A project based on

Performance Evaluation of different Machine Learning (Classification) algorithms in plant disease detection

Submitted in partial fulfilment of the Requirement for the award of the

Degree of

BACHELOR OF TECHNOLOGY

In

ELECTRONICS AND COMMUNICATION ENGINEERING

Technical Proficiency & Training - II (19TS4006)

Project submitted by

Batch-7

190040004 – A. Akhil

190040527 - T. Siddardha Rayudu

190040646 - J. V. Sri Lakshmi

190049031 – T. Eswar Abhishake

Department of Electronics and Communication Engineering

KONERU LAKSHMAIAH EDUCATIONAL FOUNDATION

Green Fields, Vaddeswaram Guntur [Dt],
A.P., India-522502

DECLARATION

We (A. Akhil, T. Siddardha, J. V. Sri Lakshmi & T. Eswar), belong to batch 7 are pursuing B. Tech under Department of Electronics and Communication, K L University, Vaddeswaram , hereby declare that all the information furnished in this report is based on our own intensive research and is genuine.

This report does not, to the best of our knowledge, contain part of our work which has been submitted for the award of our degree either of this university or any other university without proper citation.

Batch-7

190040004 – A. Akhil

190040527 - T. Siddardha Rayudu

190040646 - J. V. Sri Lakshmi

190049031 – T. Eswar Abhishake

KLEF DEPARTMENT OF E.C.E

Technical Proficiency & Training - II (19TS4006)

CERTIFICATE

This is to certify that the project-based laboratory report entitled "**Performance Evaluation of different Machine Learning(Classification) algorithms in plant disease detection**" submitted by batch – 7 students, belonging to the Department of Electronics and Communication Engineering, KL University in partial fulfilment of the requirements for the completion of a project based Laboratory in "**Technical Proficiency & Training - II"** course in III- year

B Tech EVEN Semester is a Bonafide record of the work carried out by him/her under my supervision during the academic year 2021 – 2022.

Signature of the course instructor **Signature of the course Coordinator**

ACKNOWLEDGEMENT

It is great pleasure for us to express our gratitude to our honourable President Sri. Koneru Satyanarayana, for giving the opportunity and platform with facilities in accomplishing the project-based laboratory report.

We express the sincere gratitude to our principal Dr. K. Subbarao for his administration towards our academic growth.

We record it as my privilege to deeply thank our pioneer Dr.M. SUMAN, Prof & Head of the Department for providing us the efficient faculty and facilities to make our ideas into reality.

Finally, it is pleased to acknowledge the indebtedness to all those who devoted themselves directly or indirectly to make this project report success.

Batch-7

190040004 – A. Akhil

190040527 - T. Siddardha Rayudu

190040646 - J. V. Sri Lakshmi

190049031 – T. Eswar Abhishake

INDEX:

ABSTRACT

Crop diseases are a noteworthy risk to sustenance security, however their quick distinguishing proof stays troublesome in numerous parts of the world because of the non-attendance of the important foundation.

Emergence of accurate techniques in the field of leaf-based image classification has shown impressive results.

This project is mainly about verifying the performance of different machine learning algorithms in detecting the plant disease.

Here we are using a dataset of images of different leaves which are diseased and healthy from GitHub. The dataset consists of images of different leaves with and without diseases.

Now the machine learning algorithms like Logistic regression, Support vector machine, k-nearest neighbour, random forest classifier and naïve bayes are used to classify the healthy plants and diseased plants.

Accuracy of each classification algorithm will be verified in this project.

MACHINE LEARNING MODELS

1. Logistic Regression:

Logistic regression is a process of modelling the probability of a discrete outcome given an input variable. The most common logistic regression models a binary outcome; something that can take two values such as true/false, yes/no, and so on.

Binary variables are widely used in statistics to model the probability of a certain class or event taking place, such as the probability of a team winning, of a patient being healthy, etc. and the logistic model has been the most commonly used model for binary regression since about 1970.

Binary variables can be generalized to categorical variables when there are more than two values (e.g., whether an image is of a cat, dog, lion, etc.), and the binary logistic regression generalized to multinomial logistic regression. If the multiple categories are ordered, one can use the ordinal logistic regression (for example the proportional odds ordinal logistic model). See Extensions for further extensions.

The parameters of a logistic regression are most commonly estimated by maximum-likelihood estimation (MLE). This does not have a closed-form expression, unlike linear least squares; see § Model fitting. Logistic regression by MLE plays a similarly basic role for binary or categorical responses as linear regression by ordinary least squares (OLS) plays for scalar responses: it is a simple, well-analysed baseline model; see § Comparison with linear regression for discussion

2. K Nearest Neighbours:

K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.

K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.

K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.

K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data.

It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.

KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

Suppose, we have an image of a creature that looks similar to cat and dog, but we want to know either it is a cat or dog. So for this identification, we can use the KNN algorithm, as it works on a similarity measure. Our KNN model will find the similar features of the new data set to the cats and dogs images and based on the most similar features it will put it in either cat or dog category.

3. Random Forest:

Random forest is a Supervised Machine Learning Algorithm that is used widely in Classification and Regression problems. It builds decision trees on different samples and takes their majority vote for classification and average in case of regression.

As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

Since the random forest combines multiple trees to predict the class of the dataset, it is possible that some decision trees may predict the correct output, while others may not. But together, all the trees predict the correct output. Therefore, below are two assumptions for a better Random Forest classifier

There should be some actual values in the feature variable of the dataset so that the classifier can predict accurate results rather than a guessed result. The predictions from each tree must have extremely low correlations.

Below are some points that explain why we should use the Random Forest algorithm:

It takes less training time as compared to other algorithms.
It predicts output with high accuracy, even for the large dataset it runs efficiently.
It can also maintain accuracy when a large proportion of data is missing.

4. Naive Bayes:

Naive Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems. It is mainly used in text classification that includes a high-dimensional training dataset.

Naive It is called Naive because it assumes that the occurrence of a certain feature is independent of the occurrence of other features. Such as if the fruit is identified on the bases of color, shape, and taste, then red, spherical, and sweet fruit is recognized as an apple.

Hence each feature individually contributes to identify that it is an apple without depending on each other.Bayes It is called Bayes because it depends on the principle of Bayes' Theorem.

Bayes' Theorem:
Bayes' theorem is also known as Bayes' Rule or Bayes' law, which is used to determine the probability of a hypothesis with prior knowledge. It depends on the conditional probability.

5. Support Vector Machine:

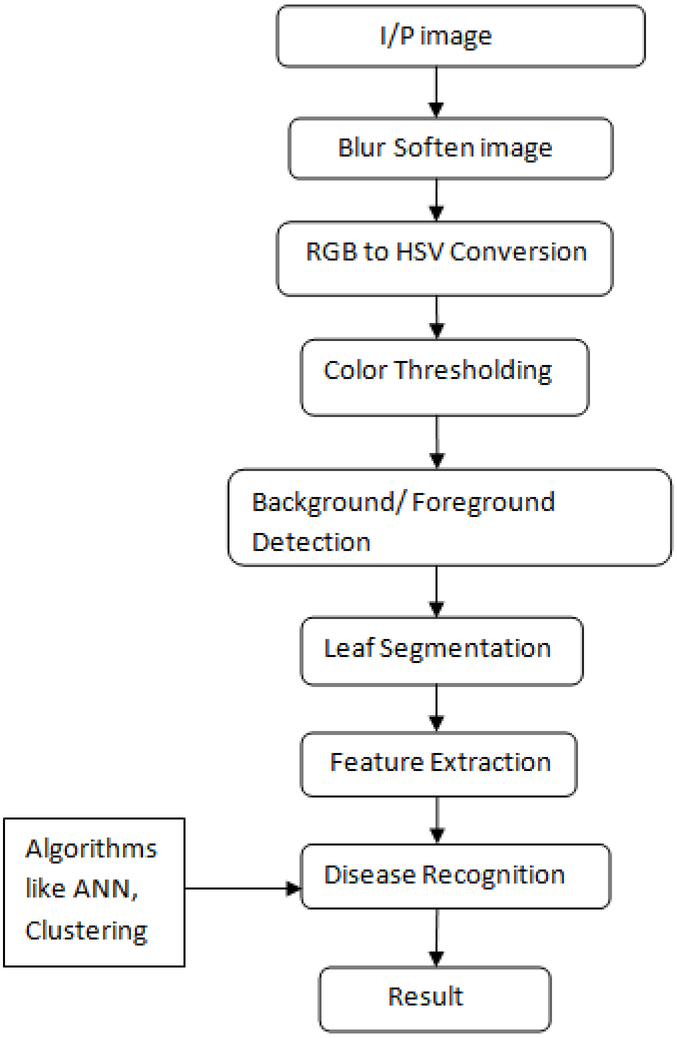
Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.

LIBRARIES USED:

- MAHOTAS
- CV2
- OS
- H5PY

FLOW CHART:



DATASET:

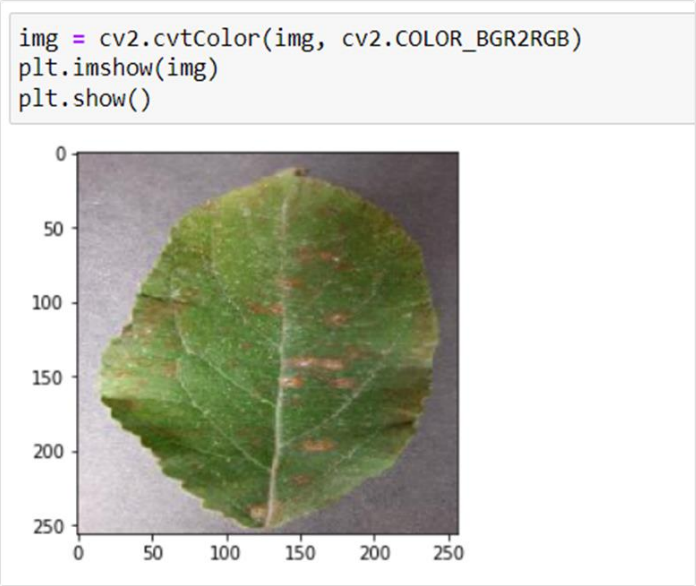
Dataset consists of images of diseased and healthy leaves in BGR format.

Link for the dataset:
https://github.com/spMohanty/PlantVillage-Dataset/tree/master/raw/color

STEPS INVOLVED:

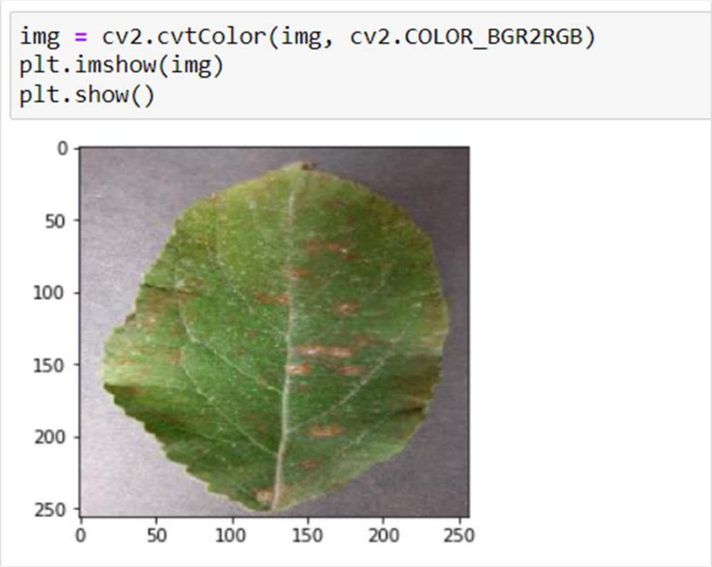
Load Original Image. A total of 800 images for each class Diseased and Healthy is fed for the machine.

Conversion of image from RGB to BGR. Since Open CV (python library for Image Processing), accepts images in RGB coloring format so it needs to be converted to the original format that is BGR format.



```
img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
plt.imshow(img)
plt.show()
```

BGR



```
img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
plt.imshow(img)
plt.show()
```

BGR to RGB

Conversion of image from BGR to HSV. The simple answer is that unlike RGB, HSV separates luma, or the image intensity, from chroma or the color information. This is very useful in many applications. For example, if you want to do histogram equalization of a color image, you probably want to do that only on the intensity component, and leave the color components alone. Otherwise you will get very strange colors. In computer vision you often want to separate color components from intensity for various reasons, such as robustness to lighting changes, or removing shadows. Note, however, that HSV is one of many color spaces that separate color from intensity (See YCbCr, Lab, etc.). HSV is often used simply because the code for converting between RGB and HSV is widely available and can also be easily implemented

Image Segmentation for extraction of Colors. In order to separate the picture of leaf from the background segmentation has to performed, The color of the leaf is extracted from the image.

Applying Global Feature Descriptor. Global features are extracted from the image using three feature descriptors namely :

Color : Color Channel Statistics (Mean, Standard Deviation) and Color Histogram

Shape : Hu Moments, Zernike Moments

Texture : Haralick Texture, Local Binary Patterns (LBP)

After extracting the feature of images the features are stacked together using numpy function “np.stack”.

According to the images situated in the folder the labels are encoded in numeric format for better understanding of the machine.

The Dataset is splitted into training and testing set with the ratio of 80/20 respectively.

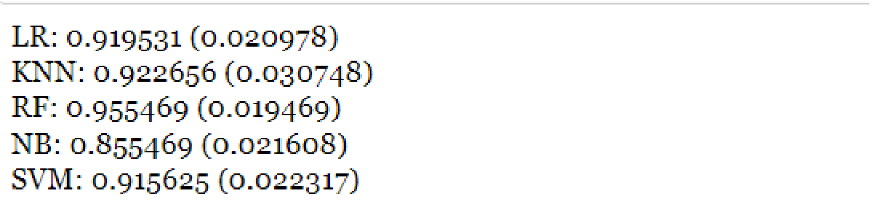
Feature Scaling Feature Scaling is a technique to standardize the independent features present in the data in a fixed range. It is performed during the data pre-processing to handle highly varying magnitudes or values or units. If feature scaling is not done, then a machine learning algorithm tends to weigh greater values, higher and consider smaller values as the lower values, regardless of the unit of the values.

Here, we have used Min-Max Scaler. This scaling brings the value between 0 and 1.

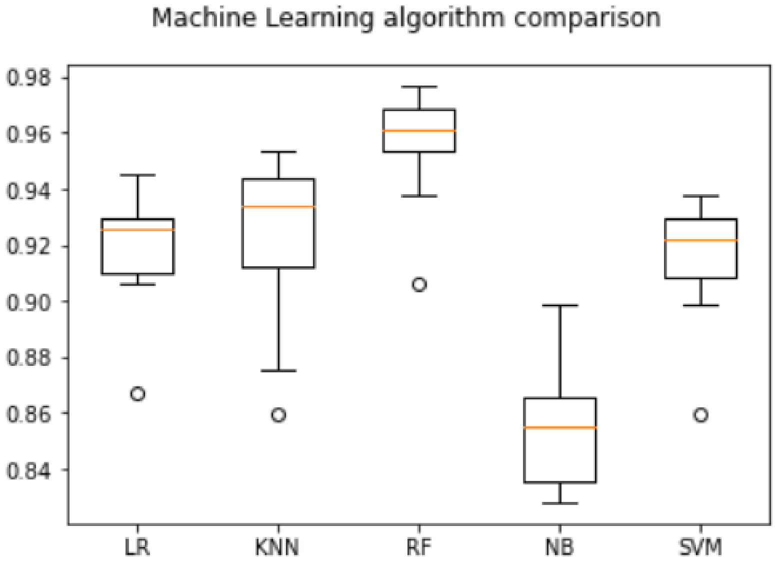
Saving the Features. After features are extracted from the images they are saved in HDF5 file. The Hierarchical Data Format version 5 (HDF5), is an open source file format that supports large, complex, heterogeneous data. HDF5 uses a "file directory" like structure that allows you to organize data within the file in many different structured ways, as you might do with files on your computer.

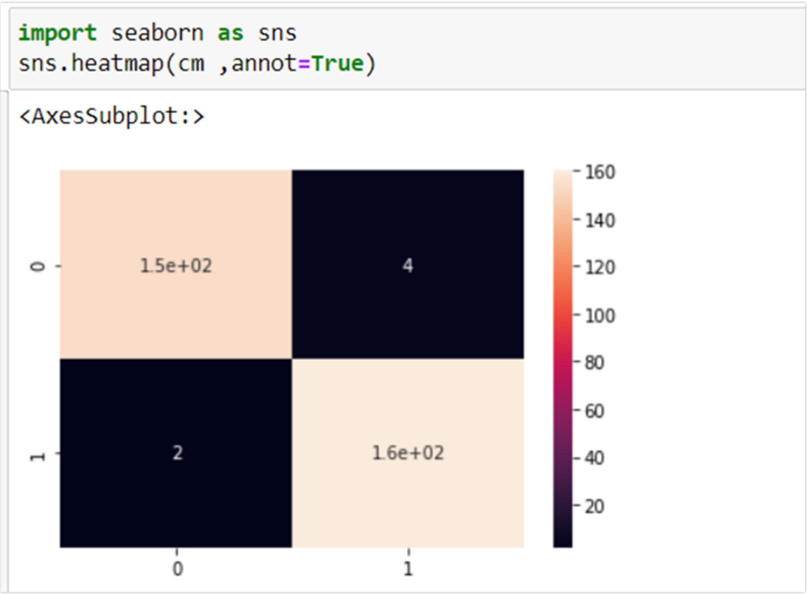
OUTPUT

Accuracy of each machine learning model



```
LR: 0.919531 (0.020978)
KNN: 0.922656 (0.030748)
RF: 0.955469 (0.019469)
NB: 0.855469 (0.021608)
SVM: 0.915625 (0.022317)
```





```
import seaborn as sns
sns.heatmap(cm ,annot=True)
```

```
<AxesSubplot:>
```

Conclusion

After testing the data with different classifiers, we get the accuracy results of each classifier.

Among all the classifiers Random Forest classifier best classifies the data.

Accuracy of Random Forest classifier is around 95%