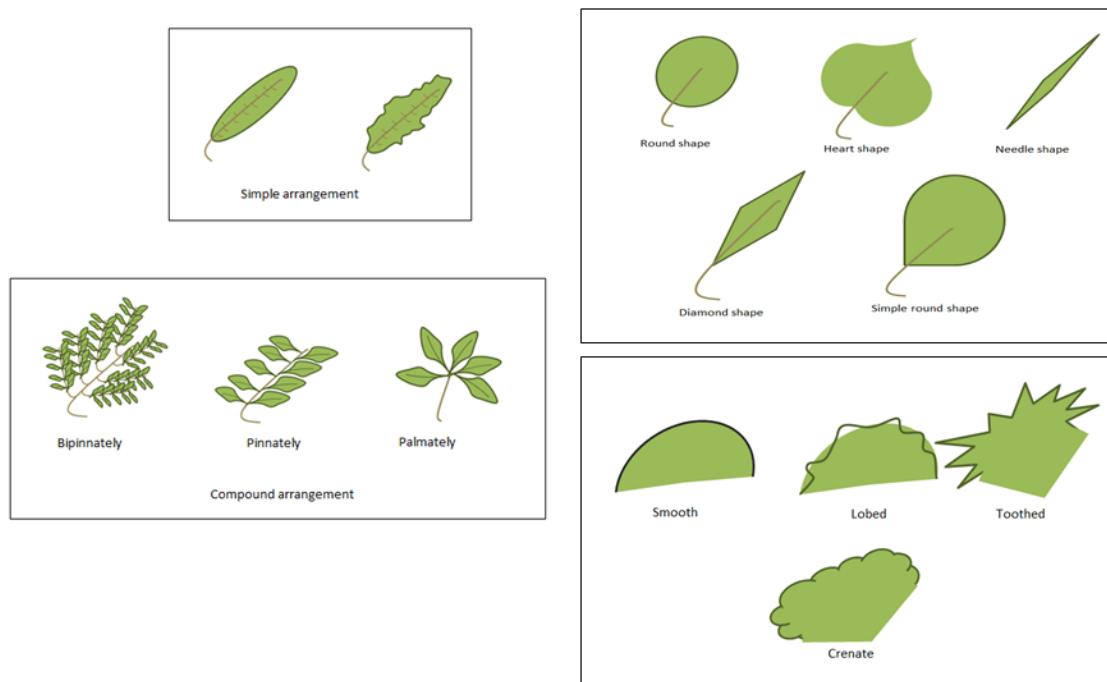
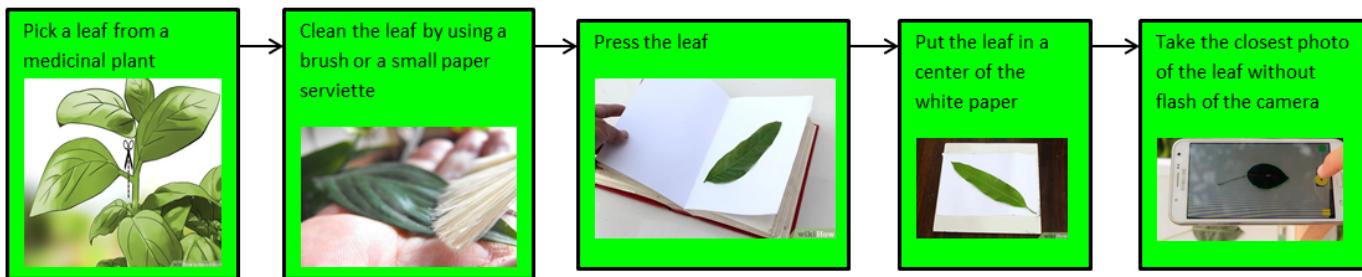


Image Acquisition

- A database of leaf images of medicinal plants in Sri Lanka **is not yet available.**
- **Establish a repository of medicinal plant images.**
- Preliminary study by using **471 medicinal plants** and recorded their characteristics like **leaf arrangement, shape, edge type etc.**



- Collected **1099** leaf images from **31 species**



MedLEA: Medicinal LEAf

MedLEA

CRAN 1.0.1 downloads 233/month

The MedLEA package provides morphological and structural features of 471 medicinal plant leaves and 1099 leaf images of 31 species and 29-45 images per species.

Installation

You could install the stable version on CRAN:

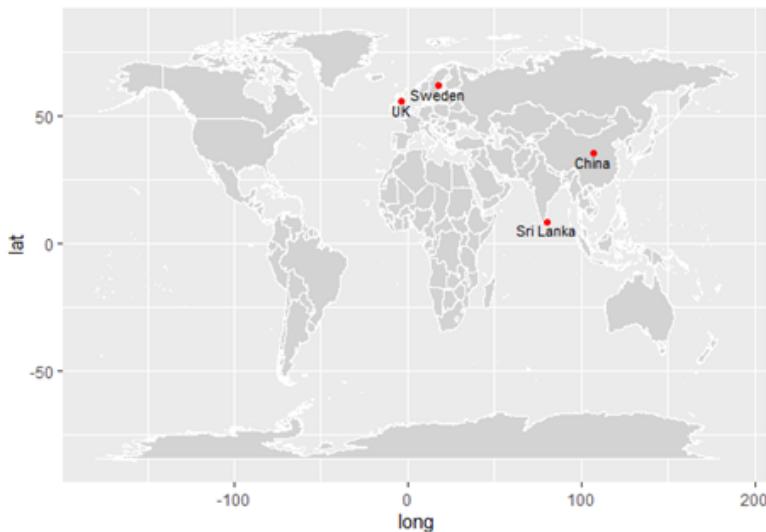
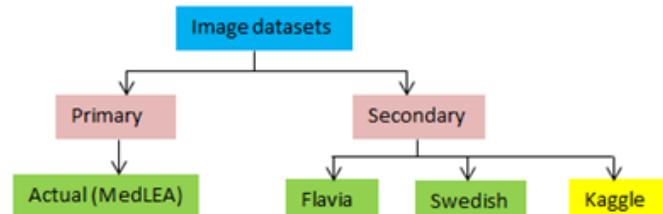
```
install.packages("MedLEA")
```

You can install the development version from GitHub with:

```
# install.packages("devtools")
devtools::install_github("SMART-Research/MedLEA")
```

repository is made available to the public through an **open-source R software** **MedLEA**, available at url(<https://CRAN.R-project.org/package=MedLEA>) for research reproducibility

Total downloads: More than 1000



Sample of Flavia dataset



Sample of Swedish dataset



Sample of Kaggle dataset



Sample of MedLEA dataset

Methodology Diagram

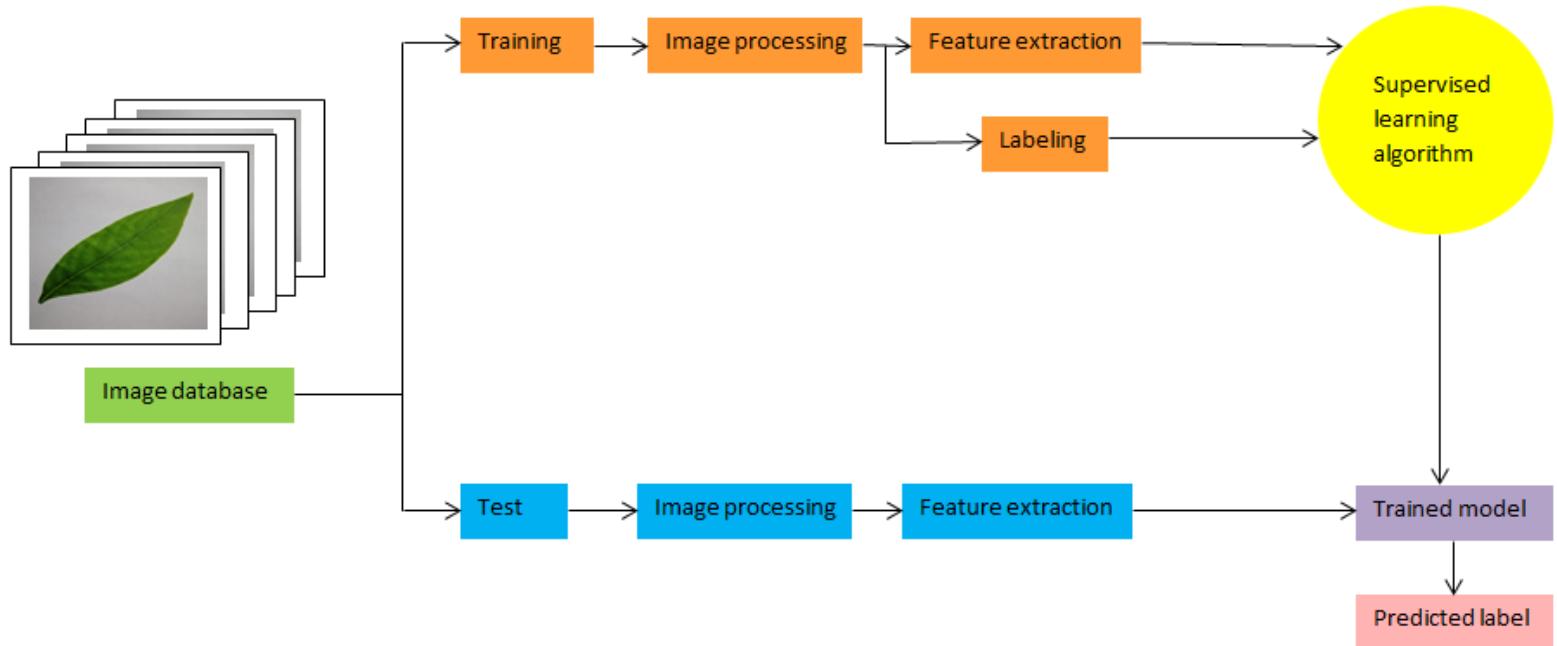
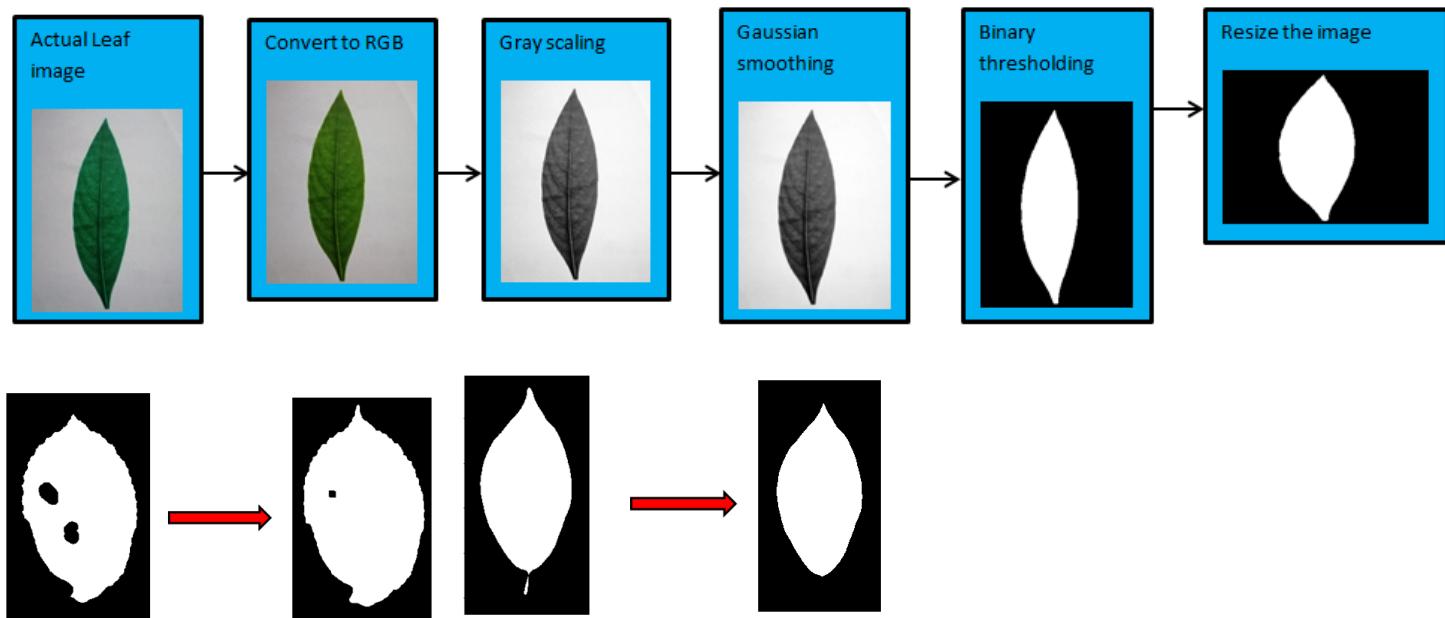
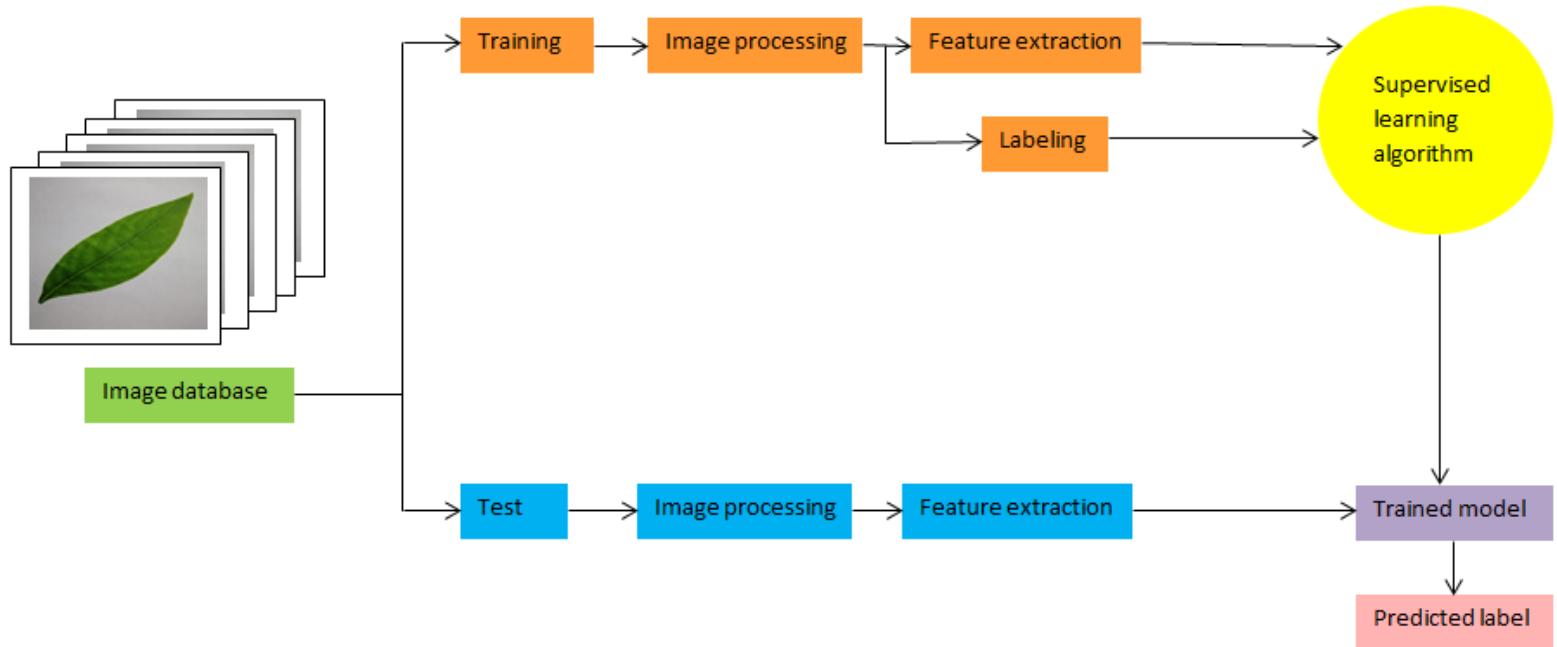


Image processing

- The image processing receives an image as input and generates a **modified image** as an output which is suitable for better **morphological analysis, feature extraction**.
- Image processing is an essential step to **reduce noise, background subtraction and content enhancement** in the identification process.



Methodology Diagram



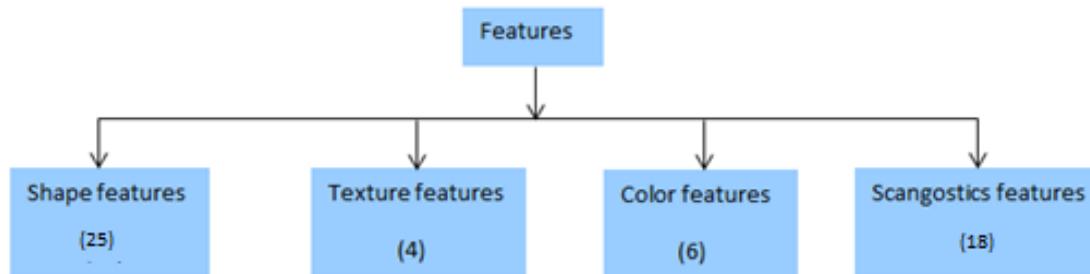
Why feature extraction is important?

- Recently, many researchers use deep learning methods like CNN (Convolution Neural Network) to classify plants - directly using plant images.
- Even though deep learning models have achieved great success, their interpretability, and transparency of the deep learning models are limited.



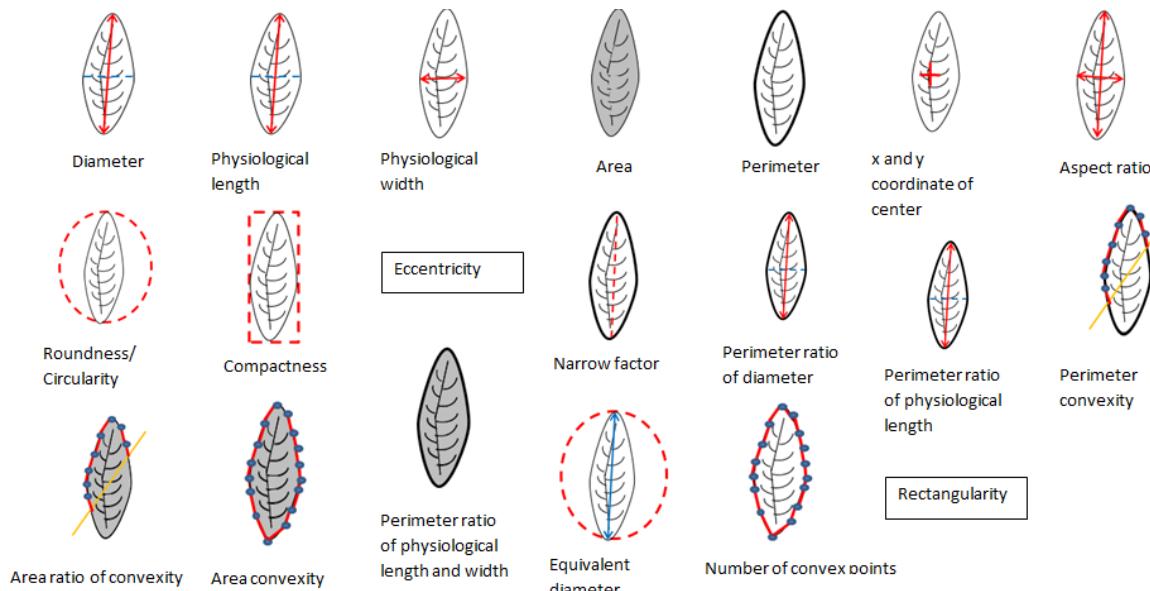
Features

- In identification of plant species by using leaf images, **features of the leaves play a main role.**
- In previous research, let the algorithm like **CNN** to extract features by itself and do the classification.
- Therefore it is so **hard to interpret and generalize** the features.
- We introduced **pre-calculate features** which can be **easy to interpret and generalize**. They are also **computational efficient**.

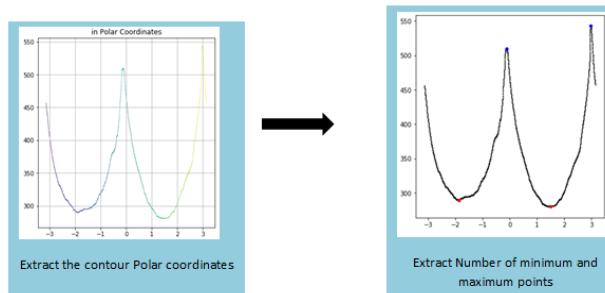


- We identified altogether **52 features**.

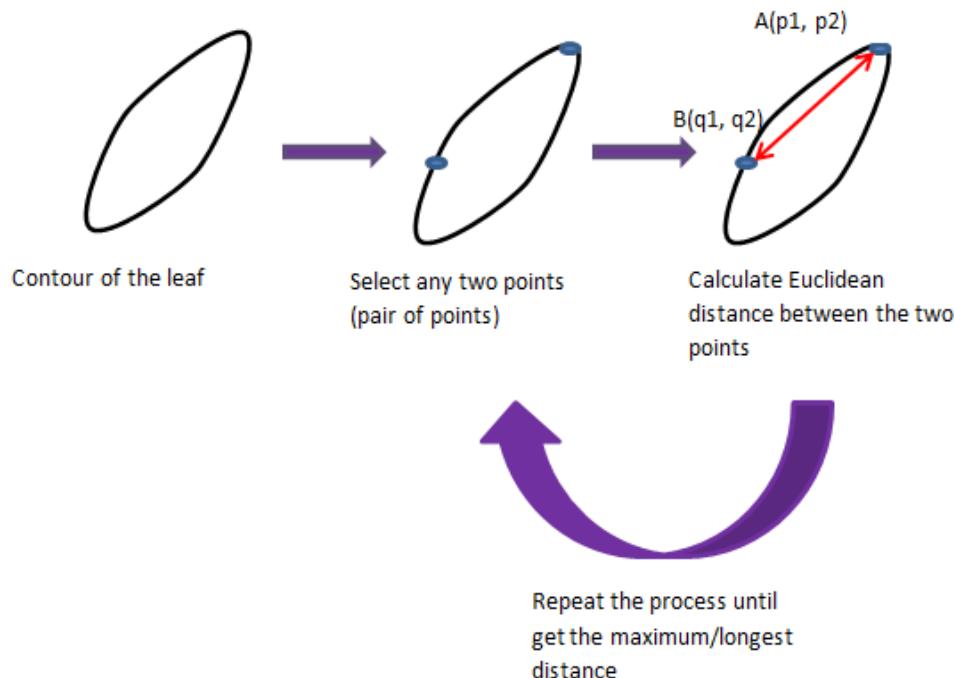
Shape features



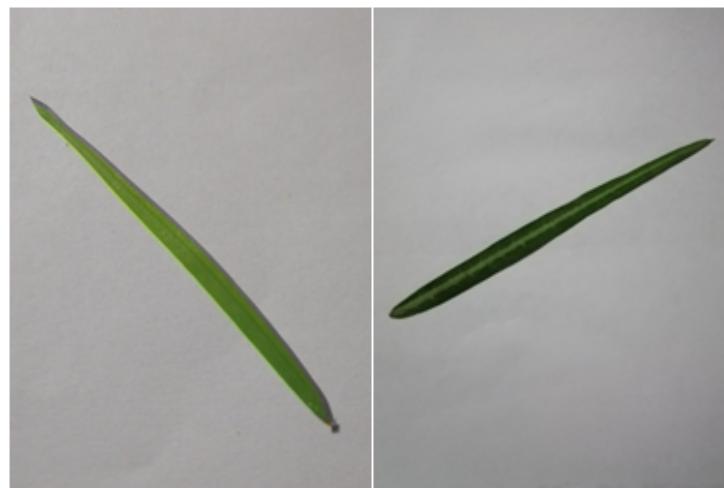
New shape features: Correlation of cartesian coordinate, number of convex points, number of minimum and maximum points



Diameter calculation



Color features



$$M = \frac{\text{Total insensity value of } r^{\text{th}} \text{ channel of the image pixels}}{\text{Total intensity value of the image}},$$

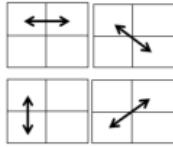
$$SD = \frac{\sqrt{\sum_{j=0}^h (r^{\text{th}} \text{ channel intensity}_j - r^{\text{th}} \text{ mean value})^2}}{\text{Total intensity value of the image}}.$$

Texture features

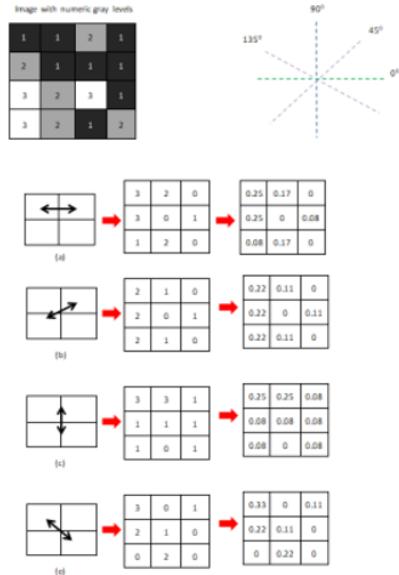
| Feature name | Detailed description | Formula | Value range |
|----------------------------|---|--|----------------|
| Contrast | Measures the relation or difference between the highest and lowest gray levels of the GLCM | $\frac{\sum_{a=1}^{\text{columns}} \sum_{b=1}^{\text{rows}} (a-b)^2 h(a,b)}{\text{Number of gray levels}-1}$ | $[0, \infty]$ |
| Entropy | Measures the randomness which means that how uniform the image is | $-\sum_{a=1}^{\text{columns}} \sum_{b=1}^{\text{rows}} h(a,b) \log_2(h(a,b))$ | $[-\infty, 0]$ |
| Correlation | Measurement of dependence of gray levels of the GLCM. | $\frac{\sum_{a=1}^{\text{columns}} \sum_{b=1}^{\text{rows}} (ab)h(a,b) - \mu_x \mu_y}{\sigma_x \sigma_y}$ | $[-1, 1]$ |
| Inverse difference moments | It measures that how a particular pixel is correlated to its neighbor pixel over the whole image It's a measure of homogeneity. Inverse level of contrast that measures how close the values of GLCM to diagonal values in GLCM | $\sum_{a=1}^{\text{columns}} \sum_{b=1}^{\text{rows}} \frac{h(a,b)}{(a-b)^2}$ | $[0, \infty]$ |

$$GLCM = \begin{bmatrix} h(1,1) & h(1,2) & \dots & \dots & h(1,n) \\ h(2,1) & h(2,2) & \dots & \dots & h(2,n) \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ h(n,1) & h(n,2) & \dots & \dots & h(n,n) \end{bmatrix}$$

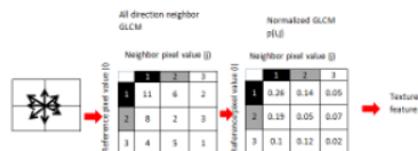
$$h(a,b) = \frac{\text{Number of times a pixel with value } a \text{ is adjacent to a pixel with value } b}{\text{Total number of such comparisons made}}$$



Four directions of adjacency as defined for calculation of the Haralick texture features

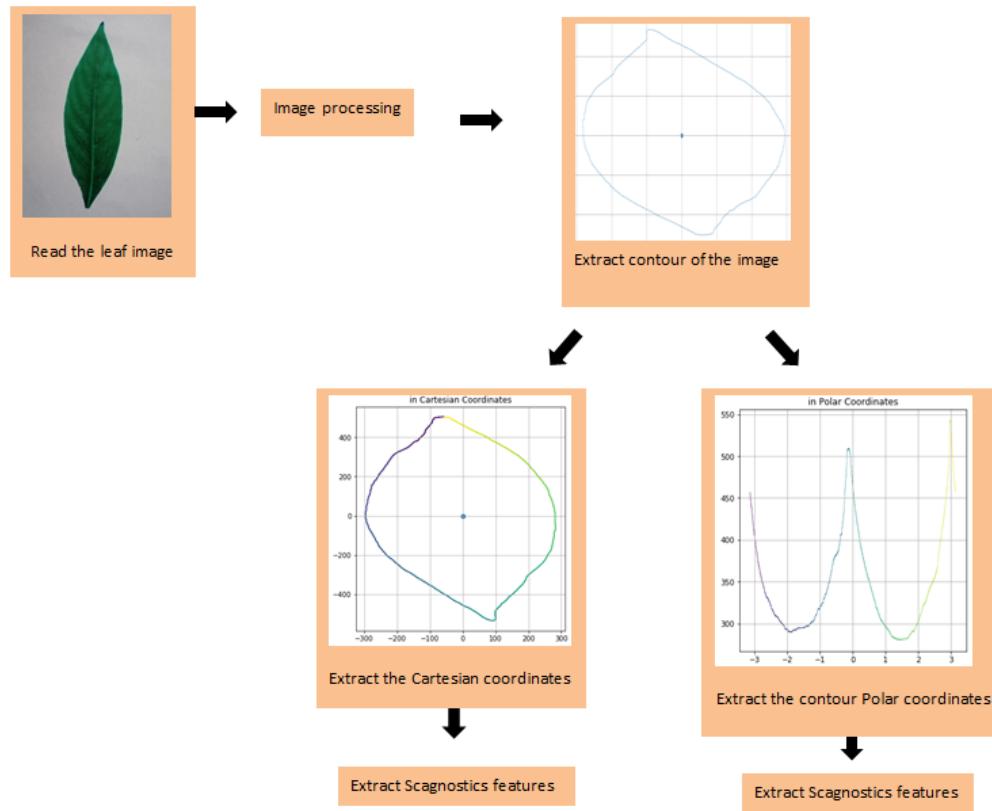


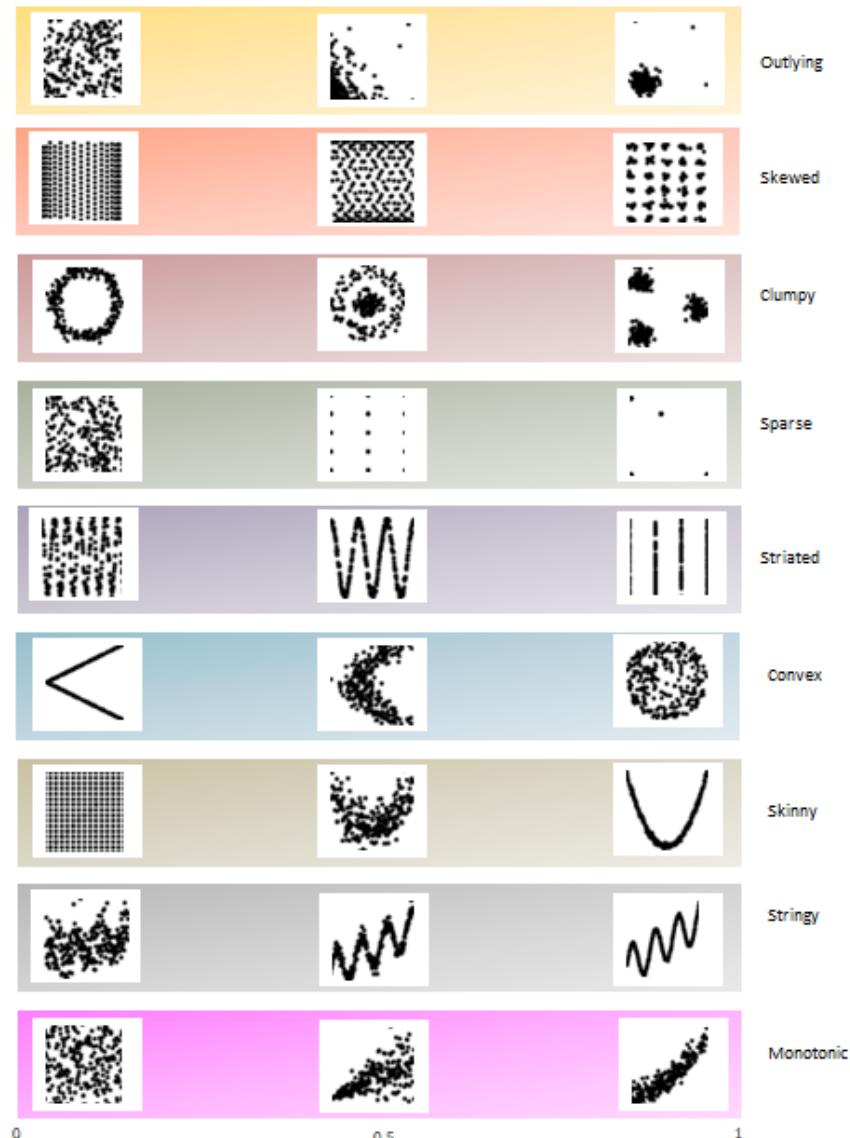
|7: Computing the Haralick texture features from a 4×4 example image step by step



Computing the Haralick texture features from a 4×4 example image with all direction

Scagnostic features

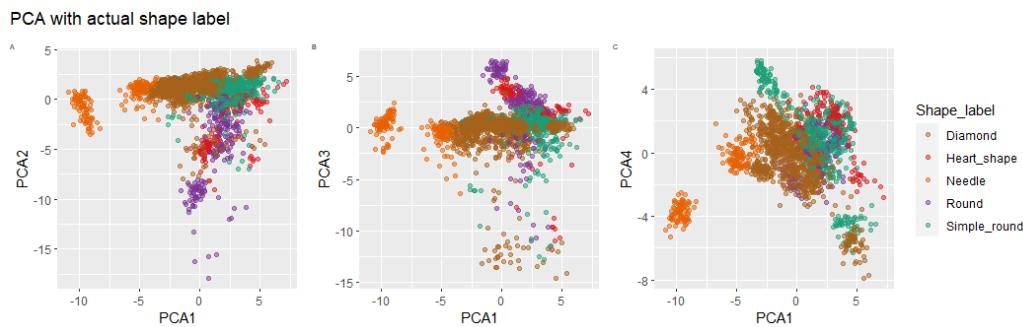




Visualization of Leaf Images in the Feature Space

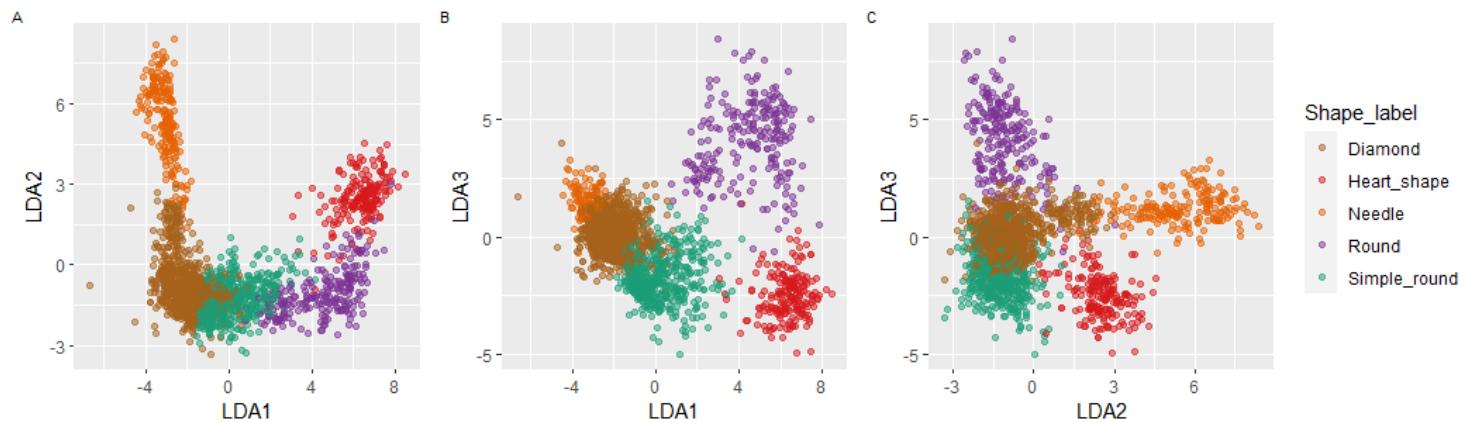
Example: Flavia

- LDA is a supervised dimensionality reduction technique, and PCA is unsupervised dimensionality reduction technique
- The first **three principal components (PCs)** accounting for approximately **83%** of the total variance in the original data

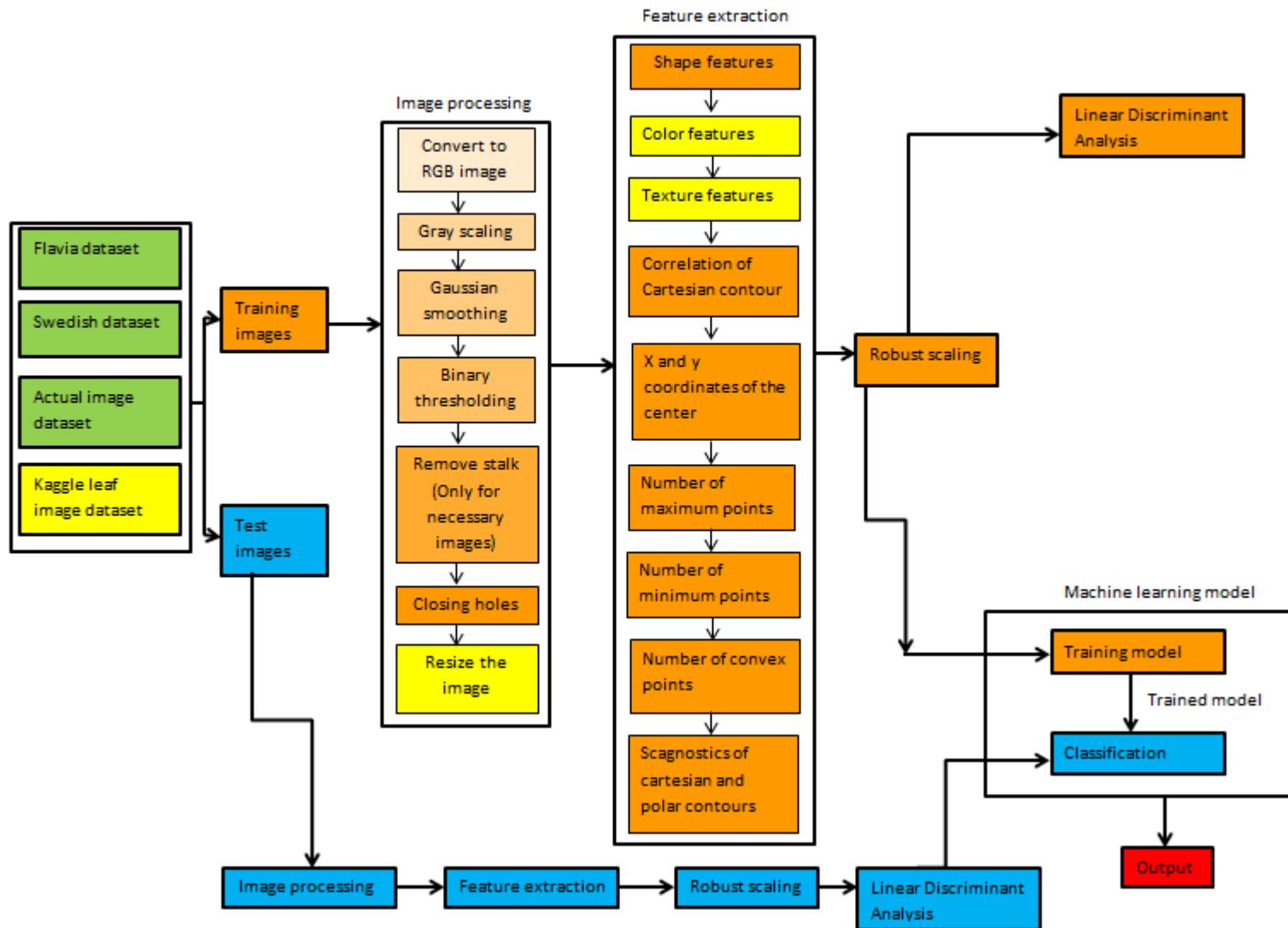


Let's see the 3D View

LDA with actual shape label



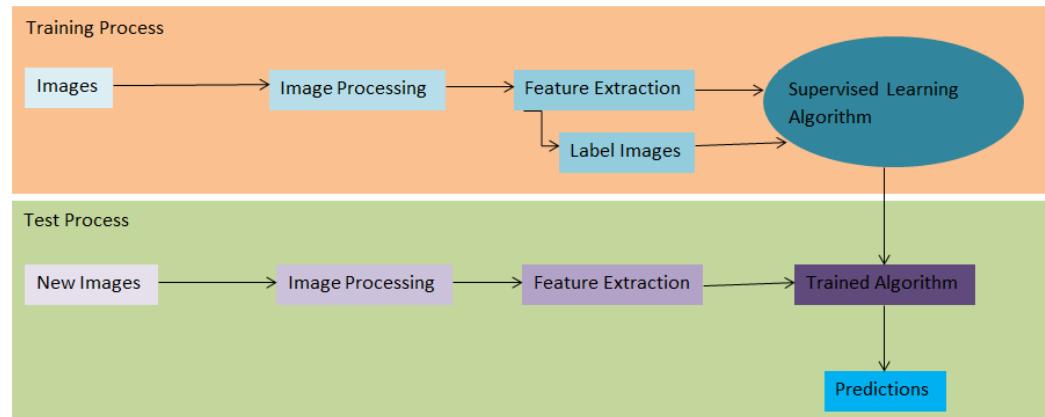
- Under both experimental settings class separation is **more clearly on the LDA space than the PCA space**. The reason could be LDA is a supervised learning algorithm while PCA space is an unsupervised learning algorithm.



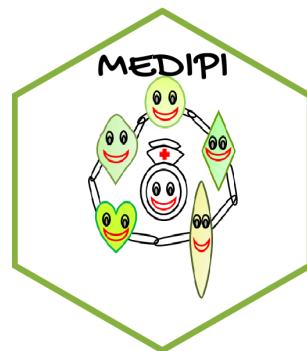
Algorithm Development



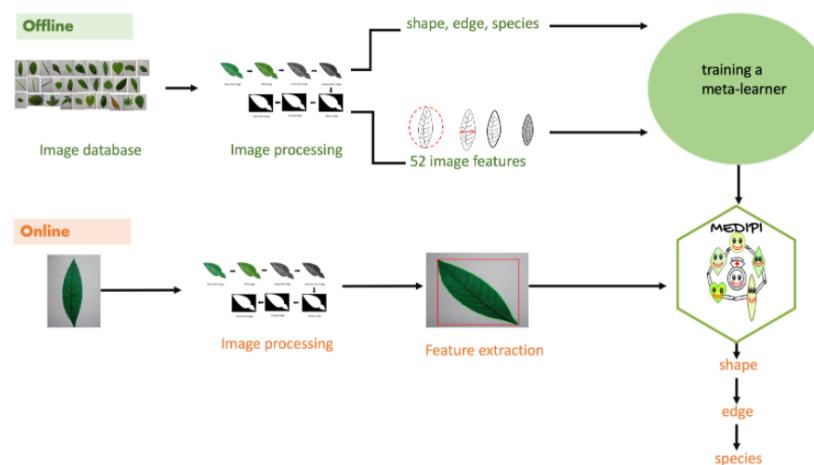
- Our medicinal plant classification algorithm contains **two process: Training process and Test process**.
- Our classification algorithm operates on the **features extracted from the image leaves**.
- The training process e of the algorithm contains four main steps: **i) Image processing, ii) Feature extraction, iii) Label images, and iv) Trained a algorithm.**
- In the test process, image processing and feature extraction steps are followed by the **new image before feed to the pre-trained model**.
- Mainly **Random Forest, Gradient Boosting and Extreme Gradient Boosting** classification algorithms are used in our research.



MEDIPI: MEDIcinal Plant Identification



Our medicinal plant classification algorithm is defined as **MEDIPI**.

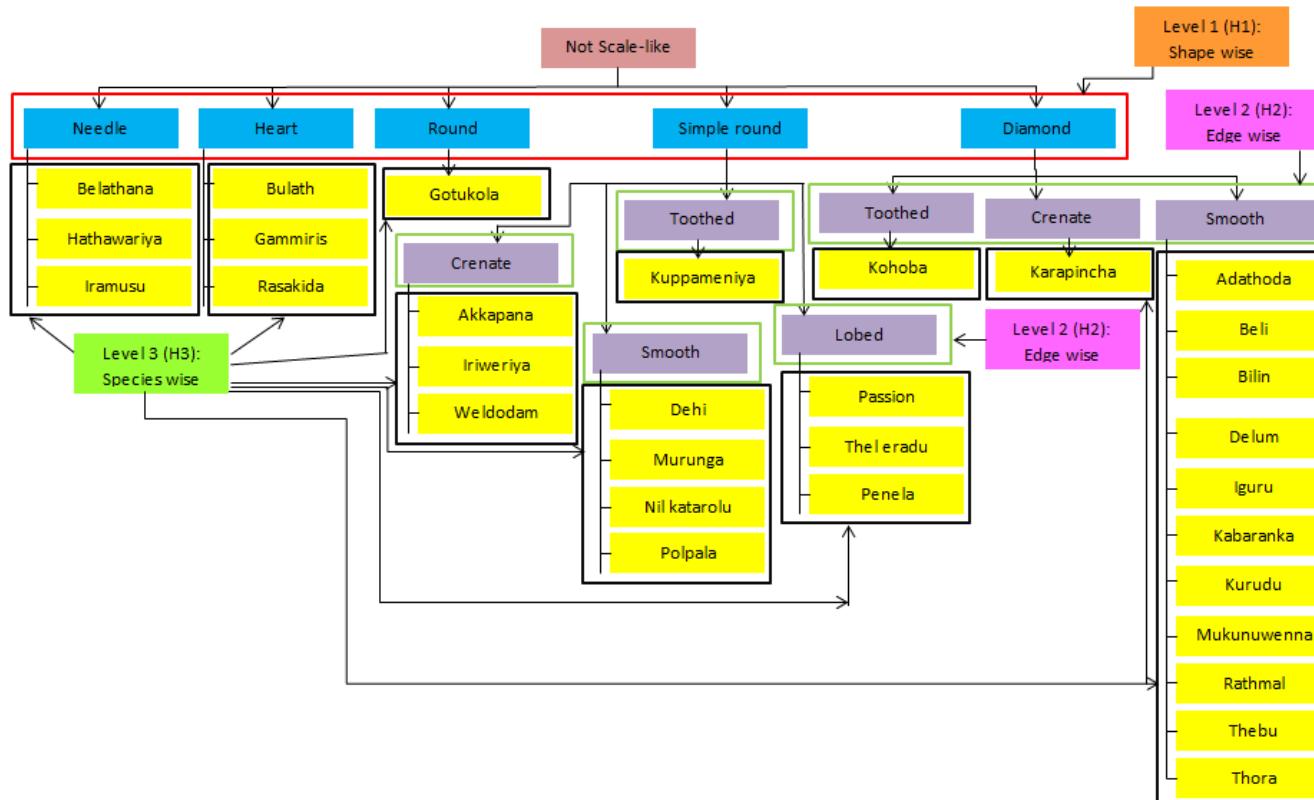


Discussion & Conclusions

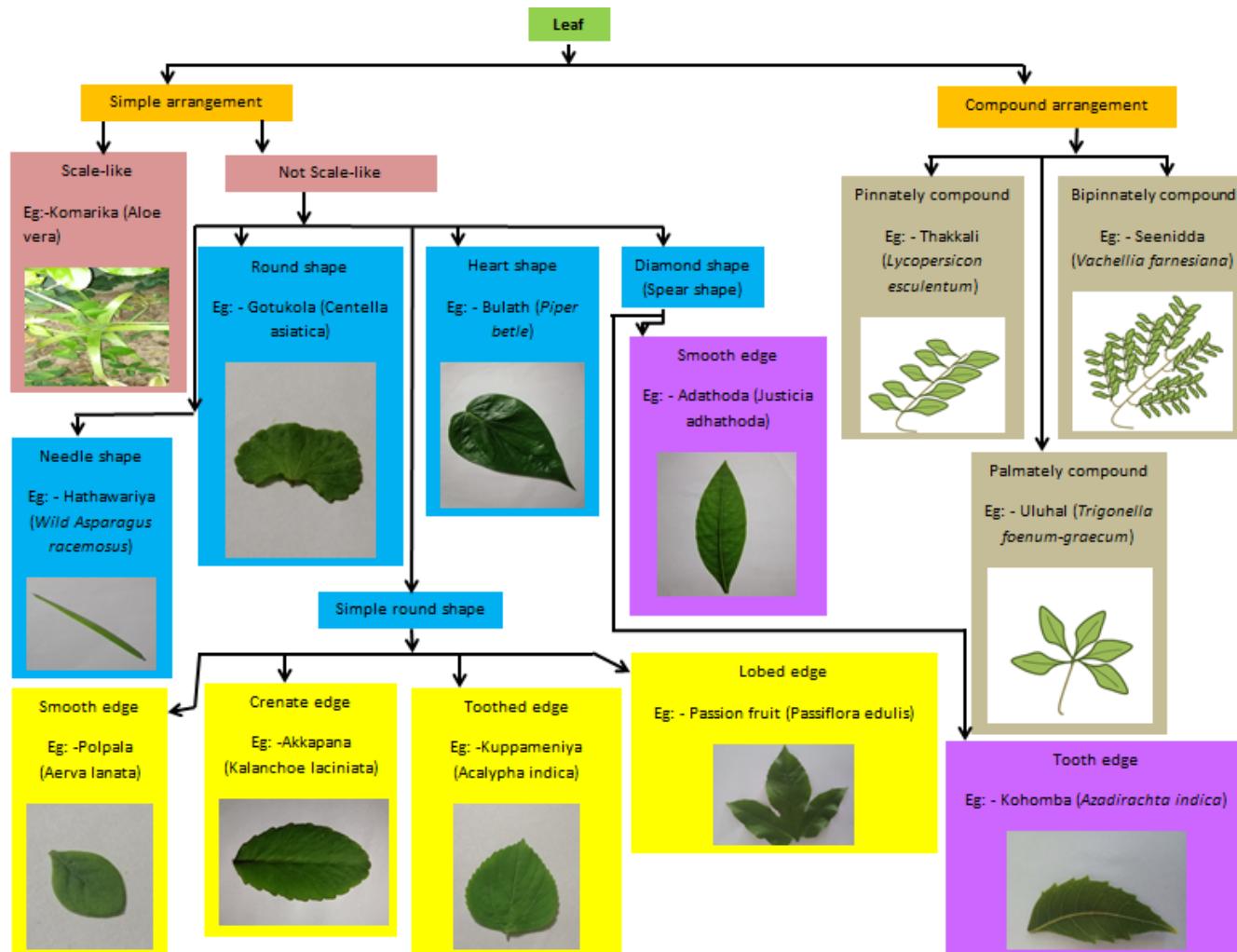


Hierarchical Approach

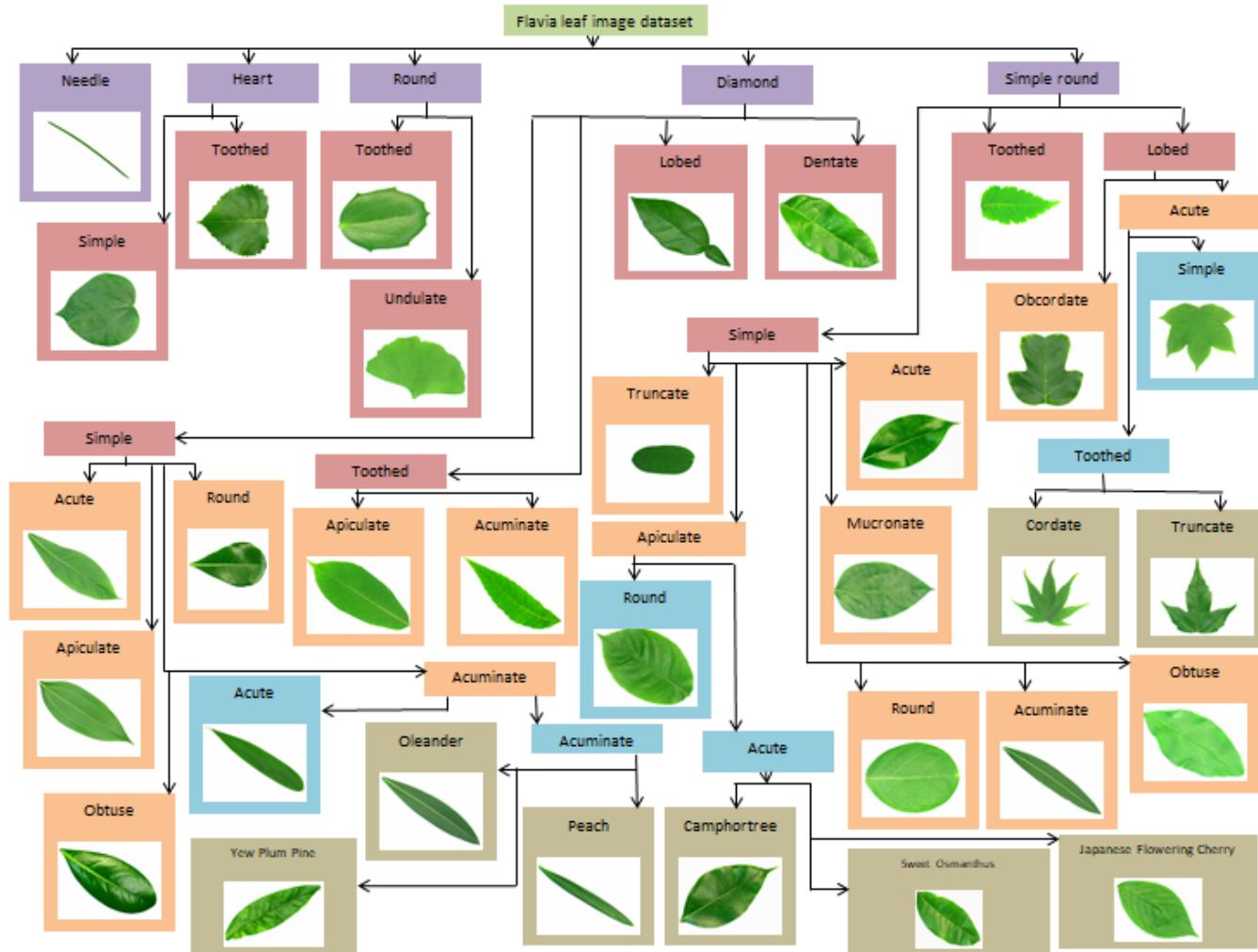
- Our algorithm works as a **hierarchical classification system**. The hierarchy contains **3 levels**. The first level classifies images according to the **shape**. The second level classifies according to the **edge types**. The bottom level classifies the **plant species**.



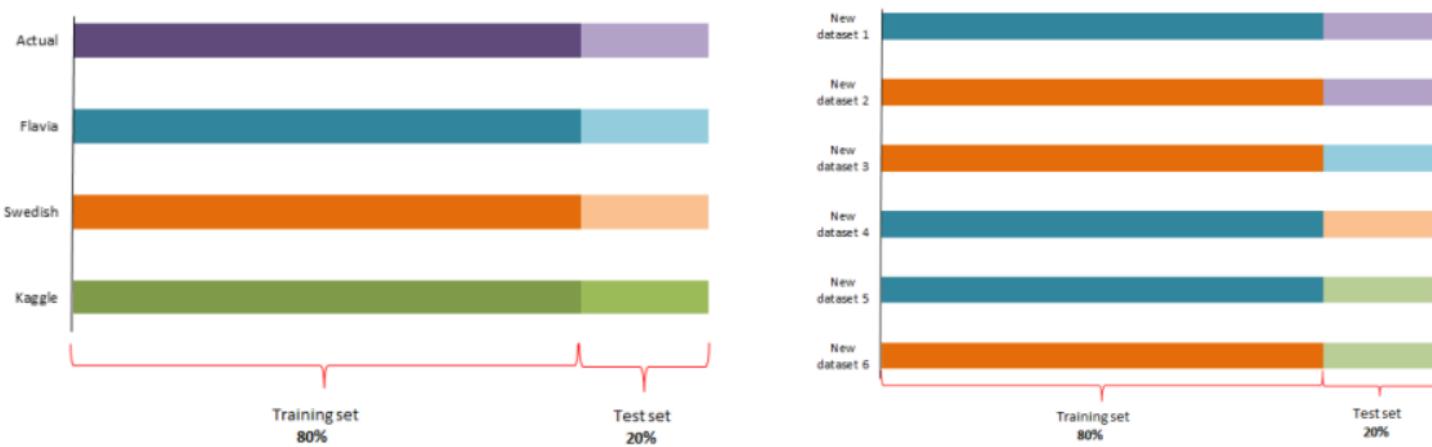
Hierarchy of Actual leaf image dataset



Hierarchy of Flavia leaf image dataset



Experiments



We have to use **training/test from same dataset** to get accurate results.

Compare results features of all categories and only with shape features

Training and test datasets from same dataset

| Dataset | Overall Classification Accuracy | | | | | | Dataset | Overall Classification Accuracy | | | | | | |
|--|---------------------------------|--------------|--------|-------------|-------|-------------------------|---------------------------|---------------------------------|----------|---------|--------------|--------|-------------|-------|
| | Diamond | Simple Round | Needle | Heart Shape | Round | Classification Accuracy | | Training Set | Test Set | Diamond | Simple Round | Needle | Heart shape | Round |
| Actual | | | | | | | | | | | | | | |
| Random Forest | 0.94 | 1.00 | 1.00 | 1.00 | 1.00 | 0.97 | Random Forest | 0.89 | 0.86 | 1.00 | 1.00 | 0.86 | 0.90 | |
| Extreme Gradient Boosting | 0.94 | 0.95 | 1.00 | 0.96 | 1.00 | 0.95 | Extreme Gradient Boosting | 0.92 | 0.89 | 0.97 | 0.97 | 1.00 | 0.92 | |
| Gradient Boosting | 0.96 | 0.96 | 1.00 | 1.00 | 1.00 | 0.97 | Gradient Boosting | 0.93 | 0.83 | 1.00 | 0.92 | 0.71 | 0.90 | |
| Flavia | | | | | | | | | | | | | | |
| Random Forest | 0.98 | 0.97 | 0.96 | 0.97 | 1.00 | 0.98 | Random Forest | 0.93 | 1.00 | 0.91 | 0.97 | 1.00 | 0.96 | |
| Extreme Gradient Boosting | 0.99 | 0.95 | 0.93 | 0.98 | 1.00 | 0.97 | Extreme Gradient Boosting | 0.96 | 0.96 | 1.00 | 0.96 | 0.98 | 0.96 | |
| Gradient Boosting | 0.98 | 0.96 | 1.00 | 1.00 | 1.00 | 0.98 | Gradient Boosting | 0.94 | 0.98 | 0.95 | 1.00 | 1.00 | 0.96 | |
| Swedish | | | | | | | | | | | | | | |
| Random Forest | 0.99 | 1.00 | 1.00 | 0.95 | 1.00 | 0.98 | Random Forest | 0.96 | 1.00 | 1.00 | 0.85 | 0.94 | 0.95 | |
| Extreme Gradient Boosting | 1.00 | 1.00 | 1.00 | 0.88 | 0.92 | 0.97 | Extreme Gradient Boosting | 0.99 | 1.00 | 1.00 | 0.82 | 1.00 | 0.97 | |
| Gradient Boosting | 0.99 | 0.98 | 0.96 | 0.86 | 1.00 | 0.96 | Gradient Boosting | 0.95 | 0.98 | 1.00 | 1.00 | 0.92 | 0.96 | |
| Kaggle | | | | | | | | | | | | | | |
| Random Forest | 0.76 | 0.69 | - | 0.82 | 0.77 | 0.74 | Random Forest | 0.96 | 1.00 | 1.00 | 0.85 | 0.94 | 0.95 | |
| Extreme Gradient Boosting | 0.75 | 0.68 | - | 0.70 | 0.70 | 0.71 | Extreme Gradient Boosting | 0.99 | 1.00 | 1.00 | 0.82 | 1.00 | 0.97 | |
| Gradient Boosting | 0.74 | 0.68 | - | 0.92 | 0.66 | 0.71 | Gradient Boosting | 0.95 | 0.98 | 1.00 | 1.00 | 0.92 | 0.96 | |
| Shape-wise and overall classification accuracy (Training and test sets are from the same dataset) | | | | | | | | | | | | | | |
| Shape-wise and overall classification accuracy (Training and test sets are from the same datasets: Only with shape features) | | | | | | | | | | | | | | |

Compare results features of all categories and only with shape features

Training and test datasets from different datasets

| | | Dataset | | | | | |
|--------------|---------------------------|---------|--------------|--------|-------------|-------|---------------------------------|
| Training Set | Test Set | Diamond | Simple Round | Needle | Heart shape | Round | Overall Classification Accuracy |
| Flavia | Actual | | | | | | |
| | Random Forest | 0.70 | 0.28 | 0.23 | 0.04 | 0.10 | 0.41 |
| | Extreme Gradient Boosting | 0.65 | 0.28 | 0.56 | 0.03 | 0.29 | 0.43 |
| | Gradient Boosting | 0.63 | 0.31 | 0.45 | 0.03 | 0.26 | 0.42 |
| Swedish | Actual | | | | | | |
| | Random Forest | 0.38 | 0.22 | 0.06 | 0.12 | 0.39 | 0.26 |
| | Extreme Gradient Boosting | 0.40 | 0.23 | 0.18 | 0.04 | 0.23 | 0.25 |
| | Gradient Boosting | 0.50 | 0.33 | 0.11 | 0.12 | 0.55 | 0.36 |
| Swedish | Flavia | | | | | | |
| | Random Forest | 0.29 | 0.43 | 0.46 | 0.0 | 0.0 | 0.27 |
| | Extreme Gradient Boosting | 0.42 | 0.34 | 0.40 | 0.01 | 0.01 | 0.31 |
| | Gradient Boosting | 0.40 | 0.37 | 0.53 | 0.12 | 0.27 | 0.37 |
| Flavia | Swedish | | | | | | |
| | Random Forest | 0.57 | 0.72 | 0.03 | 0.0 | 0.01 | 0.39 |
| | Extreme Gradient Boosting | 0.58 | 0.67 | 0.13 | 0.0 | 0.15 | 0.41 |
| | Gradient Boosting | 0.51 | 0.79 | 0.12 | 0.0 | 0.18 | 0.41 |
| Flavia | Kaggle | | | | | | |
| | Random Forest | 0.66 | 0.39 | 0.0 | 0.0 | 0.17 | 0.44 |
| | Extreme Gradient Boosting | 0.56 | 0.48 | 0.0 | 0.0 | 0.21 | 0.43 |
| | Gradient Boosting | 0.51 | 0.47 | 0.0 | 0.0 | 0.25 | 0.42 |
| Swedish | Kaggle | | | | | | |
| | Random Forest | 0.42 | 1.00 | 0.0 | 0.07 | 0.06 | 0.24 |
| | Extreme Gradient Boosting | 0.35 | 0.12 | 0.0 | 0.10 | 0.14 | 0.23 |
| | Gradient Boosting | 0.44 | 0.14 | - | 0.14 | 0.13 | 0.27 |

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

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

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

Compare algorithms

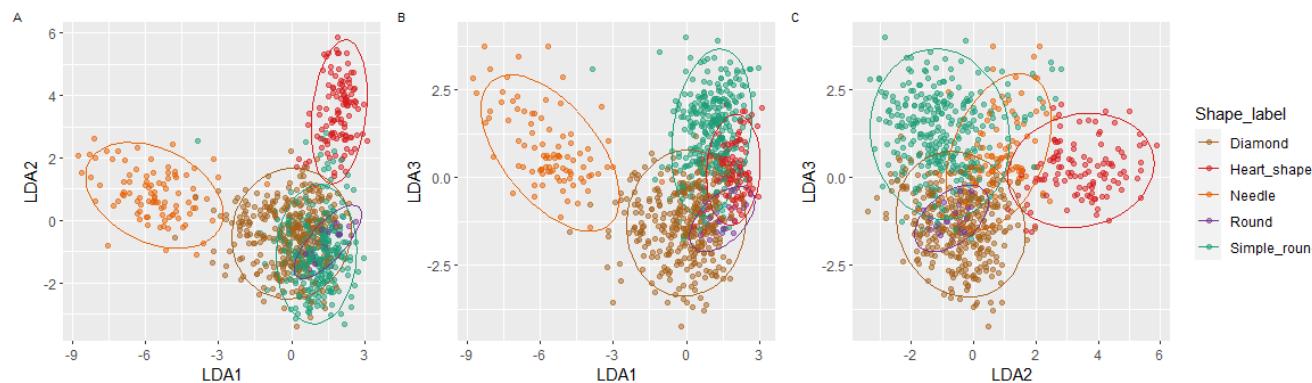
| Case | Algorithm | Overall classification | | Macro average | | Weighted average | | Average of ranks |
|--|------------------------------|------------------------|------|---------------|------|------------------|------|---------------------|
| | | Accuracy | Rank | Accuracy | Rank | Accuracy | Rank | |
| Using hierarchical level | Random forest | 1.00 | 1 | 0.99 | 1 | 1.00 | 1 | 1 |
| | 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 |
| | Random forest | 0.99 | 2 | 0.99 | 1 | 0.99 | 2 | 1.67 |
| Without using hierarchical level | Extreme gradient boosting | 0.98 | 5 | 0.98 | 5 | 0.98 | 5 | 5 |
| | 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 |

The model trained with **Random Forest** algorithm provides the highest accuracy.

Linear Discriminant Analysis

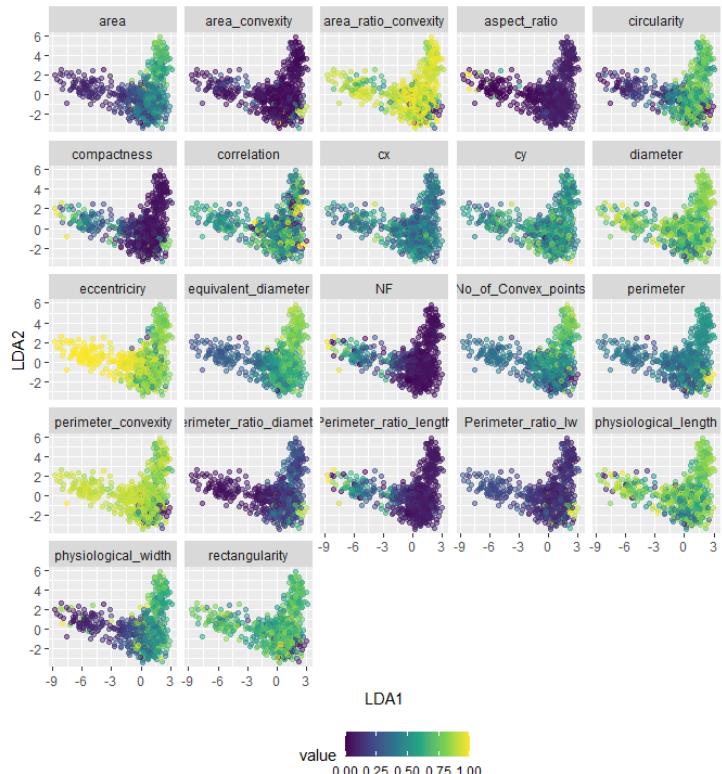
- High dimensional visualization approach
- To visualize what is **happening inside** the trained algorithm and provides **transparency** to our black-box model

LDA with Actual shape label

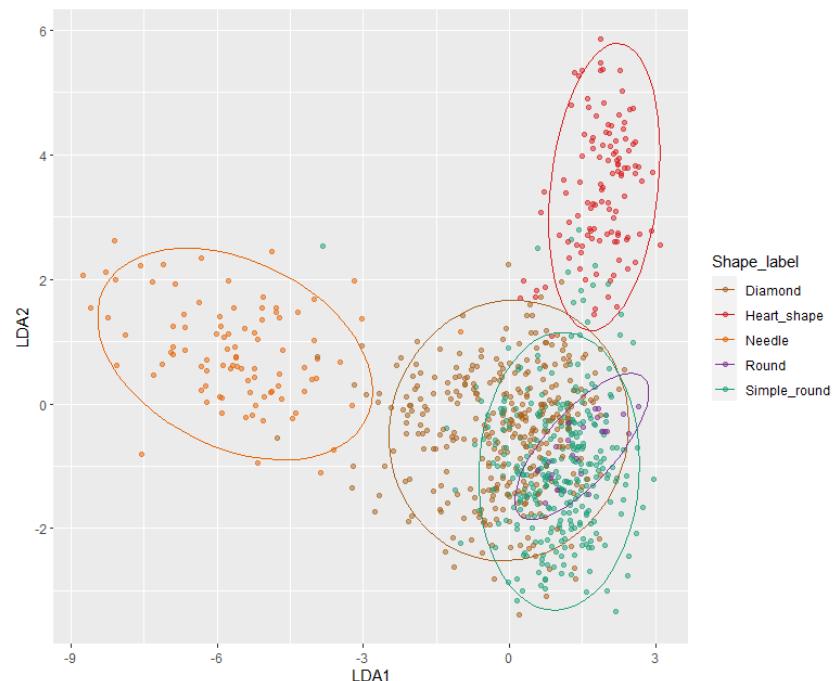


A

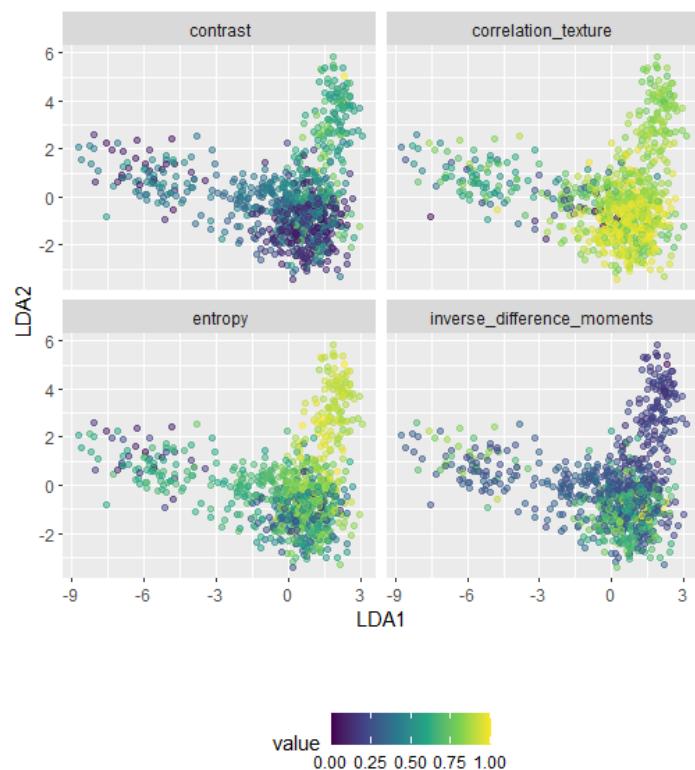
Distribution of Shape features

**B**

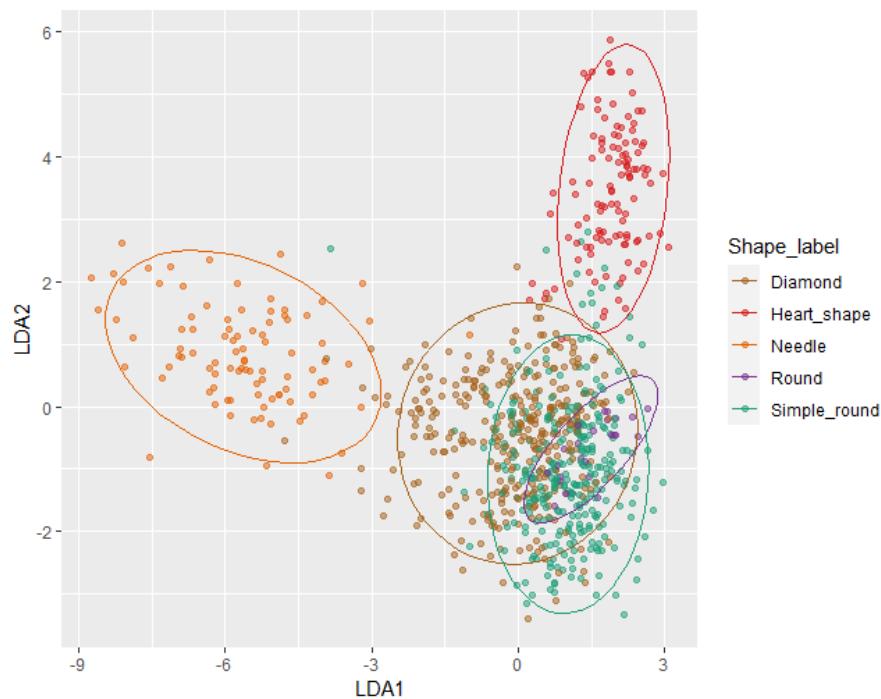
LDA1 Vs LDA2 by Actual Shape Label



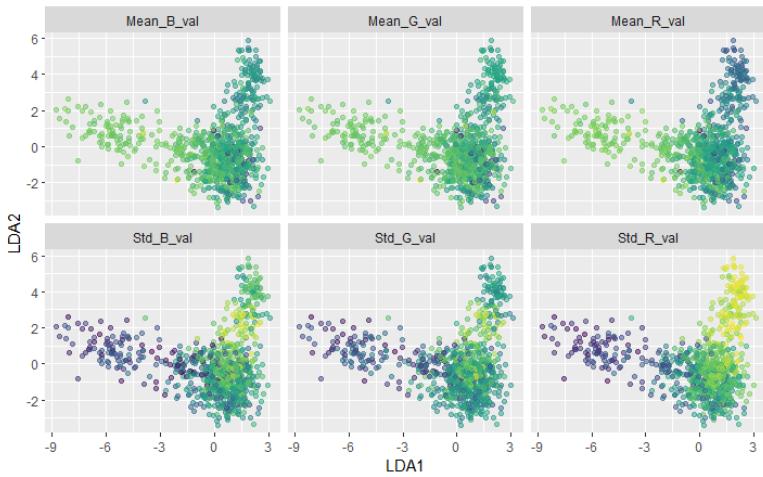
A Distribution of Texture features



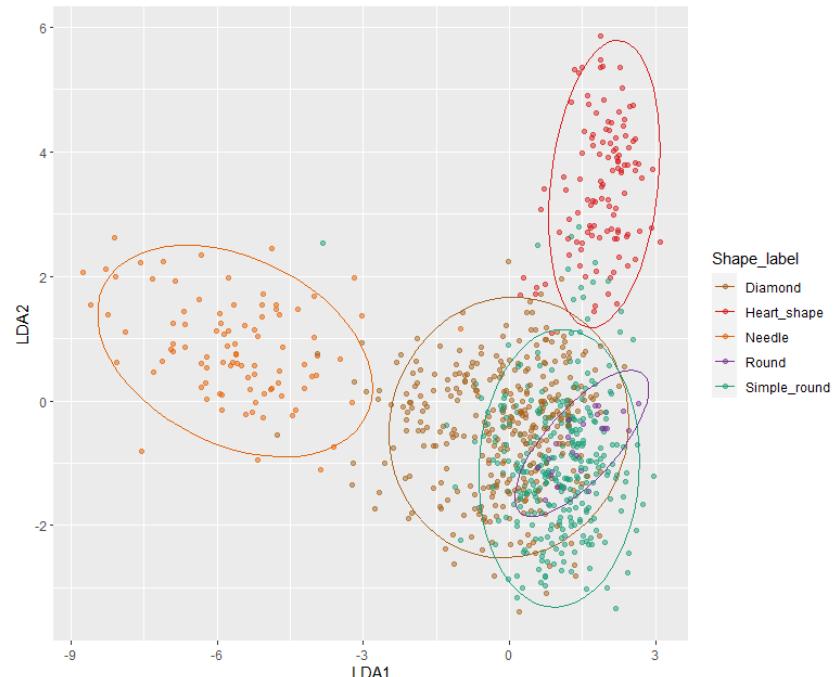
B LDA1 Vs LDA2 by Actual Shape Label



A Distribution of Color features



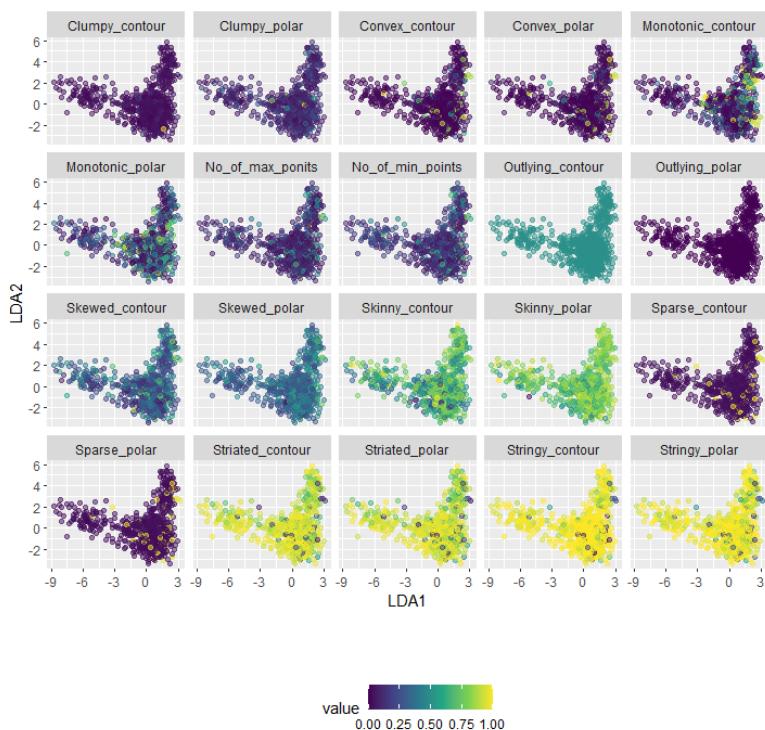
B LDA1 Vs LDA2 by Actual Shape Label



value
0.00 0.25 0.50 0.75 1.00

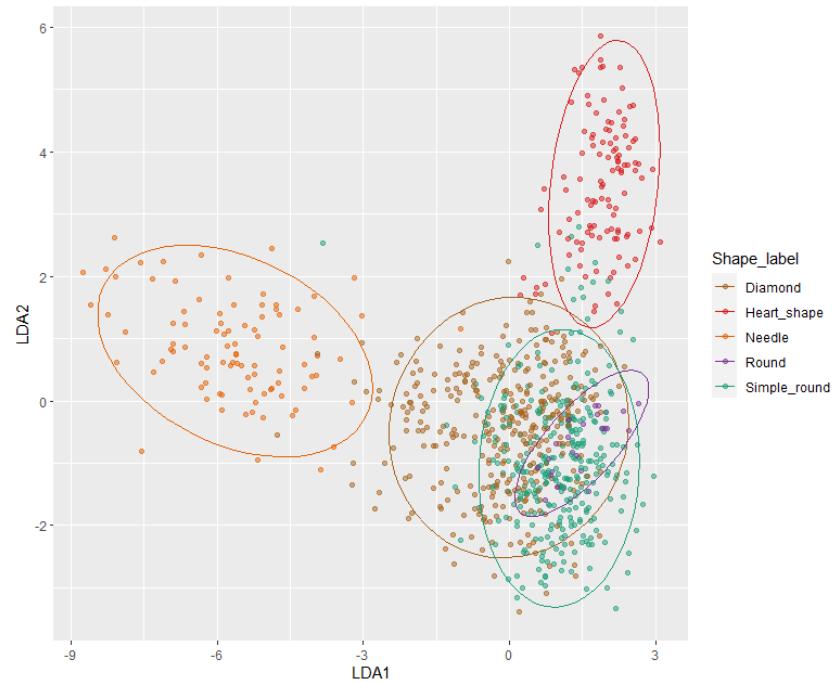
A

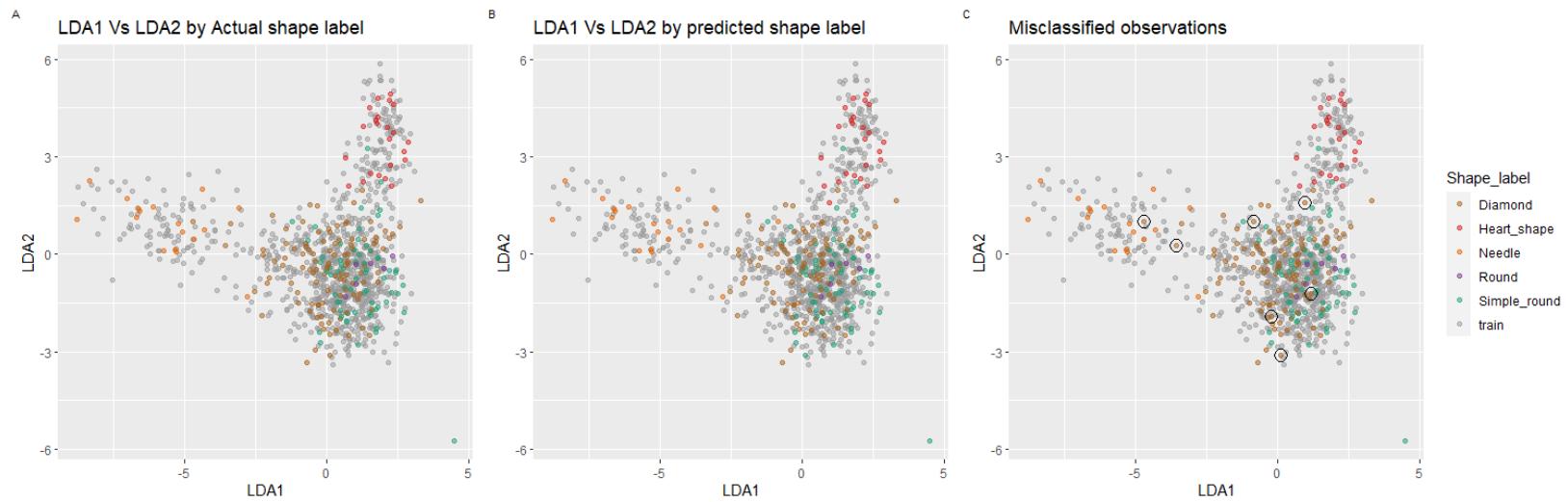
Distribution of Scagnostic features



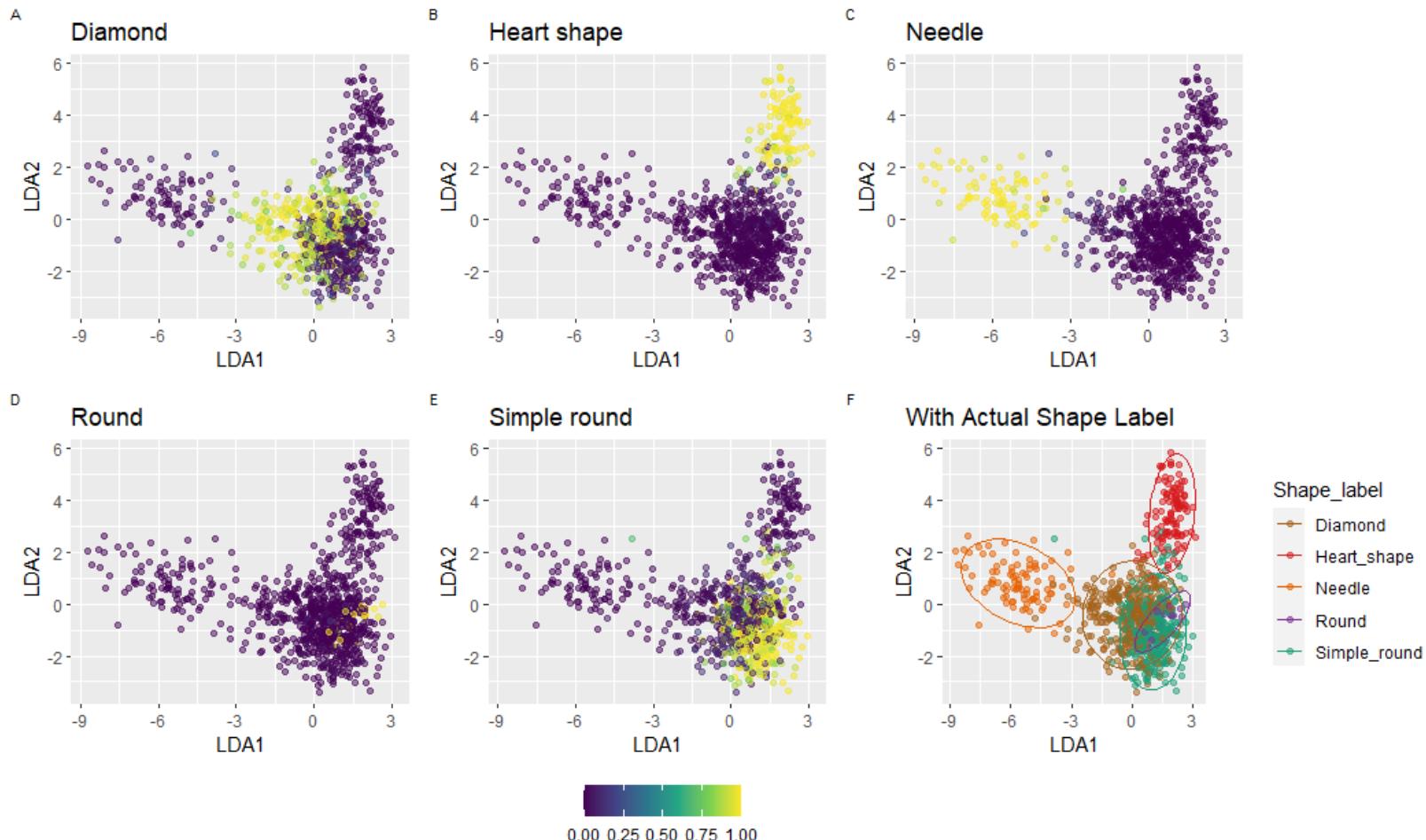
B

LDA1 Vs LDA2 by Actual Shape Label





OOB probability distribution by shape label



Conclusions

- The model trained with **random forest** algorithm provides the highest accuracy.
- Our algorithm works as a **hierarchical classification system**.
- 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, (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, (vii) Skinny contour, and (vii) Skinny contour are more important in identifying leaf species in the **bottom level** of the hierarchy.
- 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.
- We observe that **shape feature is not sufficient** to classify leaf images.

Thesis Outcome



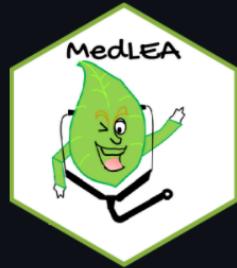
MedLEA

☰ README.md

MedLEA

CRAN 1.0.1 downloads 234/month

The MedLEA package provides morphological and structural features of 471 medicinal plant leaves and 1099 leaf images of 31 species and 29-45 images per species.



Installation

You could install the stable version on CRAN:

```
install.packages("MedLEA")
```

You can install the development version from GitHub with:

```
# install.packages("devtools")
devtools::install_github("SMART-Research/MedLEA")
```

<https://CRAN.R-project.org/package=MedLEA>

Research paper

Cornell University We gratefully acknowledge support from the Simons Foundation and member institutions.

arXiv.org > cs > arXiv:2106.08077 Search... All fields Search Help | Advanced Search

Computer Science > Computer Vision and Pattern Recognition

(Submitted on 15 Jun 2021 (v1), last revised 24 Aug 2021 (this version, v3))

Computer-aided Interpretable Features for Leaf Image Classification

Jayani P. G. Lakshika, Thiyanga S. Talagala

Plant species identification is time consuming, costly, and requires lots of efforts, and expertise knowledge. In recent, many researchers use deep learning methods to classify plants directly using plant images. While deep learning models have achieved a great success, the lack of interpretability limit their widespread application. To overcome this, we explore the use of interpretable, measurable and computer-aided features extracted from plant leaf images. Image processing is one of the most challenging, and crucial steps in feature-extraction. The purpose of image processing is to improve the leaf image by removing undesired distortion. The main image processing steps of our algorithm involves: i) Convert original image to RGB (Red-Green-Blue) image, ii) Gray scaling, iii) Gaussian smoothing, iv) Binary thresholding, v) Remove stalk, vi) Closing holes, and vii) Resize image. The next step after image processing is to extract features from plant leaf images. We introduced 52 computationally efficient features to classify plant species. These features are mainly classified into four groups as: i) shape-based features, ii) color-based features, iii) texture-based features, and iv) scagnostic features. Length, width, area, texture correlation, monotonicity and scagnostics are to name few of them. We explore the ability of features to discriminate the classes of interest under supervised learning and unsupervised learning settings. For that, supervised dimensionality reduction technique, Linear Discriminant Analysis (LDA), and unsupervised dimensionality reduction technique, Principal Component Analysis (PCA) are used to convert and visualize the images from digital-image space to feature space. The results show that the features are sufficient to discriminate the classes of interest under both supervised and unsupervised learning settings.

Comments: 31 pages
Subjects: Computer Vision and Pattern Recognition (cs.CV); Applications (stat.AP)
ACM classes: E.4; I.5.4
Cite as: arXiv:2106.08077 [cs.CV]
(or arXiv:2106.08077v3 [cs.CV] for this version)

Download:

- PDF
- Other formats

(CC BY)

Current browse context: cs.CV
< prev | next >
new | recent | 2106
Change to browse by:
cs
stat
stat.AP

References & Citations

- NASA ADS
- Google Scholar
- Semantic Scholar

DBLP - CS Bibliography
listing | bibtex

Export BibTeX Citation

Bookmark

<https://arxiv.org/abs/2106.08077>

Web Application for Leaf Image Identification

MEDIPI: Needle
Piyadi G. J. Lakshika

Upload the file
Browse... Dataset.csv Upload complete

Select the parameters below
 Header

Separator
 Comma
 Semicolon
 Tab
 Space

Select the image Id
Select
380

| Dataset | | Details about the Image | | | | | | | | |
|---------|--|-------------------------|--------------|--------------|--------------|----------------|--------------|--------------|---------------|-----------------|
| | | Outlying_polar | Skewed_polar | Clumpy_polar | Sparse_polar | Striated_polar | Convex_polar | Skinny_polar | Stringy_polar | Monotonic_polar |
| 0 | | -0.26 | -0.21 | 0.28 | 0.52 | -0.62 | 1.33 | 0.84 | -0.39 | |
| 0 | | -0.17 | -0.09 | 0.24 | 0.51 | -0.61 | 0.74 | 0.84 | -0.89 | |
| 0 | | 0.36 | -0.24 | -1.15 | -0.71 | 0.22 | 0.06 | -0.07 | -0.76 | |
| 0 | | 0.05 | -0.07 | -2.01 | -0.73 | 0.49 | -0.03 | -0.04 | -0.83 | |
| 0 | | 0.00 | -0.74 | 1.44 | 0.52 | -0.62 | 1.33 | 0.84 | -0.21 | |
| 0 | | -0.61 | 2.88 | -0.31 | -0.19 | 0.35 | -0.31 | -0.15 | 1.58 | |
| 0 | | 0.81 | -0.58 | 0.43 | 0.52 | -0.60 | -0.33 | 0.84 | -0.80 | |
| 0 | | -1.56 | 0.38 | -1.97 | 0.52 | -0.62 | 1.33 | 0.84 | -0.56 | |
| 0 | | 0.30 | 0.08 | 1.92 | 0.36 | -0.57 | -0.01 | 0.84 | -0.78 | |
| 0 | | 0.37 | -1.05 | 0.99 | 0.26 | -0.57 | 0.38 | 0.84 | -0.91 | |
| 0 | | 0.52 | -0.20 | 1.07 | 0.37 | -0.27 | -1.25 | 0.84 | -0.67 | |
| 0 | | -0.47 | -0.39 | -0.56 | -0.12 | 1.92 | -0.61 | -0.16 | 2.78 | |
| 0 | | 0.06 | -0.36 | -1.08 | 0.18 | -0.52 | -0.19 | 0.84 | -0.89 | |
| 0 | | 0.75 | -0.45 | 0.83 | 0.30 | 0.97 | -0.53 | 0.84 | -0.91 | |
| 0 | | 0.34 | 0.78 | 1.90 | 0.37 | -0.19 | -1.15 | 0.84 | 2.68 | |
| 0 | | -0.55 | 0.02 | 0.53 | 0.20 | -0.27 | -0.39 | 0.84 | -0.71 | |

MEDIPI: Needle
Piyadi G. J. Lakshika

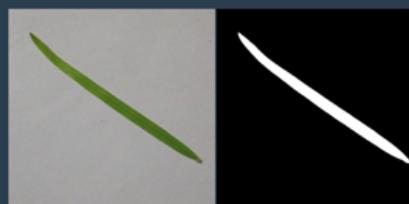
Upload the file
Browse... Dataset.csv Upload complete

Select the parameters below
 Header

Separator
 Comma
 Semicolon
 Tab
 Space

Select the image Id
Select
380

| Dataset | | Details about the Image | | | | | | | | |
|---------|--|-------------------------|--------------|--------------|--------------|----------------|--------------|--------------|---------------|-----------------|
| | | Outlying_polar | Skewed_polar | Clumpy_polar | Sparse_polar | Striated_polar | Convex_polar | Skinny_polar | Stringy_polar | Monotonic_polar |
| 0 | | 0 | 1.61 | 1.87 | 3.77 | -1.08 | 0.93 | 0.05 | 0.64 | 0.47 |



| Species | Predicted_species_label |
|------------|-------------------------|
| Hathawalya | Hathawalya |

OCTAVE

ADVANCED ANALYTICS SYMPOSIUM

FINALIST

Statistical Machine Learning for Medicinal Plant Leaves
Classification.



Jayani Lakshika
BSc in Statistics
University of Sri Jayawardenapura



9.00 AM - 1.00 PM | 30th April 2021

CLICK THE LINK IN THE CAPTION AND REGISTER TO WATCH THE WEBINAR



OCTAVE



keells.com/octave/



octave@keells.com

Applied Statistics Conference 2021 (Solvenia)

Statistical Machine Learning for Medicinal Plant Leaves Classification

Jayani P.G. Lakshika¹, Thiyantha S. Talagala²

¹ Department of Statistics, University of Sri Jayewardenepura, Nugegoda, Sri Lanka

Introduction

Medicinal plant species identification is costly, time-consuming, and requires lots of effort and expert knowledge. Recently, many researchers use deep learning models like CNN (Convolutional Neural Network) to classify plant directly using plant images. Even though deep learning models have achieved great success, their interpretability, and transparency of the deep learning models are limited. Also, deep learning models require a large memory space and become computationally prohibitive. To conquer this, we introduce an image feature-based statistical machine learning algorithm.

- Main Objective:** Develop an automatic algorithm to classify medicinal plants by using statistical machine learning approach.

Figure 1: Taking a photograph of a leaf.

- Significance of the study:** To avoid misidentifying medicinal plants.
- Why leaf images?** Leaf images are considered as they contain large number of diverse set of features.

Figure 2: Shape-based features.

- Our classification algorithm operates on the features extracted from the image leaves.

Figure 3: (a) Leaf shapes (b) Leaf margin

Methodology

Figure 4: Overview of algorithm.

- Offline phase contains 4 main steps: i) Image processing, ii) Feature extraction, iii) Label images, and iv) Train the machine learning model.
- We trained our algorithm using random forest, gradient boosting, and extreme gradient boosting.

Figure 5: Image processing

- Online phase, the pre-trained classification model is used for real-time leaf image classification.

Data

- Five leaf datasets of different countries: (i) Flavia 1907 images collected from China, (ii) Swedish 973 images collected from Sweden, and (iii) Kaggle 1584 images collected from United Kingdom (UK), and online phase.

- (iv) MedLEA 1099 images collected from Sri Lanka.

Figure 6: Datasets.

Visualization

Figure 7: Distribution of Swedish leaf images on the projection space based on principal component analysis. All projected points are colored according to their shape labels.

We explore the ability of features to discriminate the classes of interest using PCA and LDA. Under both experimental settings, clear separation of classes are visible in their projection spaces (see 3D view).

MedLEA

The MedLEA, Medicinal LEAF package is an open-source R software (<https://github.com/jayani-lakshika/MedLEA>) for research reproducibility. This repository provides morphological and structural features of 471 medicinal plant leaves and 1099 leaf images of 31 species and 29-45 images per species in Sri Lanka.

MEDIPI

Our medicinal plant classification algorithm is defined as MEDIPI: MEDical Plant Identification (<https://github.com/Jayani-Lakshika/MEDIPI>). The MEDIPI is divided into the offline phase and an online phase.

Features

We introduced 52 computationally efficient interpretable features that are useful in leaf classification (Lakshika and Talagala 2021).

Figure 8: Features.

Conclusions

- We introduced 52 computationally efficient interpretable features to classify plant species. These features are mainly classified in to four groups: (i) shape, (ii) color, (iii) texture, and (iv) scagmatics.
- The model trained with random forest algorithm provides the highest accuracy.
- Our algorithm works as a hierarchical classification system which classifies according shape, edge type (see Figure 2), and plant species. We observed that shape features are more important when classify the leaf images at the first level of hierarchy. Scagmatic features are more important identifying leaf species in the bottom level of hierarchy.
- The MEDIPI algorithm yields accurate results the state-of-the-existing techniques in the field.
- We used high dimensional visualization approach Linear Discriminant Analysis (LDA) to visualize what is happening inside the trained algorithm as it provides transparency to our black-box model.

References

Lakshika, Jayani P. G., and Thiyantha S. Talagala. 2020. "Computer-Aided Interpretable Features for Leaf Identification." <http://arxiv.org/abs/2106.08077>.

Young Scientists' Conference on Multidisciplinary Research (YSCMR 2021)

Organised by the Young Scientists' Association of the
National Institute of Fundamental Studies, Sri Lanka
(NIFS-YSA)

RESIDENTIAL ****
Young Scientists' Conference Multidisciplinary Research (YSCMR-2021)
Young Scientists' Association, National Institute of Fundamental Studies, Sri Lanka
27th October 2021

 YSA

Statistical machine learning for medicinal plant leaves classification

Jayani P.G. Lakshmi^a, Thiyanga S. Talapola^{a*}
^aUniversity of Sri Jayewardenepura, Nugegoda, Sri Lanka
jayani.lakshmi@gmail.com

Background: Medicinal plant identification involves years of experience through sensory and olfactory senses, laboratory-based testing, and automatic ways useful for those lacking knowledge.

Objectives: The main objective is to introduce an automatic approach based on statistical machine learning to classify medicinal plants using leaves.

Methods: MEDIPF is our medicinal plant classification algorithm which is divided into offline and online phase. The classification algorithm is trained in the offline phase, whereas the pre-trained classification model is used for real-time leaf image classification for general users in the online phase. Features extracted from the leaf images (Medf.EA, Flavia Swedish, and Kagggle) are used to operate our classification algorithm. First, leaf images are followed by a series of operations including cropping, image enhancement, image processing steps consist of converting the original image to an RGB image, gray-value conversion, applying binary thresholding, removing the stalk, closing hole, and resizing the image used to remove undesired distortion. Extracting features from plant leaf images is the second stage. Next, we trained our algorithm using random forest, gradient boosting, and support vector machines. We validate our algorithm by approaching it by testing (80%) and training (20%) from the same dataset and training (80%) and test (20%) from different datasets. Furthermore, our algorithm works as a hierarchical classification system that contains three levels. The level classifies images according to the shape, edge types, and plant species accordingly. We used high dimensional visualization approaches to visualize what is happening inside the trained algorithm and provides transparency to our black-box model. We established an open-source repository in R software, Medf.EA.

Results and Discussion: We introduced 22 computationally efficient features to classify plant species which are mainly classified into four groups as shape, edge types, and topogonics. Length, area are shape features, whereas monotonicity is one of topogonics features. The model trained with random forest algorithm provides the accuracy with 100% while gradient boosting, and extreme gradient boosting have 99%.

Conclusion: The MEDIPF algorithm yields accurate results to the state-of-the-existing techniques in the field for medicinal plants classification.

Thanks!

Slides created via the R package [xaringan](#).

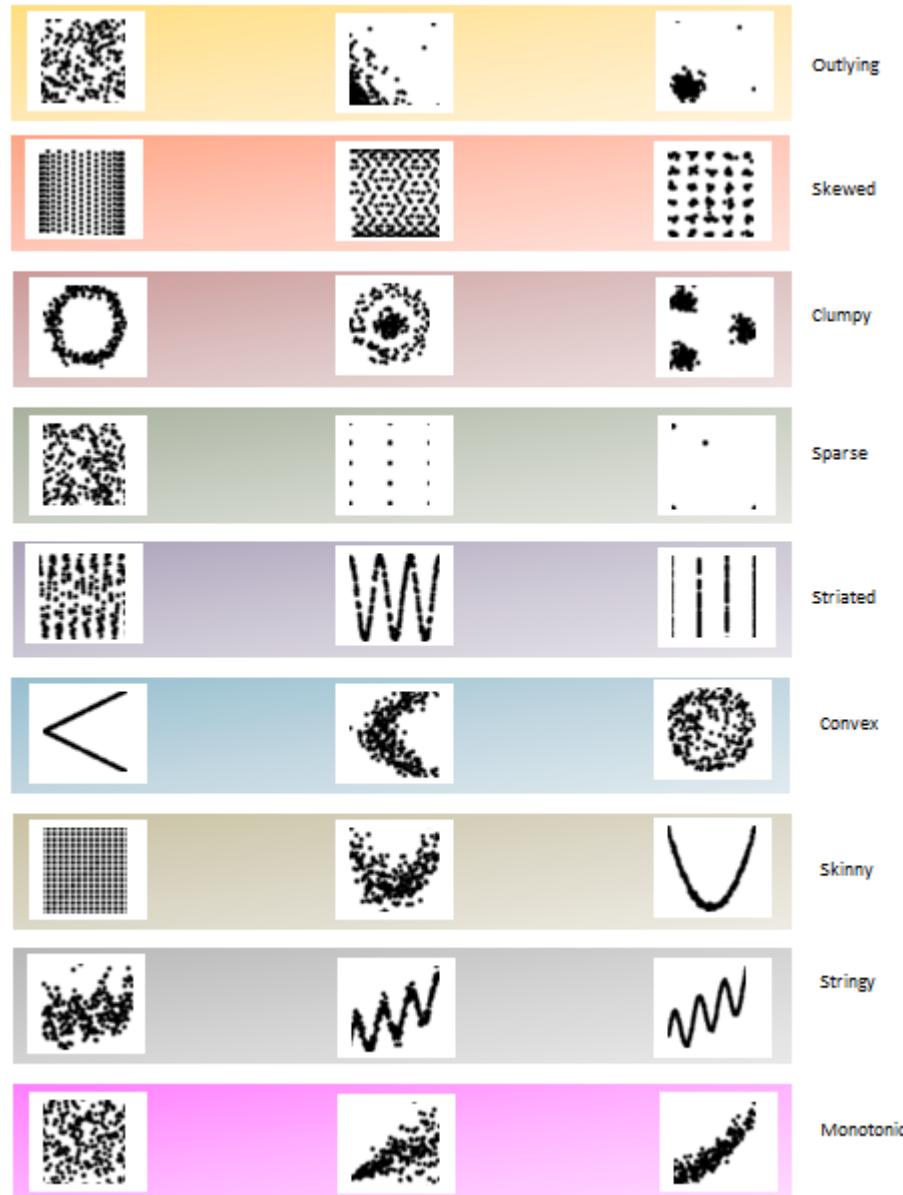
Slides are available at github: https://github.com/JayaniLakshika/Estadistica_2021

| ID | Sinhala Name | Family Name | Scientific Name | Shape | Arrangements | Bipinnately compound | Pinnately compound | Palmately compound | Edges | Uniform background | Red Margin | Shaded margin |
|----|--|------------------|---|-------|--------------|----------------------|--------------------|--------------------|-------|--------------------|------------|---------------|
| 1 | Tel kaduru (තෙලකදුරු) | EUPHORBIACEAE | <i>Sapium insigne</i> | 2 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 2 | Tehiriyia (තෙහිරියි) / Mayura manikkam (මායුර මාකික්කම්) | RHAMNACEAE | <i>Calubrina asiatica var. asatica</i> | 2 | 1 | 0 | 0 | 0 | 2 | 1 | 0 | 0 |
| 3 | Thakkali | SOLANACEAE | <i>Lycopersicon esculentum</i> | 4 | 2 | 0 | 1 | 0 | 3 | 1 | 0 | 0 |
| 4 | Thala | PEDALIACEAE | <i>Sesamum indicum</i> | 4 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 5 | Thana hal | POACEAE | <i>Setaria italica</i> | 4 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 6 | Thebu (තෙබු) / Koltan (කොතුන්) | ZINGIBERACEAE | <i>Costus speciosus</i> | 2 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 7 | Thel kola (තේලකොලා) | CONVOLVULACEAE | <i>Ipomoea littoralis</i> | 1 | 2 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 8 | Thembiloya (තෙම්බිලෝ) | MYRTACEAE | <i>Eugenia bracteata</i> | 2 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 9 | Thippili (තීපිලි) | PIPERACEAE | <i>Piper longum</i> | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 10 | Thiththa / Tolol | FLACOURTACEAE | <i>Trichadenia zeylanica</i> | 6 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 11 | Thiththa kindra | MENISPERMACAEAE | <i>Tinospora glabra</i> | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| 12 | Thiththa wel (තිත්තහ වේල්) | MENISPERMACAEAE | <i>Anamirta cocculus</i> | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 13 | Tholaboo (තොලාබූ) | AMARYLLIDACEAE | <i>Crinum asiaticum</i> | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 14 | Thumba karawila (තුඩ් කාරවිලා) / Mai thumba (මායි කාරවිලා) | CUCURBITACEAE | <i>Momordica dioica</i> | 5 | 1 | 0 | 0 | 0 | 2 | 1 | 0 | 0 |
| 15 | Yak erabudu (යාක උරඛු) / Katu kela (කටු කෙලා) | FABACEAE | <i>Erythrina fusca</i> | 2 | 2 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 16 | Yak kapu (යාක කපු) | STERCIURIACEAE | <i>Abrus Augusta</i> | 1 | 1 | 0 | 0 | 0 | 2 | 1 | 0 | 0 |
| 17 | Yak kon | MELIACEAE | <i>Syzygium febrifuga</i> | 4 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| 18 | Yakada wel (යාකද වේල්) / Manorangitham (මානරංගිත්මා) | ANNONACEAE | <i>Artobotrys hexapetalus</i> | 4 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| 19 | Yakahalu (යාකාලු) / Dummalala (දුම්මලා) | DIPTEROCARPACEAE | <i>Shorea obtongifolia</i> | 4 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 20 | Yakinaran (යාකිනරාන්) | RUTACEAE | <i>Atlantila rotundifolia</i> | 4 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 21 | Yak eraminiya | RHAMNACEAE | <i>Ziziphus napeca</i> | 4 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 22 | Yak komadu (යාක කොමදු) / Thiththa labu (තිත්තහ ලඔු) | CUCURBITACEAE | <i>Citrullus colocynthis</i> | 5 | 1 | 0 | 0 | 0 | 3 | 1 | 0 | 0 |
| 23 | Yaka bendi wel (යාක බංදී වේල්) / Wellangiriya (වේලංගිරියා) | RUTACEAE | <i>Paramigyna monophylla</i> | 4 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 24 | Yakada wel | RHAMNACEAE | <i>Ventilago madraspatana var. madraspatana</i> | 4 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| 25 | Yakinaran (යාකිනරාන්) | RUTACEAE | <i>Atalantia ceylanica</i> | 4 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 26 | Yawanasssa (යාවනසා) | LAMIACEAE | <i>Anisomeles indica</i> | 1 | 1 | 0 | 0 | 0 | 2 | 1 | 0 | 0 |
| 27 | Vanilla | ORCHIDACEAE | <i>Vanilla fragrans</i> | 4 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |

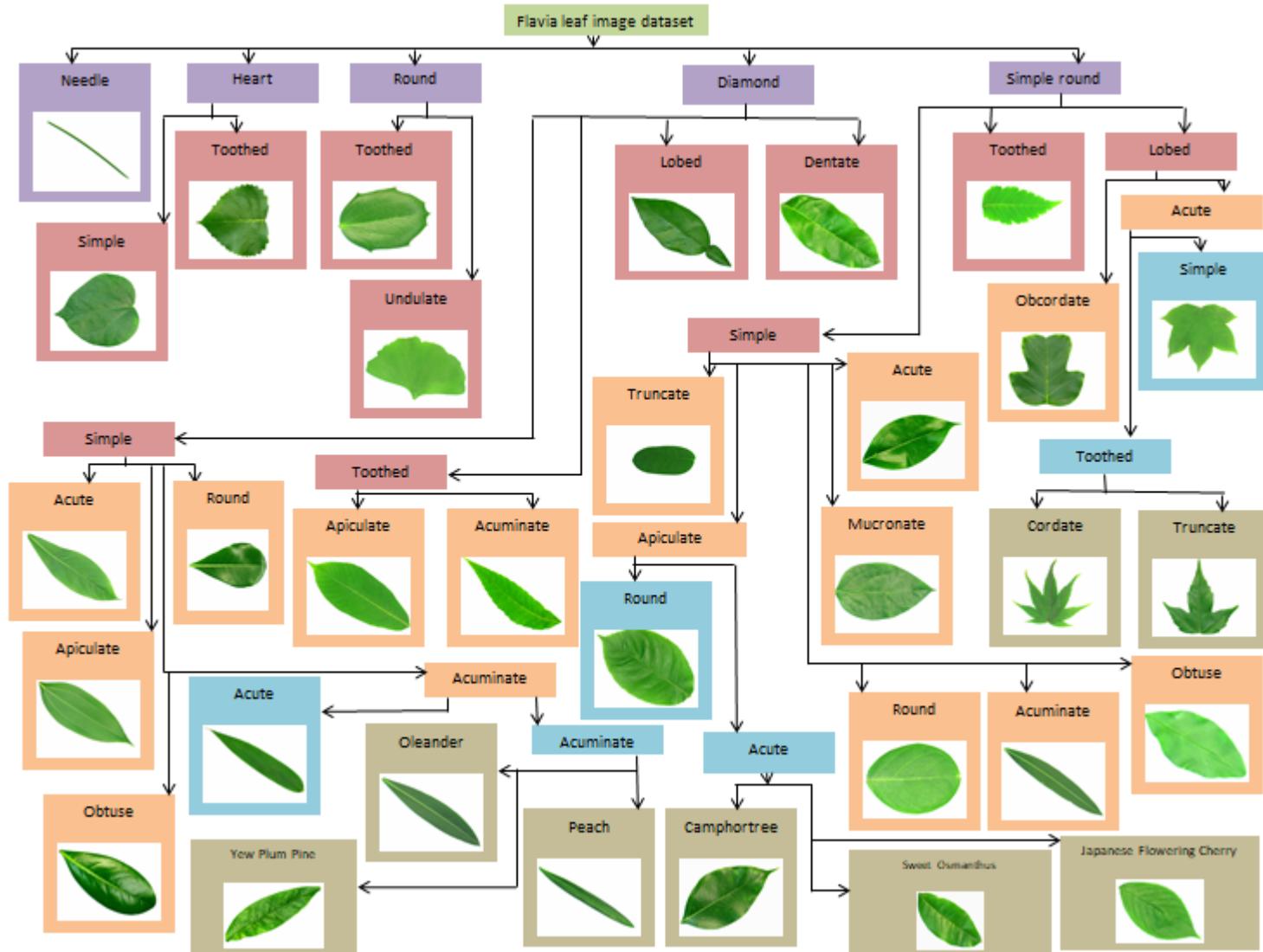
The website "AYURVEDIC MEDICINAL PLANTS OF SRI LANKA"
 (\url(<http://www.instituteofayurveda.org/plants/>))

Robust scaling

- If there are input variables that have very large values relative to the other input variables, these large values can dominate or skew some machine learning algorithms (eg: ANN, KNN, SVM etc).
- The result is that algorithm pay more attention to the large values and ignore the variables with smaller values.



Flavia hierarchy



Synthetic Minority Oversampling Technique (SMOTE)

- Imbalance dataset is a classification problem where the class distribution is biased or skewed or not uniform. Most machine learning algorithms for classification are designed under the assumption that each class has an equal number of observations. That results, especially for the minority class in the model, has poor predictive performance. Even though minority class is more important, classification errors for minority class is more sensitive than majority class.
- One remedy for class imbalance is to use an over-sampling method. SMOTE is an over-sampling method which creates synthetic (not duplicates) samples for the minority class and makes the minority class equal to the majority class. By selecting similar records and altering that record one column at a time by random amount within the difference to the neighbouring records, SMOTE handles the imbalance problem.

Key Insights



- Most of the research are based on existing databases. Even though researchers used their own database most of the time they didn't define image acquisition process in detail or it was complex. Therefore through this research, we introduced simplest and reliable approach of acquiring leaf images which can be followed without expertise knowledge.
- To classify leaf images, several reliable automatic procedures are used. But we introduced a hierarchical approach which perform better than non hierarchical approaches.
- Random Forest in hierarchical approach
- We have to use training/test from same dataset to get accurate results.

Further Research

- Develop algorithms to identify plant disease in Sri Lanka
- Expand the species collection
- Explore differences in plant features according to spatial distributions and climate conditions (For example, Gotukola leaf in Colombo is smaller than in Anuradhapura)
- Develop an algorithm to handle images with heterogeneous backgrounds