DIAGNOSIS CLASSIFICATION OF SOYBEAN DISEASE USING MACHINE LEARNING TECHNIQUES

HNIN NWE PHYO

M.C.Sc. **SEPTEMBER 2022**

DIAGNOSIS CLASSIFICATION OF SOYBEAN DISEASE USING MACHINE LEARNING TECHNIQUES

BY

HNIN NWE PHYO

B.C.Sc.

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of

Master of Computer Science

(M.C.Sc.)

University of Computer Studies, Yangon

September 2022

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research and has not been submitted for a higher degree to any other University or Institution.

…..…… ……..……

Date Hnin Nwe Phyo

ACKNOWLEDGEMENTS

I would like to thank the Minister, Ministry of Science and Technology for full facilities support during the Master course at the University of Computer Studies, Yangon.

First of all, I would like to express very special thanks to **Prof. Dr. Mie Mie Khin**, Rector, the University of Computer Studies, Yangon, for allowing me to develop this thesis and giving me general guidance during the period of my study.

I would like to express my gratitude to **Prof. Dr. Thandar Thein**, Rector of the University of Computer Studies (Maubin) for allowing me to develop this thesis and giving me general guidance during the period of my study.

I would like to thank course coordinators **Dr. Si Si Mar Win** and **Dr. Tin Zar Thaw,** Professors of Faculty of Computer Science of the University of Computer Studies, Yangon, for their support, guidance, supervision, patience, and encouragement during the period of study towards completion of this thesis.

I would like to express my deepest gratitude to my supervisor, **Prof. Dr. Myo Khaing**, Faculty of Computer Science, Head of the Department of Career, the University of Computer Studies, Mandalay, for her patient supervision, tenderness, encouragement and providing me with excellent ideas throughout the study of this thesis. I will always remember her for being a mentor to me.

I also would like to express my gratitude and appreciation to **Daw Hnin Yee Aung**, Lecturer, Department of English, University of Computer Studies, Yangon, for her advice, editing and suggestion from the language point of view.

I would like to extend my special appreciation **Dr. Khin Swe Swe Myint**, Associate Professor and Head of Department of Faculty of Computer System and Technologies of the University of Computer Studies (Maubin) for the useful comments, sharing knowledge, giving advice, and insight which are invaluable to me.

I am very grateful to **all of my teachers** from the University of Computer Studies, Yangon and the University of Computer Studies (Maubin) who had been helping me from beginning to end of my thesis. I really appreciate for their valuable comments, suggestions, helpful hints, and fullest cooperation during the seminars of my thesis.

ABSTRACT

The prevention of disease transmission in plants is largely dependent on early detection of pathogen infection. Plant diseases can be identified using machine learning techniques before they fully manifest their symptoms. The more problems have been solved, the more reliable systems have been built. This system developed the agricultural field. Machine learning is a new area of study for agricultural analysis. Machine learning is a new area of study for agricultural analysis. The use of machine learning techniques in the sector of agriculture is the main topic of this study. Different machine learning techniques are in use, such as k-Nearest Neighbors (k-NN), J48 Decision Trees, Nave Bayes and Decision Table for very recent applications of data mining techniques in the agriculture field. This thesis properly classifies the problem of soybean diseases. For this purpose, different types of machine learning techniques were evaluated on soybean disease data sets. This thesis discusses the development of an expert system to diagnose soybean disease using machine learning techniques. This system implemented the K-folds cross validation method by using K value changes.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATION

CHAPTER 1

INTRODUCTION

Machine Learning is the study and development of algorithms that can extract information from a sample dataset and use that information to generate data-driven predictions or choices on fresh data. Machine Learning requires the development of computer systems that adjust or learn when presented with fresh data, it is comparable to data mining. Machine learning uses data to detect patterns in data and adjusts program actions as appropriate, whereas data mining extracts data for human comprehension.

Data or observations, direct experience, or instruction are all used in machine learning. Machine learning generally focuses on discovering ways to improve future performance based on current experiences. The goal is to create learning algorithms that can learn on their own, without having the need for human involvement. Machine learning creates the means for the computer to design its own program based on examples provided, as opposed to merely programming it to solve the problem. Artificial intelligence's fundamental subfield is machine learning. These things are otherwise impossible to complete [12].

1.1 Objective of the System

The main objectives of this thesis are:

- To build A Smart Agricultural Expert System for soybean Disease
- To support soybean farmers by technically
- To understand machine learning fundamentals
- To prevent reducing in the yield and quantity of the agricultural product
- To develop a smart agricultural expert system for soybean disease
- To support and help farmers in an efficient way
- To be correctly and accurately when classify diseases

1.2 Related Works

In thesis book, some references are used from previous proposes papers. R.S, Michalski and R.L. Chilausky described Comparison of Expert Derived and Inductively

Derived Rules the research was supported in part grant from the National Science Foundation and in part by grant [17].

Vinita Shah, Prachi Shah discussed Comparison of four different algorithms are used multiple linear Regression, Regression Tree, k-nearest neighbor, and artificial neural network. This system crop yield prediction is an important area of research, which helps in ensuring food security all around the world [21].

Minarni, Indra Warman, Yuhendra Teknik Informatika, Institut Teknologi Padang, Indonesia presented this study produces an expert system diagnoses the peanut diseases using case-based reasoning inference and nearest neighbor similarity [10].

Aditya Pratap Indian Institute of Pulses Research 136 PUBLICATIONS 1,901 CITATIONS, Ramesh Solanki Central Arid Zone Research Institute (CAZRI) 57 PUBLICATIONS 611 CITATIONS, Jitendra Kumar Indian Institute of Pulses Research 162 PUBLICATIONS 2,480 CITATIONS proposed soybean as a crop in detail covering all major aspects related to its history and domestication, cytogenetic, breeding behavior, genetic improvement as well as its oil and nutritional quality. Keeping in view the importance of soybean as a protein and oil rich crop, its ability to improve soil quality and its multifarious uses in domestic and industrial sectors, there are enough reasons to dedicate more research efforts for its genetic improvement [1].

1.3 Motivation of the System

Early detection of pathogens in plants is a key factor in reducing the spread of disease. Proper identification and early detection are important in the systematic management of soybean disease. In this system, soybean disease can be accurately identified, which will greatly support agriculture.

1.4 Organization of the System

This dissertation is organized into five chapters. Chapter 1 includes an Introduction, Objective of thesis, Related Work, Motivation of the Research. Chapter 2 includes the theoretical background of Machine Learning, Decision tree induction, Rule induction, Bayes' Theorem and k-nearest neighbor's algorithm (k-NN). The theoretical background of moving objects Soybean database structure and about of Decision tree induction, Rule induction, Bayes' Theorem and k-nearest neighbors' algorithm (k-NN) classification methods are discussed in Chapter 3. Chapter 4 presents Design and

Implementation of the system. Conclusion, Limitations and Further Extension of this thesis are presented in Chapter 5.

CHAPTER 2

THEORETICAL BACKGROUND

In essence, machine learning converts data into knowledge. It is impossible to get information or understanding from raw data by merely looking at it. For instance, the user cannot tell if an email is spam by looking at the frequency of a single phrase; instead, must look at the frequency of a number of words, the length of the email, and other criteria. Statistics are also employed in machine learning, which may be used to any problem requiring the interpretation of data and subsequent action. The facts discovered can then be utilized to choose a new collection of data. Static programs are typically employed to tackle deterministic issues with clear solutions, however for nondeterministic problems that lack sufficient data, apply a method known as machine learning, because there were insufficient datasets to train the algorithms on the beginning, it was challenging to use machine learning to make sound conclusions. However, with the rise of sensors and their capacity to connect to the Internet, the true challenge today is to effectively sort through the voluminous free data that is accessible and use it to train machine learning algorithms [16].

2.1 Introduction of the Machine Learning

The creation of efficient general-purpose algorithms is the main objective of machine learning research. In the context of learning, one should be concerned not just with time and space efficiency but also with the amount of data that the learning algorithm requires. Learning algorithms should solve issues in a general way that makes them easily applicable to a variety of learning challenges, like those mentioned above.

A prediction rule that is accurate when making predictions on new data should be the learning process' principal goal. Machine learning algorithms are data driven and capable of analyzing vast amounts of data, they have a major benefit over static programming in that the results are frequently more accurate with machine learning than with static programming results.

Figure 2.1 shows the general process involved in a typical machine learning model [4].

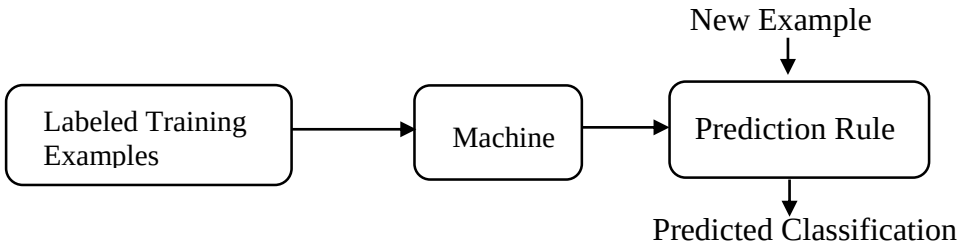

Figure 2.1. An illustration of a typical Machine Learning Process

2.2 Machine Learning Algorithms

Machine learning algorithms can model each problem differently depend on the input data. Based on their preferred learning styles, machine learning algorithms can be divided into four groups. They are

- Supervised Learning
- Unsupervised Learning
- Semi supervised Learning and
- Reinforcement Learning

Traditional classifications of learning techniques include supervised, unsupervised, semi-supervised, and reinforcement learning. The fields of signal processing, optimization, control, modeling and identification, and pattern recognition all make extensive use of supervised learning [6].

2.2.1 Supervised Learning

A criterion is required to determine when to end a learning process in order to control it. In supervised learning, the learning process is directed by an error measure that displays the discrepancy between the network output and the output from the training samples. Typically, the Mean Squared Error is used to determine the error measure (MSE).

When the MSE is sufficiently low or a failure criterion is satisfied, the learning process is over. A gradient-descent approach is actually used to reduce MSE until it is zero. The gradient-descent method consistently reaches a local minimum in close proximity to the initial network parameter solution. Two of the earliest and the most well-known supervised learning algorithms are the Least Mean Square (LMS) and back propagation techniques. The two of them [14].

2.2.2 Unsupervised Learning

Consider a machine (or living organism) which receives some sequence of inputs $x_1,x_2,x_3,\ldots$, where x_t is the sensory input at time t. This input, often called the data, could correspond to an image on the retina, the pixels in a camera, or a sound waveform. It could also correspond to less obviously sensory data, for example the words in a new story, or the list of items in a supermarket shopping basket. In supervised learning the machine is also given a sequence of desired output $y_1,y_2,\ldots$, and the goal of the machine is to learn to produce the correct output given a new input. This output could be a class label (in classification) or a real number (in regression).

In supervised learning the machine simply receives inputs $x_1,x_2,x_3,\ldots$, but obtains neither supervised target outputs, nor rewards from its environment. It may seem somewhat mysterious to imagine what the machine could possibly learn given that it doesn't get any feedback from its environment. On the basis of the idea that a machine's objective is to create representations of the input that can be utilized for decision-making, predicting future inputs, effectively transmitting the input to another machine, etc., a formal framework for unsupervised learning can be developed. In a sense, unsupervised can be thought of as finding patterns in the data above and beyond what would be considered pure unstructured noise. Two very simple classic examples of unsupervised learning are clustering and dimensionality reduction [2].

2.2.3 Semi-Supervised learning

Semi-supervised learning is a method of machine learning that, during training, blends a sizable amount of unlabeled data with a small amount of labeled data. Use small number of labeled data to label large amount of cheap unlabeled data. Basic idea: similar examples should be given the same classification. Between supervised learning (with labeled training data) and unsupervised learning is semi-supervised learning (with only labeled training data). Typical example: web page classification: unlimited amount of cheap unlabeled data, while labeling is expensive [18].

2.2.4 Reinforcement Learning

Through the use of reinforcement learning, robots can automatically decide how to behave in a given situation in order to function at their best. It needs only basic reward

feedback to learn the reinforcement signal behavior. The algorithm can be described as unsupervised learning when used with instances without labels. People can, however, add favorable or negative commentary to an example as suggested by the solution algorithm. Unlike unsupervised learning, reinforcement learning is tied to applications where the algorithm must make judgments, and those decisions have real-world effects. It is compared to learning by trial and error. One fascinating application of reinforcement learning is computers learning to play video games on their own. In this instance, an application provides instances of specific scenarios for the algorithm, such as a sequence of moves in a chess game. While attempting to prevent checkmate, the program informs the algorithm of the results of the acts it does. The chess algorithm increases its skill based on the number of games it played and the level of difficulty it encountered. This learning is a continuous improvement process [19].

2.3 Data Mining

Data mining, a branch of computer science and artificial intelligence, is the process of extracting patterns from data. Data mining is seen as an increasingly important tool by modern business to transform data into business intelligence giving an informational advantage. It is currently used in a wide range of areas, such as marketing, health, surveillance, fraud detection, and scientific discovery. This has been improved by other discoveries in computer science, such as neural networks, clustering, classification, genetic algorithms, decision tree and support vector machines. Hence data mining applies these methods to data with the intention of uncovering hidden patterns [8].

Data mining is the process of exploration and analysis, by automatic or semi-automatic means, of large quantities of data to discover meaningful patterns and rules. A typical data mining process includes data acquisition, data integration, data exploration, model building, and model validation. Both expert opinion and data mining techniques play an important role at each step of this information discovery process [9].

Data acquisition: The choice of the data types to be used is the first stage. Data mining can be carried out on a selection of variables or data samples in a larger database, even though a target data set has been generated for discovery in some applications.

Preprocessing Data: After the target data has been chosen, it is preprocessed to improve the efficiency of discovery by cleaning, scrubbing, and converting the data.

During this preprocessing step, programmers decide how to handle mission data fields and account for changes in time sequence information. They may also elect to eliminate noise or outliers if that is necessary. Additionally, the data is frequently changed to lower the actual number of variables being taken into account by either changing the kind of data (for example, turning categorical values into numerical ones) or creating new attributes (by applying mathematical or logical operators).

Data Exploration and Mode Building: The third stage of data mining includes a number of steps, including determining the type of data mining operation, the data mining techniques, the data mining algorithm, and data mining. The sort of data mining operation must be determined first. Classification, regression, segmentation, link analysis, and deviation detection are different types of data mining procedures. An appropriate data mining technique is then determined based on the operation selected for the application. The next step after selecting a data mining technique is to choose a specific algorithm for that technique. Determining which modes and parameters may be used when selecting a data mining algorithm includes a way to look for patterns in the data.

Interpretation and Evaluation: The analysis and assessment of found patterns is the fourth step in the data mining process. This work entails filtering the information to be presented by eliminating duplicate or unnecessary patterns, illustrating the useful ones graphically or logically, and translating them into terms that people can understand. The value of the knowledge that was extracted in relation to a decision-making indicator and a business objective is also assessed. The knowledge that has been collected is then applied to enhance human decision-making processes like prediction and to explain observed occurrences.

2.4 Data, Information, Knowledge, and Data Warehouses

Across a wide variety of fields, data are being collected and accumulated at a dramatic pace. There is an urgent need for a new generation of computational theories and tools to assist humans in extraction useful information (knowledge) from the rapidly growing volumes of digital data. A related field evolving from databases is data warehousing, which refers to the popular business trend of collecting and cleaning transactional data to make them available for online analysis and decision support. This

article begins by discussing in more details of data, information, knowledge, and data warehouse.

Data: Data are any information that a computer can process, including facts, figures, and language. Organizations are currently gathering enormous and expanding amounts of data in various databases.

Information: All of these data's patterns, associations, and interactions may include information. An analysis of retail point-of-sale transaction data, for instance, can reveal which products are selling and at what times.

Knowledge: Information can be transformed into knowledge about past trends and potential developments. For instance, summaries of supermarket sales data might be examined in the context of marketing initiatives to understand consumer purchasing patterns.

Data warehouses: Organizations are now able to merge their numerous databases into data warehouses thanks to dramatic improvements in data capture, computing power, data transport, and storage. Data warehousing is characterized as a method of centrally managing and retrieving data. Although the idea behind data warehousing has been around for years, the practice is relatively new, much like data mining. Maintaining a single location for all organizational data is the goal of data warehousing. In order to maximize user access and analysis, data must be centrally located. Users now have unrestricted access to this data in the data warehouses thanks to similarly impressive developments in data analysis tools.

2.5 Data Mining Function

To specify the type of pattern to be sought in data mining jobs, data mining functions are utilized. Data mining jobs can often be divided into two groups: descriptive and predictive. Inferring conclusions from the existing data is the function of descriptive mining tasks, which describe the general characteristics of the data in the database. Additionally, the user should be able to specify indications in the data mining system to focus or steer the search for intriguing patterns. The data mining functions and the variety of knowledge they discover are briefly presented in the following lists [11].

2.5.1 Association Analysis

Association analysis is the discovery of what are commonly called association rules. It studies the frequency of items occurring together in transactional database, and based on a threshold called support, identifies the frequent items sets. Another threshold, confidence which is the conditional probability than an item appears in a transaction when another item appears, is used to pinpoint association rules. Association analysis is commonly used for market basket analysis.

2.5.2 Classification

Classification means given input values must be predicted their corresponding output. For example, the users have inputted value as patient record attributes are blood sugar level, blood pressure and so on and from that record they have to predict whether patient is suffering from any disease or not. They are predicting whether the person has disease or not.

2.5.3 Prediction

Prediction has attracted considerable attention given the potential implications of successful forecasting in a business context. There are two major types of predictions: one can either try to predict some unavailable data values or pending trends or predict a class able for some data. The latter is tied to classification. Once a classification model is built based on a training set, the class label of an object can be foreseen based on the attribute values of the object and the attribute values of the classes. Prediction is however more often referred to the forecast of missing numerical values, or increase/ decrease trends in time related data. The major idea is to use a large number of past values to consider probable future values.

2.5.4 Clustering

Clustering means grouping similar data together. The users are given marks of various students and they have to group brilliant students and average students then they can apply clustering algorithm to group them together. Another example would be grouping different species of animal together.

2.5.5 Outlier Analysis

Data items that are outliers cannot be categorized into a specific class or cluster. They are also referred to as exceptions or monitored, and they are frequently crucial to recognize. In certain applications, outliers may be dismissed as noise, while in others, they may provide crucial information, making their significance and study vital.

2.5.6 Evaluation and Deviation Analysis

The examination of time-related data that varies over time is relevant to evaluation deviation analysis. Evaluation analysis simulates evolutionary tendencies in the data, which permits characterizing time-related data comparison, classification, or grouping.

On the other hand, deviation analysis takes into account discrepancies between measured values and expected values and tries to identify what is causing the deviations from the expected values. [13].

2.6 Similarity Measure

Since clustering is the grouping of similar objects, some sorts of measures that can determine whether two objects are similar or dissimilar are required.

Formally, the distance d(x,y) between any two data objects x and y is considered to be a two argument function satisfying the following conditions:

$$d(x,y) \geq 0 \text{ for every x and y} \quad (2.1)$$

$$d(x,y) = 0 \text{ for every x} \quad (2.2)$$

$$d(x,y) = d(y,x) \quad (2.3)$$

The most popular distance measures are described in Table 2.1.

Table 2.1 Popular Distance Measure

Distance Measures	Formula
Euclidean distance	$d_{euc}(x,y) = \sqrt{\sum_{i=1}^{n} (x_{1i}^{\ 2} - x_{2i}^{2})}$
Manhattan distance	$d_{man}(x,y) = \sum_{i=1}^{n} \mid x_{1i} - x_{2i} \mid$
Minkowski distance	$d_{min}(x,y) = (\sum_{i=1}^{n} (\mid x_{1i} - x_{2i} \mid)^{q})^{1/q}$
Average distance	$d_{ave}(x,y) = (\frac{1}{n} \sum_{i=1}^{n} (x_{1i} - x_{2i})^{2})^{1/2}$

2.7 Performance Metric

Accuracy: This is the most straightforward score factor. It determines the percentage of cases that are correctly categorized.

$$\text{Accurecy} = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad 2.4$$

Sensitivity (also known as Recall or True Positive Rate): Sensitivity is the percentage of true positives that the classifier properly classifies as true positives.

$$\text{Sensitivity} = \frac{TP}{(TP+FN)} \quad 2.5$$

Specificity, also known as True Negative Rate, refers to the classifier's capacity to recognize unfavorable outcomes. Think about a medical test that is meant to pinpoint a certain ailment. The percentage of people that successfully test negative for the disease and do not have it is the specificity of the test.

$$\text{Specificity} = \frac{TN}{(TN+FP)} \quad 2.6$$

Precision: This is a measure of retrieved instances that are relevant.

$$\text{Precision} = \frac{TP}{(TP+FP)} \quad 2.7$$

2.8 Cross Validation

Cross validation is another well-liked set of procedures used in model selection in addition to Information Criteria. Partitioning the data into subsets for training and testing is the standard procedure. While testing is the process of validating the fitted model by calculating the prediction error, training is the process of fitting a model.

2.8.1 Leave-One-Out Cross Validation

The procedure utilized when k = 1 is used in the aforementioned formulation, in detail due to a number of significant linkages. In this instance, the model is trained and tested using each of the I = 1,..., n possible partitions, with our test set always having cardinality 1. Let -i represent the expected value of the missing observation for each i. The cross validation estimate of error with leave-one-out is

$$\text{LOOCV} = \sum_{i=1}^{n}(y_i - y_i^{-i})^2 \qquad 2.8$$

2.8.2 Generalized Cross Validation

The generalized cross validation criterion (GCV) (Wahba, 1990) is an approximation to the LOOCV and follows from noting that tr(H) = $\sum_{i=1}^{n}(h_{ii})$followed by the approximation: $h_{ii} \approx \frac{1}{n}$ tr(H) . This is generally applicable when fitting linear methods with quadratic loss function and is a good approximation provided, h_{ii}, i = 1,…,n are not very different (Wahba, 1990). The generalized cross validation statistic becomes:

$$GCV = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{1-\frac{1}{n}tr(H)}\right)^2 \qquad 2.9$$

2.8.3 Leave-K-Out Cross Validation

In the formulation at the start of this chapter, leave-k-out cross validation (LKOCV) is the general case where the size of the test set | D | = k. As mentioned earlier, this procedure carries considerable 11 computational expense due to the $\binom{n}{k}$ possible partitions that must be left-out and is rarely used in practice.

2.8.4 K-Fold Cross Validation

An alternative procedure is K-fold cross validation and this procedure was motivated by computational expense of the leave-one-out procedure. The K-fold procedure is attractive because it balances computational cost with an increase in the estimation bias. In this procedure, the dataset D is divided into K partitions of roughly equal size, $D = \bigcup_{k=1}^{k} D_k$, and each partition is termed a "fold" of the dataset (thus there are K folds). The procedure may be understood as a leave-one-foldout procedure in analogy to the leave-one-out procedure. The model is trained on K −1 folds and the K fold is used for testing.

$$KCV = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \hat{y}^{-k(i)}\right)^2 \qquad 2.10$$

2.8.5 Choosing a Cross Validation Method

The bias and variance tradeoff affects the choice of cross validation approach, just like it does with any model selection process. As was already noted, LOOCV is asymptotically impartial. Due to some significant outcomes in, the size of the training set is a factor in our decision.

CHAPTER 3

THE PROPOSED SYSTEM ARCHITECTURE

Machine learning has developed into one of the pillars of information technology during the past two decades, becoming a significant but unnoticed aspect of our daily lives. Smart analysis is required due to the daily growth in the amount of data collected (and kept) by both individuals and businesses. Here is where machine learning takes center stage as a crucial component for technological advancement. Learning to run our experiments, the users used supervised machine learning techniques. They also made use of supervised machine learning techniques to assist in some of the research queries. Machine learning algorithms there are numerous machine learning algorithms to choose from, depending on the problem. In this chapter, supervised machine learning algorithms are presented that are commonly used for text classification.

3.1 Supervised Learning

A direct comparison between the actual network output and the anticipated output is the foundation of supervised learning. The training pattern set and the accompanying errors between the expected output and the actual network response are used to change the network parameters. The error serves as the feedback signal in a closed-loop feedback system known as supervised earning. The system is modeled using the trained network.

The test dataset is used to assess the algorithm's effectiveness after it has been trained using the training dataset. Next, a procedure known as cross-validation is employed to compare various feature selection, dimensionality reduction, and learning algorithm combinations. The most popular one is k-fold cross-validation, which involves splitting the training dataset into k subsets (k-1 subset is used for training and 1 subset can be used for testing). This splitting can assist in estimating the average error rate once the learning process is completed. Since each feature can have a distinct range of values when learning and making judgments, normalization is done to give equal weight to every feature in the dataset. It must be done on both training and test data. There are numerous learning algorithms in this work. In the next sections k-Nearest Neighbor, J48, Decision tables, and Naive Bayes will discuss. In the post-processing phase, they would test the algorithm's accuracy using test data. If the accuracy didn't

meet our expectations, they could always restart the process by giving the algorithm access to more abundant and accurate data. They can also improve the way the user gather and prepare their input datasets to get better results. The user can use the algorithm to forecast actual data once it gets the anticipated accuracy [22].

There are two types of algorithms for supervised learning. They are

- Regression and
- Classification

3.1.1 Advantages of Supervised Learning

Supervised learning allows collecting data and produces data output from previous experiences. The usages of supervised learning experience to assist optimize performance criteria. Supervised machine learning helps to solve various types of real-world computation problems.

3.1.2 Disadvantages of Supervised Learning

Supervised learning can be challenging when it comes to big data classification. The computing time required for training supervised learning is significant. Unwanted data reduces productivity. Pre-processing data presents a significant hurdle. Constantly in need of updating supervised algorithms are easily over fit by anyone.

3.2 Regression

Finding correlations between dependent and independent variables is the process of regression. It aids in the forecast of continuous variables like market trends, house values, and other things.

Regression Algorithm Types are

- Simple Linear Regression
- Multiple Linear Regression
- Polynomial Regression
- Support Vector Regression
- Decision Tree Regression and
- Random Forest Regression

3.3 Classification

Finding a function to divide the dataset into classes based on several parameters is the process of classification. Putting something or someone into a certain group or system based on certain criteria is the definition of classifying. In classification, data is divided into various classes by a computer program that has been trained on the training dataset [15].**Example:** Email spam detection is the ideal illustration of the Classification issue. Every time a new email is received, the model determines whether it is spam or not based on training data from millions of emails on various parameters. The email gets placed in the Spam folder if it is considered spam.

ML Classification Algorithm Types:

The following categories of classification algorithms exist. They are

- Logistic Regression
- K-Nearest Neighbors
- Support Vector Machines
- Kernel SVM
- Naïve Bayes classification
- Decision Tree Classification
- Random Forest Classification
- Evaluation of classifiers
- Rule induction
- Classification using association rule
- Naïve Bayesian Naïve Bayes for text classification and
- Ensemble methods : Bagging Boosting

This thesis, four important classification algorithms are

- k-Nearest Neighbors
- J48 Decision Tree
- Naïve Bayes and
- Decision Table

3.4 k-Nearest Neighbor

In this section, the first classification algorithm will be discussed: k-Nearest Neighbors. Compared to other machine learning methods, it is simple to comprehend and straightforward to implement. The letter "K" stands for the number of closest

neighbors to a new unknown variable that needs to be forecasted or categorized. The k-nearest neighbors (k-NN) algorithm is a simple, supervised machine learning algorithm that can be used to solve both classification and regression problems [5].

3.4.1 Advantages of k-Nearest Neighbor

The k-NN algorithm does not require training prior to produce predictions, the new data can be added without affecting the system's accuracy. Implementing k-NN is fairly simple. k-NN implementation just needs two parameters: the value of K and the distance function (e.g. Euclidean or Manhattan etc.). The main importance of using k-NN is that it's easy to implement and works well with small datasets.

3.4.2 Disadvantage of k-Nearest Neighbor

k-NN does not perform well with large dataset: In large datasets, the algorithm's speed suffers due to the high cost of computing the distance between each new point and each current point.

k-NN Does not perform well with high dimensional data: The k-NN algorithm performs poorly with high dimensional data because it becomes challenging for the algorithm to calculate the distance in each dimension as the number of dimensions increases.

k-NN Feature scaling is required: Before applying the KNN method to any dataset, feature scaling (standardization and normalization) must be completed. In the absence of this, KNN could produce inaccurate predictions.

3.5 Decision Trees

In this section, the next classification algorithm will be discussed: Decision Trees. It is the most commonly used machine learning technique. This section will start with an explanation about background of this algorithm followed by a flow chart which explains step wise process involved [3].

3.5.1 J48 Decision Tree

A decision tree is a supervised learning technique that has a pre-defined target variable and is often used in classification problems. Building a model of classes from a set of records that have class labels is the process of classification. Decision Tree

Algorithm is to find out the way the attributes vector behaves for a number of instances. The classes for the freshly generated instances are also being discovered on the basis of the training instances. The rules for the target variable's prediction are generated by this algorithm. With the help of tree classification algorithm, the critical distribution of the data is easily understandable. An extension of ID3 is J48. Accounting for missing values, decision tree pruning, continuous attribute value ranges, the development of rules, etc. are further characteristics of J48.

3.5.2 Advantages of J48 Decision Tree

The main benefits of decision trees are that they are computationally affordable and that people can readily interpret the data. Additionally, the construction of a decision tree is not significantly impacted by missing values in the data. Technical teams and stakeholders may easily understand a decision tree model because it is so simple.

3.5.3 Disadvantages of J48 Decision Tree

The decision tree's structure can drastically change in response to little changes in the data, which might cause instability. Sometimes calculations for a decision tree can be significantly more complicated than for other methods. Decision trees usually require more time throughout the model training process.

3.6 Naïve Bayes Classification

Naive Bayesian classifier is based on Bayes' theorem with the independence assumptions between predictors. A Naive Bayesian model is simple to construct and does not require time-consuming iterative parameter estimation, making it especially beneficial for very large datasets. They would make the best prediction and give it a probability under the Naive Bayes model [7].

1. Let D be a training set of tuples and their associated class labels. Each tuple is represented by an n-dimensional attribute vector, $X = (x_1, x_2, \ldots\ldots\ldots, x_n)$.Depicting n measurements made on the tuples from n attributes , respectively, $A_1, A_2, \ldots\ldots\ldots, A_n$, here xn refers to the value of attribute An for tuple X.

2. Suppose, m is a classes, variable C1, $C_2, \ldots\ldots, C_m$ .Given a tuple, X , the classifier will predict that X belongs to the class having the highest posterior probability , conditioned on X. Predicts that tuple X belongs to the class Ci if and only if

$$P(Ci|X) > P(Cj|X) \text{ for } 1 \leq j \leq m, j \neq i \quad 3.1$$

The class Ci for which P(Ci|X) is maximized is called the maximum posteriori hypothesis

$$P(Ci|X) = \frac{P(X|Ci) \,.\, P\,(Ci)}{P\,(X)} \quad 3.2$$

3. P(X) is constant for all classes, only P(X|Ci).P(Ci) needs to be maximized. It is commonly assumed that the classes are equally likely, that is P(C1) = P(C2) = … = P(Cm) therefore maximize P(X|Ci). Otherwise maximize P(X|Ci) . P(Ci)

4. Given data sets with attributes, extremely computationally expensive to compute P(X|Ci).

To reduce computation in evaluating P(X|Ci) the naïve assumption of class-conditional independence is made.

$$P(X|Ci) = \prod_{k=1}^{n} P\,(x_k|Ci) \quad 3.3$$

$$P(x_1|Ci) \times P(x_2|Ci) \times \ldots\ldots\ldots\ldots .. P(x_n|Ci) \quad 3.4$$

The probability $P(x_1|Ci)$, $P(x_2|Ci)$,…, $P(x_n|Ci)$ from the training tuple. Recall that here x_k refers to the value of attribute Ak for tuple X.

5. To predict the class label of X, P(X|Ci) P(Ci) is evaluated for each class Ci .The classifier predicts that the class label of tuple X is the class Ci if and only if

$$P(X|Ci)\, P(Ci) > P(X|Cj)\, P(Cj) \text{ for } 1 \leq j \leq m\,,\, j \neq i \quad 3.5$$

The predicted class label is the class Ci for which P(X|Ci) P(Ci) is the maximum.

3.6.1 Advantages of Naïve Bayes Classification

The main benefit of Naive Bayes' is that it can accommodate several classes and work with less data. Evaluation of the conditional probability is simple. Naive Bayes' is very quick - no iterations necessary because the probabilities may be calculated directly. Therefore, this method is helpful in situations where training speed is crucial.

3.6.2 Disadvantages of Naïve Bayes Classification

Naive Bayes assumes that all predictors (or features) are independent, rarely happening in real life. If the test data set has a categorical variable of a category, it is

not present in the training data set. The Naive Bayes model will assign it zero probability and will not be able to make any predictions in this regard.

3.7 Rule Induction

Rule induction is a data mining process of deducing if-then rules from a data set. These symbolic decision rules explain an inherent relationship between the attributes and class labels in the data set [31].

Rule-based classifiers, where the learned model is represented as a set of IF-THEN rules.

Using IF-THEN Rules for Classification

An IF-THEN rule is an expression of the form

IF condition THEN conclusion.

3.7.1 Decision Table

Decision tables are used to model complicated programming logic. They can make it simple to understand that all potential conditions have been taken into account. The decision table are composed of 4 parts: conditions, actions, condition alternatives and actions for the rules. A decision table that includes every conditional statement. Decision tables are used to lay out in tabular form all possible situations which a business decision may encounter. A matrix of causes and effects is listed in a decision table. A different combination is represented by each column. The goal of decision tables to structure logic.

3.7.2 Advantages of Decision Table

Decision tables are very helpful in test design technique. It helps testers to search the effects of combinations of different inputs and other software states that implement business rules. It provides a regular way of stating complex business rules which benefits the developers as well as the testers.

3.7.3 Disadvantages of Decision Table

The decision table is not equivalent to complete test cases containing set-by-step instructions of what to do in what order. If the user has a lot of combination, it may not possible or sensible to test every combination.

3.8 K-Fold Cross Validation

One of the most prominent techniques frequently employed by data scientists is k-fold cross-validation. It is a method of data partitioning that enables you to make the most of the user dataset while creating a more comprehensive model. Any type of machine learning has as its major goal the creation of a broader model that can function well with unknown input. On the training data, a perfect model can be created with 100% accuracy or 0 errors, but it might not generalize to new data. As a result, it is a poor model. The training data are over fit by it. Machine learning is all about generalization, hence the performance of the model can only be evaluated using data points that were not utilized in the training phase. This system implemented the K-folds cross validation method by using K value changes (k= 2 3 4 5).

3.9 Datasets for four Methods Comparison

This soybean disease dataset was obtained from the R. S. Michal ski's research while affiliated with University of Illinois, (Donor: Ming Tan & Jeff Schlimmer (Jeff.Schlimmer%cs.cmu.edu)). The soybean dataset has 683 instances, 35 attributes and 19 classes [31].

3.9.1 Data Description

The soybean disease dataset which has 638 instances, 35 attributes and 19 classes.

Table 3.1 Sample attributes of Soybean Disease Dataset

date	plant-stand	precip	temp	hail	crop-hist	area-damaged	severity
october	normal	gt-norm	norm	yes	same-lst-yr	low-areas	pot-severe
august	normal	gt-norm	norm	yes	same-lst-two-yrs	scattered	severe
July	normal	gt-norm	norm	yes	same-lst-yr	scattered	severe
July	normal	gt-norm	norm	yes	same-lst-yr	scattered	severe
october	normal	gt-norm	norm	yes	same-lst-two-yrs	scattered	pot-severe
september	normal	gt-norm	norm	yes	same-lst-sev-yrs	scattered	pot-severe
september	normal	gt-norm	norm	yes	same-lst-two-yrs	scattered	pot-severe
august	normal	gt-norm	norm	no	same-lst-yr	scattered	pot-severe
october	normal	gt-norm	norm	yes	same-lst-sev-yrs	scattered	pot-severe
august	normal	gt-norm	norm	yes	same-lst-two-yrs	scattered	severe
october	normal	lt-norm	gt-norm	yes	same-lst-yr	whole-field	pot-severe
august	normal	lt-norm	norm	no	same-lst-yr	whole-field	pot-severe
July	normal	lt-norm	norm	yes	same-lst-yr	upper-areas	pot-severe
october	normal	lt-norm	norm	no	same-lst-sev-yrs	whole-field	pot-severe
october	normal	lt-norm	gt-norm	yes	same-lst-yr	whole-field	pot-severe
september	normal	lt-norm	gt-norm	no	same-lst-sev-yrs	whole-field	pot-severe
october	normal	lt-norm	gt-norm	no	diff-lst-year	upper-areas	pot-severe
october	normal	lt-norm	gt-norm	yes	same-lst-yr	whole-field	pot-severe
august	normal	lt-norm	norm	no	same-lst-yr	whole-field	pot-severe
July	normal	lt-norm	norm	yes	same-lst-yr	upper-areas	pot-severe
october	normal	lt-norm	norm	no	same-lst-sev-yrs	whole-field	pot-severe
october	normal	lt-norm	gt-norm	yes	same-lst-yr	whole-field	pot-severe
september	normal	lt-norm	gt-norm	no	same-lst-sev-yrs	whole-field	pot-severe
october	normal	lt-norm	gt-norm	no	diff-lst-year	upper-areas	pot-severe
october	normal	lt-norm	gt-norm	yes	same-lst-yr	whole-field	pot-severe

The attributes are date, plant-stand, precip, temp, hail, crop-hist , area-damaged, severity, seed-tmt, germination, plant, leaves, leafspot-halo, leafspot-marg, leafspot-size, leaf-shread, leaf-malf, leaf-mild, stem, lodging, stem-cankers, canker-lesion, fruiting-bodies, external decay, mycelium**,** int-discolor, sclerotia, fruit-pods, fruit spots, seed, mold-growth, seed-discolor, seed-size, shriveling and roots. The types of the diseases are diaporthe-stem-canker**,** charcoal-rot, rhizoctonia-root-rot, phytophthora-rot, brown-stem-rot, powdery-mildew, downy-mildew, brown-spot, bacterial-blight,

bacterial-pustule, purple-seed-stain, anthracnose, phyllosticta-leafspot, alternarialeaf-spot, frog-eye-leafspot, diaporthe-pod-&-stem-blight, cyst-nematode, 2-4-d-injury and herbicide-injury.

Table 3.2 Interface for Preprocessing

No	Features Name	Description	
1	date	october	0
		auguest	1
		july	2
		september	3
		may	4
		june	5
		april	6
2	plant-stand	normal	0
		It-normal	1
3	precip	gt-norm	0
		lt-norm	1
		norm	2
4	temp	norm	0
		gt-norm	1
		lt-norm	2
5	hail	yes	0
		no	1
6	crop-hist	same-lst-yr	0
		same-lst-two-yrs	1
		same-lst-sev-yrs	2
		diff-lst-year	3
7	area-damaged	low-areas	0
		low-areas	1
		scattered	2
		whole-field	3
		upper-areas	4
8	severity	pot-severe	0
		severe	1
		minor	2
9	seed-tmt	none	0
		fungicide	1
		other	2

No	Features Name	Description	
10	germination	90-100	0
		80-89	1
		lt-80	2
11	plant-growth	abnorm	0
		norm	1
12	leaves	abnorm	0
		norm	1
13	leafspots-halo	absent	0
		no-yellow-halos	1
		yellow-halos	2
14	leafspots-marg	dna	0
		w-s-marg	1
		no-w-s-marg	2
15	leafspot-size	dna	0
		w-s-marg	1
		no-w-s-marg	2
16	leaf-shread	absent	0
		present	1
17	leaf-malf	absent	0
		present	1
18	leaf-mild	absent	0
		upper-surf	1
		lower-surf	2
19	stem	abnorm	0
		norm	1
20	lodging	no	0
		yes	1
21	stem-cankers	above-sec-nde	0
		absent	1
		below-soil	2
		above-soil	3
22	canker-lesion	brown	0
		dna	1
		tan	2
		dk-brown-blk	3
23	fruiting-bodies	present	0
		absent	1

No	Features Name	Description	
24	external-decay	firm-and-dry	0
		absent	1
		watery	2
25	mycelium	absent	0
		present	1
26	int-discolor	none	0
		black	1
		brown	2
27	sclerotia	absent	0
		present	1
28	fruit-pods	norm	0
		dna	1
		diseased	2
		few-present	3
29	fruit-spots	dna	0
		absent	1
		colored	2
		brown-w/blk-specks	3
30	seed	norm	0
		abnorm	1
31	mold-growth	absent	0
		present	1
32	seed-discolor	absent	0
		present	1
33	seed-size	norm	0
		lt-norm	1
34	shriveling	absent	0
		present	1
35	roots	norm	0
		rotted	1
		galls-cysts	2

CHAPTER 4

SYSTEM DESIGN AND IMPLEMENTATION

This chapter describes the detail of system design and implementation. In the system implementation, the characteristics of datasets classification are described. Then the comparison of performance of k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table is described. Finally, the result of this system is expressed in term of user interface.

4.1 Overview of the System

The overview of the system is illustrated in figure 4.1.

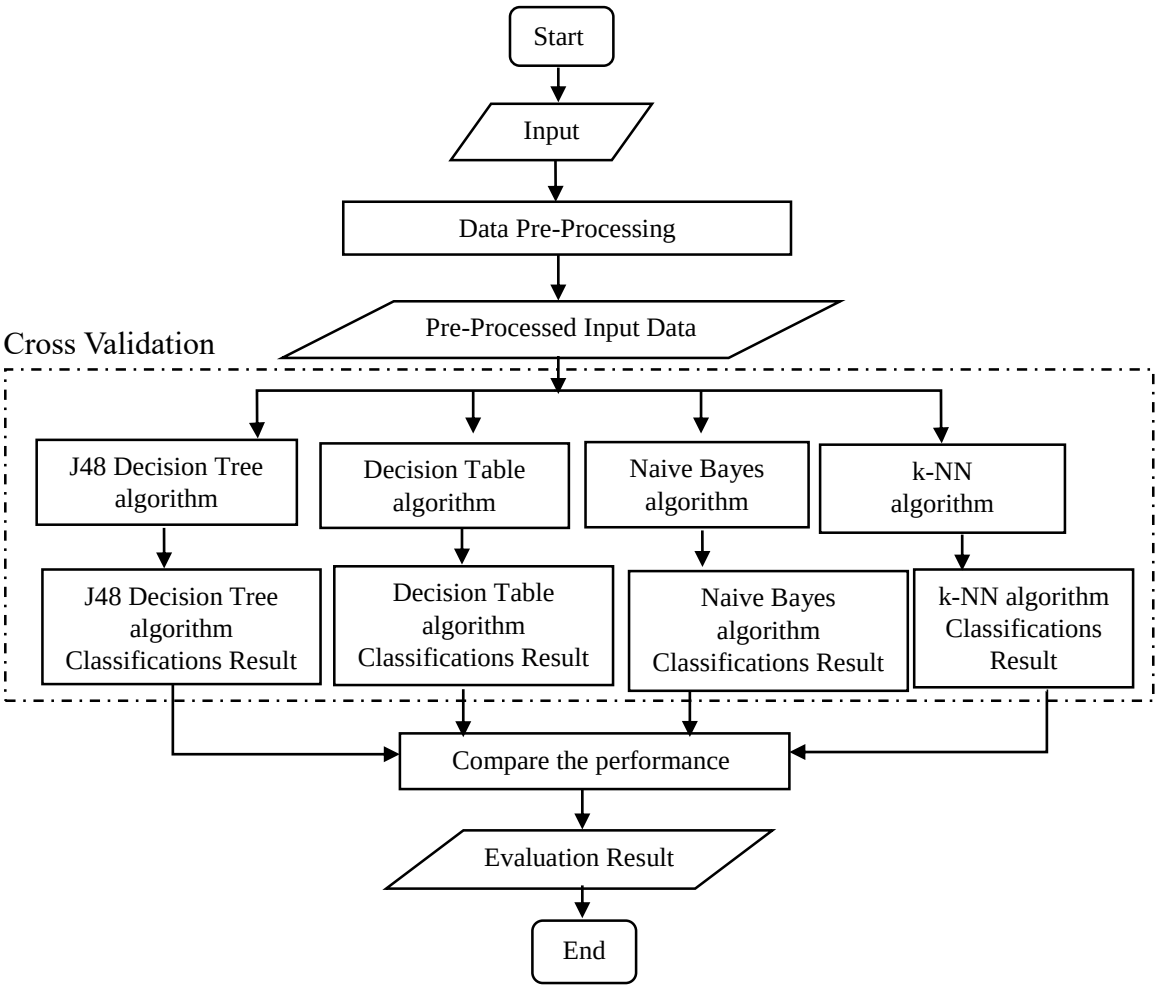

Figure 4.1 Overview of the System

The overview system is to compare k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table classification algorithm based on classification of datasets such as soybean disease. Moreover, the performance of each method with each dataset is also compared and analyzed.

Data preprocessing stage of in this system have four steps. Step one involves importing the row dataset. Step two verify the values which are missing. View the categorical values in step three. Dividing the data set into a training and test set in step four.

Data normalization is generally considered the development of clean data. It increases the cohesion of entry types leading to cleansing, lead generation, segmentation and higher quality data. This thesis, data are normalized by data normalization process.

4.2 Implementation of the System

The performance of k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table classification methods are compared and analyzed in term of execution time by using soybean disease dataset. In this system, datasets are obtained from UCI (University of California Irvine) Machine learning repository [20].

4.3 User Interface

This section describes the user interface of the system implementation.

4.3.1 Main Form

There are two menus in menu bar: (i) Classification and (ii) about. Figure 4.2 describes the main form of the system.

Figure 4.2 shows main frame of the system.

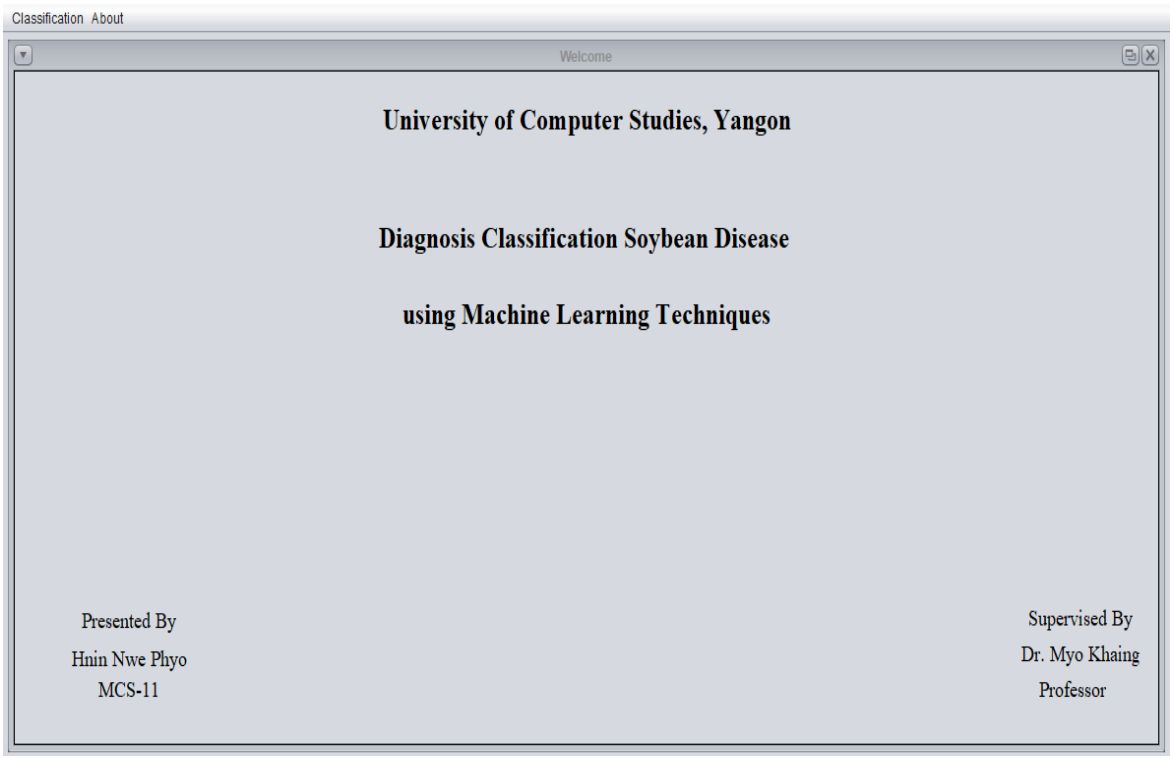

Figure 4.2 Main Frame of the System

4.3.2 Interfacing with Import Dataset

Figure 4.3 shows interface of import dataset.

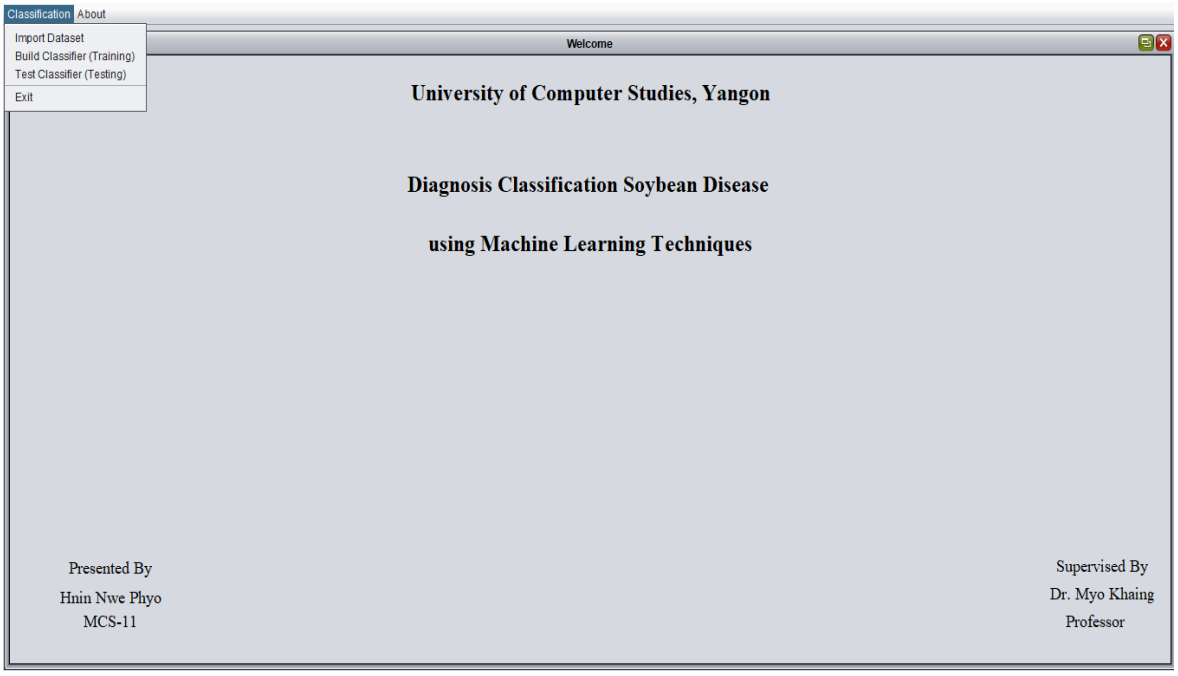

Figure 4.3 Interface of Import Dataset

4.3.3 Interfacing with Choose Dataset

Figure 4.4 shows choose dataset

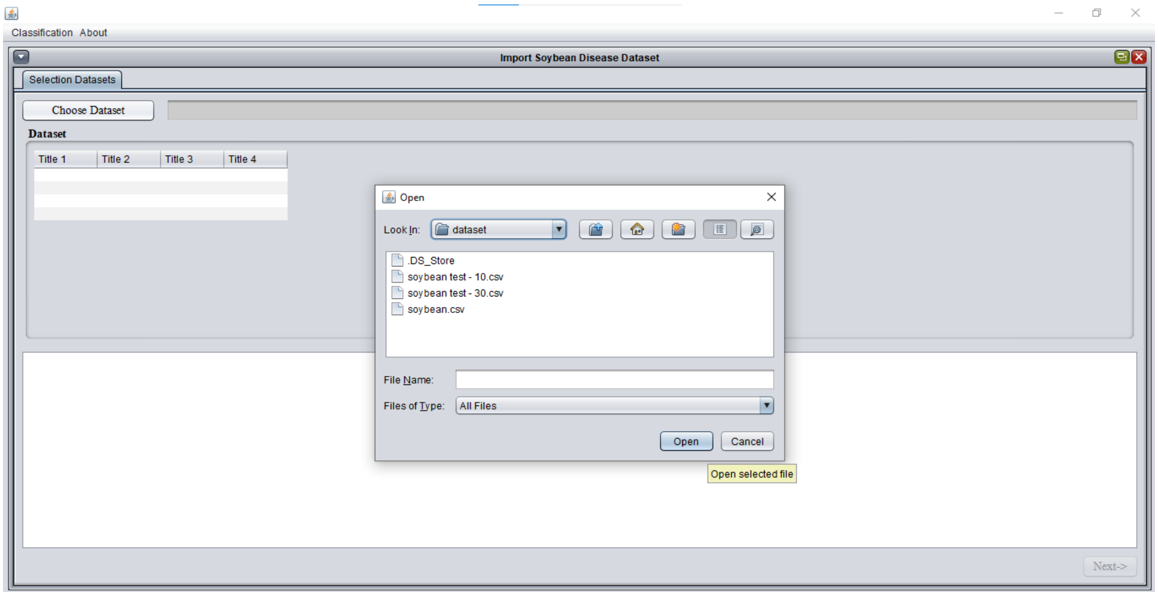

Figure 4.4 Interface of Choose Dataset

Figure 4.5 shows the soybean disease dataset which has 10 instances, 35 attributes and 19 classes.

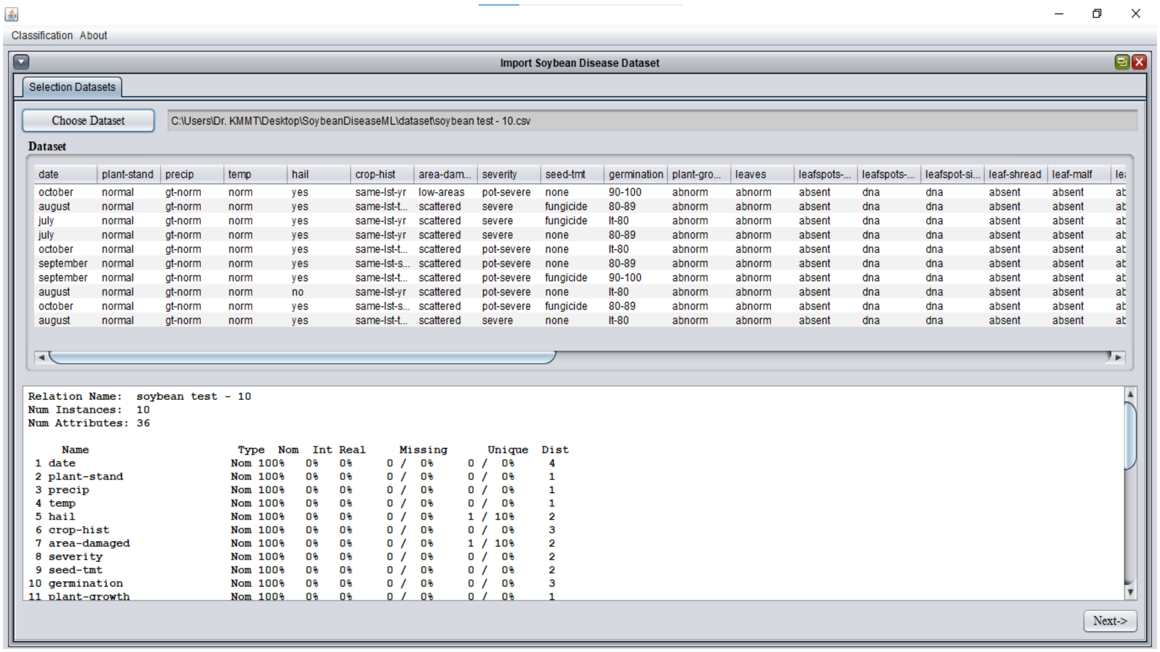

Figure 4.5 Interface of Soybean Disease test - 20 Dataset soybean

Figure 4.6 shows the soybean disease dataset which has 30 instances, 35 attributes and 19 classes.

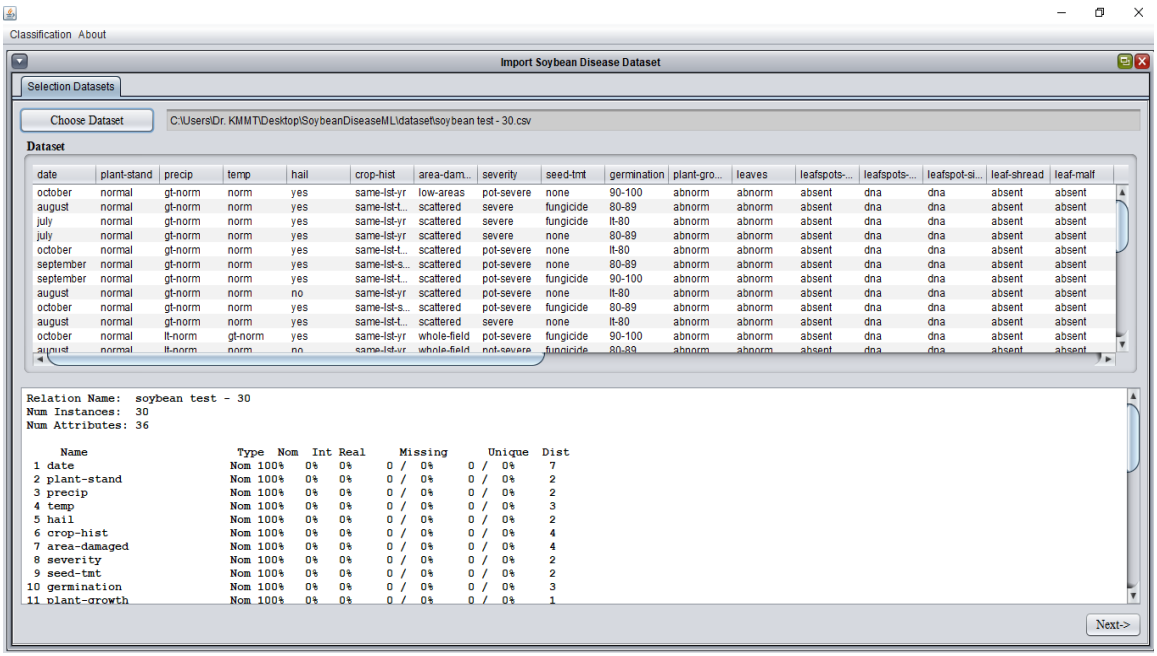

Figure 4.6 Interface of Soybean Disease test - 30 Dataset

Figure 4.7 shows Interface of Soybean Disease Dataset

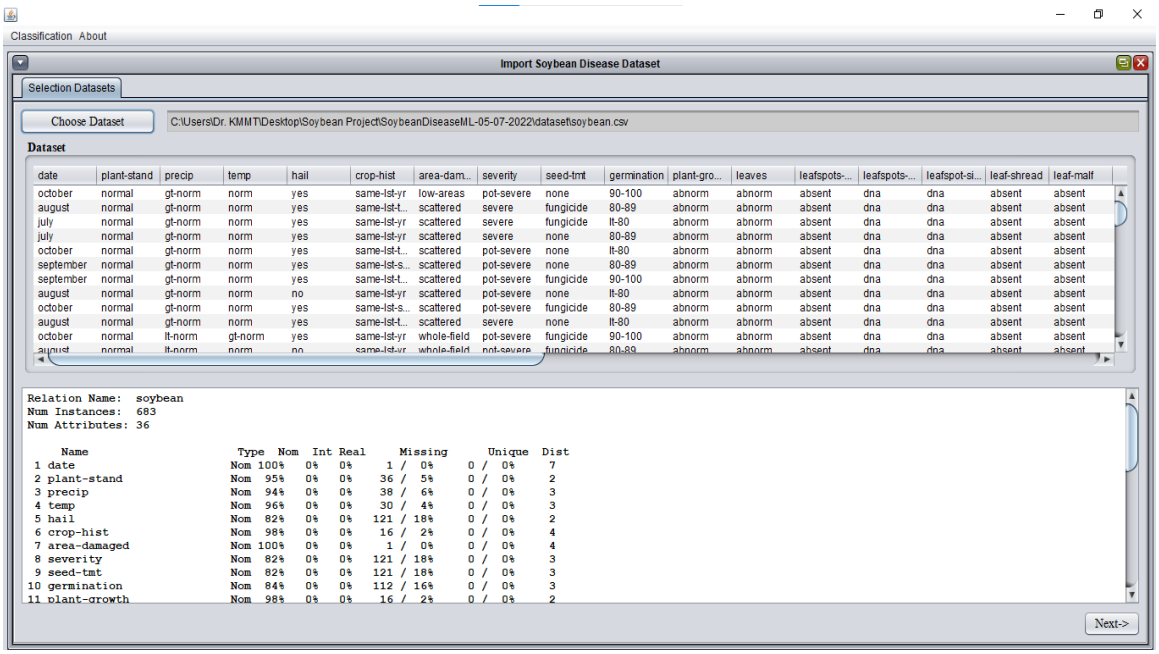

Figure 4.7 Interface of Soybean Disease Dataset

Figure 4.7 shows the soybean disease dataset which has 638 instances, 35 attributes and 19 classes. In this figure, date, plant-stand, precip, temp, hail, crop-hist ,

area-damaged, severity, seed-tmt, germination, plant, leaves, leafspot-halo, leafspot-marg, leafspot-size, leaf-shread, leaf-malf, leaf-mild, stem, lodging, stem-cankers, canker-lesion, fruiting-bodies, external decay, mycelium**,** int-discolor, sclerotia, fruit-pods, fruit spots, seed, mold-growth, seed-discolor, seed-size, shriveling and roots represents 35 attributes of the soybean disease dataset. In this figure, 19 classes are diaporthe-stem-canker**,** charcoal-rot, rhizoctonia-root-rot, phytophthora-rot, brown-stem-rot, powdery-mildew, downy-mildew, brown-spot, bacterial-blight, bacterial-pustule, purple-seed-stain, anthracnose, phyllosticta-leafspot, alternarialeaf-spot, frog-eye-leafspot, diaporthe-pod-&-stem-blight, cyst-nematode, 2-4-d-injury and herbicide-injury.

4.3.4 Interfacing with Normalization Dataset

Figure 4.8 shows Normalization Dataset

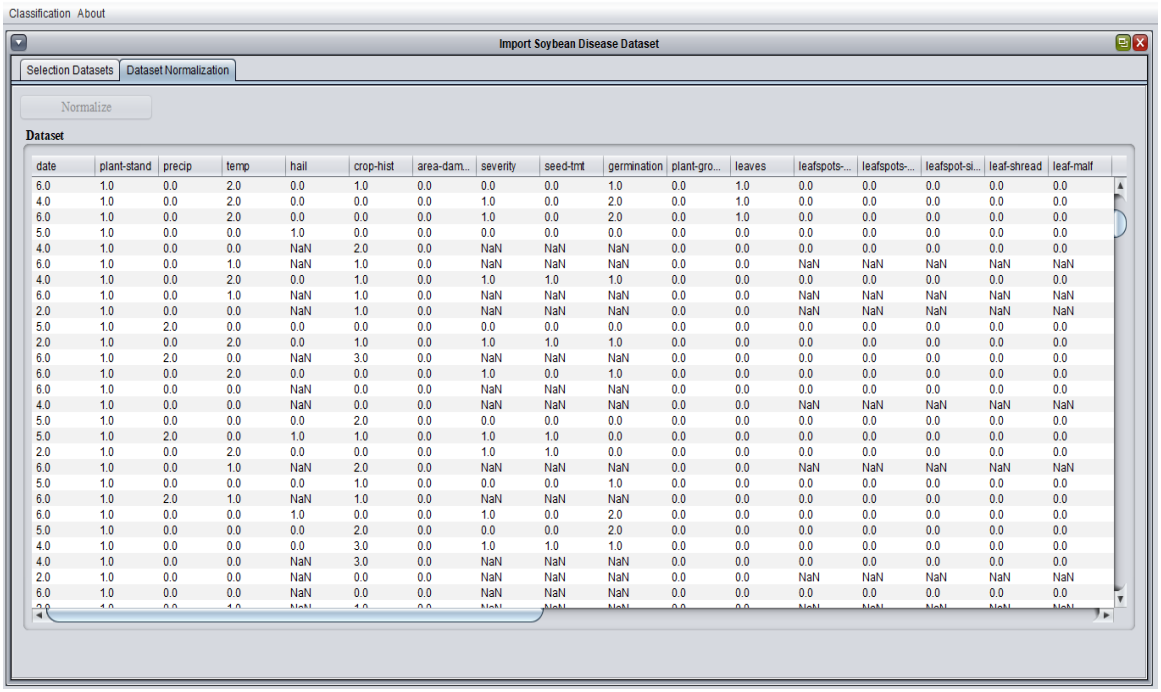

Figure 4.8 Interface of Normalization Dataset

4.3.5 Interfacing with Build Classifier (Training Data)

Figure 4.9 shows build classifier (Training data)

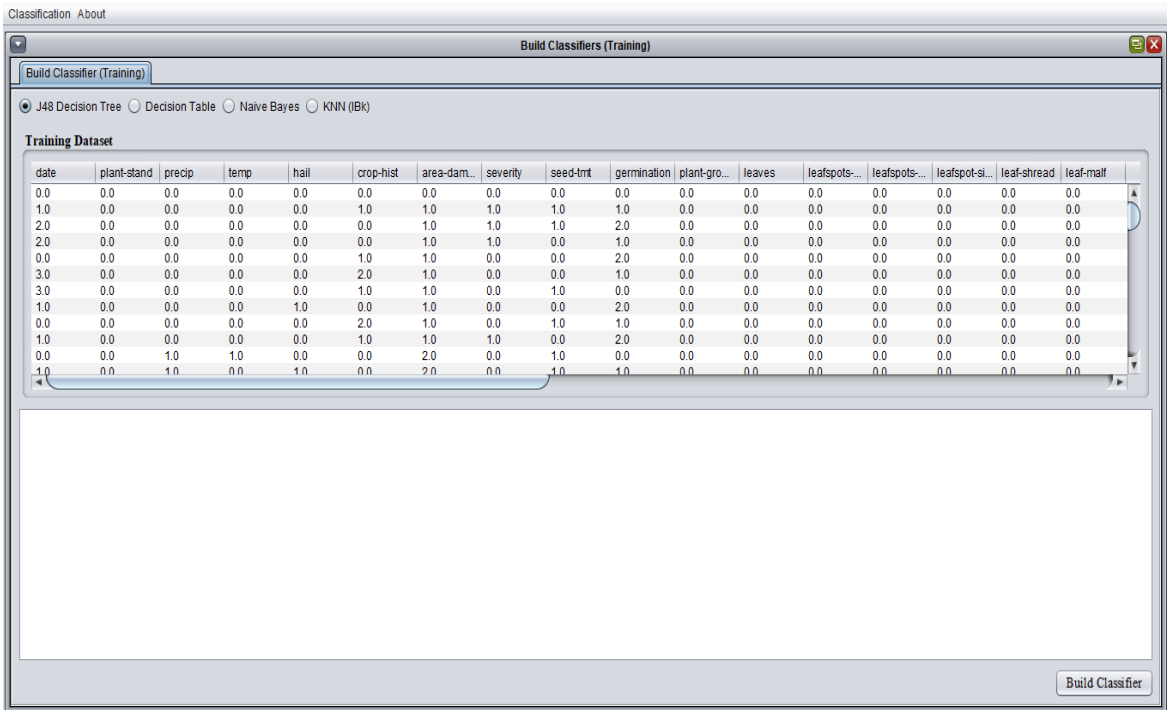

Figure 4.9 Interface of Build Classifier (Training data)

Figure 4.10 shows J48 decision tree build classifier

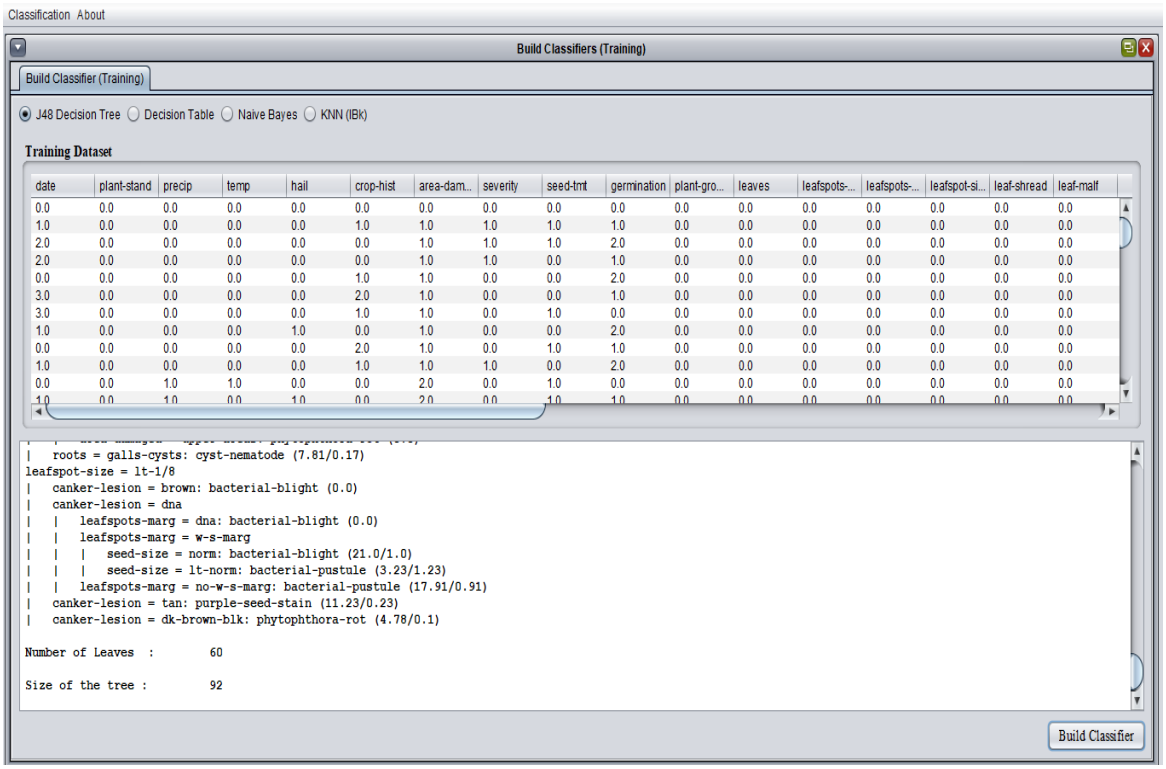

Figure 4.10 Interface of J48 Decision Tree Build Classifier

Figure 4.11 shows decision table build classifier

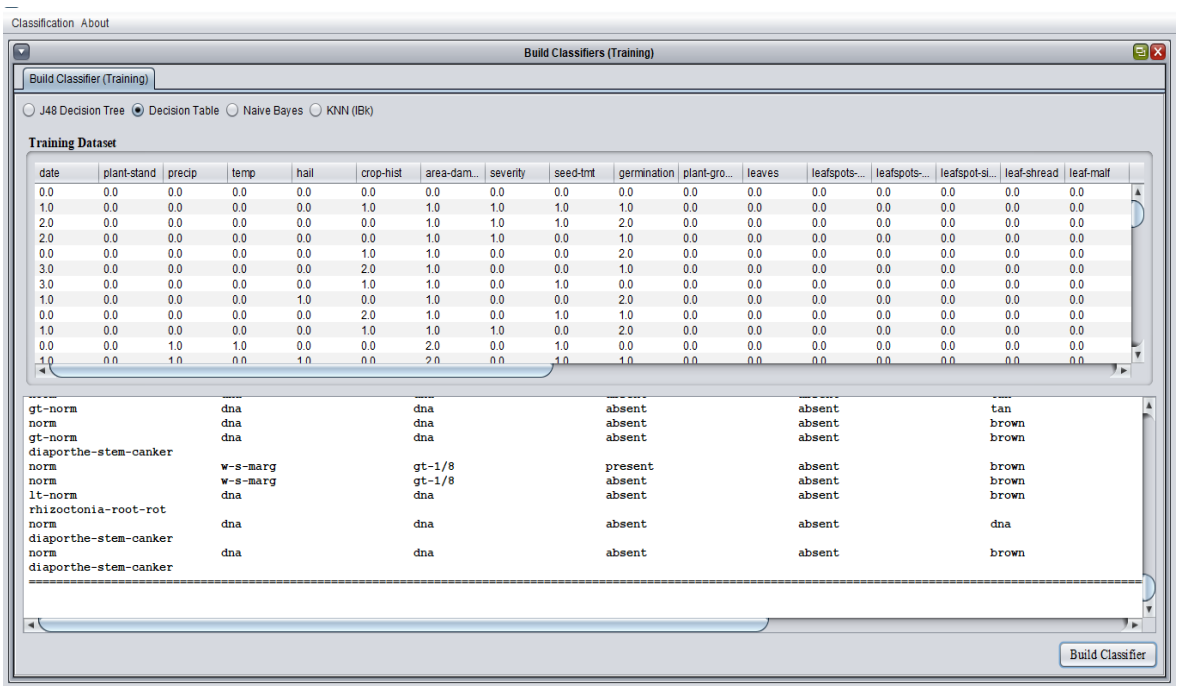

Figure 4.11 Interface of Decision Table Build Classifier

Figure 4.12 shows naïve bayse build classifier

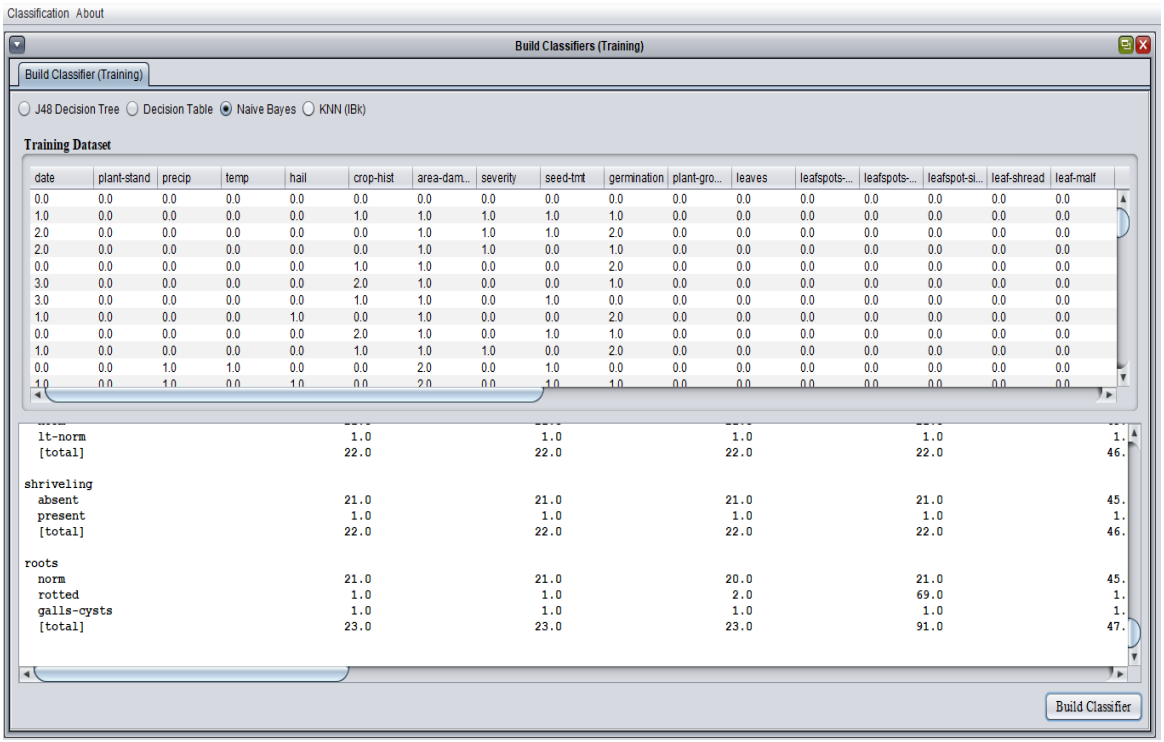

Figure 4.12 Interface of Naïve Bayse Table Build Classifier

Figure 4.13 shows k-NN build classifier

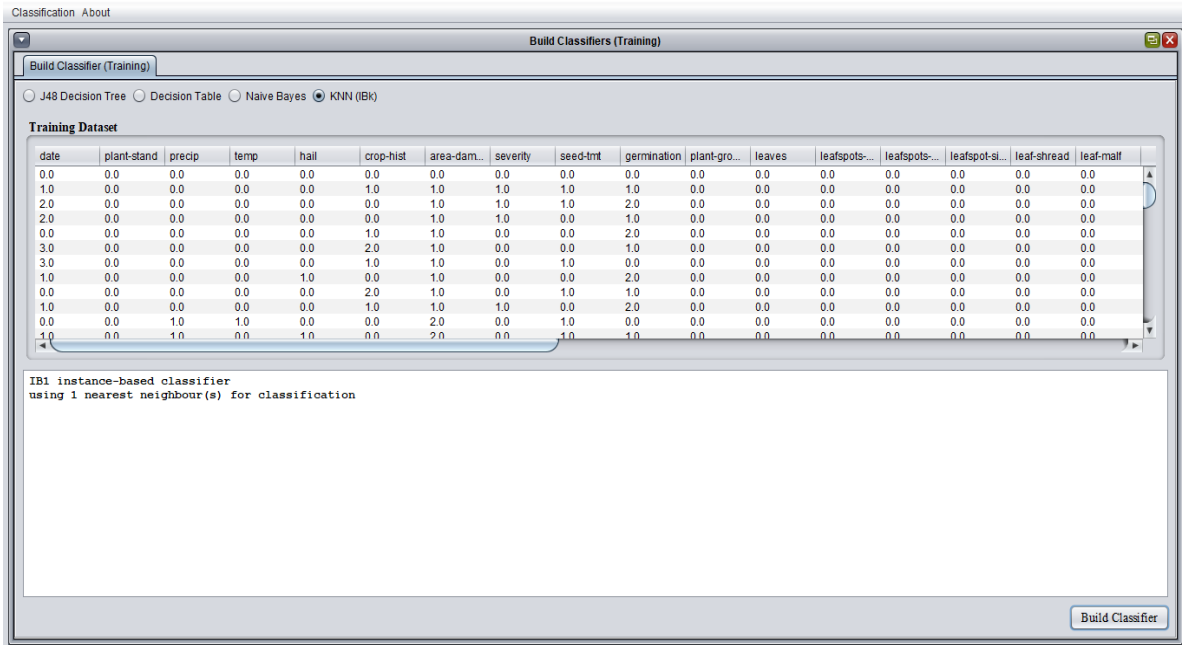

Figure 4.13 Interface of k-NN Build Classifier

4.4 Comparing with k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table Depend on Iteration for "Soybean Disease" Dataset

Comparing of four algorithm showed in the result of k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table for soybean disease dataset that depend on iteration.

4.4.1 Interfacing with the K-Folds Cross Validation

Figure 4.14 shows of the K-folds cross vaidation.

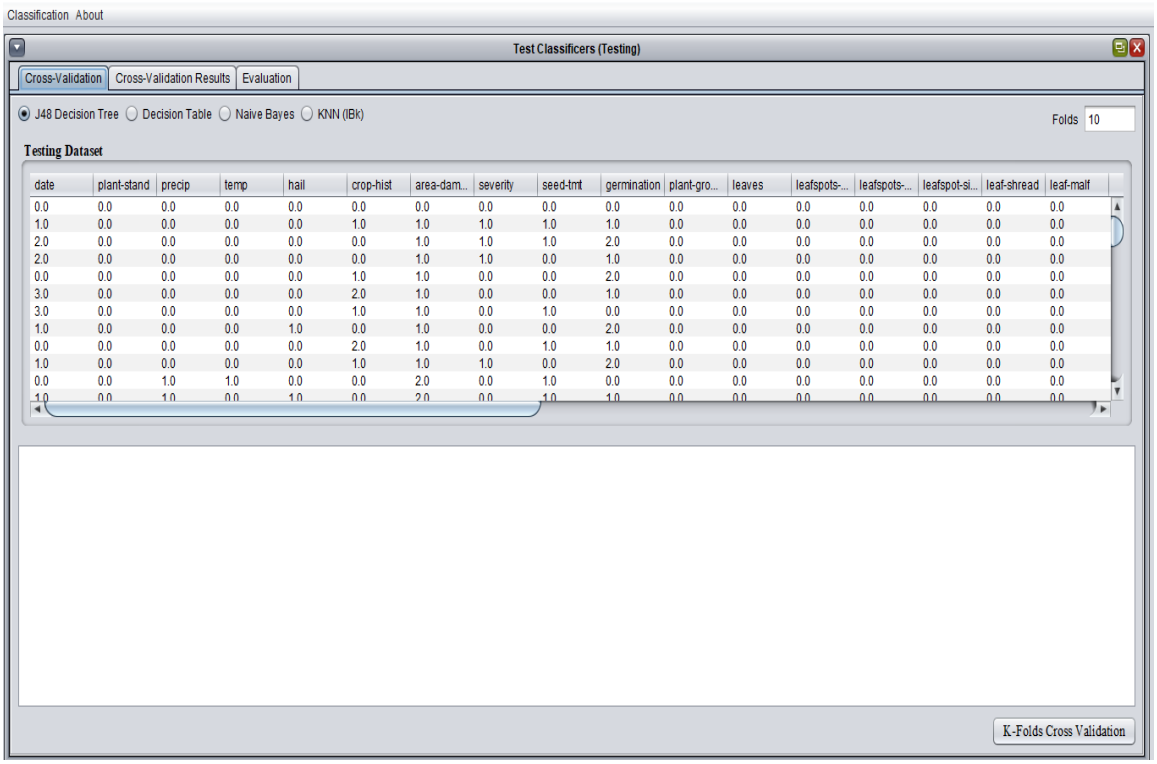

Figure 4.14 Interface of the K-Folds Cross Validation

Figure 4.15 shows the J48 Decision Tree for 2-folds cross vaidation.

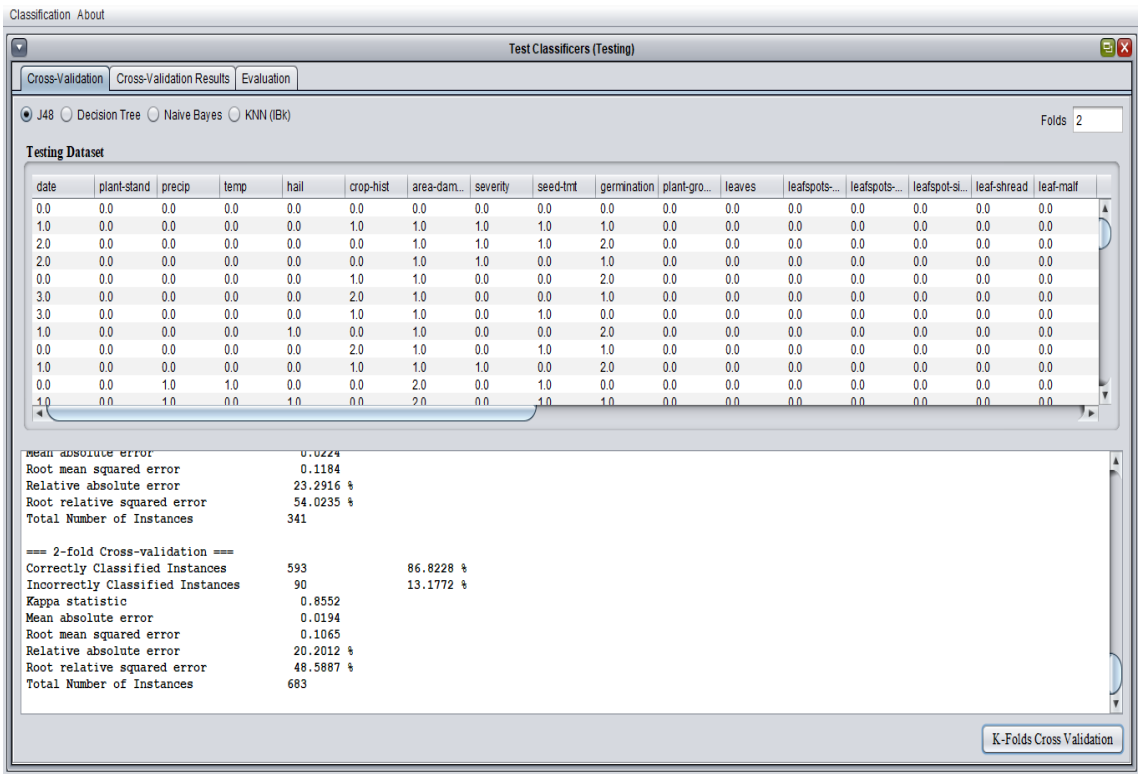

Figure 4.15 Interface of the J48 Decision Tree for 2-Folds Cross Validation

Figure 4.16 shows the Decision Table for 2-folds cross vaidation.

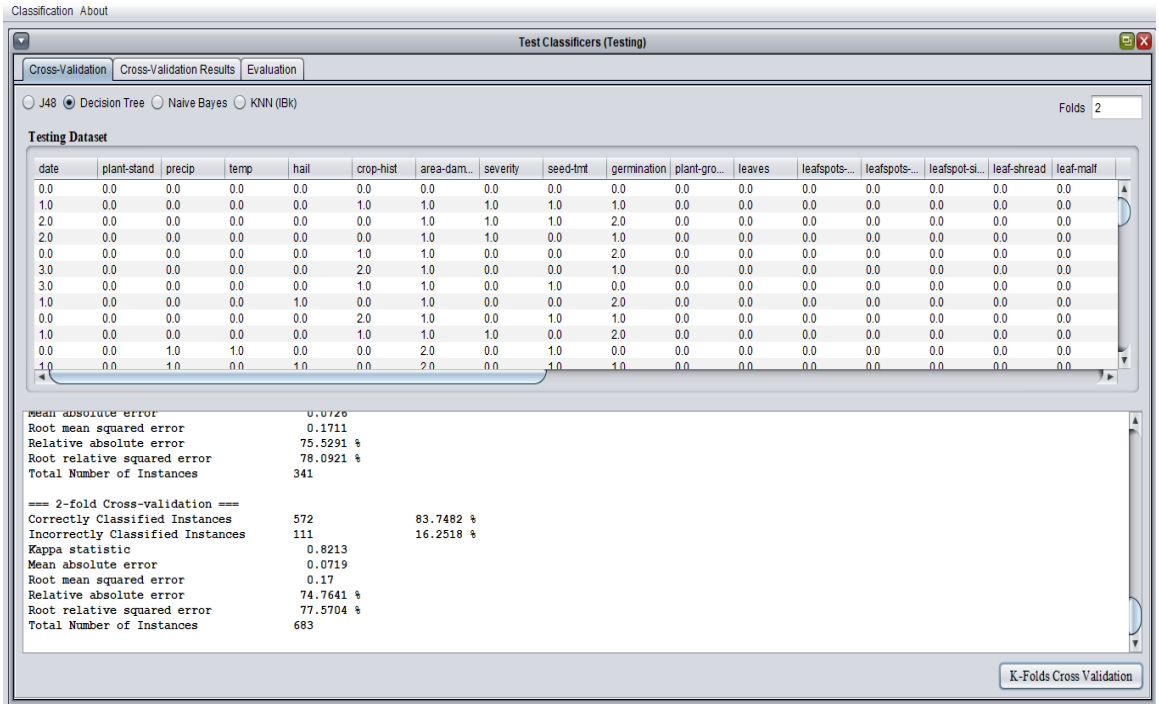

Figure 4.16 Interface of the Decision Table for 2-Folds Cross Validation

Figure 4.17 shows the Naïve Bayes for 2-folds cross vaidation.

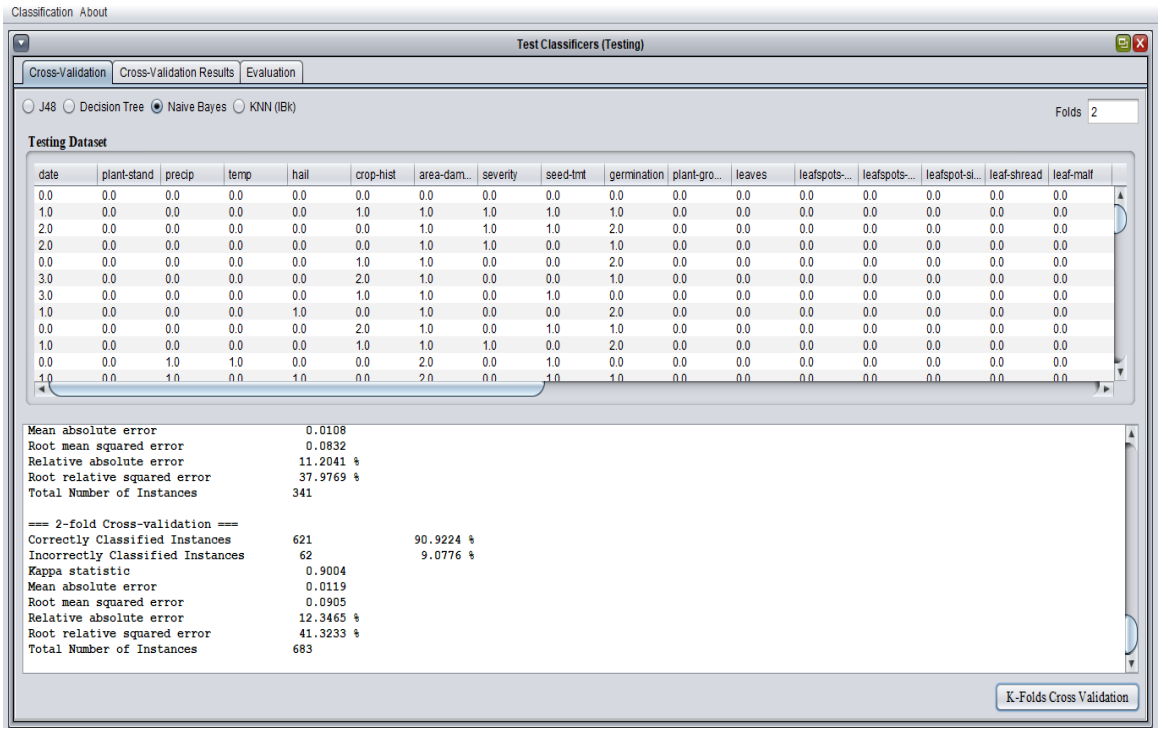

Figure 4.17 Interface of the Naïve Bayes for 2-Folds Cross Validation

Figure 4.18 shows the k-NN for 2-folds cross vaidation.

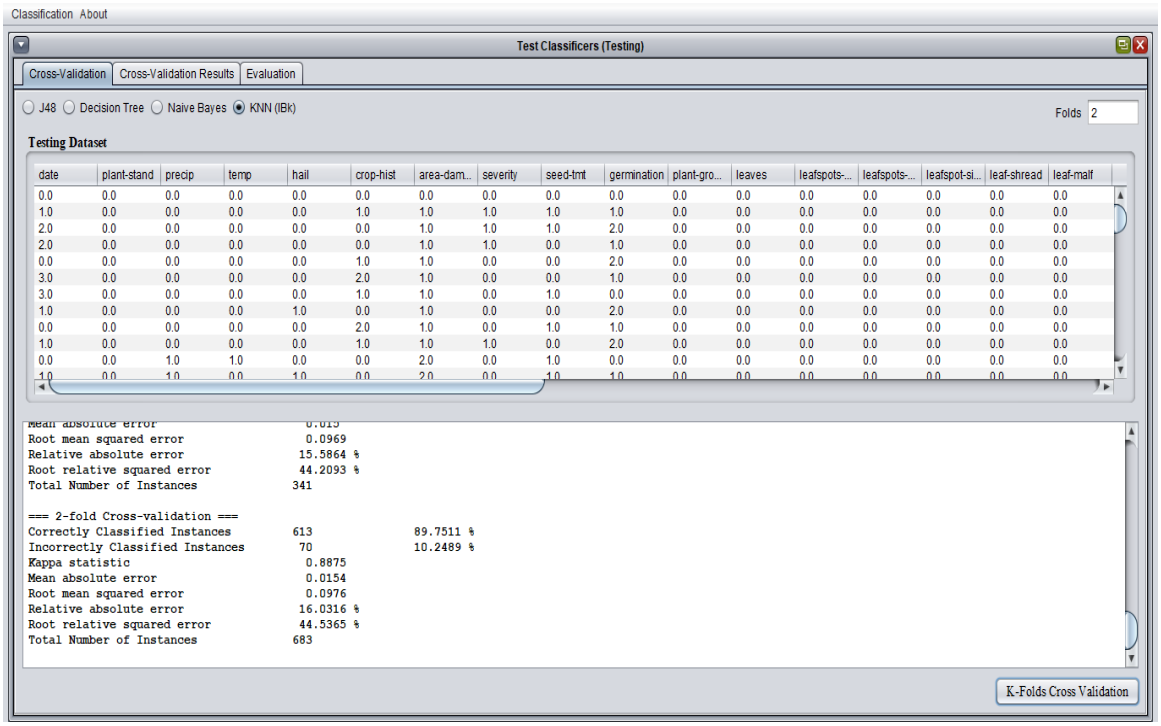

Figure 4.18 Interface of the k-NN for 2-Folds Cross Validation

Figure 4.19 shows the J48 Decision Tree for 4-folds cross vaidation.

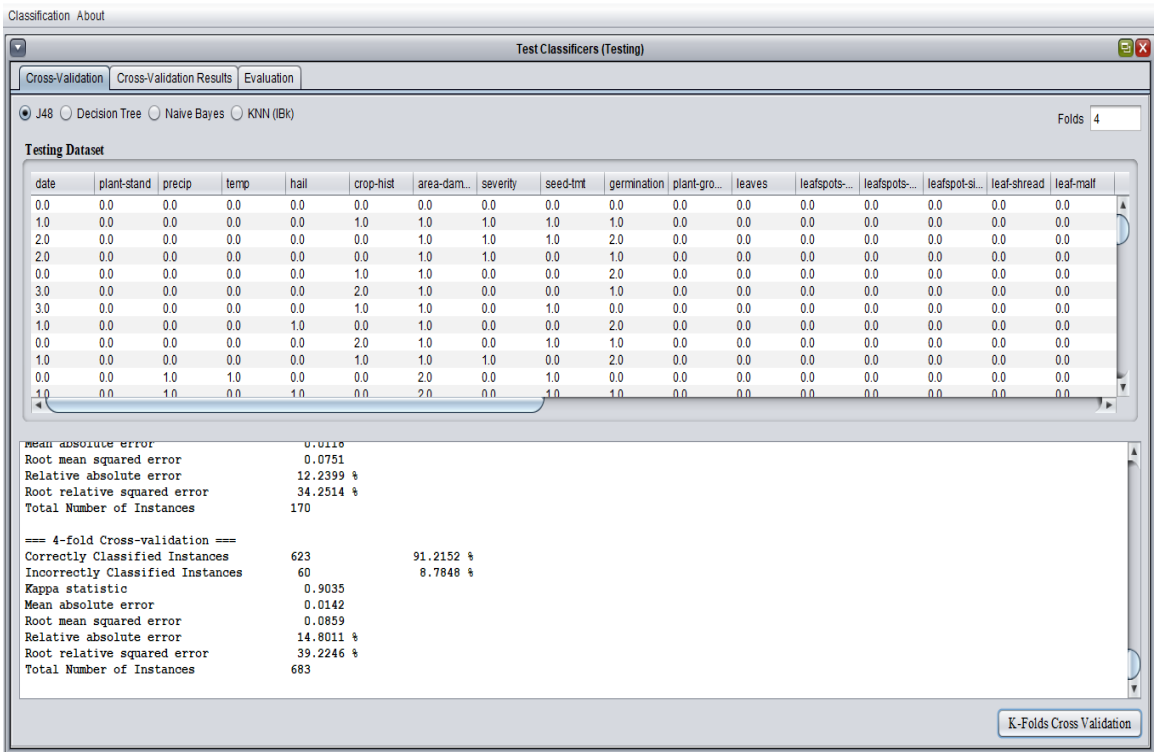

Figure 4.19 Interface of the J48 Decision Tree for 4-Folds Cross Validation

Figure 4.20 shows the Decision Table for 4-folds cross vaidation.

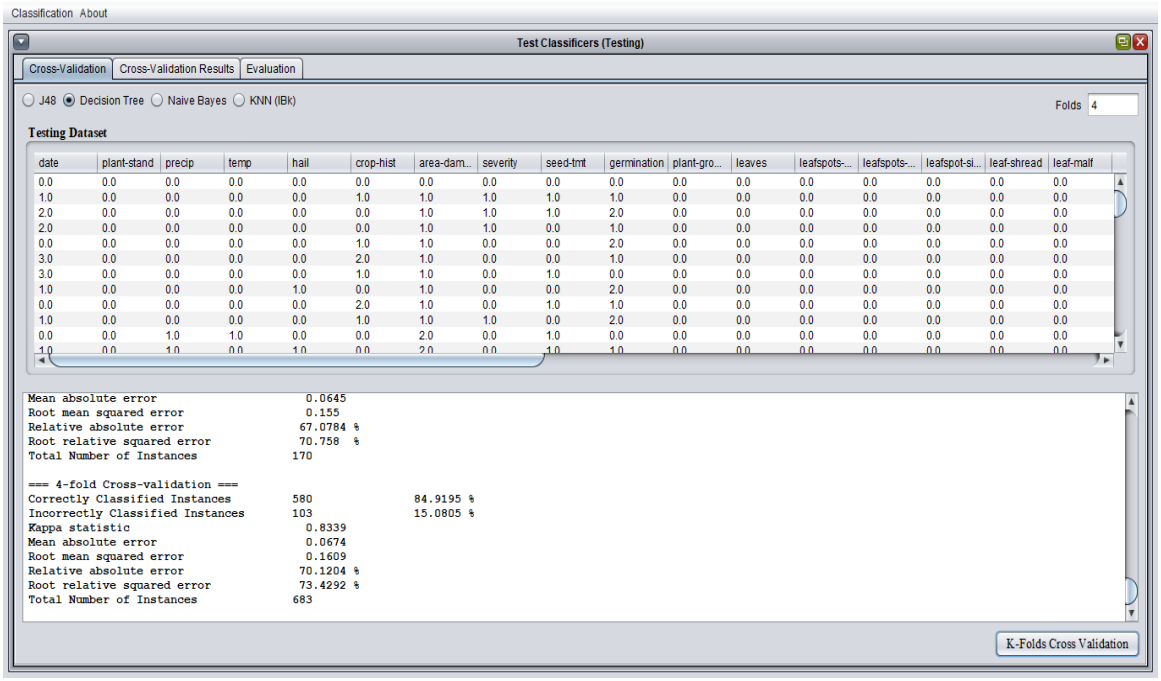

Figure 4.20 Interface of the Decision Table for 4-Folds Cross Validation

Figure 4.21 shows the Naïve Bayes for 4-folds cross vaidation.

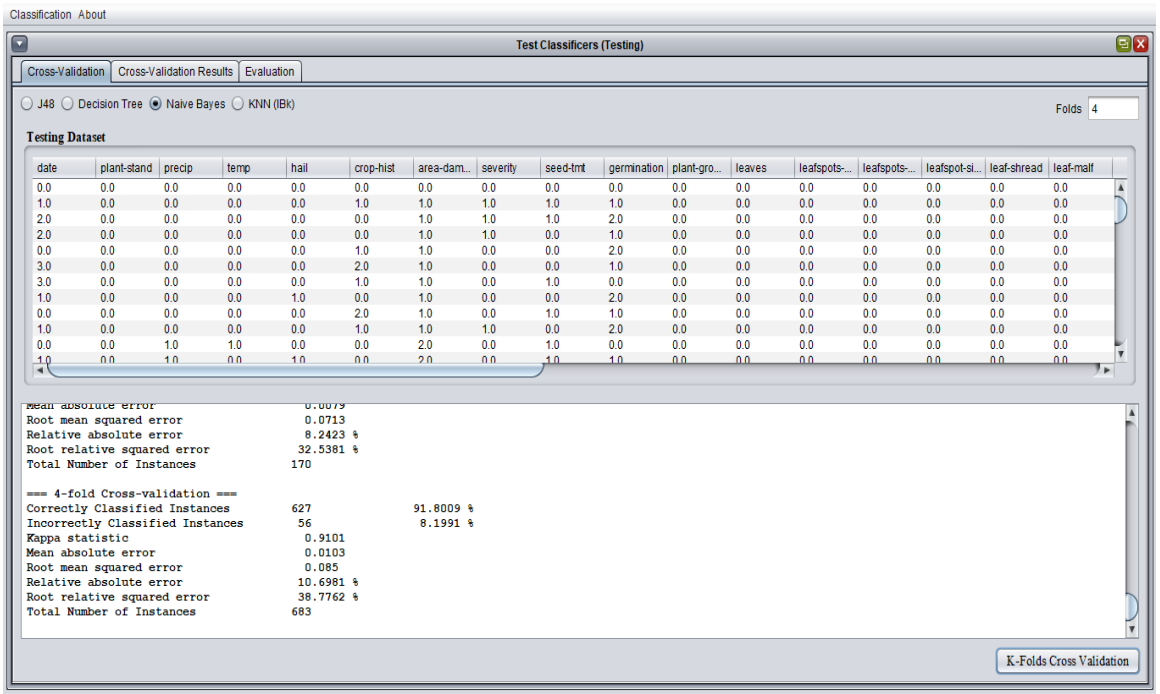

Figure 4.21 Interface of the Naïve Bayes for 4-Folds Cross Validation

Figure 4.22 shows the k-NN for 4-folds cross vaidation.

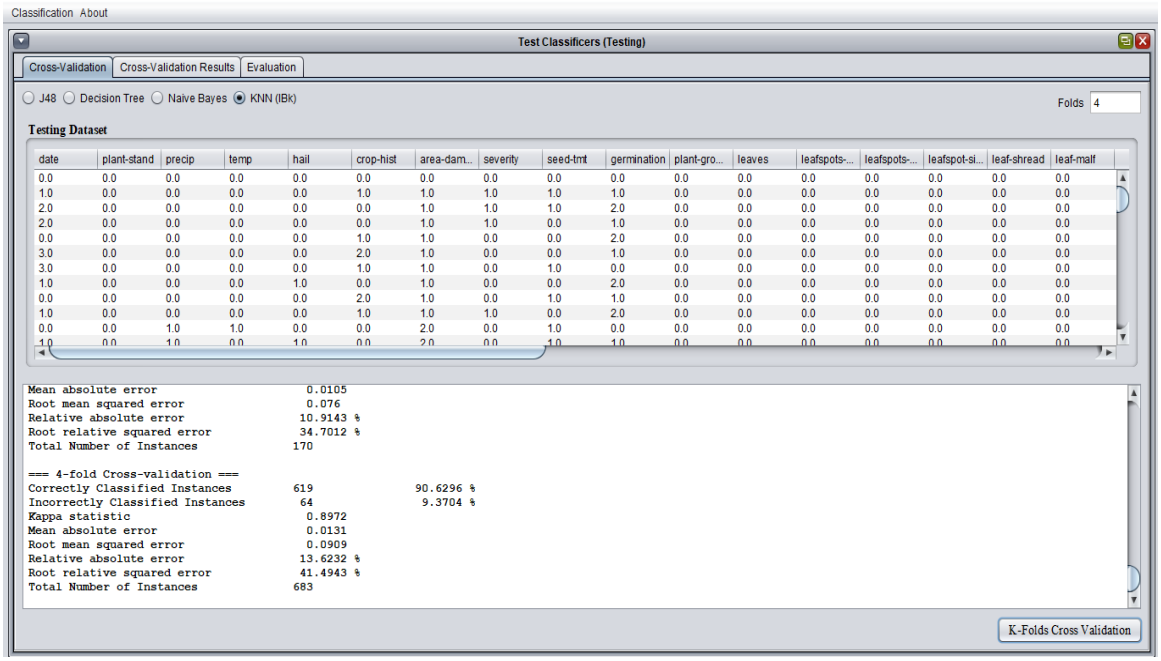

Figure 4.22 Interface of the k-NN for 4-Folds Cross Validation

Figure 4.23 shows the J48 Decision Tree for 8-folds cross vaidation.

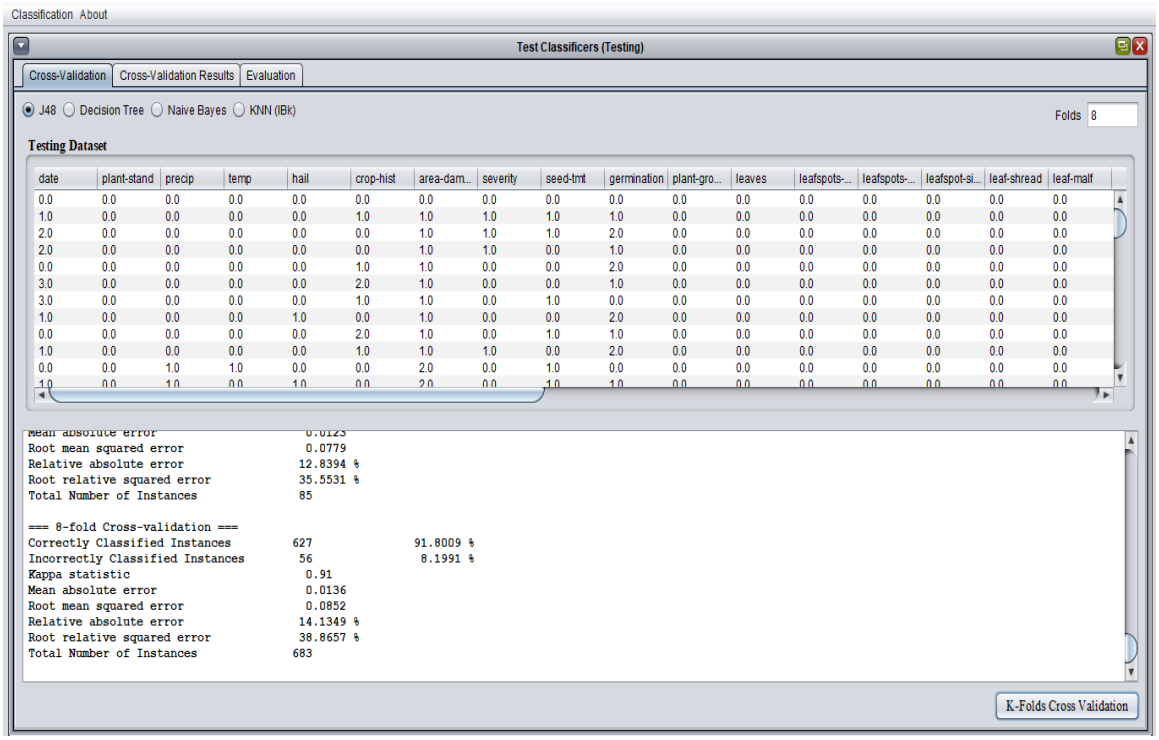

Figure 4.23 Interface of the J48 Decision Tree for 8-Folds Cross Validation

Figure 4.24 shows the Decision Table for 8-folds cross vaidation.

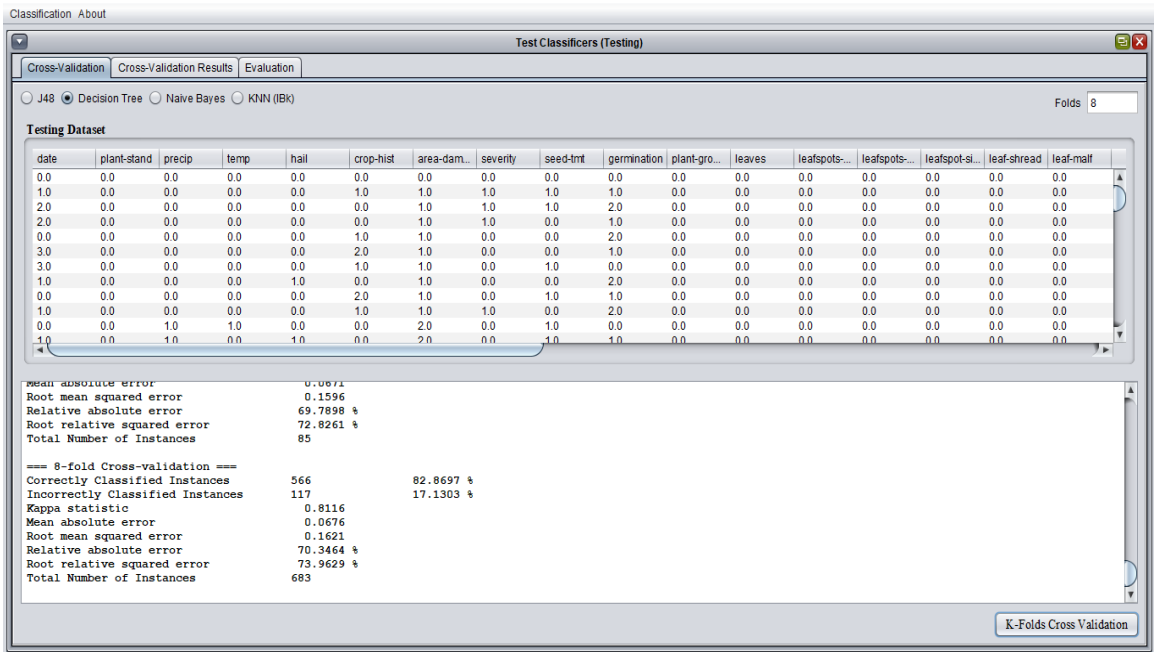

Figure 4.24 Interface of the Decision Table for 8-Folds Cross Validation

Figure 4.25 shows the Naïve Bayes for 8-folds cross vaidation.

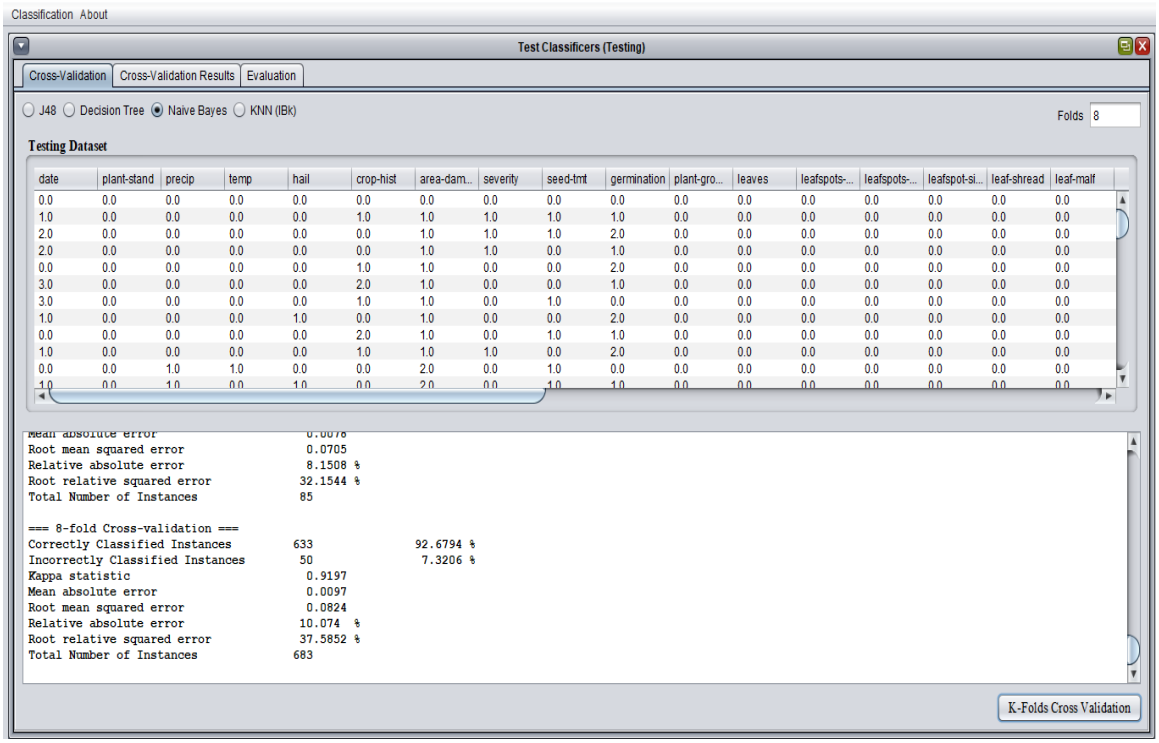

Figure 4.25 Interface of the Naïve Bayes for 8-Folds Cross Validation

Figure 4.26 shows the k-NN for 8-folds cross vaidation.

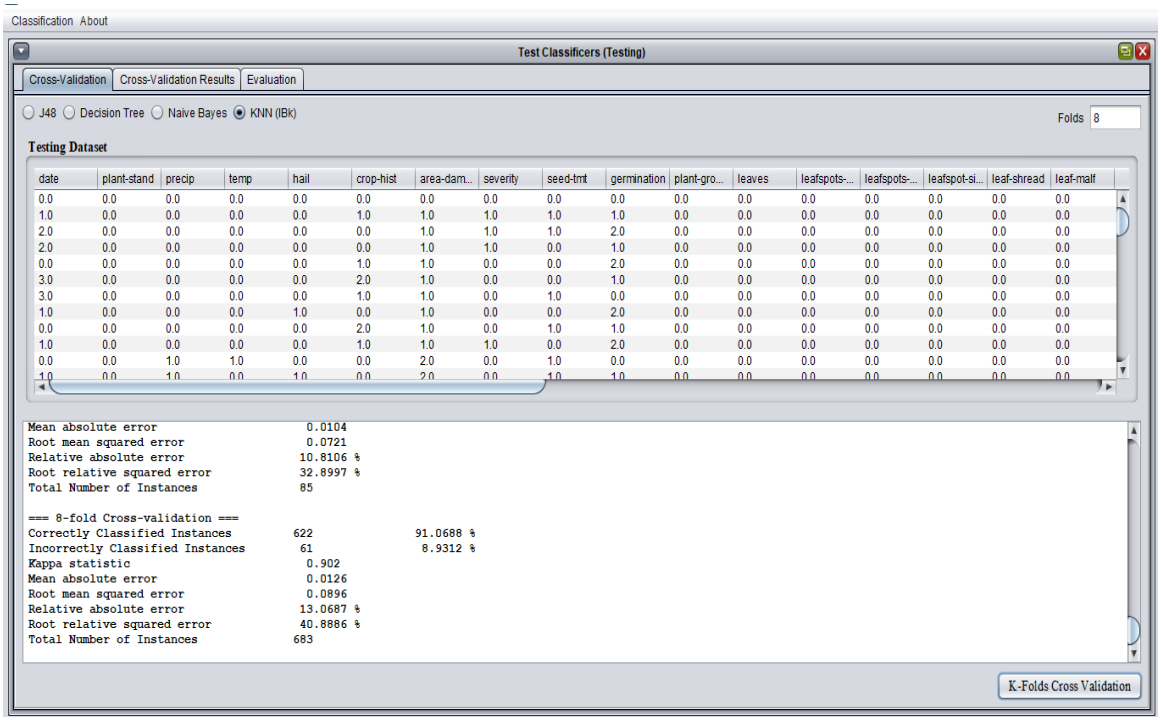

Figure 4.26 Interface of the k-NN for 8-Folds Cross Validation

Figure 4.27 shows the J48 Decision Tree for 10-folds cross vaidation.

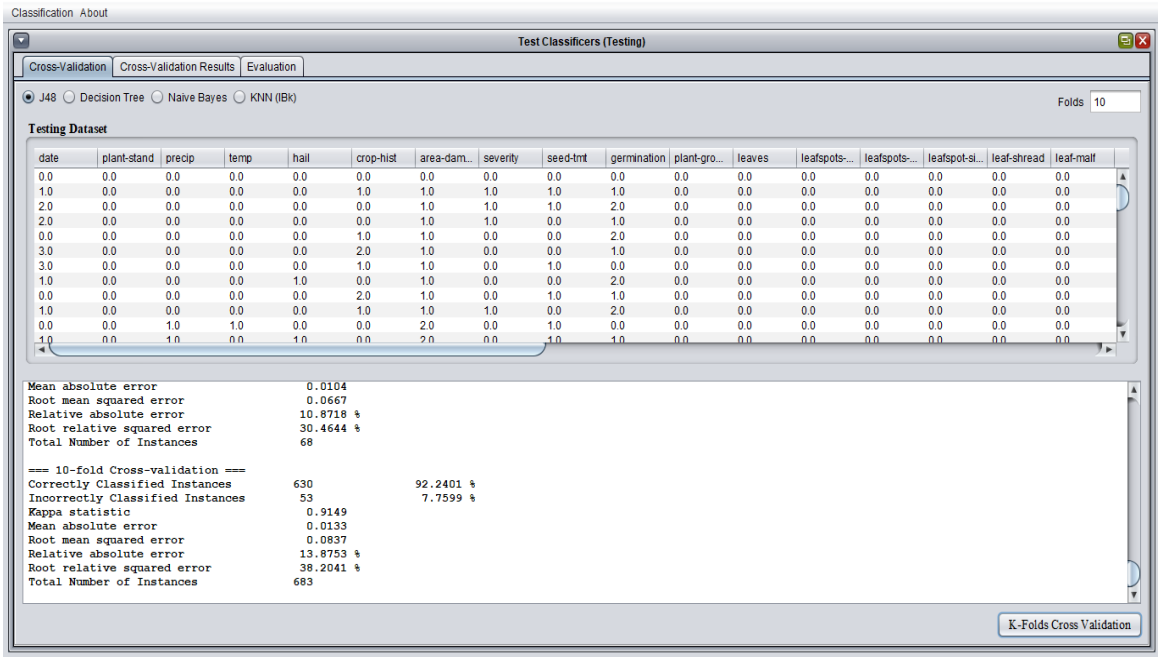

Figure 4.27 Interface of the J48 Decision Tree for 10-Folds Cross Validation

Figure 4.28 shows the Decision Table for 10-folds cross vaidation.

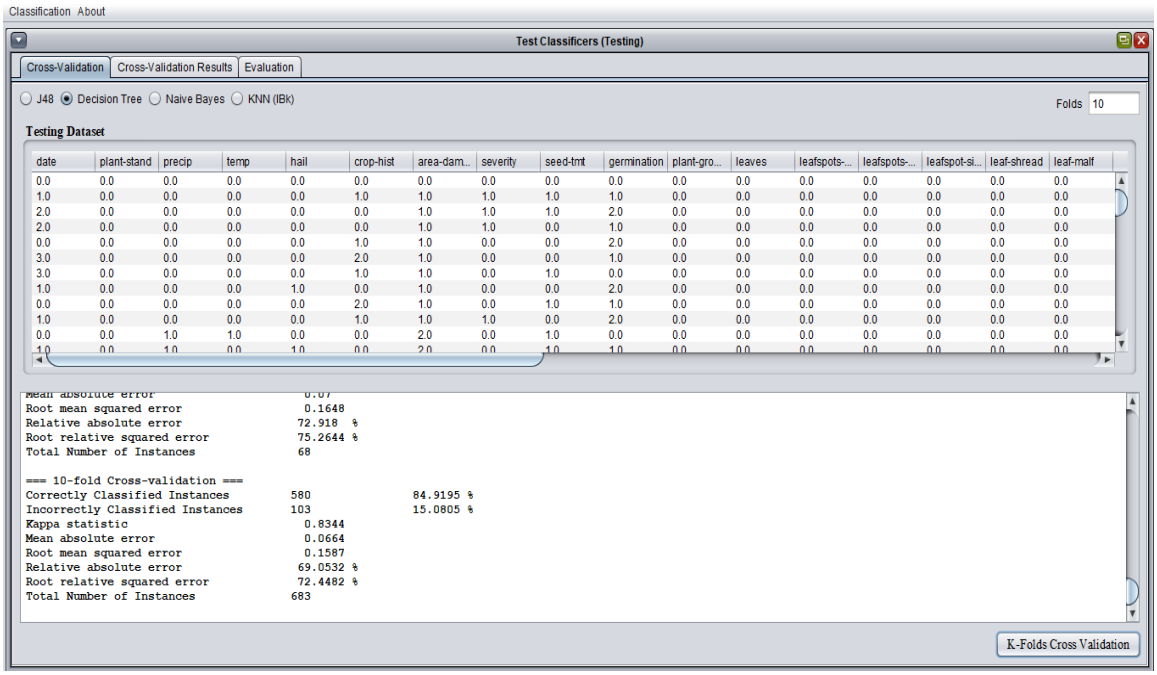

Figure 4.28 Interface of the Decision Table for 10-Folds Cross Validation

Figure 4.29 shows the Naïve Bayes for 10-folds cross vaidation.

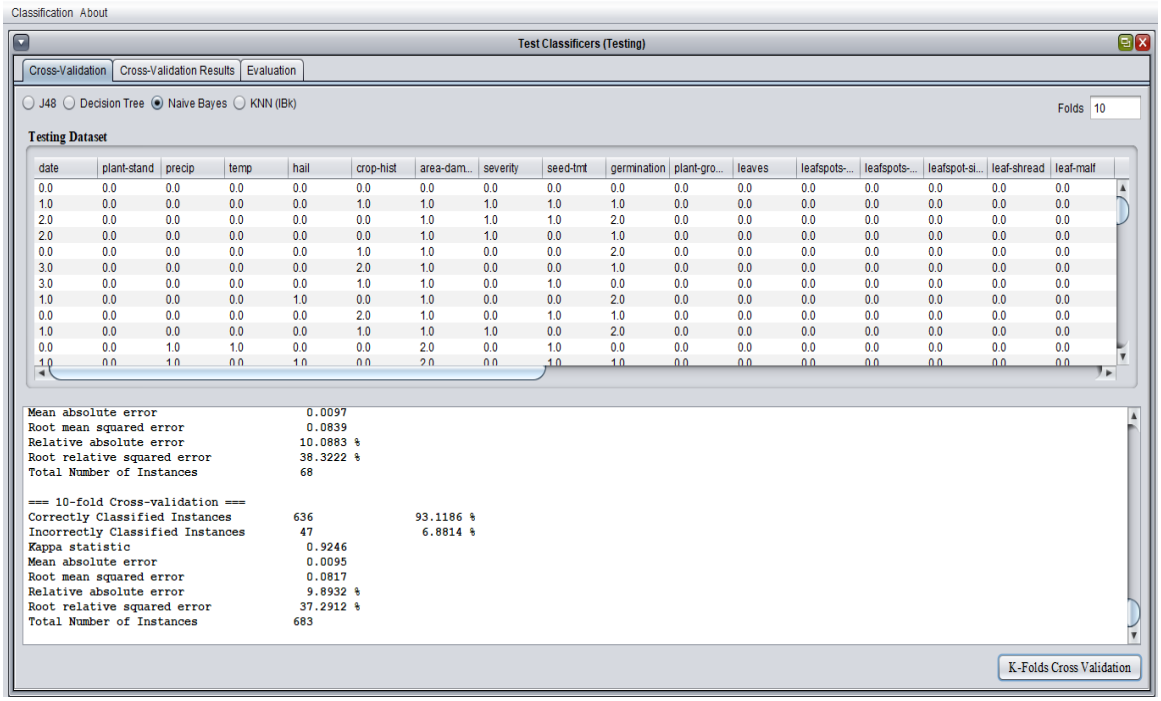

Figure 4.29 Interface of the Naïve Bayes for 10-Folds Cross Validation

Figure 4.30 shows the k-NN for 10-folds cross vaidation.

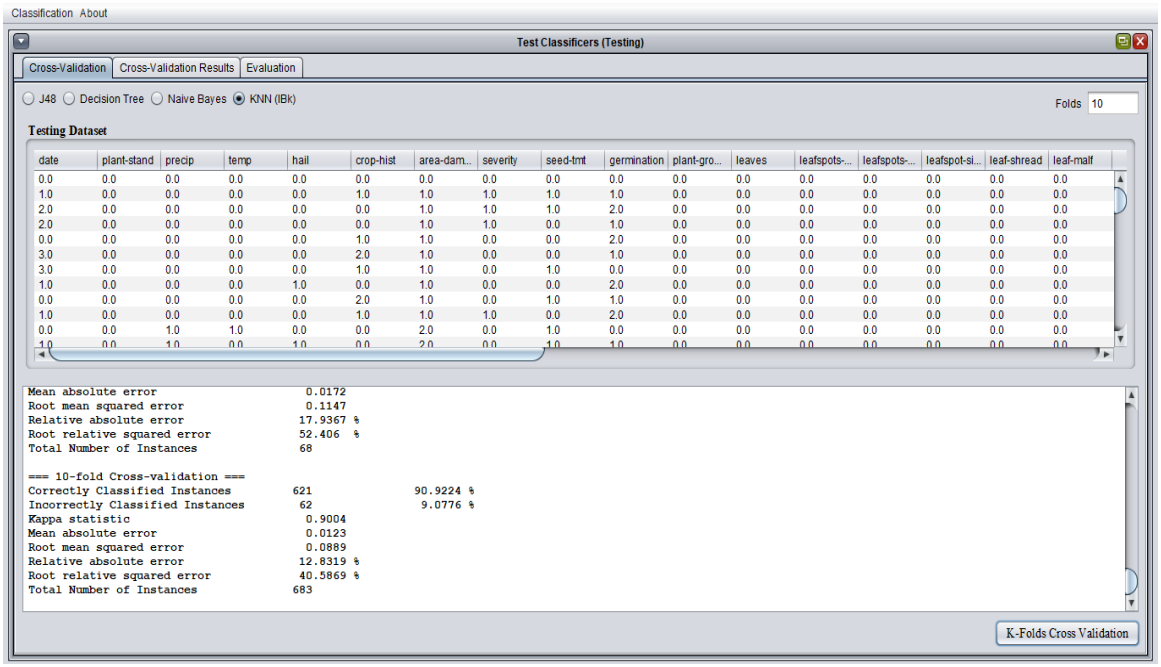

Figure 4.30 Interface of the k-NN for 10-Folds Cross Validation

4.4.2 Comparing with k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table

Figure 4.31 Comparing the k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table Depend on Iteration for “Soybean Disease” Dataset using 2-Cross Validation.

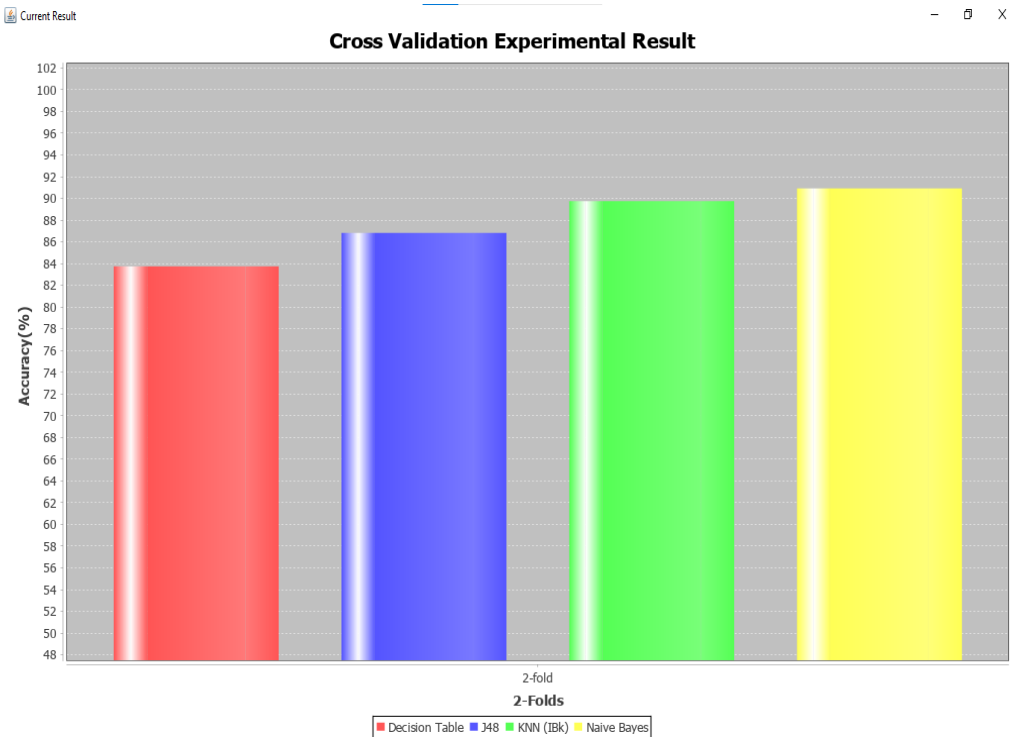

Figure 4.31 Interface of the compare 2-cross validation result k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table

Figure 4.32 Comparing the k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table Depend on Iteration for "Soybean Disease" Dataset using 4-Cross Validation.

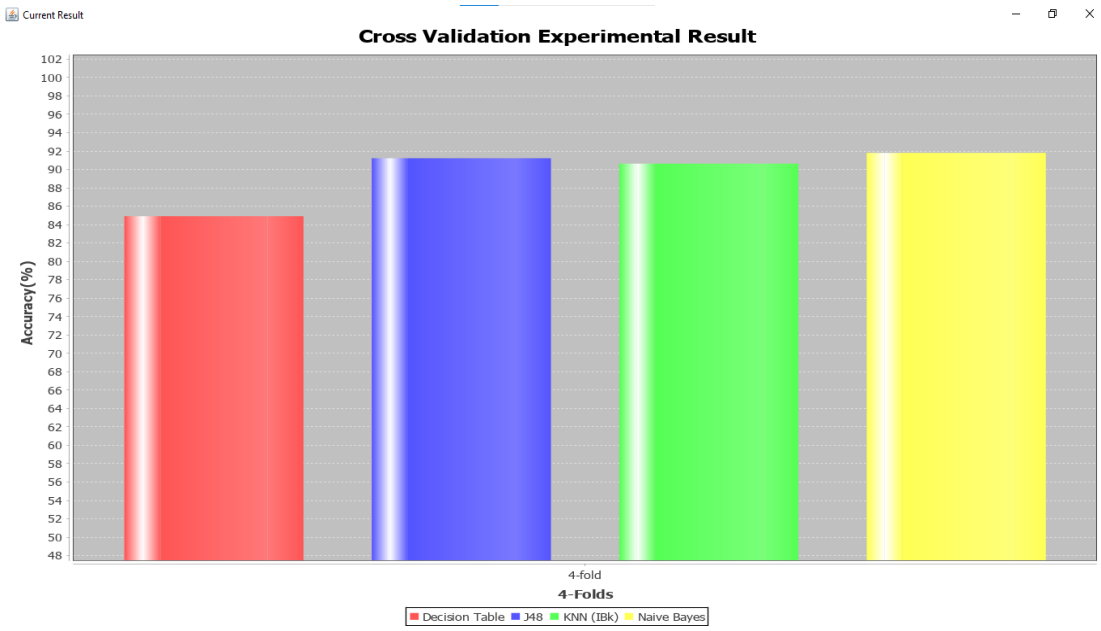

Figure 4.32 Interface of the compare 4-cross validation result k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table

Figure 4.33 Comparing the k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table Depend on Iteration for “Soybean Disease” Dataset using 8-Cross Validation.

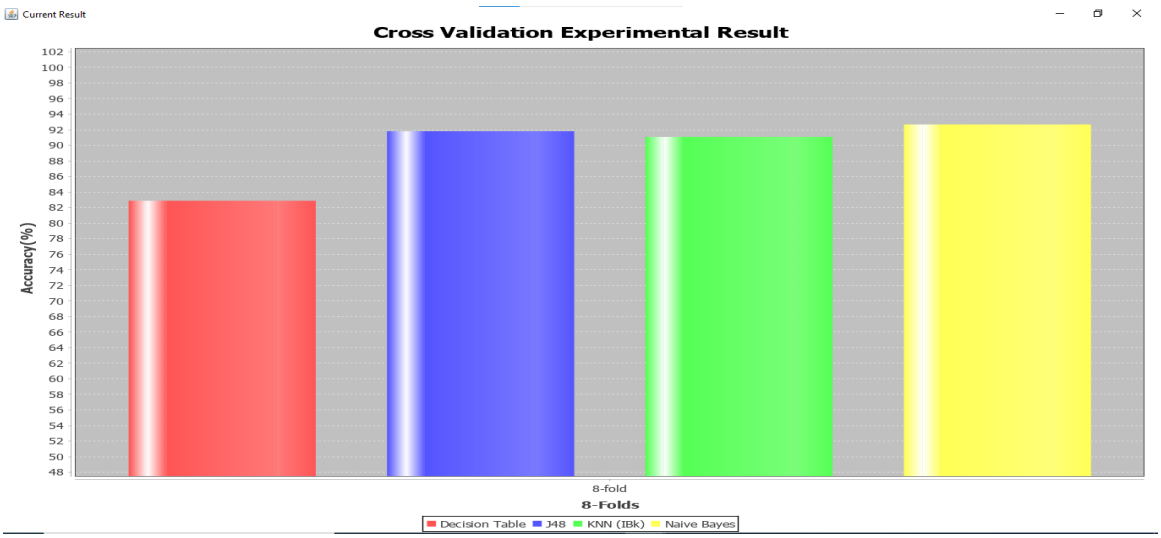

Figure 4.33 Interface of the compare 8-cross validation result k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table

Figure 4.34 Comparing the k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table Depend on Iteration for “Soybean Disease” Dataset using 10-Cross Validation.

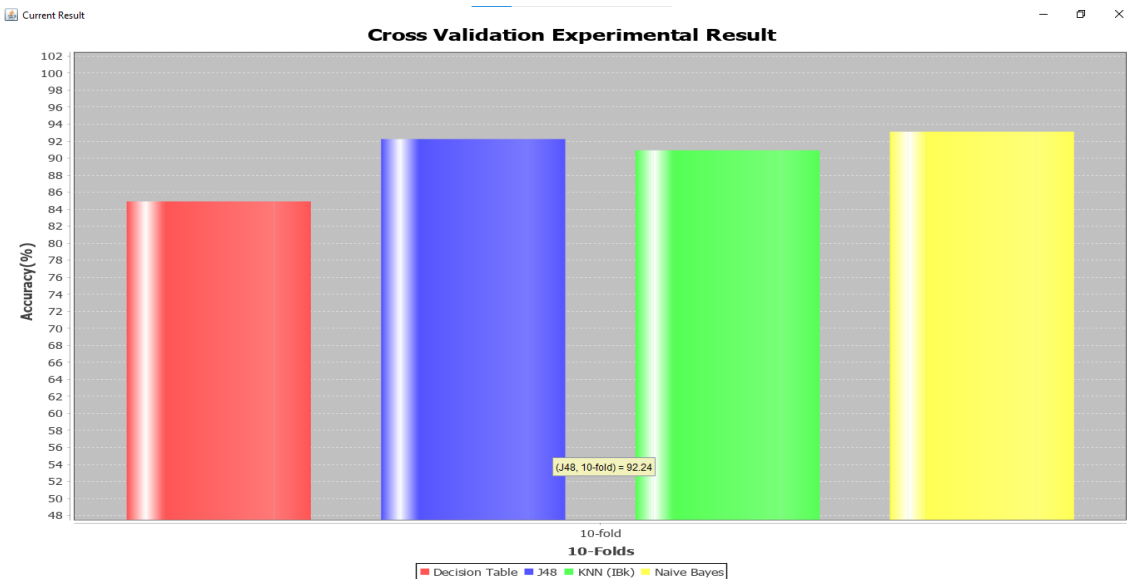

Figure 4.34 Interface of the compare 10-cross validation result k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table

Figure 4.35 Comparing the k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table Depend on Iteration for “Soybean Disease” Dataset using Cross Validation.

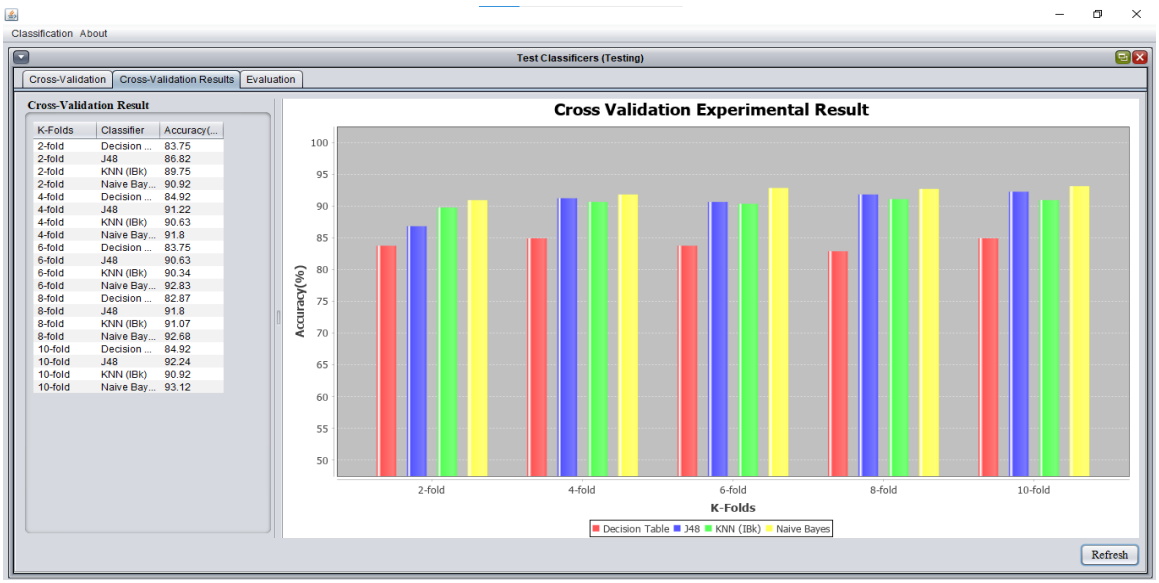

Figure 4.35 Interface of the compare cross-validation result k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table

4.5 Experimental Results

The experimental result in the analysis is done by using an Intel(R) Core(TM) i7-5500U CPU with @ 2.40GHz 2.40 GHz processor along with 4 GB of RAM and Apache Net Beans IDE 12.6 programming language along with Java version 11.0.1.

Comparing of four algorithm showed in the result of k- Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table for soybean disease dataset that depend on iteration.

In the cross-validation result, there are 2-cross validation. If J48 is used, the result is 86.82%. If Decision Table is used, the result is 83.75%, If Naive Bayes is used, the result is 90.92%. If k-NN is used, the result is 89.75%. On the other hand, there are 4-cross validation. If J48 is used, the result is 91.22%. If Decision Table is used, the result is 84.92%, If Naive Bayes is used, the result is 91.80%. If k-NN is used, the result is 90.63%.On the other hand, there are also 6-cross validation. If J48 is used, the result is 90.6296%. If Decision Table is used, the result is 83.7482%. If Naive Bayes is used, the result is 92.8258%. If k-NN is used, the result is 90.3367%. In this section, there are also 8-cross validation. If J48 is used, the result is 91.8009%. If Decision Table is

used, the result is 82.8697%. If Naive Bayes is used, the result is 92.6794%. If k-NN is used, the result is 91.0688%. In 10-cross validation result, If J48 is used, the result is 92.2401%. If Decision Table is used, the result is 84.9195%. If Naive Bayes is used, the result is 93.1186%. If k-NN is used, the result is 90.9224%.

Depending on the dataset, therefore, the evaluation results may change.

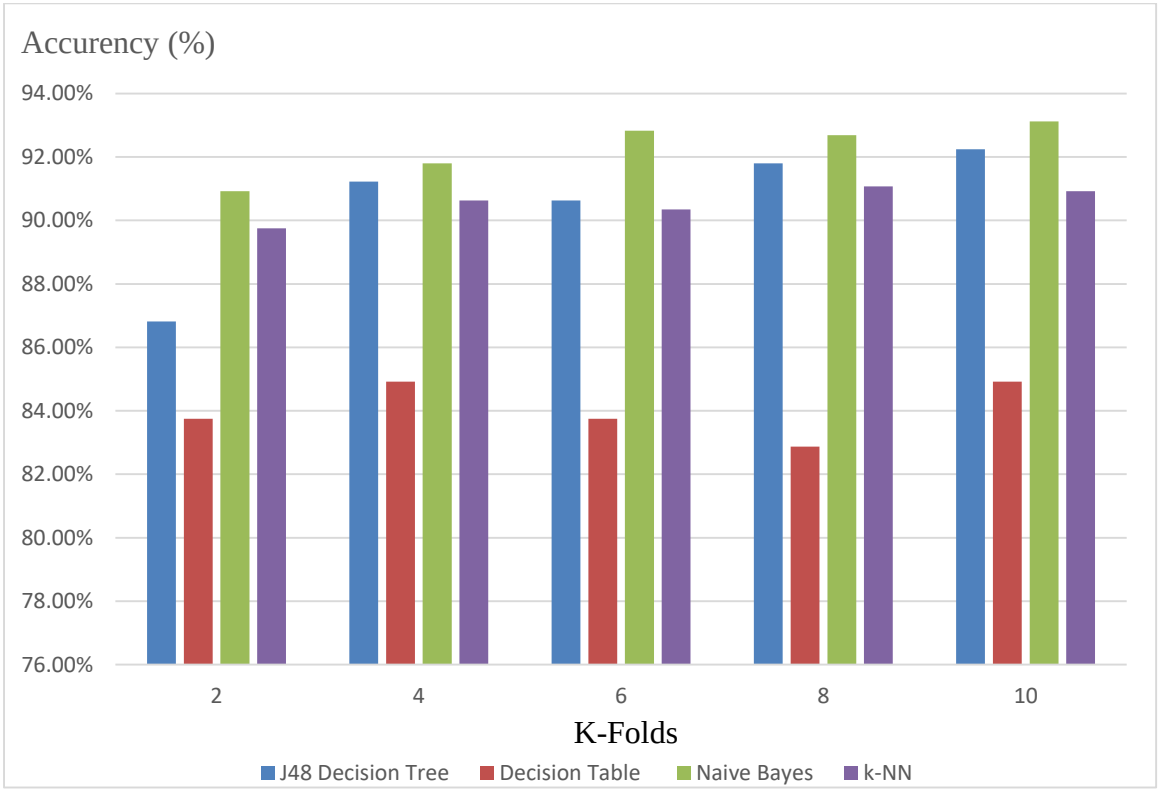

Figure 4.46 The compare result four different algorithms are used cross-validation

CHAPTER 5

CONCLUSION

There are many real-life use cases to create unique machine learning projects. If the users are still having trouble coming up with an actual use case, look for something unique and useful, such as a machine learning project where they can compare several machine learning classification algorithms.

Classification in machine learning refers to training a model to identify the category to which a given entry belongs. Since there are so many different categorization methods in machine learning, it would be a fantastic and original machine learning project for a novice if the user could present a thorough comparison of these techniques. To complete this job, they must first select a classification-based issue statement and list all possible classification algorithms.

Four kinds of algorithm are used in this paper. Machine Learning techniques are in use, such as k-Nearest Neighbors (k-NN), J48 Decision Trees, Naïve Bayes and Decision Table. Among them Naïve Bayes Algorithm is the best by according the results.

This thesis has been performed the experiments in order to determine the classification accuracy of four algorithms in terms of which is the better predictive algorithm of user's decision making, with the help of an attractive data mining tool known as Python, and Apache Net Beans IDE. The system will give diagnosis for respective diseases to famers.

5.1 Advantages of the System

In the developing countries, 50-80% of the population is directly engaged in agriculture and they also have the lowest agricultural output. The best ways to increase the food supply should be pursued in order to feed these individuals. One of these strategies is disease protection for the crops. The types of plants and enterprises in a region may be restricted by plant diseases. Plant diseases lower the output of plants, both qualitatively and quantitatively. Crops or plants that have diseases may become poisonous to both people and animals.

This thesis the potential yield benefits can be achieved by managing risks to early planting with improved technologies and tools. Early detection of soybean disease

can prevent farmers from contracting the disease and increase yields. It is very important to be able to accurately diagnose the disease. This thesis has presented accurately describes soybean disease in four of classification algorithm.

5.2 Limitations and Further Extension

This system is implemented by soybean disease dataset from only UCI (University of California) Machine learning dataset by using k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table classification algorithms. The study of datasets is automatically classified based on k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table classification algorithms. k-Nearest Neighbors, Decision Trees, Naïve Bayes and Decision Table classification algorithms have been developed by using java programming language. This thesis can be extended in various ways. Although there are many various versions of classification can be used advanced in the literature. Thus, any other extended versions of algorithm can be used for classification. Other fitness functions should be tested.

AUTHORS PUBLICATION

[1] Hnin Nwe Phyo, Myo Khaing, “Diagnosis Classification of Soybean Disease Using Machine Learning Techniques”, Parallel and Soft Computing (PSC), UCSY, Yangon, Myanmar, 2022.

REFERENCES

[1] Aditya Pratap Indian Institute of Pulses Research 136 PUBLICATIONS 1,901 CITATIONS, Ramesh Solanki Central Arid Zone Research Institute (CAZRI) 57 PUBLICATIONS 611 CITATIONS, Jitendra Kumar Indian Institute of Pulses Research 162 PUBLICATIONS 2,480 CITATIONS. "Soybean" January 2012 DOI: 10.1007/978-1-4614-0356-2_12.

[2] AK Jain, RPW Duin, Jianchang Mao Statistical pattern recognition: a review. IEEE Trans Pattern Analysis and Machine Intelligence - 2000. 22(1):4–37.

[3] C. Andrew, "Buiding Decision Trees with the ID3 Algorithm", Dr. Dobbs Journal, Jun 1996.

[4] C. Aggarwal, "Towards Effective and Interpretable Data Mining by Visual Interaction", ACM SIGKDD Exploration Vol., pp.11-12. 2002.

[5] Cover TM, Hart PE. Nearest neighbor pattern classification. IEEE Transactions on Information Theory. 1967;13(1):21–27.

[6] EE I9900 - Master's Thesis Submitted in partial fulfillment of the requirement for the degree Master of Engineering (Electrical) Spring 2017 At The City College of New York "Brief Study of Classification Algorithms in Machine Learning" Ramesh Sankara Subbu CUNY City College.

[7] I. Rish. An empirical study of the naive Bayes classifier. IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence, 22: 41-46, 2001.

[8] J. Han and J. Kamber, "Data Mining Concepts and Techniques", Morgan Kaufmann Publishers, USA,2001.

[9] Kusrini and Luthfi ET 2009 Algoritma Data Mining (Andi Offset, Yogyakarta)

[10] Minarni, Indra Warman, Yuhendra Teknik Informatika, Institut Teknologi Padang, Indonesia "Implementation of Case-Based Reasoning and Nearest Neighbor Similarity for Peanut Disease Diagnosis" to cite this article: Minarni et al 2019 J. Phys.: Conf. Ser. 1196 012053.

[11] MucherinoPetraqPapajorgji, P. M. Pardalos, "A survey of data mining techniques applied to agriculture", Springer, 2009.

[12] Mitchell, T. (1997). Machine Learning. McGraw Hill. p. 2. ISBN 0-07-042807-7.

[13] M. Tom Mitchell, “Machine Learning”, Singapore,WCB McGraw-Hill, 52-81,1997 app.

[14] Peter Harrington. (2010). Machine Learning in Action. Manning Publications Co. ISBN 9781617290183.

[15] Quinlan, J. R. (1986). Induction of decision trees. Machine Learning. 1, 81-106.

[16] Ramesh Sankara Subbu “Brief Study of Classification Algorithms in Machine Learning” EE I9900 - Master’s Thesis Submitted in partial fulfillment of the requirement for the degree Master of Engineering (Electrical) Spring 2017.

[17] R.S, Michal ski and R.L. Chilausky “An Experimental Comparison of the Two Methods of Knowledge Acquisition in the Context of Developing an Expert System for Soybean Disease Diagnosis” Vol.4, No.2, 1980.

[18] R.O. Zaiane, “Introduction to Data Mining”, COMPUT690 Principles of Knowledge Discovery in Databases.

[19] S. M. Kamruzzaman. Text Classification using Artificial Intelligence. Journal of Electrical Engineering, 33, No. I & II, December 2006.

[20] UCI Machine Learning Repository Irvine, CA: University of California, School of Information and Computer Science.

[21] Vinita Shah, Prachi Shah “Groundnut Crop Yield Predication Using Machine Learning Techniques” © 2018 IJSRCSEIT | Volume 3 | Issue 5 | ISSN: 2456-3307.

[22] Witten IH and Frank E 2005 Data Mining: Practical Machine Learning Tools and Techniques (Morgan Kaufmann, San Francisco).