# CHAPTER 3

# EFFECT OF FEATURE SELECTION PROCESS ON CLASSIFICATION RESULTS

## 3.1 Introduction

With the increase in human population, human beings are building their abode in the areas on earth inhabited by trees, herbs and shrubs. In this process, the vegetation is losing its existence and even some of the plant species are on the verge of extinction. Therefore, it has become quite imperative that different plant species must be preserved for future. But the biggest hurdle in preserving the different plant species lies in first knowing them and then taxonomically classifying them according to their species. There are millions of plant species on this planet earth. A considerable number are still unknown or have regional or geographical variants untouched by the biologists, and very soon will be wiped out of this planet, due to negligence or due to the human needs for homes, roads and bridges etc. For classifying the plants, the different parts of the plants roots, shoots, seeds and flowers have been studied either independently or in groups. The biologists have been doing commendable work in taxonomically classifying and preserving the different plant species for future use. With the advancement in the technology, computer scientists and technologists are playing a crucial role by providing newer tools for understanding such species. The computer scientists are trying to understand the different plant species in a different way by applying image processing and machine learning techniques. These techniques involve fetching the features from the image data through different devices, sensors, statistical observations and analyzing these characteristic features for a meaningful plant species classification. The concept of plant classification using image processing techniques involves studying the images of the leaves, flowers and their placement on the plant. The classification of plants leaf needs the study of its geometrical shape, venation pattern, color and texture through their digital images.

According to Liu et al. (1998), the size of the dataset can be measured in two dimensions and they are: number of features (N) and number of instances (P). In the present scenario, both N and P are enormously large. This leads to the fact that there is a need to identify the subset of features according to a certain criterion and this can be used as tool to study the datasets. When N (*i.e*. the number of features) is reduced, the value of P (*i.e*. the number of instances) gets automatically reduced, and the size of the overall dataset shrinks and a small unique dataset is formed which is devoid of unwanted, irrelevant and duplicate features and is ready for further analysis. Therefore, it can be stated that the basic motto of feature selection techniques is to find and learn some unique functions from the patterns available from the dataset undertaken for study and then make the new pattern recognized by these learning functions.

According to Blachnik et al. (2009) and Hall (1999), the process of feature selection helps in improving the predictive classification accuracy and reduces the time required for computation, thereby making the dataset comprehensible. The feature selection methods have been grouped into three major categories. Firstly, in the case of embedded methods for feature selection, such methods are part of the predictors as in the case of decision trees and neural networks. In the case of filtering methods, they are independent of predictors and base themselves on the ranking of the features and measures the indices of the relevance of the feature quality from the subset of features selected. In the case of filtering techniques, correlation, chi-square and probability of distribution have often been used as methods to find the relevant features. The third category is called wrappers which wrap around certain predictors and check the performance on them. The dataset is divided into training set and testing set to test the predictor's performance.

The concept of feature selection algorithms is very popular in machine learning problems involving digital images. The digital images have been studied for image classification, enhancement, image compression and image segmentation purposes. Each digital image is made up of pixels of different intensity values and is placed in a pattern. The human beings can detect or distinguish different objects through their eyes using certain peculiar characteristic features of the objects like color, shape, geometry or texture features. When the new subset of the object is presented to the human beings, the object becomes discernible as the brain makes a comparative analysis of the new subset of features with that of the existing feature set in the brain. In the case of machine learning, there is a need for a feature set containing different characteristic features of the object of interest, may be digital images or any measurable object. The characteristic feature set of such objects is studied and subjected to feature selection and further classification process.

In this chapter, it is proposed to study the effect of feature selection algorithm on the predicative classification accuracy results of algorithms used for discriminating the different plant leaf images. The process involves extracting the important texture features from the digital images and then subjecting them to feature selection and further classification process. Section 3.2.2 describes about the methodology adopted to find the features from the leaf images of different plant species and preparation of the feature set. Section 3.2.3 describes about the application of feature selection algorithm in extracting useful features. Section 3.2.4 describes about the application of classification algorithms like KNN, J48, CART and RF on two different sets of data one representing all the features extracted from the leaf images and the second one representing the chosen few features. Section 3.3 represents the result analysis and comparative study with other works of similar nature. This chapter has been adapted from Kumar et al. (2015).

# 3.2 Methodology Adopted

## 3.2.1 Data preprocessing

The present work involves studying the effect of feature selection algorithm on the predictive accuracy of the classifiers working with digital leaf image datasets. In this work, 250 images of 10 plant species have been taken for the experimental purpose and a sample of the leaf image dataset is shown in Figure 3.1. The colored leaf images were converted to gray scale and the size of all the images was reduced to 256 × 256. Each gray scaled slice of a leaf image was preprocessed and background was removed and its contrast and intensity values were enhanced as mentioned by Gonzalez et al.(2001).

Figure 3.2 shows the Slice-1 in gray scale and its enhanced image has been shown as Slice-1E, the next part of Figure 3.2 shows the histogram of the Slice-1 and that of the Slice-1E. The histogram of Slice-1 shows that the distribution of the pixels concentrated in a region whereas the histogram of Slice-1E shows that the pixel intensities have been distributed over a wider region.

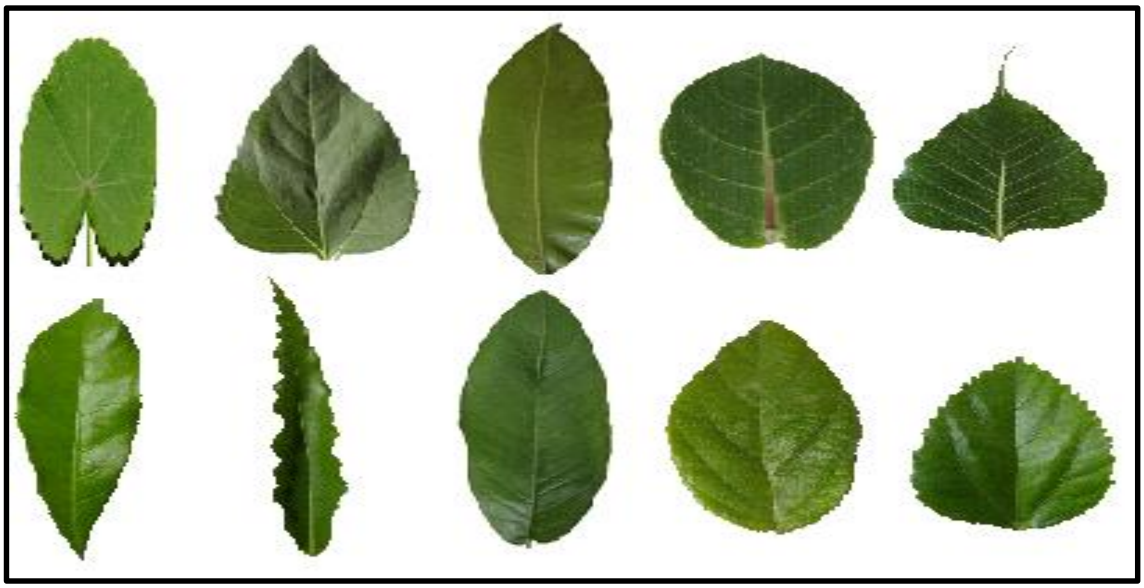

Figure 3.1 A sample of Leaf image dataset

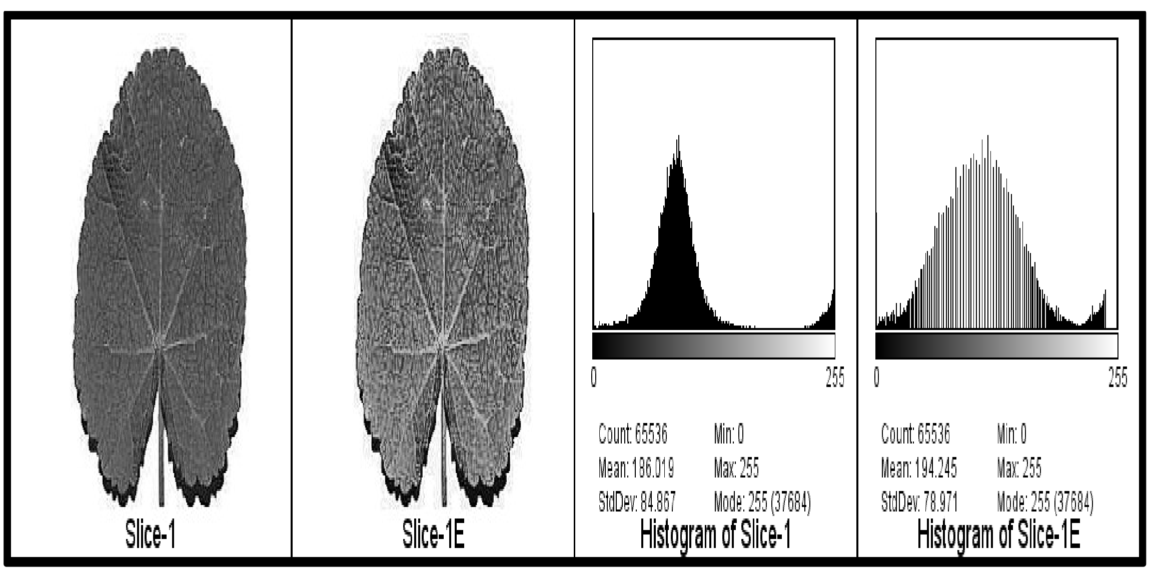


Figure 3.2 Leaf image for Slice-1 and its enhanced leaf image Slice-1E with their respective histograms

### 3.2.2 Gabor texture feature extraction

The term Gabor filter has been coined after the name of Dennis Gabor who in the year 1946 experimented and subsequently proposed the representation of the signals. In image processing tasks, Gabor filters have been extensively been used for feature extraction for the digital leaf images. The frequency and orientation representation as used in Gabor filters, are useful for texture representation and discrimination and the same concept is used in human visual system. The most important properties are related to invariance to illumination, rotation, scale, and translation. The Gabor filters have several advantages in feature extraction process over other techniques used for the similar purpose. The Gabor feature vectors can be used directly as input to a classification or segmentation operator or they can first be transformed into feature vectors, which are then used as input for another stage. The Gabor features have created a niche for themselves in the areas of face recognition, character recognition, browsing and retrieving of image data. The concept of Gabor theory [Arivazhagan et al.(2006)] has been represented in Eqs. (3.1), (3.2) and (3.3).

$$g(x,y;\lambda,\theta,\psi,\sigma,\gamma)=\exp\left(\frac{-x'^{2}+\gamma^{2}y'^{2}}{2\sigma^{2}}\right)\exp\left(i\left(2\pi\frac{x'}{\lambda}+\psi\right)\right) \tag{3.1}$$

$$g(x,y;\lambda,\theta,\psi,\sigma,\gamma)=\exp\left(\frac{-x'^{2}+\gamma^{2}y'^{2}}{2\sigma^{2}}\right)\cos\left(2\pi\frac{x'}{\lambda}+\psi\right) \tag{3.2}$$

The Eq. (3.2) represents the real part of the Gabor representation.

$$g(x,y;\lambda,\theta,\psi,\sigma,\gamma)=\exp\left(\frac{-x'^{2}+\gamma^{2}y'^{2}}{2\sigma^{2}}\right)\sin\left(2\pi\frac{x'}{\lambda}+\psi\right) \tag{3.3}$$

The Eq. (3.3) represents the imaginary part of the Gabor representation, where $x' = x\cos\theta + y\sin\theta$ and $y' = -x\sin\theta + y\cos\theta$.

In Eq. (3.1), (3.2) and (3.3), $\lambda$ represents the wavelength of the sinusoidal factor, $\theta$ represents the orientation of the normal to the parallel stripes of a Gabor function, $\psi$ is the phase offset, $\sigma$ is the standard deviation of the Gaussian envelope and $\gamma$ is the spatial aspect ratio, and specifies the ellipticity of the support of the Gabor function. The contrast enhanced image with the help of histogram equalization process undergo Gabor transform as mentioned in Eq. (3.2). The Gabor transform of an image $I(x, y)$ is defined as the convolution of a Gabor filter $g(x, y)$ as shown in Eqs. (3.4) and (3.5).

$$R(x,y) = g(x,y) * I(x,y) \tag{3.4}$$

Therefore,

$$R(x,y) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} g(m,n).I(x-m, y-n) \tag{3.5}$$

The operator $*$ in Eq. (3.4), denotes the two dimensional linear convolution, $M$ and $N$ are the sizes of the Gabor filter space. The Gabor filters extract characteristics in specific orientation and frequency bands called Gabor coefficients as described by Zhang et al. (2012) and Casanova et al. (2009) and denoted by Eq. (3.6).

$$G(s,\theta)_{(x,y)} = \sqrt{E^2_{(s,\theta)(x,y)} + O^2_{(s,\theta)(x,y)}} \tag{3.6}$$

where $E_{(s,\theta)} = I * g_{e(s,\theta)}$ and $O_{(s,\theta)} = I * g_{o(s,\theta)}$

Here, scale $s$ and orientation $\theta$ are odd symmetric and even symmetric filters for the image $I$ and in Eq. (3.6) $*$ denotes the convolution operation. A single Gabor filter will detect patterns in the leaf images with a certain fixed frequency and an orientation value. To capture the entire texture feature available in a digital image, the Gabor filter bank is tuned at different frequency and orientation values. According to Kyrki et al.(2004), the properties of Gabor Filters are as follows:

- **Translation Invariance:** One of the central properties of the Gabor filter is the translation invariance property. Let $s$ be a translated version of some 2-D signal $s_1$:

$$s_2 = s_1\left(x - x_1, y - y_1\right) \tag{3.7}$$

Then, through Eq. (3.7), it can be shown that the Gabor response of s2 is a translation of the Gabor response of $s_1$. The Eq. (3.8) demonstrate response of translation invariance using gabor.

$$\begin{aligned} r_{s2}\left(x, y; f, \theta\right) &= \iint s_2\left(x - x_\tau , y - y_\tau \right) g\left(x_\tau , y_\tau ; f, \theta\right) dx_\tau , dy_\tau \\ &= \iint s_2\left(x - x_1 - x_\tau , y - y_1 - y_\tau \right) g\left(x_\tau , y_\tau ; f, \theta\right) dx_\tau , dy_\tau \\ &= r_{s1} \left(x - x_1, y - y_1; f, \theta\right) \end{aligned} \tag{3.8}$$

- **Scale Invariance:** 2-D Gabor filters are also scale invariant . Let the 2-d signal $s_3$ be a homogeneously scaled version of the signal $s_1\left(x, y\right)$, as in Eq. (3.9).

$$s_3\left(t\right) = s_1\left(ax, ay\right) \tag{3.9}$$

$$r_{s3} \left(x, y; f, \theta\right) = \iint g\left(x - x_\tau , y - y_\tau ; f, \theta\right) s3\left(x_\tau , y_\tau \right) dx_\tau dy_\tau$$

$$= \iint s_3(x - x_\tau, y - y_\tau) g(x_\tau, y_\tau; f, \theta) dx_\tau dy_\tau$$

$$= \iint s_1(ax - ax_\tau, ay - ay_\tau) g(x_\tau, y_\tau; f, \theta) dx_\tau dy_\tau \quad (3.10)$$

Let $\hat{x}_\tau = ax_\tau, \hat{y}_\tau = ay_\tau \Rightarrow dx_\tau = \frac{d\hat{x}_\tau}{a}$ and $dy_\tau = \frac{d\hat{y}_\tau}{a}$, in Eq. (3.10), then the integral can be redefined using Eq. (3.11).

$$r_{s3}(x, y; f, \theta) = \iint s_1(ax - \hat{x}_\tau, ay - \hat{y}_\tau) g\left(\hat{x}_\tau, \hat{y}_\tau; \frac{f}{a}, \theta\right) d\hat{x}_\tau d\hat{y}_\tau$$

$$= r_{s1}\left(ax, ay; \frac{f}{a}, \theta\right) \quad (3.11)$$

The Eq. (3.11) represents the scale property of Gabor filters, i.e. the filter response of a scaled version of a signal at a particular location is equivalent to a scaled version (determined by the scaling factor of the signal) of the filter response.

- **Rotation and illumination invariance:** Let $s_2(x, y)$ be a rotated version of an image $s_1(x, y)$, where $s_1$ is rotated anti-clockwise around the spatial location $(x_0, y_0)$ by an angle φ, as shown through Eq. (3.12).

$$s_2(x, y) = s_1(\hat{x}, \hat{y}) \quad (3.12)$$

where

$$\hat{x} = (x - x_0) \cos\varphi + (y - y_0) \sin\varphi + x_0$$

$$\hat{y} = -(x - x_0) \cos\varphi + (y - y_0) \sin\varphi + y_0$$

The Gabor filter response of the rotated image in terms of the rotated parameters has been shown through Eq. (3.13).

$$r_{s2}\left(x_0, y_0; f, \theta\right) = \iint g\left(x_0 - x_\tau, y_0 - y_\tau; f, \theta\right) s_2\left(x_\tau, y_\tau\right) dx_\tau dy_\tau \qquad (3.13)$$

By changing the integration axes to $\left(x_\tau', y_\tau'\right)$ which are correspondingly rotated around the point $\left(x_0, y_0\right)$ by the angle $-\varphi$ Eq. (3.13) can be rewritten as Eq. (3.14):

$$= \iint g\left(x_0 - \hat{x}_\tau, y_0 - \hat{y}_\tau; f, \theta - \varphi\right) s_1\left(x_\tau', y_\tau'\right) dx_\tau' dy_\tau' = \mathrm{r}_{s1}\left(x_0, y_0; f, \theta - \varphi\right) \qquad (3.14)$$

where

$$\hat{x}_\tau = \left(x_0 - x_\tau'\right) cos\left(\theta - \varphi\right) + \left(y_0 - y_\tau'\right) sin\left(\theta - \varphi\right)$$
$$\hat{y}_\tau = -\left(x_0 - x_\tau'\right) \sin\left(\theta - \varphi\right) + \left(y_0 - y_\tau'\right) \cos\left(\theta - \varphi\right)$$

The rotation property in Eq. (31.4) shows that the response of the Gabor filter for a rotated image is equal to the response of the correspondingly rotated filter for the original image without rotation. Let $s_5\left(x, y\right)$ be a differently illuminated version of the image $s\left(x, y\right)$. The uniform illumination change can be modeled as a multiplication by a constant, implying that $s_5\left(x, y\right) = cs\left(x, y\right)$. Based on the linearity of the convolution, it can be written through Eq.(3.15).

$$r_{s5}\left(x, y; f, \theta\right) = cr_s\left(x, y; f, \theta\right) \qquad (3.15)$$

The Eq. (3.15) shows the illumination invariance property, i.e. the Gabor filter response for an illuminated image (the original image multiplied by a constant) is equal to the correspondingly filter response (multiplied by the same constant) of the original image.

The present work involves studying the effect of feature selection algorithm on the predictive accuracy of the classifiers working with digital leaf image datasets. In this work, 250 images of 10 plant species have been taken for the experimental purpose and a sample of the leaf image dataset has been shown in Figure 3.1. The colored leaf images were converted to gray scale and the size of all the images was reduced to $256 \times 256$. Each gray scaled slice of a leaf image was preprocessed and background was removed and its contrast and intensity values were enhanced as described by Gonzalez et al. (2001) as shown in Figure 3.2. Figure 3.2 shows the Slice-1 in gray scale and its enhanced image has been shown as Slice-1E, the next part of Figure 3.2 shows the histogram of the Slice-1 and that of the Slice-1E. The histogram of Slice-1 shows that the distribution of the pixels concentrated in a region whereas the histogram of Slice-1E shows that the pixel intensities have been distributed over a wider region.

To pursue, work in the area of digital images, there is a need to extract important features. Texture is a basic characteristic visual feature, which helps the human visual system in segmentation and recognition based processes, performed by human brain. Texture based features in computer vision science have been playing its role for the last couple of years. Textures can be divided into two categories, namely, tactile and visual textures. Tactile textures refer to the immediate tangible feel of a surface. Visual textures refer to the visual impression that textures produce to human observer, which are related to local spatial variations of simple stimuli like color, orientation and intensity in an image [Wikimedia(2015)]**.**

In this work, Gabor features have been extracted from the enhanced images. A single Gabor filter will detect patterns in the leaf images with a certain fixed frequency and an orientation value. To capture the entire texture features available in the digital image, the Gabor filter bank is tuned at different frequency and orientation values. For each image in the dataset, the Gabor filter generates 32 real images for 4 different values of Scale (2, 4, 8, 16) and 8 different orientation values (22˚, 44˚, 66˚, 88˚, 110˚, 132˚, 154˚, 176˚). The time domain plot for Gabor filter at different orientation values has been shown in Figure

3.3 for Slice-1. Figure 3.4 shows the 32 images obtained by convolving the enhanced image Slice-1 with Gabor Filter.

After the images had undergone the process of image enhancement, a stack of enhanced images was prepared. This stack was subjected to the process of feature extraction using Gabor filters [Wikimedia(2015)] and six texture features namely Mean, Energy, Standard Deviation, Skewness, Contrast and Kurtosis were derived and the measured values were stored in separate CSV file, for classification purpose as mentioned in next sections.

After the images had undergone the process of image enhancement, a stack of enhanced images was prepared. This stack was subjected to the process of feature extraction using Gabor Filters [Wikimedia(2015)] and six texture features namely Mean(GTF1), Energy(GTF2), Standard Deviation(GTF3), Skewness(GTF4), Contrast(GTF5) and Kurtosis(GTF6) were derived and the measured values were stored in separate CSV file, for classification purpose as mentioned in next sections.

- **Mean(GTF1):** It is denoted as mentioned in Eq. (3.16).

$$\mu_{(s,\theta)} = \frac{1}{NM} \sum_{x=1}^{N} \sum_{y=1}^{M} G_{(s,\theta)}(x,y) \tag{3.16}$$

- **Energy(GTF2):** The texture energy is expressed as $E(x,y)$ and is mentioned in Eq. (3.17).

$$E(x,y) = \frac{1}{M} \sum_{(a,b \in w)} \left| R(a,b) - \mu \right| \tag{3.17}$$

- **Standard Deviation(GTF3):** The standard deviation has been calculated as mentioned in Eq. (3.18).

$$\sigma_{(s,\theta)} = \sqrt{\frac{1}{NM}\sum_{x=1}^{N}\sum_{y=1}^{M}\left(G_{(s,\theta)(x,y)} - \mu_{(s,\theta)}\right)^2} \tag{3.18}$$

- **Skewness(GTF4):** Skewness is the measure of asymmetry and is denoted by $\gamma$ **,** it can be positive which means that the distribution tends towards right and if it is negative when the distribution tends towards left and is represented by Eq. (3.19).

$$\gamma_{(s,\theta)} = \frac{\mu^3_{(s,\theta)}}{\sigma^3_{(s,\theta)}} \tag{3.19}$$

- **Contrast (GTF5) and Kurtosis(GTF6):** Contrast is expressed as $\psi_{(s,\theta)}$ as mentioned in the Eq. (3.20).

$$\psi_{(s,\theta)} = \frac{\mu_{(s,\theta)}}{k^{0.25}_{(s,\theta)}} \tag{3.20}$$

where $k_{(s,\theta)} = \frac{\mu^4_{(s,\theta)}}{\sigma^4_{(s,\theta)}}$ is the kurtosis or the degree of peakedness in a dataset.

The Gabor Texture Feature Dataset(GTFD) has been prepared using all the six Gabor features extracted and has been shown in Eq. (3.21).

$$GTFD = \left(GTF1, GTF2, GTF3, GTF4, GTF5, GTF6\right) \tag{3.21}$$

Here $GTF1, GTF2, ..., GTF6$ indicate that all the six different values of Gabor texture features as mentioned above in Eqs. (3.16), (3.17), (3.18), (3.19) and (3.20).

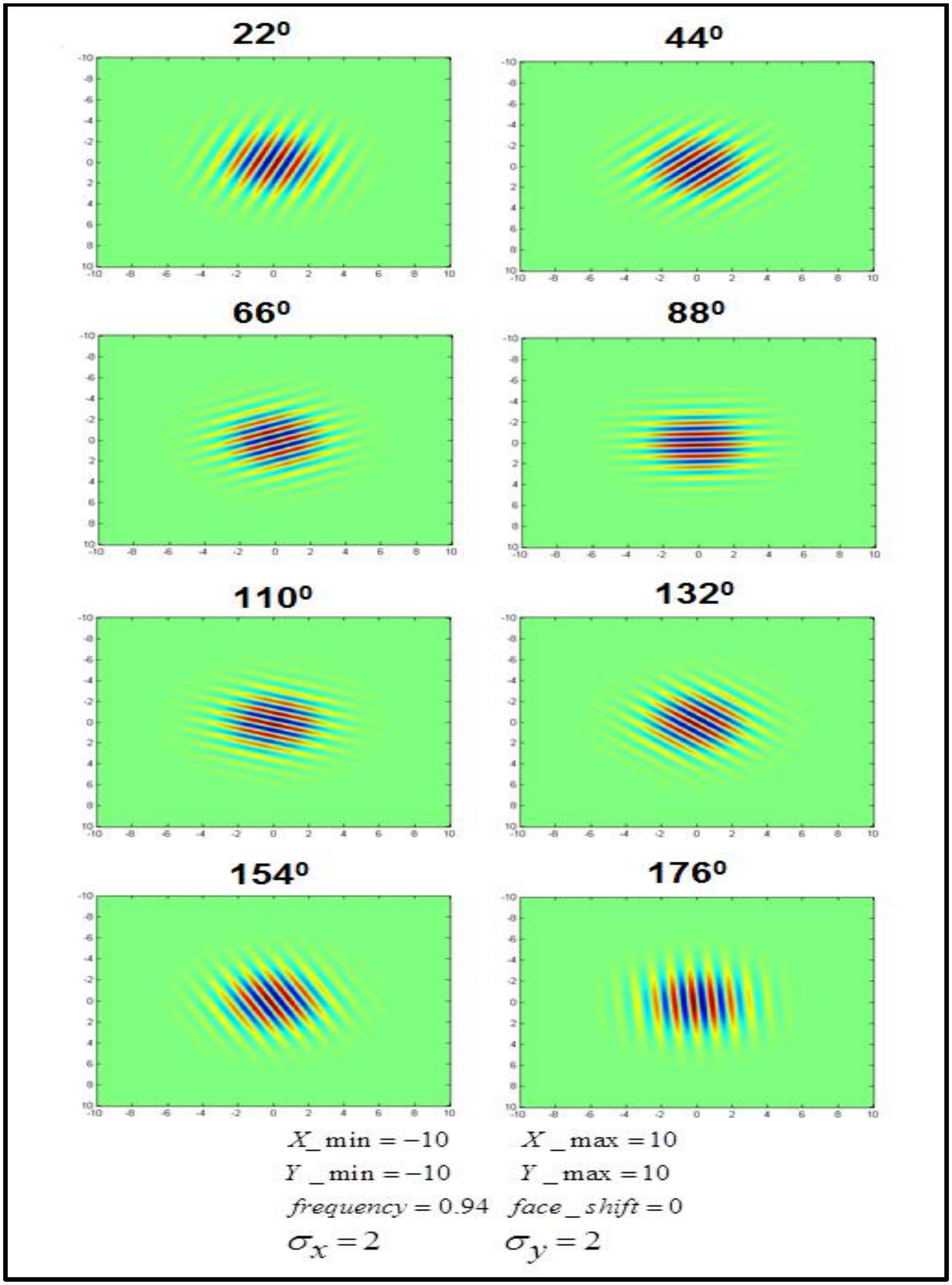


Figure 3.3 Time domain diagram for Gabor filter for Slice-1 at different orientation values

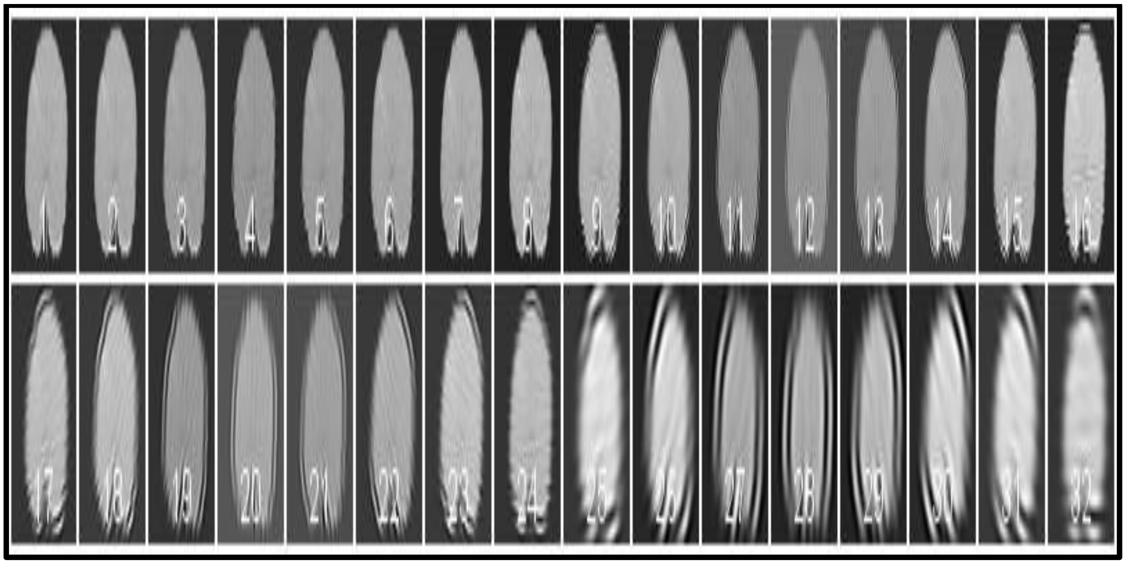


Figure 3.4 Slice-1 convolved with Gabor filter, generates 32 images at different scales and orientation values

### 3.2.3 Application of Random Forest algorithm for feature selection

Random forest is an ensemble classifier that consists of many decision trees and outputs the class that is the mode of the class's output by individual trees. Random forests are often used when we have very large training datasets and a very large number of input variables (hundreds or even thousands of input variables). A random forest model is typically made up of tens or hundreds of decision trees. In this algorithm, the basic principle is to construct a predictor ensemble with a set of decision trees that grow in randomly selected subspace of features and then calculating the best split based on these features in the training set.

According to White (2005), random forest is a meta learner technique having many individual learners or trees as shown in Figure 3.5. The random forest uses multiple random trees classifications to vote on an overall classification for the given set of inputs. Each individual meta-learner vote has equal vote. The forest chooses the individual classification that has the most votes. The Figure 3.5 shows the Meta Learners or the trees voting (v1 etc.), but the final tree chooses the maximum vote criteria.

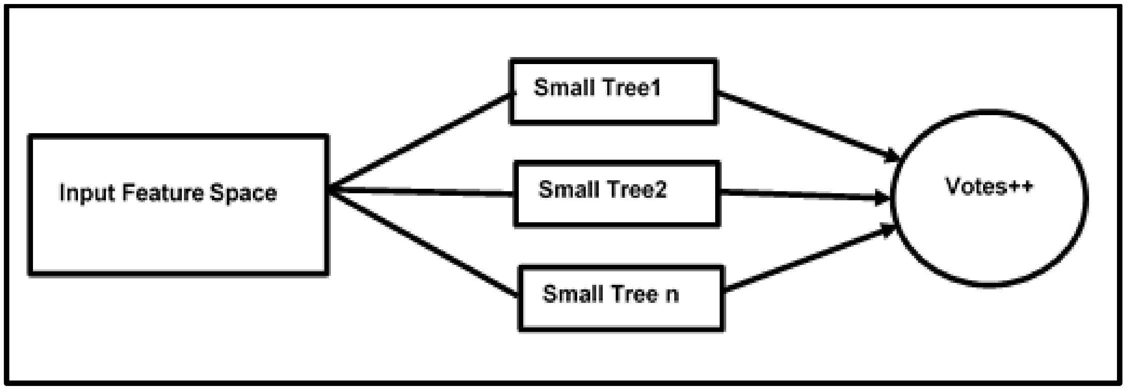


Figure 3.5 Concept of meta learners in Random Forest

A two copies of CSV(Comma Separated Values) file of data with six features has been prepared using Eq. (3.21). One copy of the file has been preserved for classification purpose and the other copy has been subjected to feature selection algorithm. Automatic feature selection methods can be used to build many models with different subsets of a dataset and identify those attributes that are not required to build an accurate model as mentioned by Liu et al. (1998). In this algorithm, dataset is divided into training and test sets. The training set is trained over all the predicators. The predictive accuracy of the unknown sample is calculated and then the variable importance is calculated. Now the training set is again trained over the few important variable having higher ranking than others and the predictive accuracy of the unknown sample is again calculated. The appropriate number of predictors are identified and the model is prepared over the new set of predictors. The variable importance depends upon the interaction of the variables with each other. The Random Forest algorithm estimates the importance of variables by looking at how the prediction error increases, when the out of bag (OOB) data are permuted while others are left unchanged.

In this present work, there are six predictor variables as mentioned in section 3.2.2, they are subjected to random feature elimination algorithm to find the best features, avoiding correlated variables as far as possible. Figure 5 shows the resampling performance over a subset size using 10-fold cross-validation technique. The Figure 3.6 shows the use of number of variables on the x-axis plotted against predictive accuracy values (10-fold cross validated). From Figure 3.6, it is clear that out of six features, it is appropriate to choose five features which shall provide comparable accuracy results as are provided by using six features.

Figure 3.7 shows the plot for variable importance (VIMP). Figure 3.7(a) shows mean decrease in the accuracy values plotted against the features of the dataset and it has been observed that $GTF_3$ has the highest value of 158.024 which makes it the most important variable and $GTF_2$ has the lowest value of 109.50 making it the least important variable in the dataset.

The mean decrease in accuracy a variable causes is determined during the out of bag error calculation phase. The more the accuracy of the random forest decreases due to the exclusion (or permutation) of a single variable, the more important that variable is deemed, and therefore variables with a large mean decrease in accuracy are more important for classification of the data. The mean decrease in Gini coefficient is a measure of how each variable contributes to the homogeneity of the nodes and leaves in the resulting random forest. Each time a particular variable is used to split a node, the Gini coefficient for the child nodes are calculated and compared to that of the original node. The Gini coefficient is a measure of homogeneity from 0 (homogeneous) to 1 (heterogeneous). The changes in Gini are summed for each variable and normalized at the end of the calculation. Variables that result in nodes with higher purity have a higher decrease in Gini coefficient.

A type 1 variable importance plot shows the mean decrease in accuracy, while a type 2 plot shows the mean decrease in Gini.as mentioned by Dins Lab (2015). The five best predictor variables found by using random forest algorithm are $GTF_3$, $GTF_5$, $GTF_6$, $GTF_1$ and $GTF_2$ placed in highest to the lowest variable importance order. These five features were subjected to find the predictive classification accuracy.

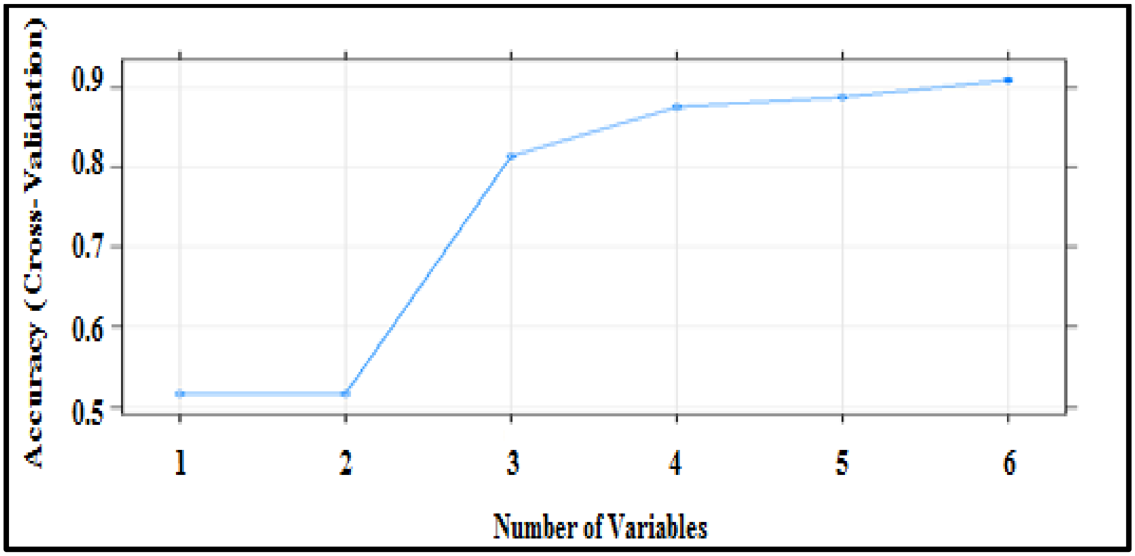


Figure 3.6 Resampling over subset size

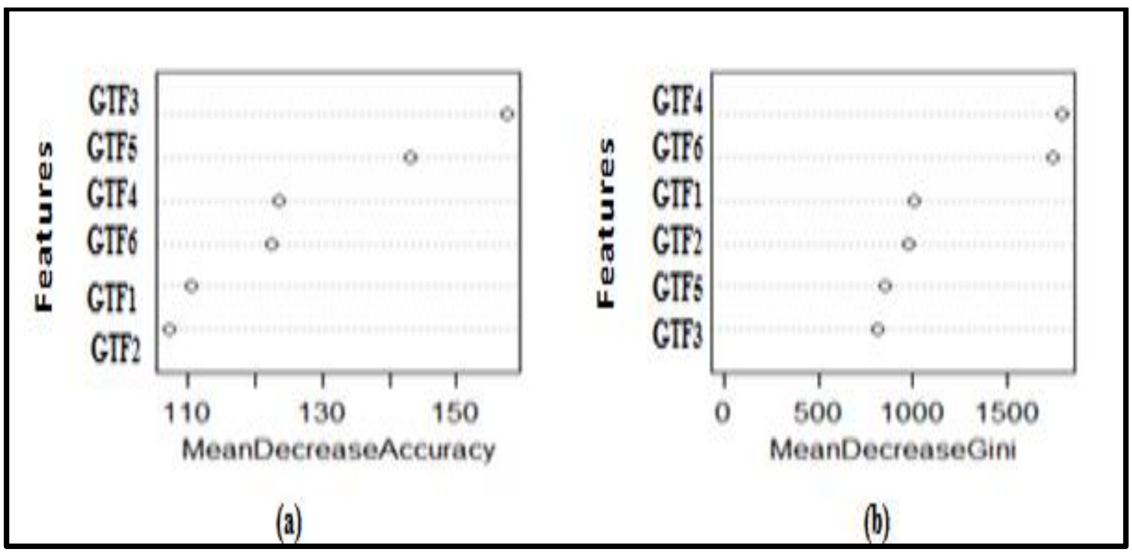


Figure 3.7 Visualization of variable importance

### 3.2.4 Application of classification algorithms

The process of feature selection is proceeded by methods and techniques to classify the data sets into appropriate classes using classification algorithms. The following four classification algorithms have been used for the classification of leaf images: K-Nearest

Neighbor (KNN), J48, Classification and Regression Trees(CART), Random Forest(RF) using R [A I.3]and ImageJ[A I.2].

Each data set was split into two groups (Training and Test sets) in the ratio 75:25. The training data set contains the class labels, whereas the testing dataset does not contain the class labels. The pre-processing of the data involved centring and the scaling of the data matrix. In the classification procedure, a 10-fold cross validation technique has been applied which is repeated three times for validating any predictive model. Predictive accuracy and kappa values have been adopted as a measurable parameter for the classification process. Kappa is defined as the degree of right predictions of a model. This is originally a measure of agreement between two classifiers and is calculated with Eq. (3.22).

$$\kappa = \frac{\Pr(a) - \Pr(e)}{1 - \Pr(e)} \tag{3.22}$$

In broad terms a kappa below 0.2 indicates poor agreement and a kappa above 0.8 indicates very good agreement beyond chance as described by Sim et al. (2005).

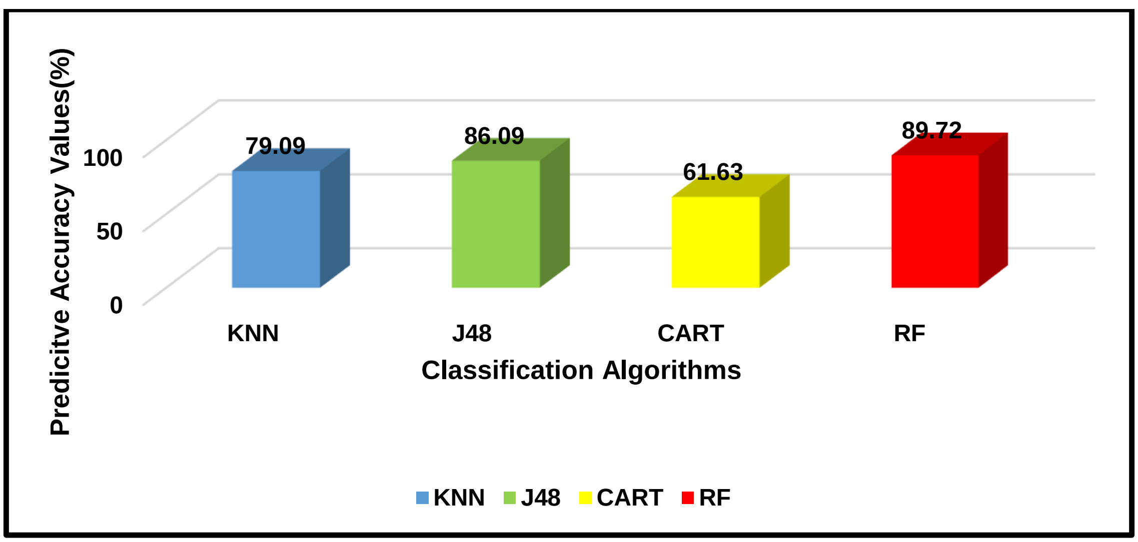


Figure 3.8 Predictive accuracy chart for the complete feature set

## 3.3 Results and Discussion

The predictive accuracy values calculated for the feature set containing all the six features show that, RF algorithm gives the highest predictive accuracy value of 89.72%, closely followed by J48 algorithm at 86.09%, as shown in the Figure 3.8.

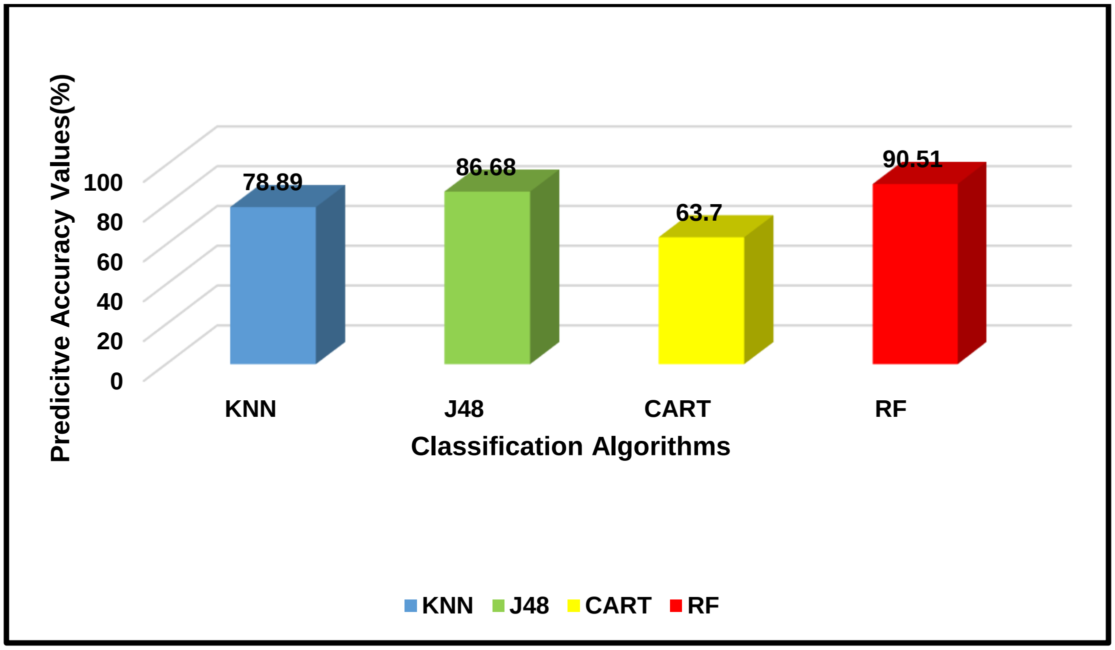


Figure 3.9 Predictive accuracy chart for five feature set

In the case of model where the features have been chosen through RF technique and then the percentage accuracy value has been calculated, again the RF classification algorithm gives the highest predictive accuracy value of 90.51% and closely followed by J48 algorithm at 86.68%, as shown in Figure 3.9.

A margin is a measure of the certainty of classification. This method calculates the difference between the support of a correct class and the maximum support of an incorrect class. A margin is the measurement of certainty of the classification; it is computed by the support of the correct class and the maximum support of the incorrect class. The formula of margins is represented in Equation (3.23).

$$\text{margin}(x_i) = \text{support}_c(x_i) - \max_{j \neq c} \text{support}_j(x_i) \tag{3.23}$$

In the Random Forest classifier the margin [Dins Lab and Yu et al.(2015)] for the data points was also calculated and has been shown through Figure 3.10. The margin of a data point is defined as the proportion of votes for the correct class minus maximum proportion of votes for the other classes. Here, the margin of the $\mathbf{x_i}$ sample equals the support of a correctly classified sample (**c** denotes the correct class) minus the maximum support of a sample that is classified to **class j** (**where j≠c and j=1…k**). Therefore, correctly classified examples will have positive margins and misclassified examples will have negative margins. If the margin value is close to one, it means that correctly classified examples have a high degree of confidence. On the other hand, examples of uncertain classifications will have small margins.

Thus under majority votes, positive margin means correct classification, and vice versa as shown in the Figure 3.10. In the dataset there are eight thousand tuples shown on x-axis and y-axis show the proportion of correct votes for the class minus the maximum votes for other classes.

Figure 3.11 shows the error rate over the trees. In the case of Random Forest classification method, the number of trees prepared were 500(ntree). The curve shows the number of trees constructed on the x-axis and corresponding errors on y-axis per leaf image class in different colors. The OOB error has been shown in black color. The OOB data is used to get a running unbiased estimate of the classifier errors as trees are added to the forest per leaf image class.

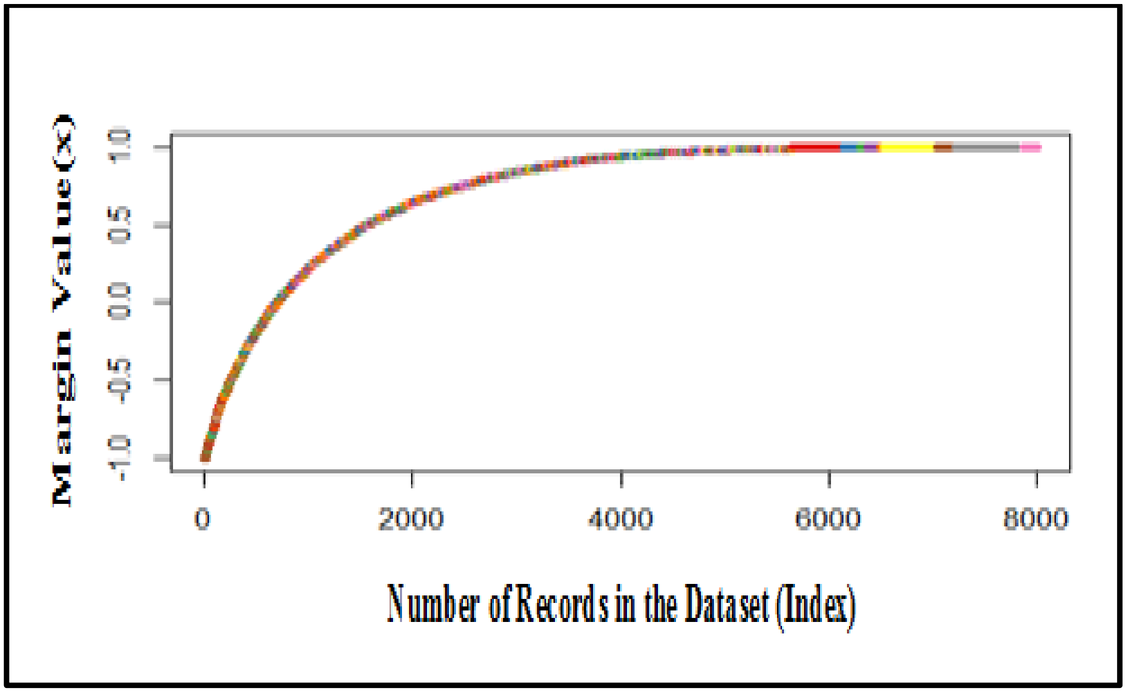


Figure 3.10 Number of records plotted against predicted margin values for each leaf image class

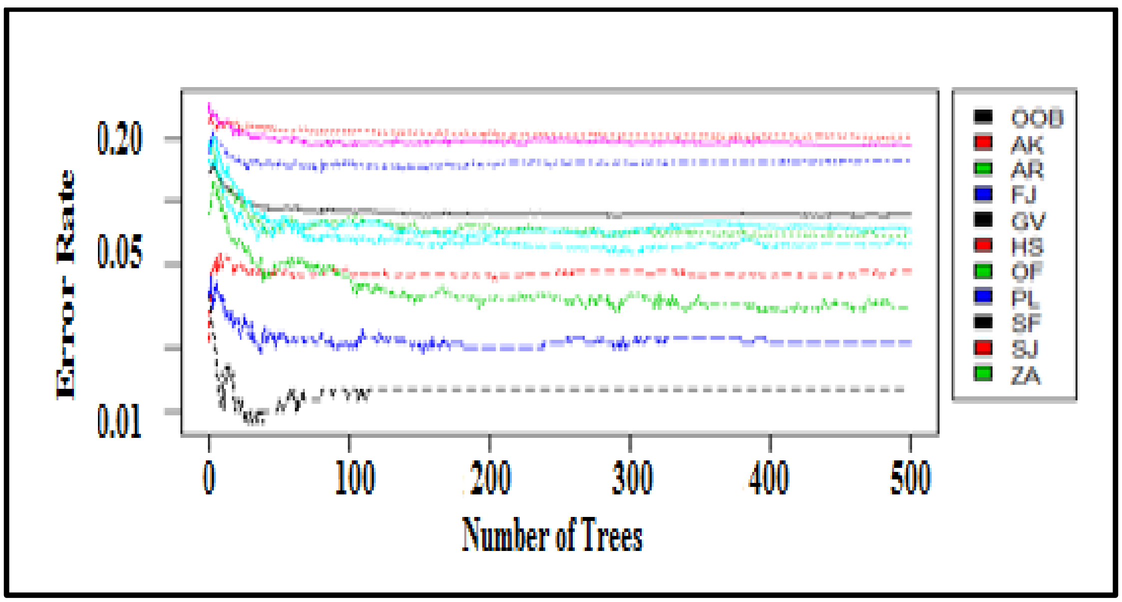


Figure 3.11 The number of trees constructed corresponding to the error rate generated per class

Table 3.1 Kappa values for two different feature sets

| **Classification Algorithm** | **Predictive Kappa Values (%)** | |
|---|---|---|
| | **Complete set of features (6 Features)** | **Selected set of features (5 Features)** |
| KNN | 76.77 | 76.54 |
| J48 | 84.54 | 85.20 |
| CART | 57.37 | 59.67 |
| RF | 88.58 | 89.46 |

## 3.4 Conclusion

The predictive accuracy value in case of selected features is higher as compared to prediction values calculated upon all the features of the leaf image dataset as discussed in Section 3.3. The higher performance has also been expressed through the kappa values mentioned in Table 3.1 and further through the margin values represented in Figure 3.10. The results discussed in Section 3.3, have proved the assumption of this study that feature selection has an incremental effect on predictive accuracy values for leaf image classification. The size of the dataset has also been reduced due to selected features. The future scope of the leaf image classification lies in studying genetic algorithms and optimization techniques.