BİLGİ KEŞFİ VE İRİS VERİ SETİ ÜZERİNDE VERİ MADENCİLİĞİ ARAÇLARININ KARŞILAŞTIRILMASI

KNOWLEDGE DISCOVERY AND A COMPARISON OF DATA MINING TOOLS ON IRIS DATASET

DİDEM TOKMAK

Thesis Submitted
in Partial Fulfillment of Requirements
for the Degree of Master of Science
in Department of Computer Engineering
at Başkent University

2011

Institute of Science and Engineering

This thesis has been approved in partial fulfillment of the requirements for the degree of MASTER OF SCIENCE IN COMPUTER ENGINEERING by committee members.

Chairman (supervisor) : Prof. Dr. A. Ziya AKTAŞ

Member : Assoc. Prof. Dr. Hasan OĞUL

Member : Asist. Prof. Dr. Pınar ŞENKUL

Approval

This thesis is approved by committee members on …../…../ 2011

..../..../ 2011

Prof. Dr. Emin AKATA

Director of Institute of Science and Engineering

ÖZ

BİLGİ KEŞFİ VE İRİS VERİ SETİ ÜZERİNDE VERİ MADENCİLİĞİ ARAÇLARININ KARŞILAŞTIRILMASI

Didem TOKMAK
Başkent Üniversitesi Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Günümüzde birçok alan ile ilgili ve çeşitli veritabanlarında tutulan çok fazla miktarda veri bulunmaktadır. Bu büyük veri yığınlarından birbirleri ile ilişkili, anlamlı, önceden bulunmamış veya bilinmeyen bilgiler çıkarmak bilgi keşfi (veya araması) dediğimiz sürecin temel amacıdır. Bilgi keşfi sürecinin en önemli adımlarından birisini de veri madenciliği oluşturur. Bir bakıma veri madenciliğini bilgi keşfinin bir aracı olarak tanımlamak da mümkündür. Bu tez çalışmasında, "veri madenciliği" ve "veritabanlarında bilgi keşfi" adı ile bilinen iki yaklaşım arasındaki ilişkiyi incelemek ve irdelemek temel amaçtır. Gereken inceleme ve araştırmalardan sonra çeşitli veri madenciliği yazılımları kullanılarak "IRIS veri seti" denilen bir veri demeti üzerinde veri madenciliği uygulaması yapılmıştır. Yapılan bu uygulamalar sonucunda bu yazılımlar karşılaştırılmıştır. Çalışmanın özü ileride yapılacak çalışmalar için bir ön adım oluşturmaktır.

ANAHTAR SÖZCÜKLER: Veri Madenciliği, Veritabanlarında Bilgi Keşfi, Sınıflandırma, Karar Ağaçları
Danışman: Prof. Dr. A.Ziya AKTAŞ, Başkent Üniversitesi, Bilgisayar Mühendisliği Bölümü

ABSTRACT

KNOWLEDGE DISCOVERY AND A COMPARISON OF DATA MINING TOOLS ON IRIS DATASET

Didem TOKMAK
Baskent University Institute of Science and Engineering
Department of Computer Engineering

Nowadays, there are large amount of data being kept on many databases that are related to many areas. One of the basic aims of KDD (knowledge discovery in databases) is to extract the knowledge from these large amount of data associated with each other, meaningful and not found before. During the process of KDD, the resulting such information or rather knowledge by interpreting and combining with other knowledge if necessary, is knowledge discovery in databases. DM (data mining) on the other hand is one of the crucial steps of KDD. In this thesis study, main objective is to show the relationship between knowledge discovery in databases (KDD) and data mining (DM). After the needed review and research, a data mining application on an available data which is called “IRIS dataset” is performed using some data mining tools. Results of these performed applications are compared with each other. The key objective of the study is to prepare an initial step for further applications with real data.

KEYWORDS: Classification, Data Mining, Decision Trees, Knowledge Discovery in Databases
Supervisor: Prof.Dr.A.Ziya AKTAŞ, Baskent University, Department of Computer Engineering

TABLE OF CONTENT

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
BIRCH	Balanced Iterative Reducing and Clustering using Hierarchies
CRISP-DM	Cross- Industry Standard Process for Data Mining
CTQ	Critical to Quality
DFSS	Design for Six Sigma
DM	Data Mining
DMADV	Define, Measure, Analyse, Design, Verify
DMAIC	Define, Measure, Analyse, Improve, Control
FN	False Negative
FP	False Positive
KDD	Knowledge Discovery in Databases
ODM	Oracle Data Miner
SEMMA	Sample, Explore, Modify, Model, Assess
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
WEKA	Waikato Environment for Knowledge Analysis

1. INTRODUCTION

1.1 Statement of the Problem

Data, information and knowledge are closely related but different terms. Comparing these three for their meaning value it has been concluded that knowledge has the highest value as noted already by AKTAŞ [1], BECERRA-FERNANDEZ et al. [2], AWAD and GHAZIRI [3] and others.

The huge volume of data produced in parallel with the technological developments of Information and Communications Technologies (ICT) field are being stored in the large databases and data warehouses. Trying to find the valuable knowledge buried in these data oceans has been the subject of recent research works.

There are two trends in these research works: one under the title of ‘knowledge discovery in databases’ and other under title of ‘data mining’. Although there are some claims that they are the same, it is the belief that they are not exactly the same is the key reason of this research.

During the thesis work, first knowledge discovery in databases and later data mining topics are elaborated and their interrelationships are defined.

1.2 Previous Work and Objectives of the Study

In the literature survey stage of this study, a number of references about data mining and knowledge discovery in databases have been studied and some sample data sets are investigated. Some of these references are summarized in the following sections.

BECERRA-FERNANDEZ et al. [2] had a chapter on knowledge discovery systems in their book on knowledge management. Knowledge management is defined as having four basic processes such as Knowledge Discovery, Knowledge Capture, Knowledge Sharing and Knowledge Application. The authors claim that knowledge discovery technologies can be very powerful for organizations wishing to obtain an advantage

over their competition. They define knowledge discovery in databases as the process of finding and interpreting patterns from data, involving the application of algorithms to interpret the patterns generated by these algorithms. They also state that although the majority of the practitioners use KDD and DM interchangeably, for some, KDD is defined to involve the whole process of knowledge discovery including the application of DM techniques.

LUTZ [4] had a relatively recent book on the knowledge discovery with support vector machines (SVM) algorithm. In Chapter 1 of his book, LUTZ defined data mining as a specific kind of knowledge discovery process that aims at extracting information (knowledge) from databases. He also added that knowledge discovery is a highly interdisciplinary undertaking ranging from domain analysis, data cleansing and visualization to model evaluation and deployment.

BANDYOPADHYAY and et al. [5] presented the basic concepts and basic issues of KDD in their book. They discussed the challenges that data mining researchers are facing. They also made a review of recent trends in knowledge discovery such as Content-based Retrieval, Text Retrieval, Image Retrieval, Web mining, Biological data mining, Distributed data mining, Mining in sensor, Peer-to-peer Networks, Case-based reasoning and mining techniques based on soft computing approaches.

MAIMON and ROKACH [6] discussed data mining and knowledge discovery in databases extensively in their handbook published in 2005. The handbook also included a chapter on data mining in medicine and another on data mining for software testing.

MAIMON and ROKACH [7] discussed decomposition methodology for knowledge discovery and data mining in their book. The book has a chapter where taxonomy of data mining methods is given.

As claimed by FAYYAD and et al [8] data mining and knowledge discovery in databases have been attracting a significant amount of research, industry, and media attention during the last decades since the late nineties. Their article provides an overview of this emerging field, clarifying how data mining and knowledge discovery

in databases are related both to each other and to related fields, such as machine learning, statistics, and databases. The article included particular real-world applications, specific data-mining techniques, challenges involved in real-world applications of knowledge discovery, and current and future research directions in the field.

An interesting study is reported by GOEBEL and GRUENWALD [9] which includes a feature classification scheme to study knowledge and data mining software. The scheme is based on software's general characteristics, database connectivity and data mining characteristics. It is applied on 43 software products which are either research prototypes or commercially available.

1.3 Organization of the Thesis

In the thesis, Chapter 2 provides background on the methodology known as KDD (Knowledge Discovery in Databases).

Chapter 3 presents information on Data Mining (DM) giving an introduction of Data Mining and presenting its tasks. The confusing KDD and Data Mining relationship is later debated. This chapter also presents available methodologies or processes on Knowledge Discovery and Data Mining. One of the methodologies named as KDD will be discussed in Chapter 2 of the thesis. Other methodologies are CRISP-DM[1], SEMMA[2] and Six Sigma[3]. All these methodologies are explained in that chapter and later compared to each other in a table.

Chapter 4 contains a summary of data mining tools: Oracle Data Miner, WEKA, R, RapidMiner, and ToolDiag.

Chapter 5 explains the data set, namely "IRIS dataset" which is used during the thesis study.

---

[1] http://www.crisp-dm.org/CRISPWP-0800.pdf
[2] http://www.sas.com/offices/europe/uk/technologies/analytics/datamining/miner/semma.html
[3] http://web.archive.org/web/20080723015058/http://www.onesixsigma.com/node/7630

Chapter 6 provides application with various DM tools on IRIS[4] dataset. Applications are explained step by step for all tools.

Chapter 7 explains the discussion of results obtained by data mining software tool applications and their comparison.

Chapter 8 presents the summary and conclusions of the thesis and also explains the extension of the study.

[4] http://archive.ics.uci.edu/ml/datasets/Iris

2. KNOWLEDGE DISCOVERY IN DATABASES

Knowledge discovery may be defined as a process of picking up useful information from huge amount of data that are too much to be investigated manually as noted by LUTZ [4].

KDD is developed as a combination of research in databases, machine learning, pattern recognition, statistics, artificial intelligence, expert systems, information retrieval, signal processing, high performance computing and networking as noted by BANDYOPADHYAY, et al [5].

According to the FAYYAD and et al. [10] the knowledge discovery process is iterative and interactive, consisting of nine steps which are named as Domain Understanding and KDD Goals, Selection and Addition of Data a Set, Pre-processing and Data Cleaning, Data Transformation, Data Mining Task, Choosing the Data Mining Algorithm, Employing the Data Mining Algorithm, Evaluation and Interpretation, and Discovered Knowledge, respectively as given in Figure1. An interesting comment made by FAYYAD et al. is that “One cannot present one formula or make a complete taxonomy for the right choices for each step and application type.”

In the following subsections each of these steps are summarized referring to MAIMON and ROKACH [6].

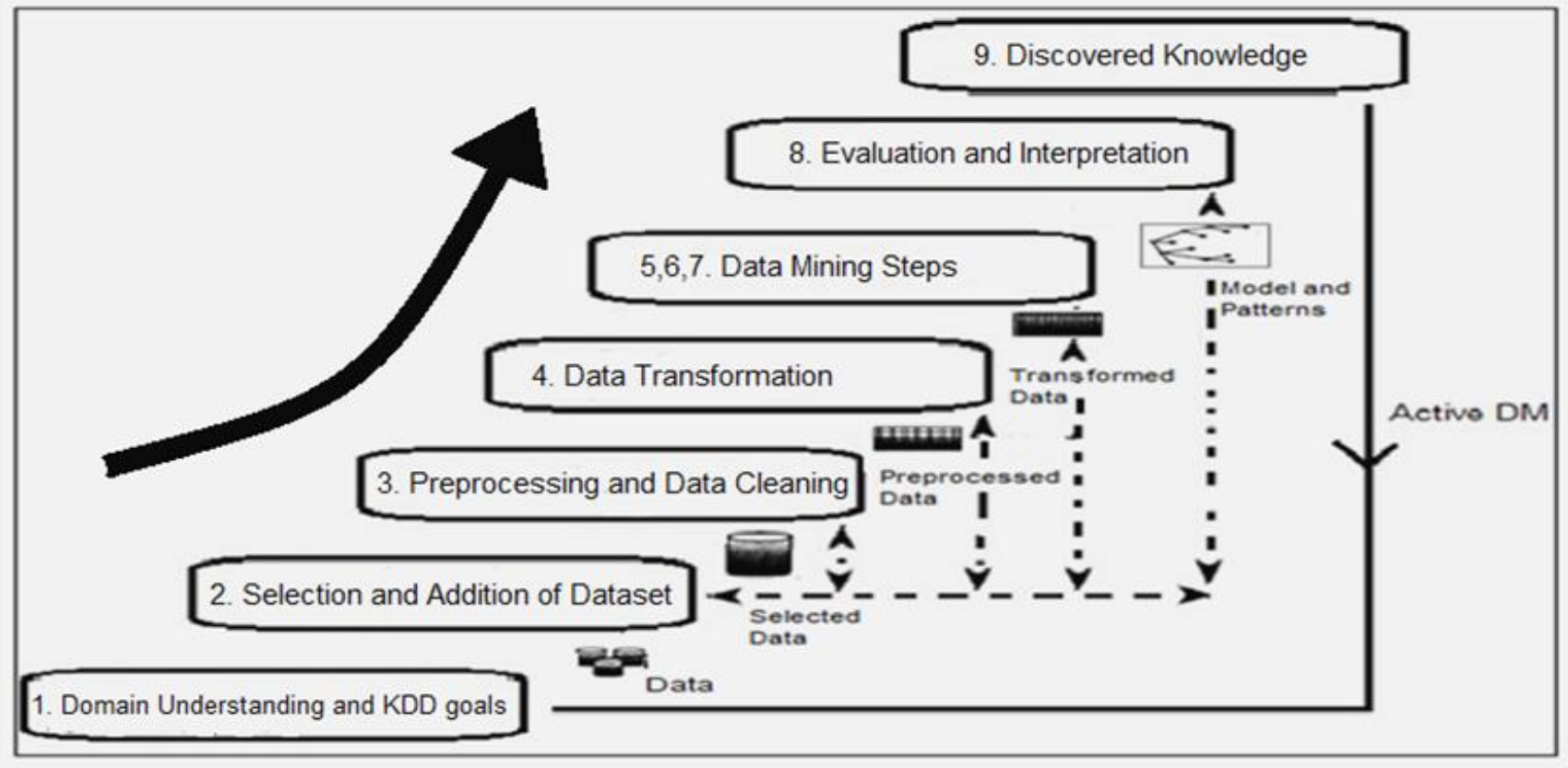


Figure 1. The Process of Knowledge Discovery in Databases

2.1 Domain Understanding and KDD Goals

While charging KDD project, one needs to understand and define the relevant prior knowledge and the goals of the end-user and the environment.

2.2 Selection and Addition of Data Set

Selection of which data is available and which data is necessary to add, and then aggregate all the data for the knowledge discovery into one data set is the objective of the this step. Data set includes the attributes that will be considered for the process. If some important attributes are missing, then the entire study may fail.

2.3 Pre-processing and Data Cleaning

In this step, data reliability is improved. To make this, data is cleaned, that is missing values are handled and removal of noise or outliers to get reliable study are performed.

2.4 Data Transformation

In this step, in order to prepare data for Data Mining, data transformation methods that include dimension reduction (such as feature selection and extraction and record sampling), and attribute transformation (such as discretization of numerical attributes and functional transformation) are applied.

2.5 Data Mining Task

In this step, which type of Data Mining to use for data set is to be decided. One chooses one of two major types in Data Mining: prediction or description.

2.6 Choosing the Data Mining Algorithm

In this step, the specific method to be used for searching patterns of selected Data Mining type is selected.

2.7 Employing the Data Mining Algorithm

Data Mining algorithm is reached and it is needed to employ several times until a suitable result is gained.

2.8 Evaluation and Interpretation

In this step, gained results of mined patterns are evaluated and interpreted according to the goals defined in the first step. Then discovered knowledge is documented for usage.

2.9 Discovered Knowledge

In this step, the knowledge becomes active in the sense into another system for action that may make changes to the system and measure the effects. Generally the success of this step relates the effectiveness of the entire KDD process.

3. DATA MINING

Data Mining is the technology of exploring data in order to discover previously unknown patterns in huge amounts of datasets. Data Mining can also be considered as a central step of the overall process of the KDD process as noted by MAIMON and ROKACH [7].

Data Mining process of discovering unknown and potentially useful patterns from large amounts of data is performed with different types of tools such as associations, sub graphs, trees, anomalies and regression. Discovered patterns being interesting or useful are related with the domain and interested user. It is true that, such information may be valuable for one user and completely useless to another.

As defined by BANDYOPADHYAY, et al. [5] a model, a preference criterion and search algorithm are the three components of any Data Mining techniques. The flexibility of the model for representing the underlying data and the interpretability of the human model in human terms is determined by model representation.

The preference criterion is used to determine, depending on the underlying data set, which model to use for mining, by associating some measure of goodness with the model functions.

Finally, once the model and the preference criterion are selected, specification of the search algorithm is defined in terms of these along with the given data.

3.1 Tasks in Data Mining

Data Mining tasks are mostly divided into two main categories such as discovery-oriented and verification-oriented. Discovery oriented methods are divided into two as descriptive and predictive techniques. The descriptive techniques support a summary of the data and characterize its general properties to be used for some conclusions. Predictive techniques on the other hand learn from the data to make predictions

about the behaviour of new data set. Verification oriented methods that are common statistics methods are the evaluation of a hypothesis proposed by an external source. Figure 2 shows the classification of Data Mining that is illustrated as referring the referring to MAIMON and ROKACH [6].

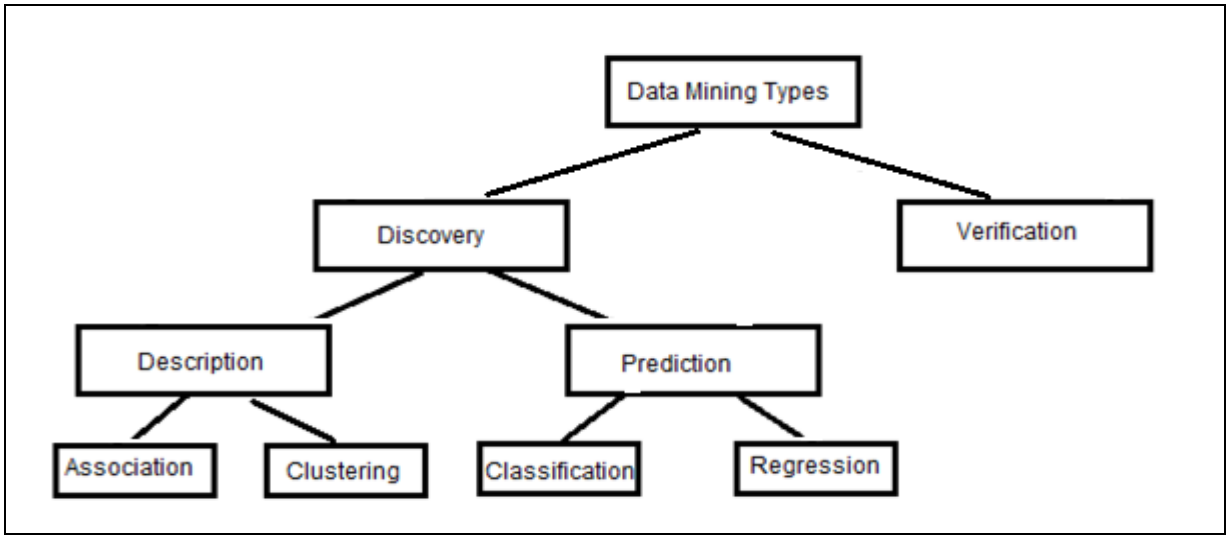


Figure 2. Classification of Data Mining

3.1.1 Descriptive techniques

In the following some of the descriptive techniques are summarized referring to MAIMON and ROKACH [6]:

a) Association

Association rule mining task can be named as market basket or transaction data analysis. Association rules are finding with given a set of transactions that predicting occurrence of an item based on occurrences of other items in transaction data set.

Some association rule mining techniques may be defined as:

a) Find all rules having in given set of transaction

b) Generate strong association rules from finding rules

b) Cluster Analysis

Cluster analysis can be applied dataset having a set of attributes and similarity measure among them. Data points is finding in one cluster are more similar to another or separating in less similar to one another. The process of clustering is shown in Figure3 as given by KOUTSONIKOLA et al. [11].

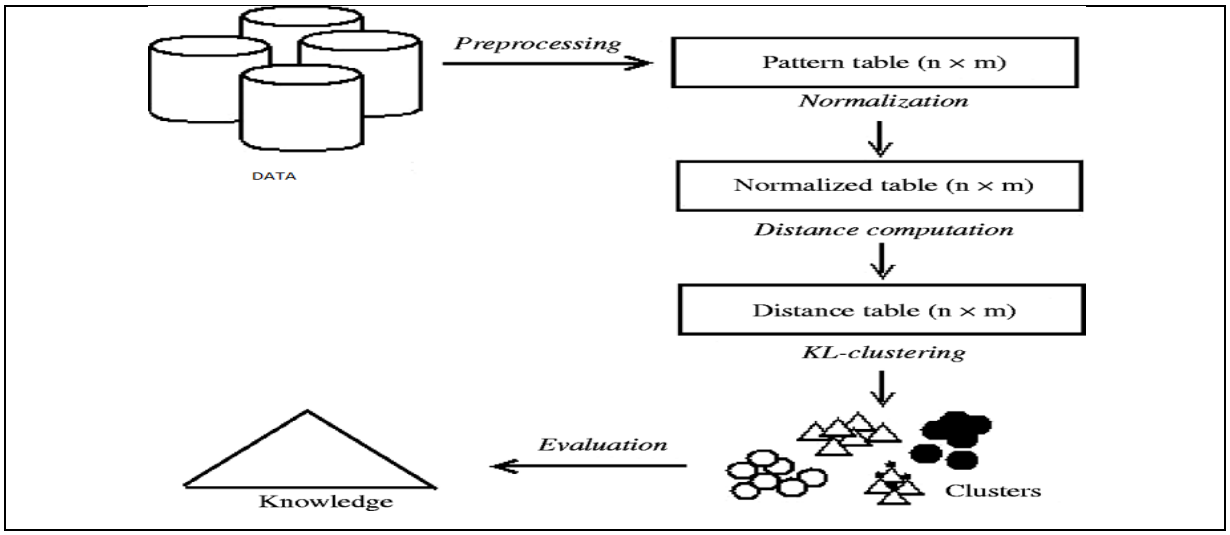


Figure 3. The Process of Clustering

Some clustering task algorithms are given below:

- Euclidean distance
- *K*-means algorithm
- CLARA
- Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH)

3.1.2 Predictive techniques

In the following some of the predictive techniques are summarized referring to MAIMON and ROKACH [6]:

a) Classification

A typical pattern recognition system consists of three phases such as acquisition, feature extraction and classification respectively. In the data acquisition phase, depending on the environment within which the objects are to be classified, data are gathered using a set of sensors. These are then passed on to the feature extraction phase, where the dimensionality of the data is reduced by measuring/retaining only some characteristic features or properties. Finally, in the classification phase, the extracted features are passed on to the classifier that evaluates the incoming information and makes a final decision.

Classification task finds a model for class attribute as a function of the values of other attributes. Datasets are divided into training and test set, with training data set used to build a model and test set used to validate the finding model. Figure 4 shows the process of classification.

The problem of classification is basically one of partitioning the feature space into regions, one region for each category of input. Classifiers are usually, but not always, designed with labelled data, in which case these problems are sometimes referred to as supervised classification.

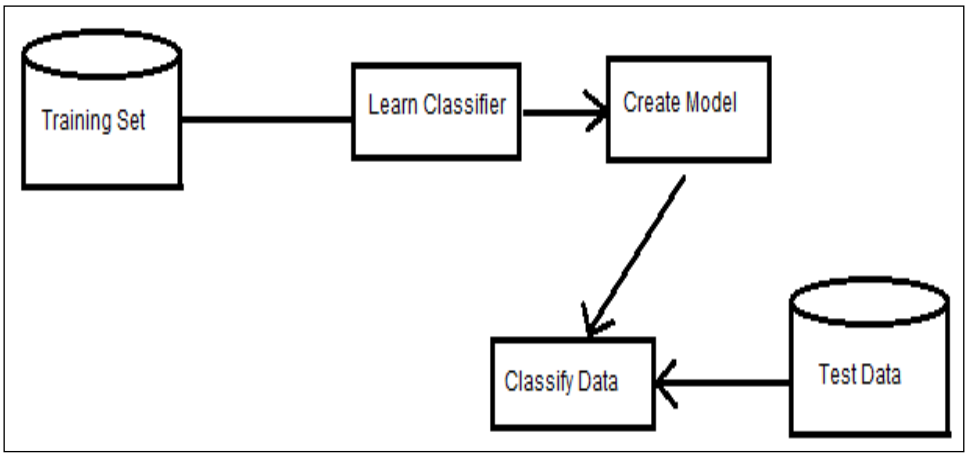


Figure 4. The Process of Classification

Some of the commonly used classification algorithms are given below:

- Decision Trees
- Naïve Bayes
- Support vector machines
- Rule-based methods

Since decision tree algorithm is relatively simple and commonly used, during this study, decision tree algorithm is used. It generates rules which are conditional statements that can easily understand and used within a database to release dataset. Support and confidence are the metrics which measure the strength of the rules. Support determines how often a rule applicable to a given data set. On the other hand, confidence determines how frequently items in Y appear in transactions that contain X.

$$Support(X \rightarrow Y) = \frac{number\ \ of\ X\ and\ Y}{total} \tag{3.1}$$

$$Confidence(X \rightarrow Y) = \frac{number\ \ of\ X\ and\ Y}{number\ \ of\ X} \tag{3.2}$$

There are many measures that can be used to determine best split of the decision tree. Best split are often based on the degree of impurity of child nodes. Some examples of impurity measures are defined such as gini, entropy and classification error.

Equation for gini index for a given node t where $p(j/t)$ is the relative frequency of class j at node t is given by TAN et al. [12].

$$Gini(t) = 1 - \sum_j [p(j|t)]^{\,2} \tag{3.3}$$

Entropy is measure of information or the degree of randomness in data. Equation for entropy index for a given node t where $p(j/t)$ is the relative frequency of class j at node t.

$$Entropy(t) = -\sum_j p(j|t) \log_2 p(j|t) \tag{3.4}$$

Classification error measures misclassification error made by a node. Equation for error index for a given node t where $p(j/t)$ is the relative frequency of class j at node t.

$$Error(t) = 1 - \max_{j} [p(j|t)] \quad (3.5)$$

b) Regression

Regression is a technique used to predict a numerical variable while building an equation to a dataset. The simplest form of regression is linear regression which uses the formula of a straight line (f = ax + b) and determines the suitable values for 'a' and 'b' to predict the value of f by using a given value of x. These relationships between predictors and given value of target are composed in a regression model, which can then be applied to a different dataset in which the target values are unknown. Figure 5 shows the process of regression[5].

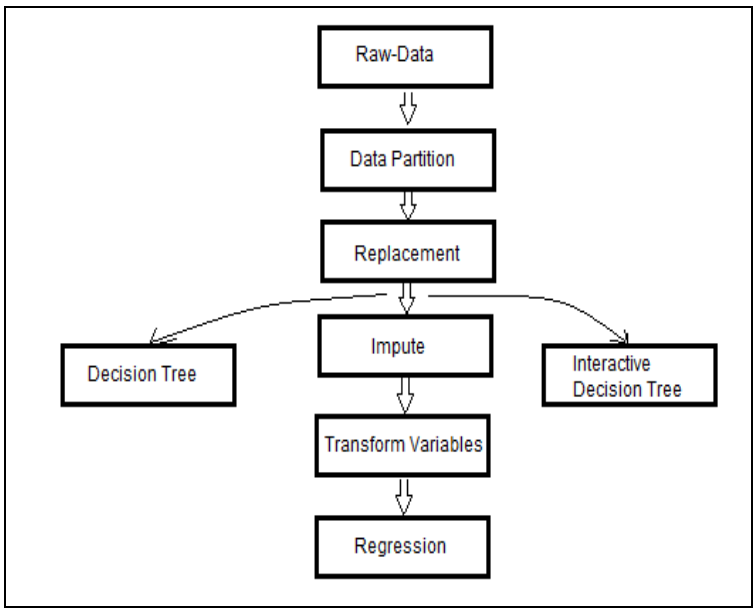


Figure 5. The Process of Regression

[5]http://support.sas.com/documentation/cdl/en/emgsj/61207/HTML/default/viewer.htm#p0tad07m88xmotn1c78zunl7sm5i.htm

3.2 A Comparison of Knowledge Discovery in Databases(KDD) and Data Mining(DM)

There are a variety of names that may be given to discovering useful patterns in data. These are data mining, data archaeology, data pattern processing, knowledge extraction, information discovery and information harvesting. Data Mining is the most widely used term.

As noted earlier, to elaborate the difference between KDD and DM is the one of the major objectives of this study.

Referring to Figure 1 in Chapter 1, one notes that KDD is the whole process of the discovering the useful information in large databases. On the other hand, Data Mining (DM) is the particular set of steps in such a KDD process. Data Mining uses specific algorithms to find the model from data.

KDD has a relationship between research areas such as artificial intelligence, pattern recognition, databases, statistics, machine learning, and knowledge acquisition for expert systems. Its goal is take out the high level knowledge from meaningless data in the large datasets. Data Mining, as a part of the KDD, provides patterns by using known algorithms in the steps of KDD process already as shown in Figure1. Data Mining consists of applying data analysis and discovery algorithms which are tasks of Data Mining.

3.3 Available Methodologies for Data Mining

There exist different methodologies for Data Mining in order to perform KDD. Most common of them are CRISP-DM, SEMMA and Six Sigma methodologies.

3.3.1 CRISP-DM (Cross- Industry Standard Process for Data Mining)

CRISP-DM[6] methodology for Data Mining provides a life cycle that is an overview of the project that was conceived in late 1996. It has six stages in life cycle. Figure 6[6] shows that the phases of life cycle methodology. The sequence of the phases is not strict. Moving back and forth between each phases, that is, iteration is allowed.

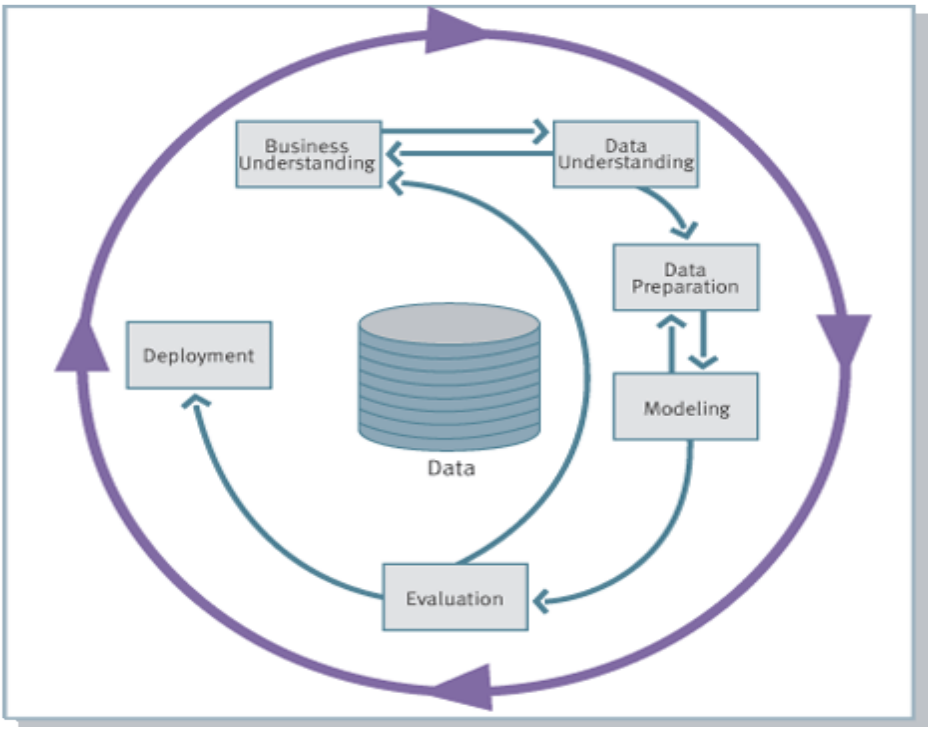


Figure 6. CRISP-DM Life Cycle

In the following subsections the six stages of the CRISP-DM are summarized:

a) Business understanding

This initial phase focuses on understanding the project requirements from a business side, and then converting this knowledge into a Data Mining problem definition, and a preliminary plan designed to achieve the objectives.

[6] http://www.crisp-dm.org/CRISPWP-0800.pdf

b) Data understanding

The data understanding phase starts with an initial data collection and proceeds with activities in order to get familiar with the data, to identify data quality problems, to discover first insights into the data, or to detect interesting subsets to form hypotheses for hidden information.

c) Data preparation

The data preparation phase covers all activities to construct the final dataset from the initial raw data.

d) Modeling

In this phase, various modeling techniques are selected and applied, and their parameters are calibrated to optimal values.

e) Evaluation

At this stage in the project you have built a model that appears to have high quality, from a data analysis perspective. Before proceeding to final deployment of the model, it is important to more thoroughly evaluate the model, and review the steps executed to construct the model, to be certain it properly achieves the business objectives. A key objective is to determine if there is some important business issue that has not been sufficiently considered. At the end of this phase, a decision on the use of the Data Mining results should be reached**.**

f) Deployment

Creation of the model is generally not the end of the project. Depending on the requirements, the deployment phase can be as simple as generating a report or as complex as implementing a repeatable Data Mining process.

3.3.2 SEMMA (Sample, Explore, Modify, Model, Assess)

The SAS Company claims that SEMMA[7] is not a Data Mining methodology but rather a logical organization of the function tool set of SAS Enterprise Miner for core tasks of Data Mining. However, it may also be considered to use it is a methodology. SEMMA is a cycle with five stages for process. Figure 7 shows that the phases of the methodology.

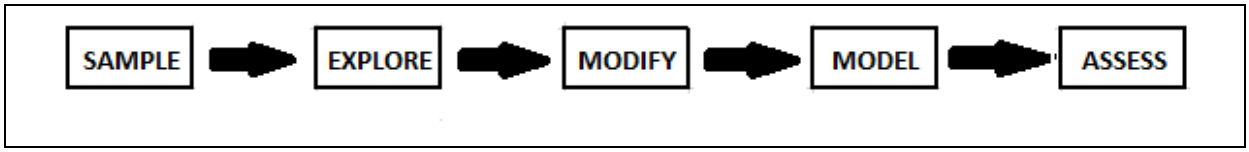


Figure 7. The Phases of SEMMA Methodology

In the following SEMMA phases are summarized:

a) Sample

Sample your data by extracting a portion of a large data set big enough to contain the significant information, yet small enough to manipulate quickly. Data partition nodes are: training which is used for model fitting, validation which is used for assessment and to prevent over fitting, test which is used to obtain an honest assessment of how well a model generalizes.

b) Explore

Explore your data by searching for unanticipated trends and anomalies in order to gain understanding and ideas. Exploration helps refine the discovery process.

c) Modify

Modify your data by creating, selecting, and transforming the variables to focus the model selection process. Based on your discoveries in the exploration phase, you

[7] http://www.sas.com/offices/europe/uk/technologies/analytics/datamining/miner/semma.html

may need to manipulate your data to include information such as the grouping of customers and significant subgroups, or to introduce new variables.

d) Model

Model your data by allowing the software to search automatically for a combination of data that reliably predicts a desired outcome.

e) Assess

Assess your data by evaluating the usefulness and reliability of the findings from the Data Mining process and estimate how well it performs. A common means of assessing a model is to apply it to a portion of data set aside during the sampling stage.

3.3.3 Six Sigma

Six Sigma[8] is a business management strategy originally developed by Motorola. Six Sigma seeks to improve the quality of process outputs by identifying and removing the causes of errors and minimizing variability in manufacturing and business processes.

It uses a set of quality management methods, including statistical methods, and creates a special infrastructure of people within the organization.

Six Sigma projects follow two project methodologies each composed of five phases called DMAIC (Define, Measure, Analyse, Improve, Control) and DMADV(Define, Measure, Analyse, Design, Verify) as noted by DE et al [13].

[8] http://web.archive.org/web/20080723015058/http://www.onesixsigma.com/node/7630

3.3.3.1 DMAIC (Define, Measure, Analyse, Improve, Control)

DMAIC is used for projects for improving an existing business process which is used in the projects for creating new product or process designs and it has the following phases as defined by DE et al. [13].

a) Define

This step is voice of the customer and the project goals, specifically.

b) Measure

Key aspects of the current process and collect relevant data.

c) Analyze

Investigating and verify cause-and-effect relationships. Determine what the relationships are, and attempt to ensure that all factors have been considered. Seek out root cause of the defect under investigation.

d) Improve

Improving the current process based on data analysis using techniques such as design of experiments or mistake proofing, and standard work to create a new, future state process.

e) Control

Controlling is the future state process to ensure that any deviations from target are corrected before they result in defects. Implement control systems such as statistical process control, production boards, and visual workplaces, and continuously monitor the process.

3.3.3.2 DMADV (Define, Measure, Analyse, Design, Verify)

The DMADV project methodology is also known as DFSS (“Design for Six Sigma”) and features five phases:

a) Define

Define design goals that are consistent with customer demands and the enterprise strategy.

b) Measure

Measure and identify CTQs (characteristics that are Critical to Quality), product capabilities, production process capability, and risks.

c) Analyze

Analyze to develop and design alternatives, create a high-level design and evaluate design capability to select the best design.

d) Design details

Optimize the design, and plan for design verification. This phase may require simulations.

e) Verify the design

Set up pilot runs, implement the production process and hand it over to the process owner(s).

As noted above DMADV has two phases on design such as design details and verify the design compared to DMAIC.

3.4 A Comparison of KDD, SEMMA, CRISP-DM and Six Sigma

Similar to AZEVEDO and SANTOS [14] and adding Six Sigma one may compare SEMMA, CRISP-DM and Six Sigma to KDD process as given in the Table1.

Table 1. Comparison between KDD, SEMMA, CRISP-DM and Six Sigma Methodologies

KDD	SEMMA	CRISP-DM	Six Sigma
Pre KDD		Business Understanding	Define
Selection	Sample	Data Understanding	Measure
Pre-processing	Explore		Analyze
Transformation	Modify	Data Preparation	
Data Mining	Model	Modelling	Improve
Interpretation	Assessment	Evaluation	Control
Post KDD		Deployment	

4. AN OVERVIEW OF DATA MINING TOOLS

4.1 Oracle Data Mining

Oracle Data Mining (ODM)[9] is an option to the Enterprise Edition of Oracle Database. Oracle Data Mining enables to easily build and deploy applications that deliver predictive analytics and new insights. It includes programmatic interfaces for SQL, PL/SQL, and Java. It also supports a spreadsheet add-in.

a) SQL Functions

The Data Mining functions are SQL language operators for the deployment of data mining models. They allow data mining to be easily incorporated into SQL queries, and thus into SQL-based applications. Application developers can rapidly build next-generation applications using ODM's SQL APIs that automatically mine Oracle data and deploy results in real-time-throughout the enterprise. Because the data, models and results remain in the Oracle Database, it is claimed by ORACLE that data movement is eliminated, security is maximized and information latency is minimized. Oracle Data Mining models can be included in SQL queries and embedded in applications to offer improved business intelligence.

b) Oracle Data Miner

Oracle Data Miner is the graphical user interface for Oracle Data Mining. Oracle Data Miner provides wizards that guide you through the data preparation, data mining, model evaluation, and model scoring process. Data analysts can quickly access their Oracle data using the optional Oracle Data Miner graphical user interface and explore their data to find patterns, relationships, and hidden insights. Oracle Data Mining provides a collection of in-database data mining algorithms that solve a wide range of business problems. Anyone who can access data stored in an Oracle Database can access Oracle Data Mining results-predictions, recommendations, and discoveries using SQL-based query.

---

[9] http://www.oracle.com/technetwork/database/options/odm/index.html#learnmore

c) PL/SQL Packages

The Oracle Data Mining PL/SQL API is implemented in the following PL/SQL packages:

- DBMS_DATA_MINING — Contains routines for building, testing, and applying data mining models.
- DBMS_DATA_MINING_TRANSFORM — Contains routines for transforming the data sets prior to building or applying a model. Users are free to use these routines or any other SQL-based method for defining transformations. The routines in DBMS_DATA_MINING_TRANSFORM are simply provided as a convenience.
- DBMS_PREDICTIVE_ANALYTICS — Contains automated data mining routines for PREDICT, EXPLAIN, and PROFILE operations.

d) Java API

The Oracle Data Mining Java API is an Oracle implementation of the JDM standard Java API for data mining. The Java API is layered on the PL/SQL API, and the two APIs are fully interoperable.

4.1.1 Oracle Data Mining Techniques and Algorithms

ODM provides several data mining techniques and algorithms to solve many types of business problems:

a) Classification

Classification is the most commonly used technique for predicting a specific outcome. It has four algorithms to create model such as Decision Tree, Naïve Bayes, Support Vector Machine (SVM) and Logistic Regression.

b) Regression

Regression is the technique for predicting a continuous numerical outcome such as customer lifetime value, house value, process yield rates. Multiple Regression and Support Vector Machine are the algorithms of regression technique.

c) Attribute Importance

This technique ranks attributes according to strength of relationship with target attribute. Use cases include finding factors most associated with customers who respond to an offer, factors most associated with healthy patients. Minimum Description Length is the algorithm that considers each attribute as a simple predictive model of the target class

d) Anomaly Detection

Anomaly Detection identifies unusual cases based on deviation from the norm. Common examples include health care fraud, expense report fraud, and tax compliance. One-Class Support Vector Machine is the algorithm of anomaly detection.

e) Clustering

Clustering is useful for exploring data and finding natural groupings. Members of a cluster are more like each other than they are like members of a different cluster. Enhanced K-Means and Orthogonal Partitioning Clustering are algorithms for clustering.

f) Association

Association technique finds rules associated with frequently co-occurring items, used for market basket analysis, cross-sell, and root cause analysis. Algorithm of the association is Apriori algorithm.

g) Feature Extraction

This technique produces new attributes as linear combination of existing attributes. It is applicable for text data, latent semantic analysis, data compression, data decomposition and projection, and pattern recognition. Non-negative Matrix Factorization algorithm used for next generation, maps the original data into the new set of attributes.

4.2 WEKA

"WEKA"[10] stands for the Waikato Environment for Knowledge Analysis, which was developed at the University of Waikato in New Zealand. WEKA is a free software. WEKA is extensible and has become a collection of machine learning algorithms for solving real-world data mining problems. It is written in Java and runs on almost every platform. The WEKA contains a collection of visualization tools and algorithms for data analysis and predictive modeling, together with graphical user interfaces for easy access to this functionality as noted by HOLMES and et al [15].

WEKA supports several standard data mining tasks, more specifically, data preprocessing, feature selection, classification, regression, clustering and visualization.

Besides actual learning schemes, WEKA also contains a large variety of tools that can be used for pre-processing datasets, so that you can focus on your algorithm without considering too much details as reading the data from files, implementing filtering algorithm and providing code to evaluate the results.

WEKA has some user interfaces such as Explorer, Knowledge Flow and Experimenter. Its main user interface is the Explorer. On the other hand, Knowledge Flow interface is including the command line. There is also the Experimenter, which allows the systematic comparison of the predictive performance of WEKA's machine learning algorithms on a collection of datasets.

[10] http://www.cs.waikato.ac.nz/ml/weka/

The Explorer interface has several panels that are Preprocess, Classify, Associate, Cluster, Select Attributes and Visualize. The Preprocess panel is for importing data from a file. The Classify panel enables to apply classification and regression algorithms to the resulting dataset, to estimate the accuracy of the resulting predictive model. The Associate panel provides access to association rule learners that attempt to identify all important interrelationships between attributes in the data. It has apriori algorithm. The Cluster panel gives access to the clustering techniques in WEKA, for example, simple k-means algorithm. The Select attributes panel provides algorithms for identifying the most predictive attributes in a dataset. The Visualize panel shows a scatter plot matrix and analysed further using various selection operators.

4.3 R

R[11] is a language and environment for statistical computing and graphics. It is a GNU (GNU's Not Unix) project which was developed at Bell Laboratories.

R provides a wide variety of statistical such as classification, clustering, linear and nonlinear modeling and graphical techniques, and is highly extensible.

One of R's strengths is the ease with which well-designed publication-quality plots can be produced, including mathematical symbols and formulae where needed. Great care has been taken over the defaults for the minor design choices in graphics, but the user retains full control.

R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS.

The R environment features are given below:

[11] http://cran.r-project.org/doc/manuals/R-intro.html

- an effective data handling and storage facility,
- a suite of operators for calculations on arrays, in particular matrices,
- a large, coherent, integrated collection of intermediate tools for data analysis,
- graphical facilities for data analysis and display either on-screen or on hardcopy, and
- a well-developed, simple and effective programming language which includes conditionals, loops, user-defined recursive functions and input and output facilities.

The term “environment” is intended to characterize it as a fully planned and coherent system, rather than an incremental expansion of very specific and inflexible tools, as is frequently the case with other data analysis software.

R allows users to add additional functionality by defining new functions. For computationally-intensive tasks, C, C++ and Fortran code can be linked and called at run time. Advanced users can write C code to manipulate R objects directly.

It is an environment within which statistical techniques are implemented. R can be extended via packages. There are many packages for data mining in R such as Party, e1071, RandomForest etc. Also it supports the all WEKA algorithms in package Rweka.

4.4 RapidMiner

RapidMiner[12] is an open source learning environment for data mining and machine learning. This environment can be used to extract meaning from a dataset. There are hundreds of machine learning operators to choose from, helpful pre and post processing operators, descriptive graphic visualizations, and many other features. It is written in the Java programming language and therefore can work on all popular operating systems. It also integrates learning schemes and attributes evaluators of the WEKA learning environment. RapidMiner was successfully applied on a wide range of applications where its rapid prototyping abilities demonstrated their

[12] http://rapid-i.com/content/view/181/190/lang,en/

usefulness, including text mining, multimedia mining, feature engineering, data stream mining and tracking drifting concepts, development of ensemble methods, and distributed data mining.

RapidMiner can be used as stand-alone program on the desktop with its graphical user interface (GUI), on a server via its command line version, or as data mining engine for your own products and Java library for developers.

This environment has a steep learning curve, especially for someone who does not have a background in data mining. It allows experiments to be made up of a large number of arbitrarily nestable operators, described in XML files which are created with RapidMiner's graphical user interface. RapidMiner is used for both research and real-world data mining tasks.

The Community Edition of RapidMiner's (formerly known as "YALE") strengths reside in part in its ability to easily define analytical steps (especially when compared with R), and in generating graphs more easily than e.g., R, or more effectively than MS Excel.

Some of important features of RapidMiner are given below:

1) RapidMiner is based on modular operator concept which facilitates rapid prototyping of data mining processes by way of nesting operator chains and using complex operator trees.

RapidMiner provides flexible operators for data input and data output in different file formats such as excel files, files, SPSS files, data sets from well known databases such as Oracle, mySQL, PostgreSQL, Microsoft SQL Server, Sybase, and dBase. It also accepts sparse file formats such as SVMight, mySVM; standard data mining and learning scheme formats such as csv, Arff, and C4.5.

2) RapidMiner follows a multi-layered data view concept which enables it to store different views on the same data table and therefore facilitates cascading multiple views in layers through a central data table. RapidMiner data core is typically similar to a standard database management system.

3) RapidMiner has a flexible interactive design which lets user to additional Meta data on the available data sets to enable automated search and optimized preprocessing which are both needed for an effective data mining processes.

4) RapidMiner also acts as a powerful scripting language engine along with a graphical user interface. Since using RapidMiner, data mining processes are designed as operator trees defined in XML, where operators are not defined in a graph layout so as to be positioned and connected by a user. Therefore data flow normally follows “depth first search,” resulting in optimization of data mining processes.

4.5 TOOLDIAG

TOOLDIAG[13] is a collection of methods for statistical pattern recognition. The main area of application is classification. The application area is limited to multidimensional continuous features, without any missing values. No symbolic features (attributes) are allowed. The program is implemented in the ‘C’ programming language and was tested in several computing environments. The user interface is simple, command-line oriented, but the methods behind it are efficient and fast.

Possibilities of TOOLDIAG are classification, feature selection, feature extraction, performance estimation and some statistics. Classification provide some algorithms such as K-nearest neighbor, linear machines, Q* algorithm and etc. Feature selection provides best features, sequential forward selection, sequential backward selection etc. Feature extraction algorithms are such as linear discriminate analysis. Several performance estimation methods can be combined with all available classifier paradigms such as cross validation methods, accuracy, precision etc.

Only databases with continuous attributes and no missing values are allowed as input to TOOLDIAG.

Application with TOOLDIAG cannot be able to apply because of limited information about TOOLDIAG software tool.

[13] http://sites.google.com/site/tooldiag/Home

5. APPLICATION DATA: IRIS DATASET

In this study, special data set named “IRIS dataset” which is taken from UC Irvine Machine Learning Repository[14] is used as an example application.

Iris is a garden flower. It has three types as iris-setosa, iris-versicolor and iris-virginica. Iris dataset has four attributes such as petal length, petal width, sepal length, sepal width of the Iris in centimetres. The sepals of a flower are the outer structures that protect the more fragile parts of the flower, such as the petals. In many flowers, the sepals are green, and only the petals are colourful. For Irises, however, the sepals are also colourful. As an example Figure 8 shows iris-versicolor.

Figure 8. Iris-versicolor

Dataset has 150 instances and these instances’ associated task is classification. The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant such as iris-setosa, iris-versicolour and iris-virginica. The data in the dataset follows same order as setosa-versicolor-virginica. Figure 9a and 9b show the partial data for iris dataset.

[14] http://archive.ics.uci.edu/ml/datasets/Iris

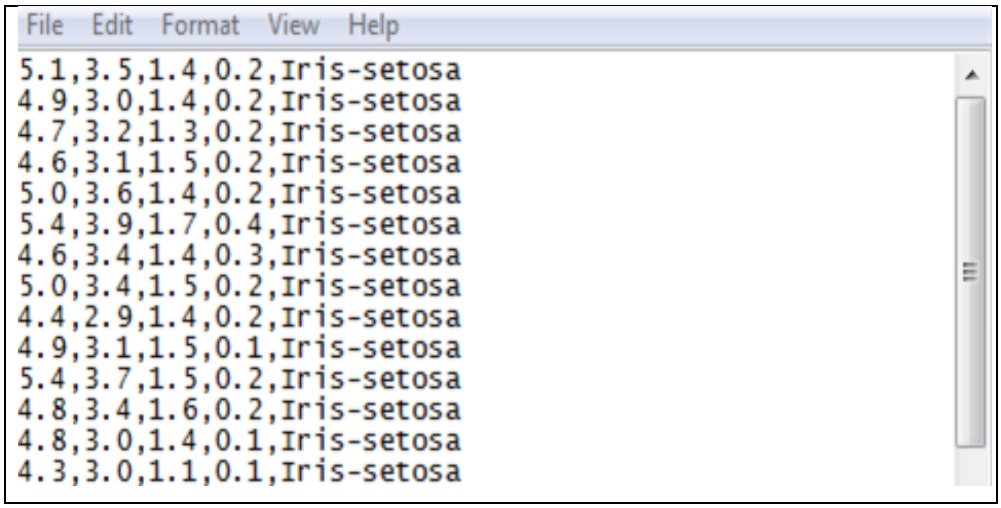


Figure 9a. Screen Shot of Partial Data for IRIS Dataset

Sepal_width	sepal_ length	petal_width	petal_ length	Class
5.1	3.5	1.4	0.2	Iris-setosa
4.9	3	1.4	0.2	Iris-setosa
6.1	3.0	4.6	1.4	Iris-versicolor
5.8	2.6	4.0	1.2	Iris-versicolor
6.4	3.1	5.5	1.8	Iris-virginica
6.0	3.0	4.8	1.8	Iris-virginica

Figure 9b. Partial Data for IRIS Dataset

Whole data is divided to three data sets. 90 of them are for training data set. 40 of them are for test data set and 20 of them are for applying data. This division is made randomly for the application.

6. APPLICATIONS

6.1 Introduction

In this chapter, IRIS dataset is used for data mining software tools. Data set includes the four attributes as noted in Chapter 5 and its attributes are not missing. In this study class attribute of IRIS is used for understanding how they are classified with three different types.

As noted in Chapter 3 Discovery-oriented methods are divided into two as descriptive and predictive. In this study, predictive technique is applied to find flowers' type from decision tree which will be applied with training dataset.

Feature extraction that is one of the data transformation methods is built with DM tools for IRIS training dataset to prepare data for data mining.

6.2 Application with Oracle Data Miner

Before starting the study, data set is imported in ODM. Training set, test set and apply set are imported in the same way. Figure 10 shows the import data to ODM that has Data tab and below of this tab select import and follow the directions of file import wizard. In this wizard, data is specified with field delimiter that may be comma, space etc.

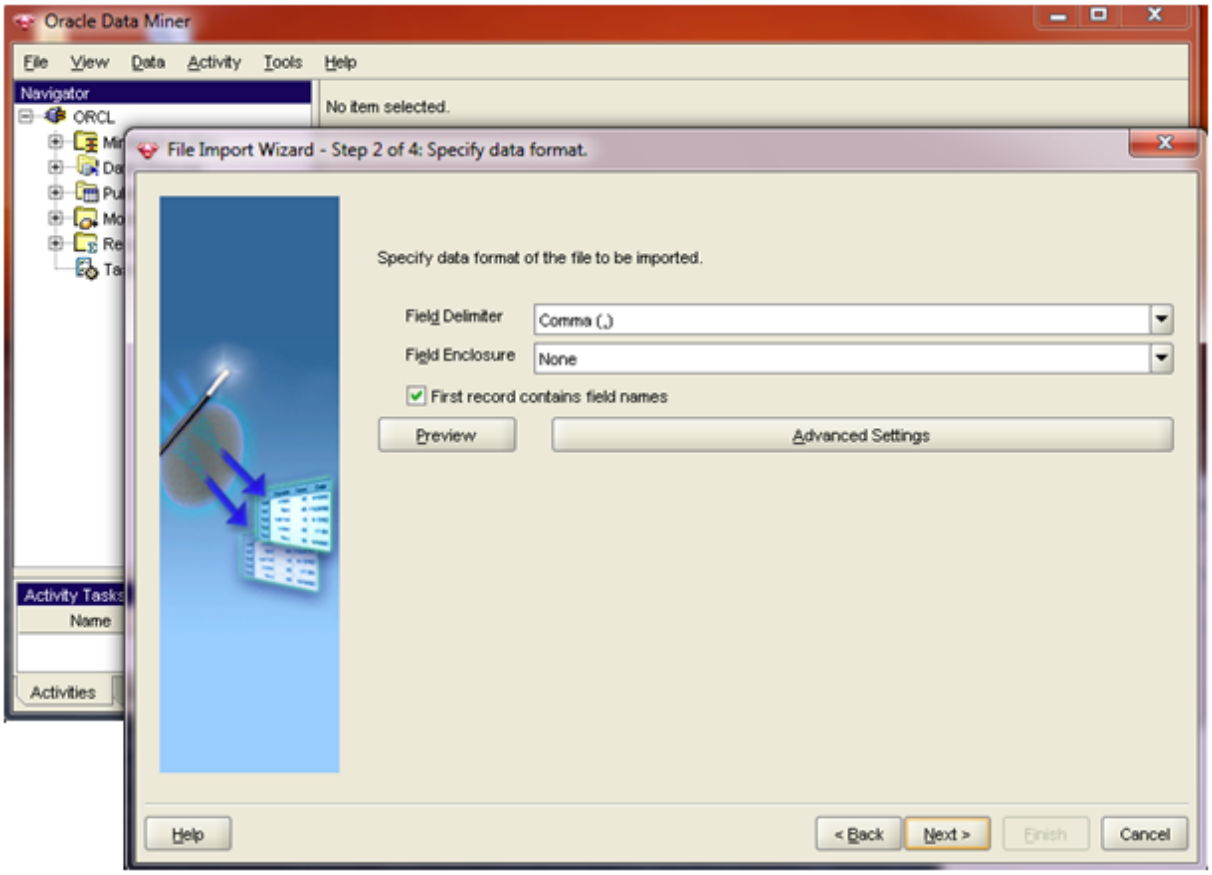


Figure 10. Import Data to ODM

Partial view including only 25 instances of the training dataset for model building that consists of attributes of data after imported in ODM is given as Figure 11.

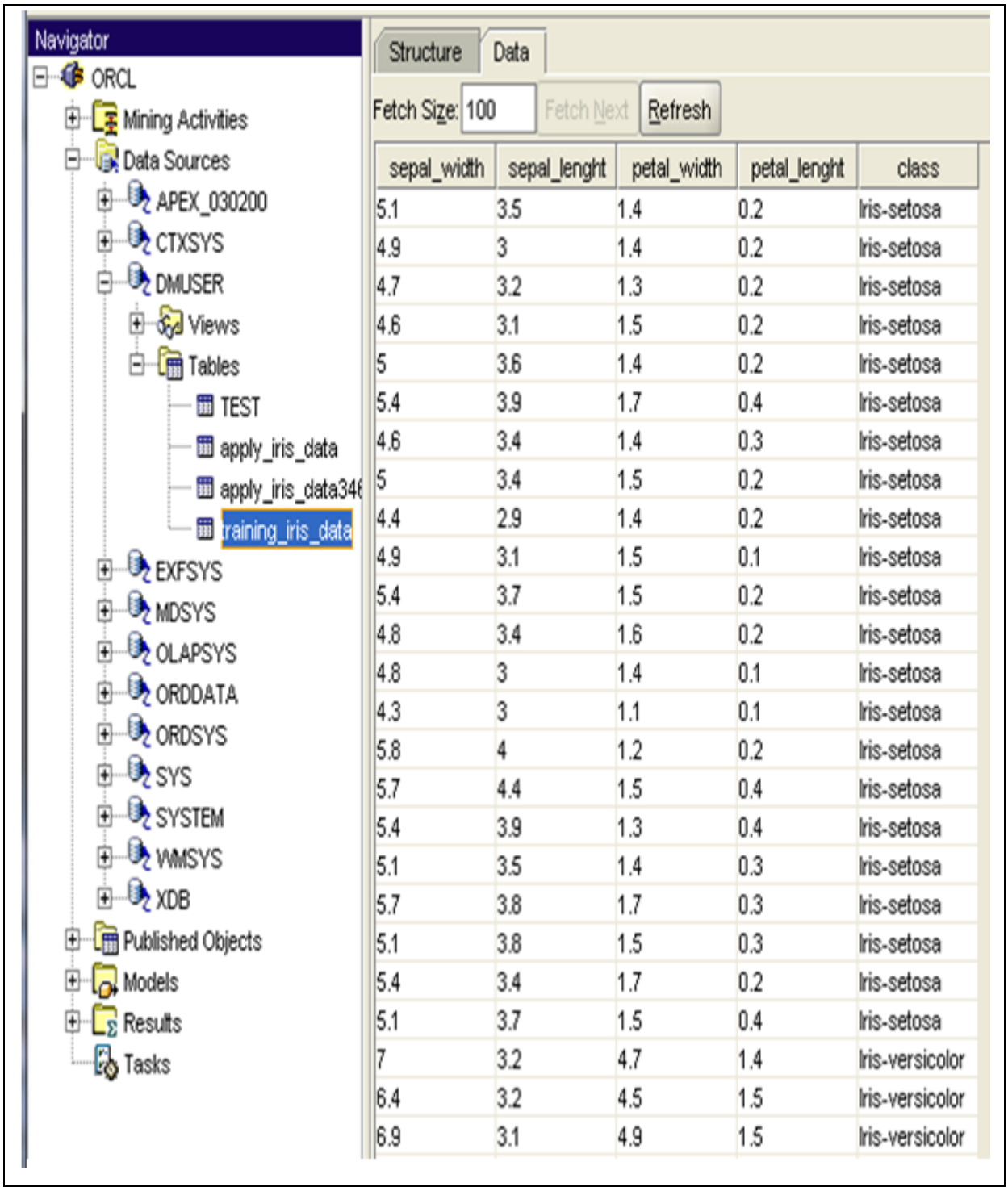


Figure 11. Partial View of the IRIS Dataset and Its Attributes

After importing the training set, feature extraction which stands in activity – build tab is built. Figure 12 shows the mining activity screen.

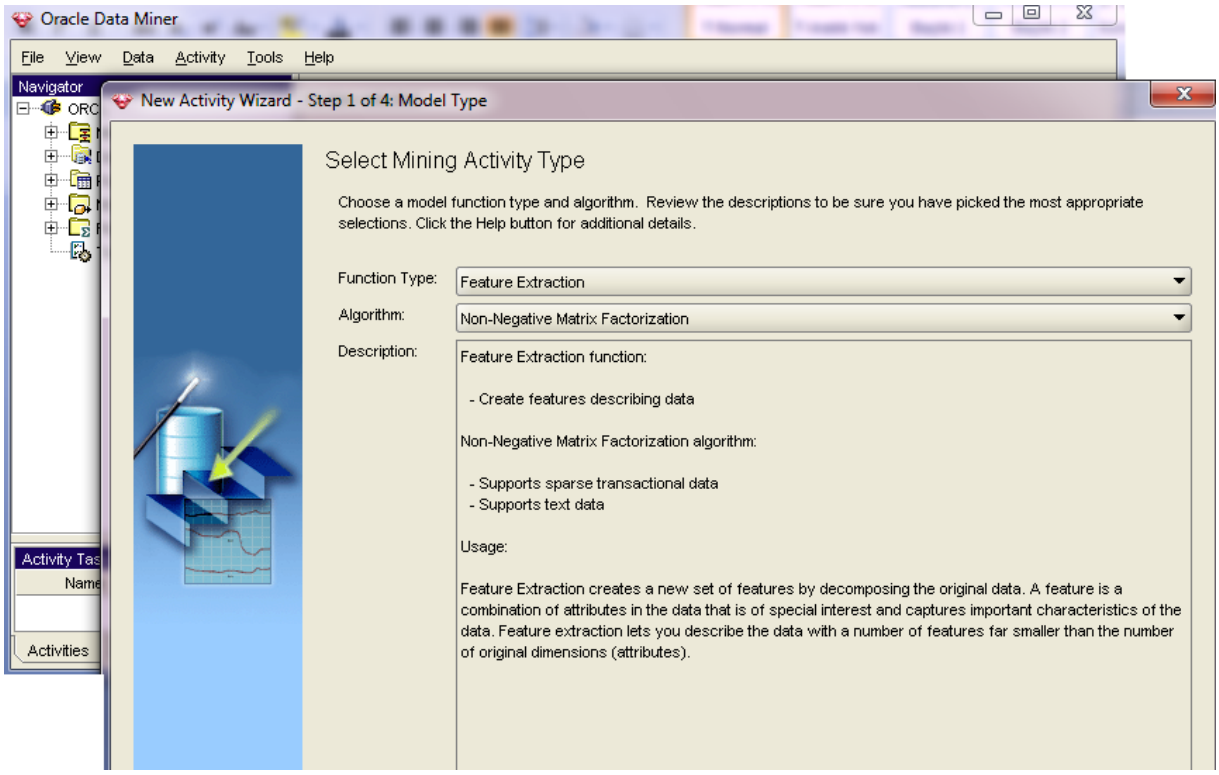


Figure 12. Feature Extraction Step for ODM

After this method, petal length and petal width are the useful features for this dataset that is shown in Figure 13 and decision tree was created according to this extraction.

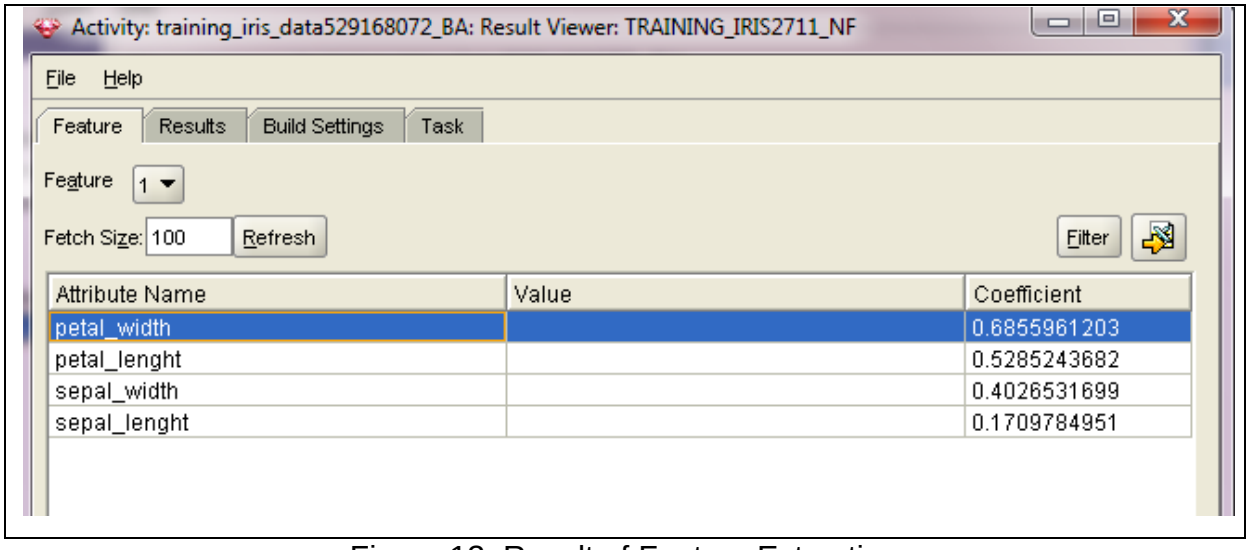


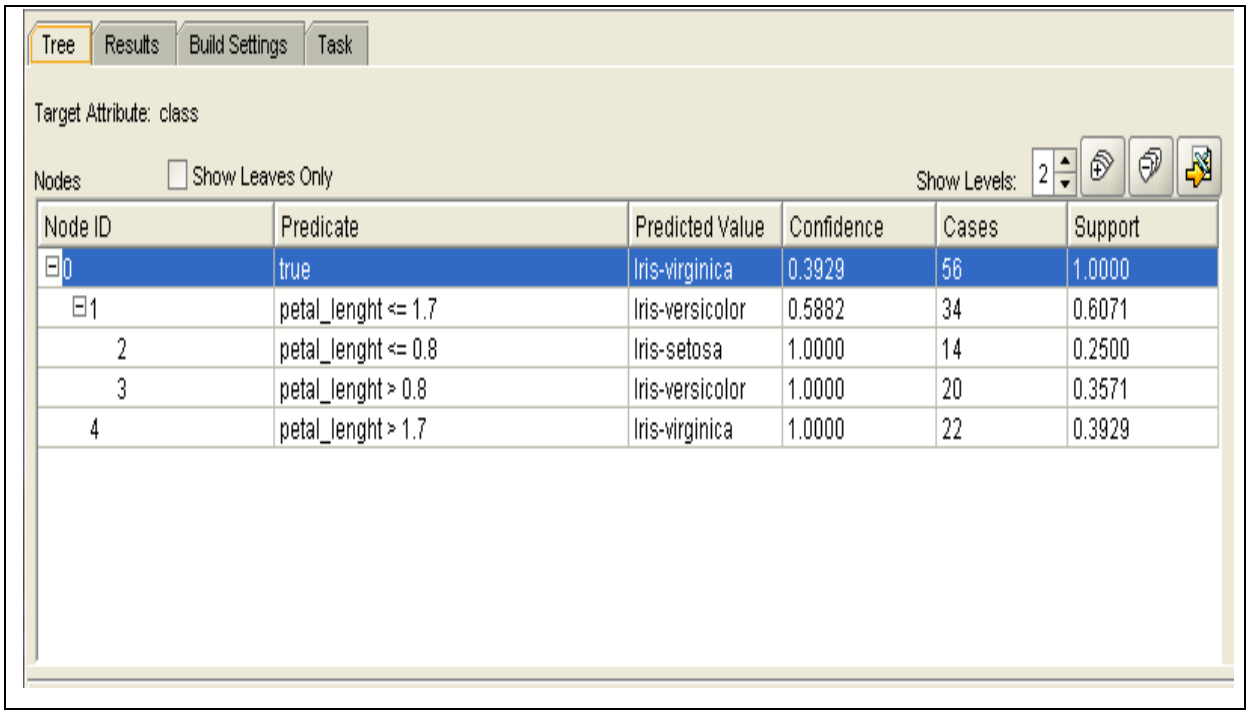
Figure 13. Result of Feature Extraction

As noted in Chapter 3.1 Discovery-oriented methods are divided into two as predictive and descriptive. In this study, predictive technique is applied to find flowers' type from decision tree which will be applied with training dataset.

ODM provides predictive techniques of Data mining such as classification and regression. In this study, decision tree algorithm is used for searching patterns of Data Mining classification task. As noted earlier in Chapter 3, classification assigns items in a collection to target classes. Its goal is to accurately predict the target class for each instance in the dataset. ODM provides two metrics such as gini and entropy for decision tree algorithm. In this study gini is used.

Decision tree classification model is built and tested with ODM in activity-build tab. The randomly selected training dataset is selected as input to mining activity. Class attribute is chosen for target value.

The decision tree model for the training data of Iris dataset defined by ODM is given in Figure 14. The decision tree defined by ODM is shown graphically in Figure 15.



Node ID	Predicate	Predicted Value	Confidence	Cases	Support
0	true	Iris-virginica	0.3929	56	1.0000
1	petal_lenght <= 1.7	Iris-versicolor	0.5882	34	0.6071
2	petal_lenght <= 0.8	Iris-setosa	1.0000	14	0.2500
3	petal_lenght > 0.8	Iris-versicolor	1.0000	20	0.3571
4	petal_lenght > 1.7	Iris-virginica	1.0000	22	0.3929

Figure 14. The Decision Tree Model of Iris Dataset

Some of the terms in Figure 14 such as Node ID, Predicate, Predicted Value, Confidence, Cases and Support are defined below:

- Predicate shows the conditional statements in the data.

- Predicted value is the names of types of iris plant which are the member of target values.

- Confidence and support are already defined in Chapter 3. In this figure, for instance, confidence of the node 2 is 100% that means at this case is satisfied with this decision tree.

- Cases are the number of instances on nodes.

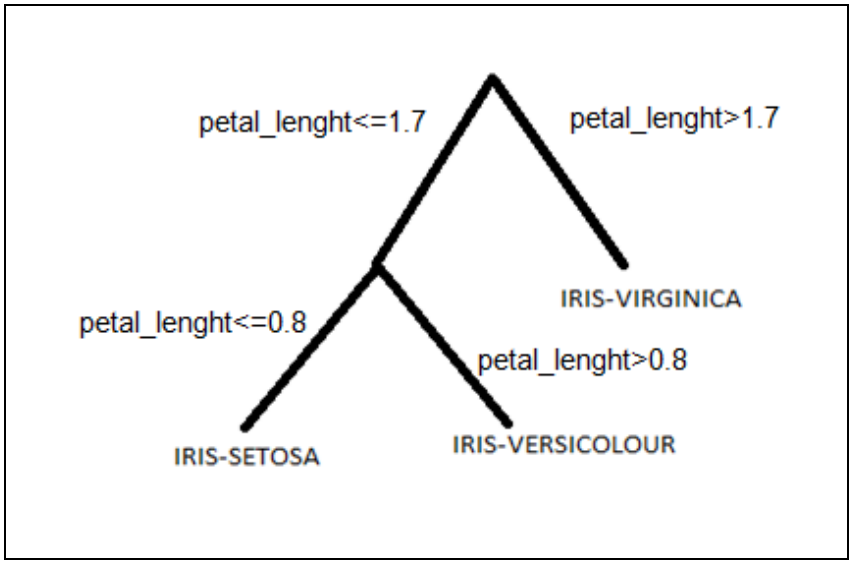


Figure 15. Visualization of the Decision Tree

In the decision tree classification algorithm of ODM, the default settings are chosen such calculation metric gini, maximum depth 7, minimum records in a node 10, minimum percent of records in a node 0.05, minimum records for a split 20, and minimum percent of records for a split 0.1. The user may change these default values if necessary.

Figure 16-18 show the generated rules by decision tree which according to target attribute of the data set.

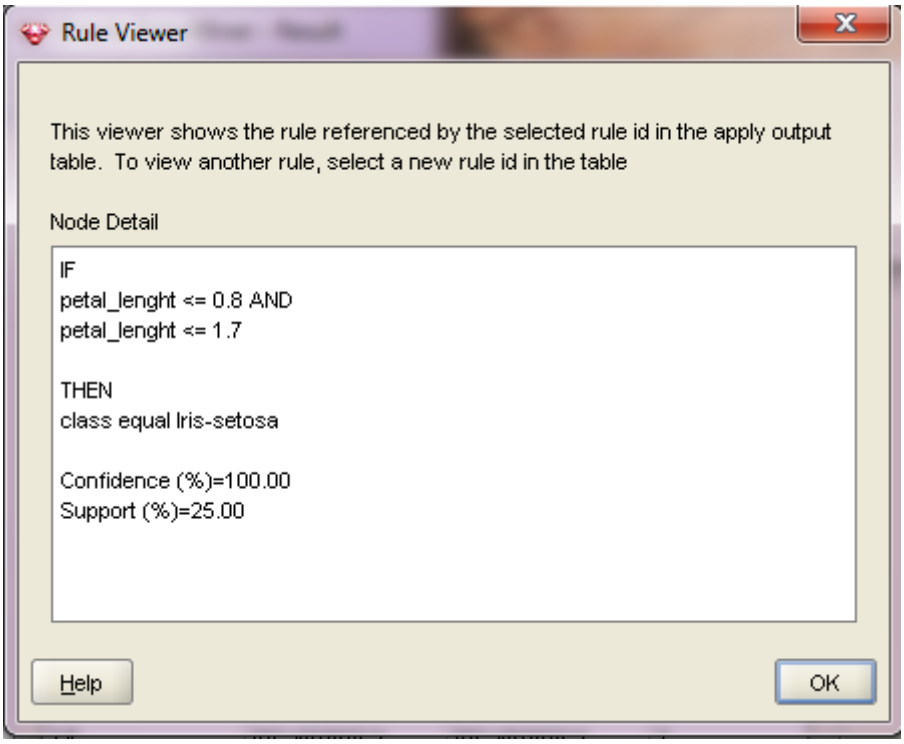


Figure 16. Rule View for Iris-Setosa

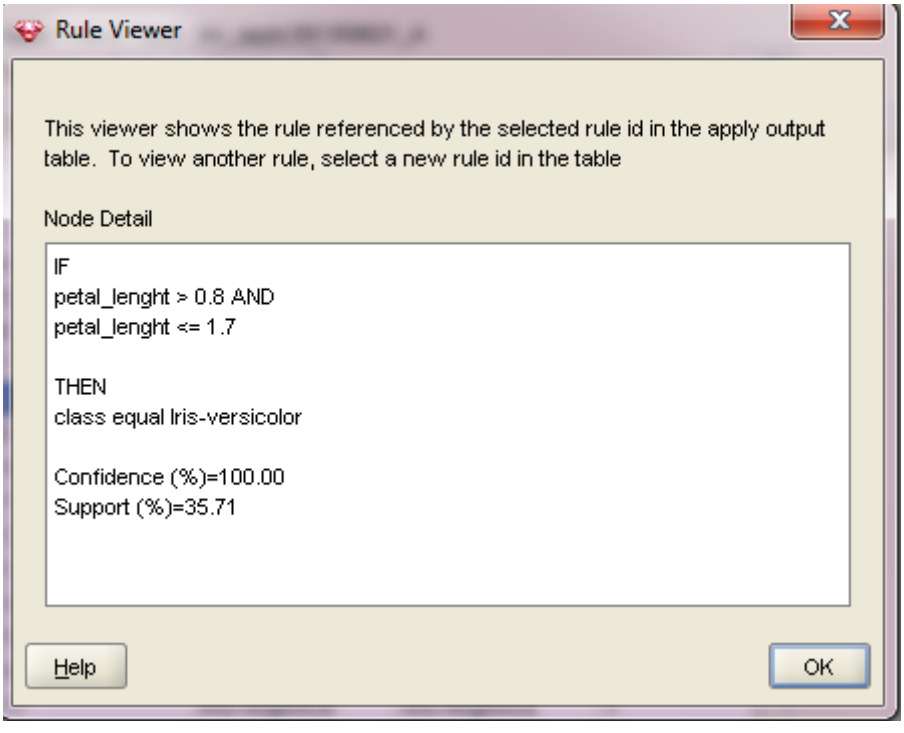


Figure 17. Rule View for Iris-Versicolor

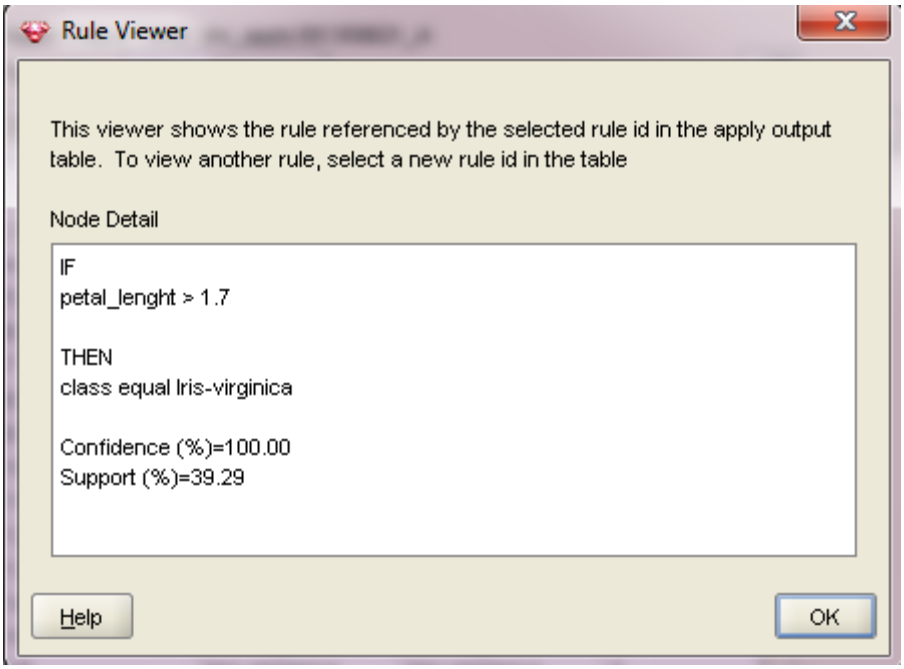


Figure 18. Rule View for Iris-Virginica

Generated decision tree is used for applying the test data. Decision tree predicts a target value of dataset by asking sequence questions that are generated by rules. Performance of classification model and predictive confidence is used to decide model is good for this dataset or not. Predictive confidence of decision tree model is shown in Figure 19. In this study, predictive confidence is 85.45%. Predictive confidence of ODM calculation is shown below:

$$\text{Predictive Confidence} = 1 - ((\text{Error of Predict}) / (\text{Error of naive model})) \quad (6.1)$$

$$\text{Error of Predict} = 1 - \frac{A1 + A2 + A3}{N} \quad (6.2)$$

A1 is accuracy for target class 1, A2 is accuracy for target class 2, A3 is accuracy for target class 3 and N is the number of target classes.

$$\text{Error of naive model} = (N - 1)/N \quad (6.3)$$

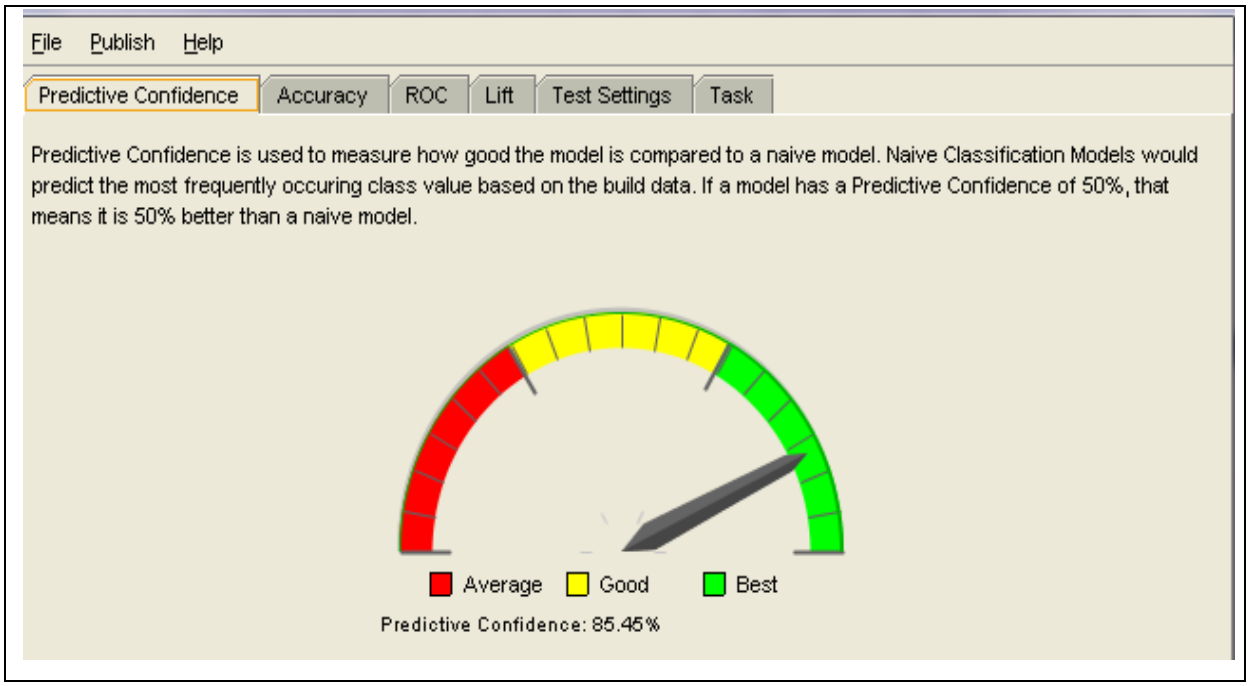


Figure 19. Predictive Confidence of Decision Tree Model

Accuracy presents the percentage of correct predictions made by the decision tree model when compared with the actual values in the test data shown in Figure 20. In this study, iris- versicolour target is correctly predicted 90.91%. Iris-setosa is correctly predicted 100%. Iris- virginica is correctly predicted 80%.

Confusion matrix displays the number of correct and incorrect predictions in the test data. In this matrix columns are predicted values, rows present actual values. When looking the iris-versicolour case, model predict six of them true but others are not. It predicts them like iris-virginica. In this case the model misclassified.

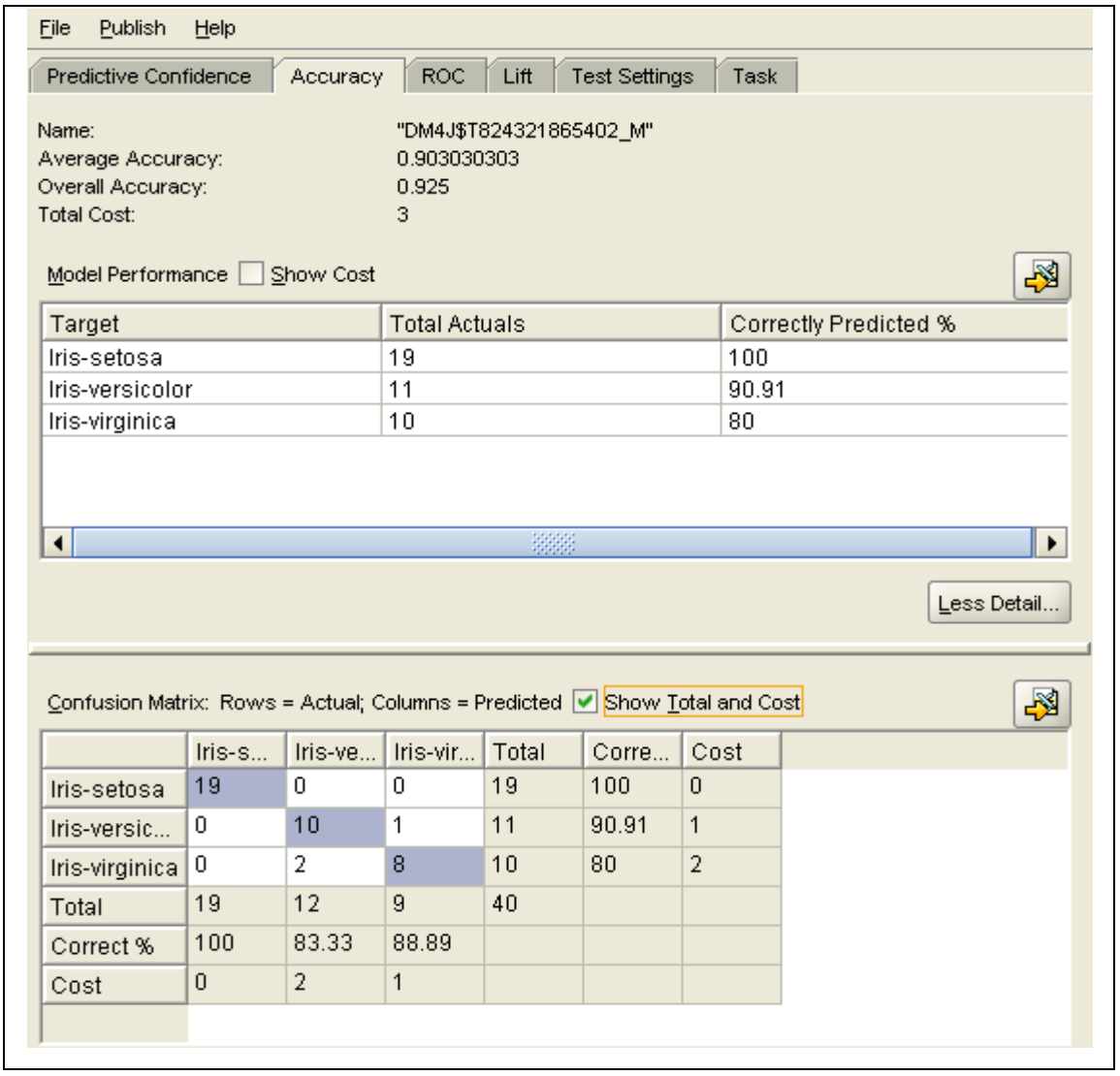
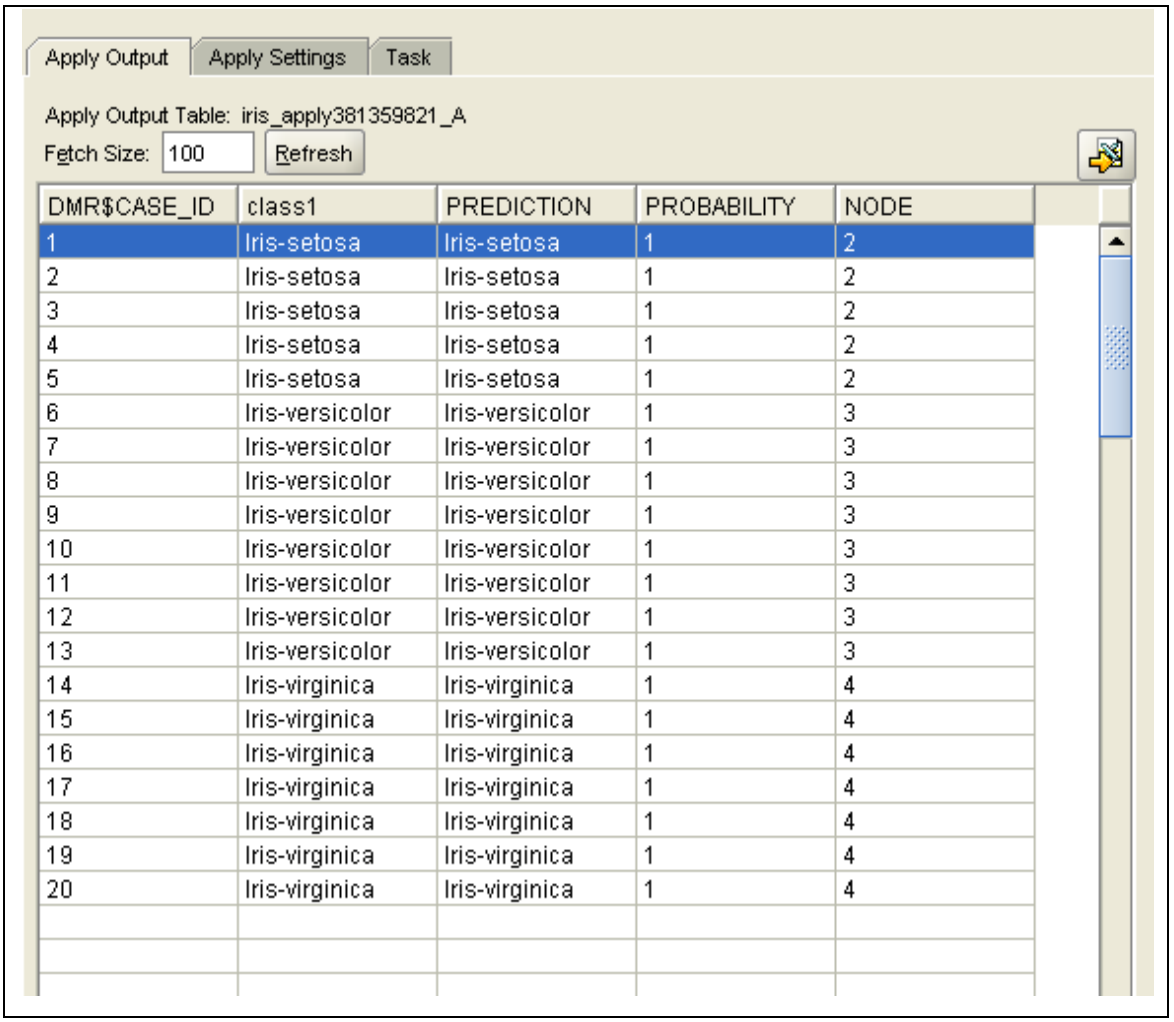


Figure 20. Accuracy Table and Confusion Matrix of Decision Tree Model

In Figure 21 the applied data and predictions of them are shown.



DMR$CASE_ID	class1	PREDICTION	PROBABILITY	NODE
1	Iris-setosa	Iris-setosa	1	2
2	Iris-setosa	Iris-setosa	1	2
3	Iris-setosa	Iris-setosa	1	2
4	Iris-setosa	Iris-setosa	1	2
5	Iris-setosa	Iris-setosa	1	2
6	Iris-versicolor	Iris-versicolor	1	3
7	Iris-versicolor	Iris-versicolor	1	3
8	Iris-versicolor	Iris-versicolor	1	3
9	Iris-versicolor	Iris-versicolor	1	3
10	Iris-versicolor	Iris-versicolor	1	3
11	Iris-versicolor	Iris-versicolor	1	3
12	Iris-versicolor	Iris-versicolor	1	3
13	Iris-versicolor	Iris-versicolor	1	3
14	Iris-virginica	Iris-virginica	1	4
15	Iris-virginica	Iris-virginica	1	4
16	Iris-virginica	Iris-virginica	1	4
17	Iris-virginica	Iris-virginica	1	4
18	Iris-virginica	Iris-virginica	1	4
19	Iris-virginica	Iris-virginica	1	4
20	Iris-virginica	Iris-virginica	1	4

Figure 21. Predictions of Decision Tree Model

Finally, generated rules and decision tree show that one class is linearly separable from the others however the latter are not linearly separable from each other.

If the predictive confidence of dataset is enough for user of system, there can be a desktop application with using generated rules. This application used for writing the attributes of iris plant and finding which iris plant it is.

6.3 Application with WEKA

WEKA explorer is imported data using open file tab and choosing data set files. Figure 22 shows the importing data to WEKA and Figure 23 shows the general information about data.

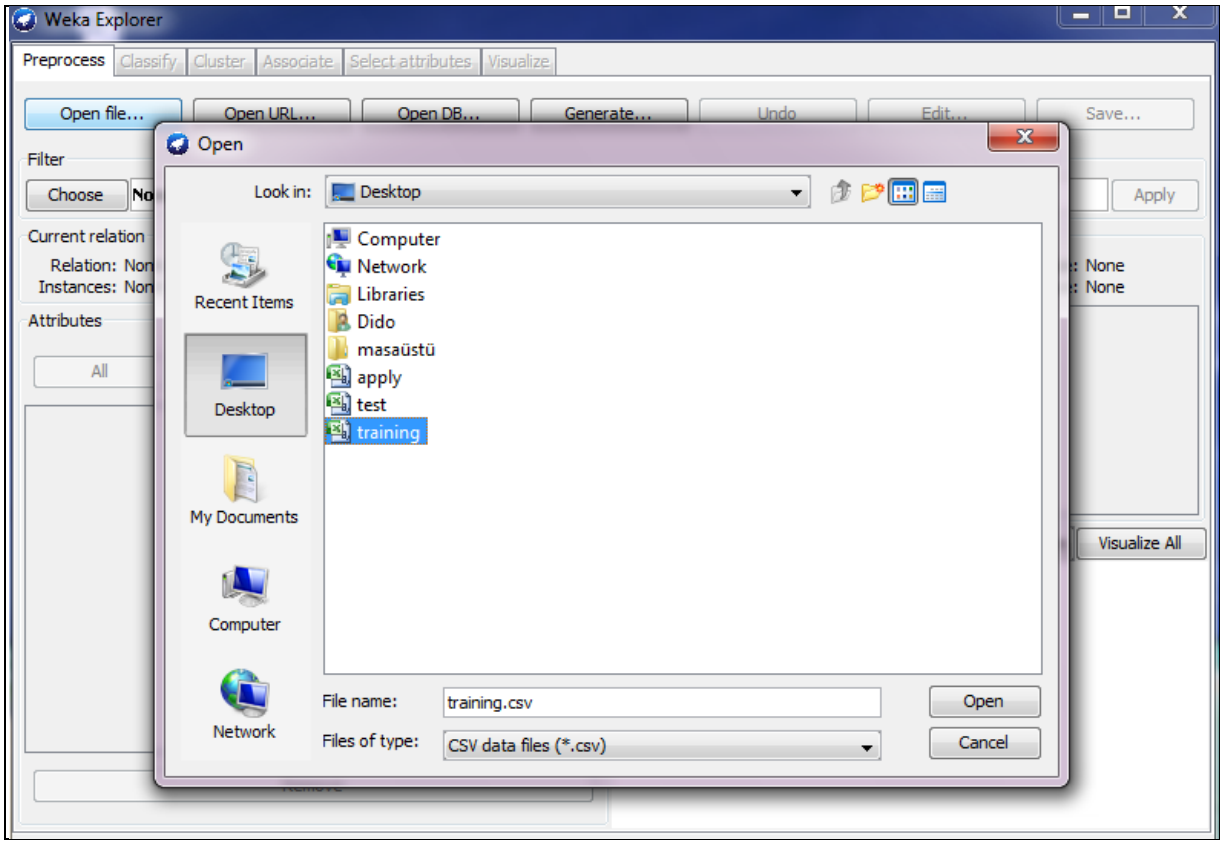


Figure22. Import Data to WEKA

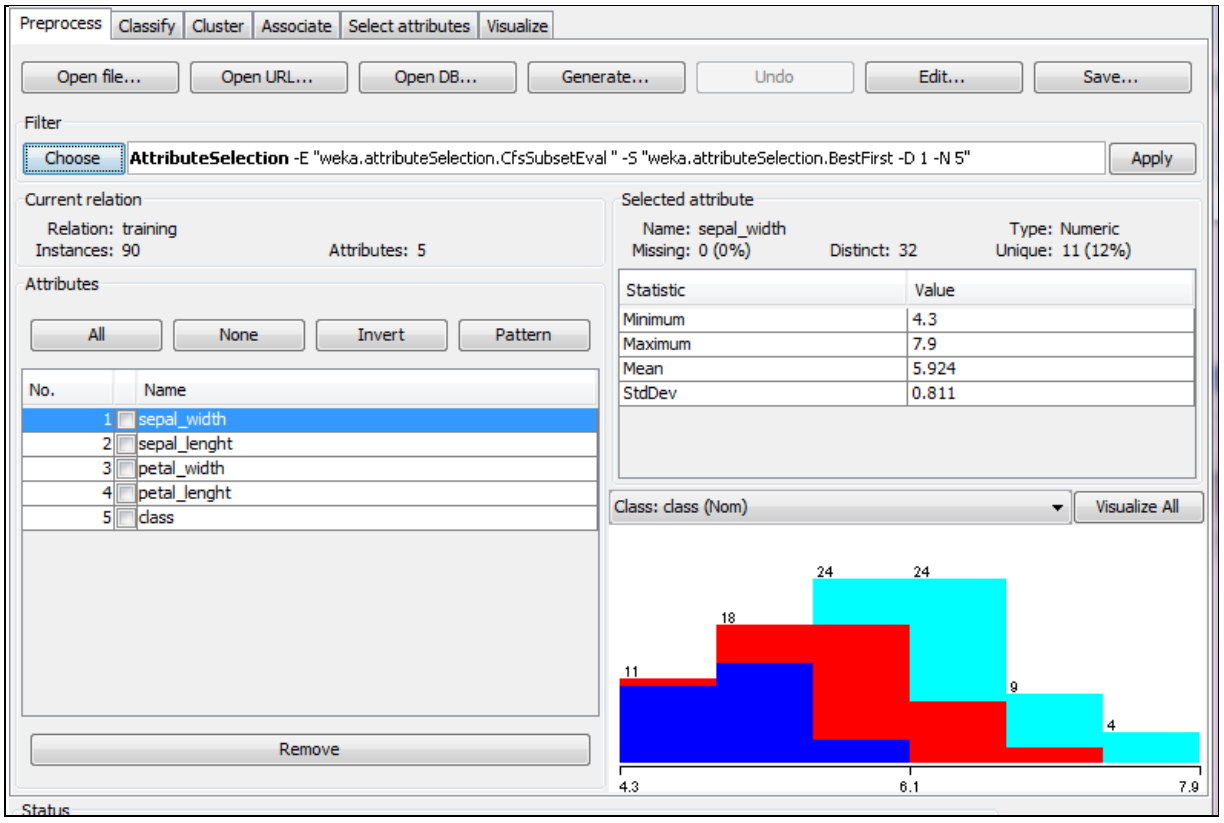


Figure 23. Visualization of Data in WEKA

To make attribute selection for WEKA, selecting filter tab and choosing AttributeSelection algorithm for iris data gives the selected attributes are petal width and petal length is shown in Figure 24.

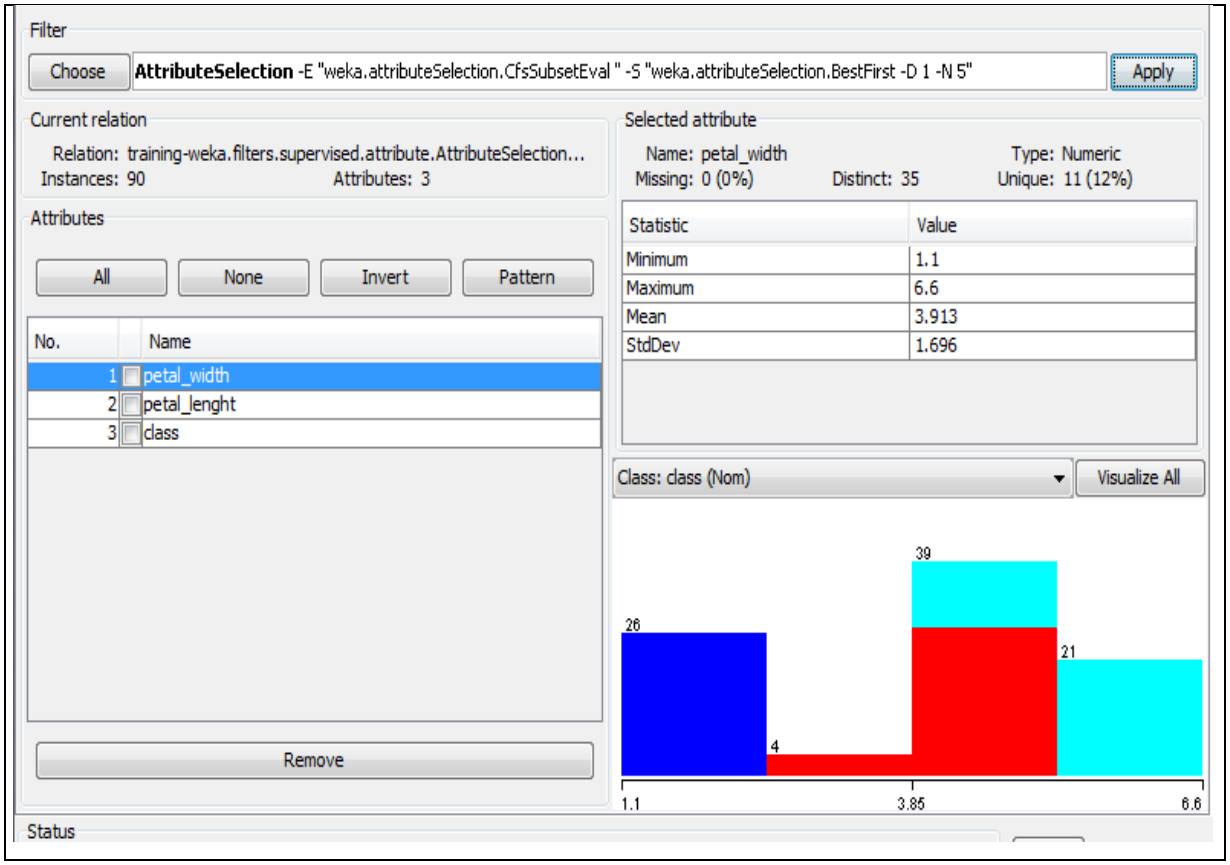


Figure 24. Attribute Selection for WEKA

After attribute selection for iris data in Classify section choosing the trees and NBTree(Naive Bayes Tree) algorithm for classification of data is shown in Figure 25.

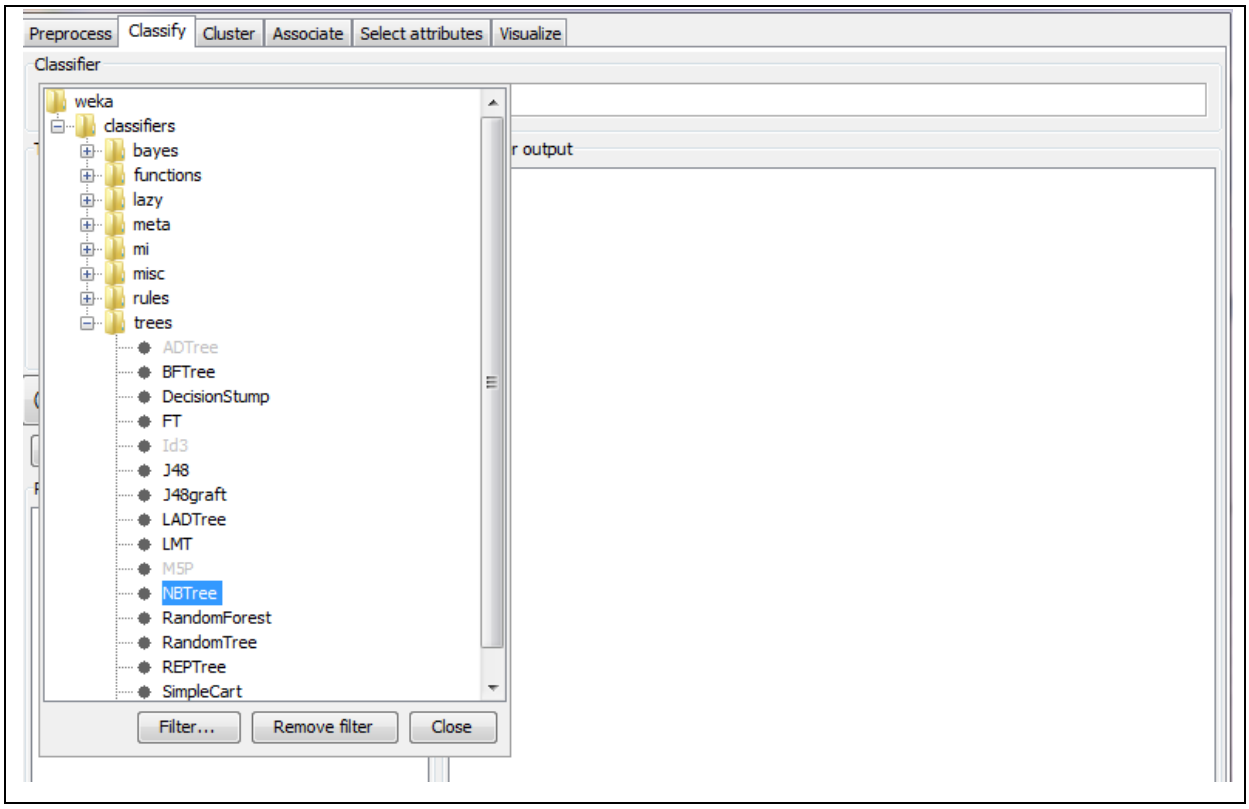


Figure 25. Visualization of algorithms for Classification of WEKA

After performed NBTree algorithm, only one instance is classified in correctly is shown in Figure 26. While performing the NBTree, test option is tenfold cross-validation which is mainly used in setting where goal is prediction. Accuracy of the tree is 98.88%.

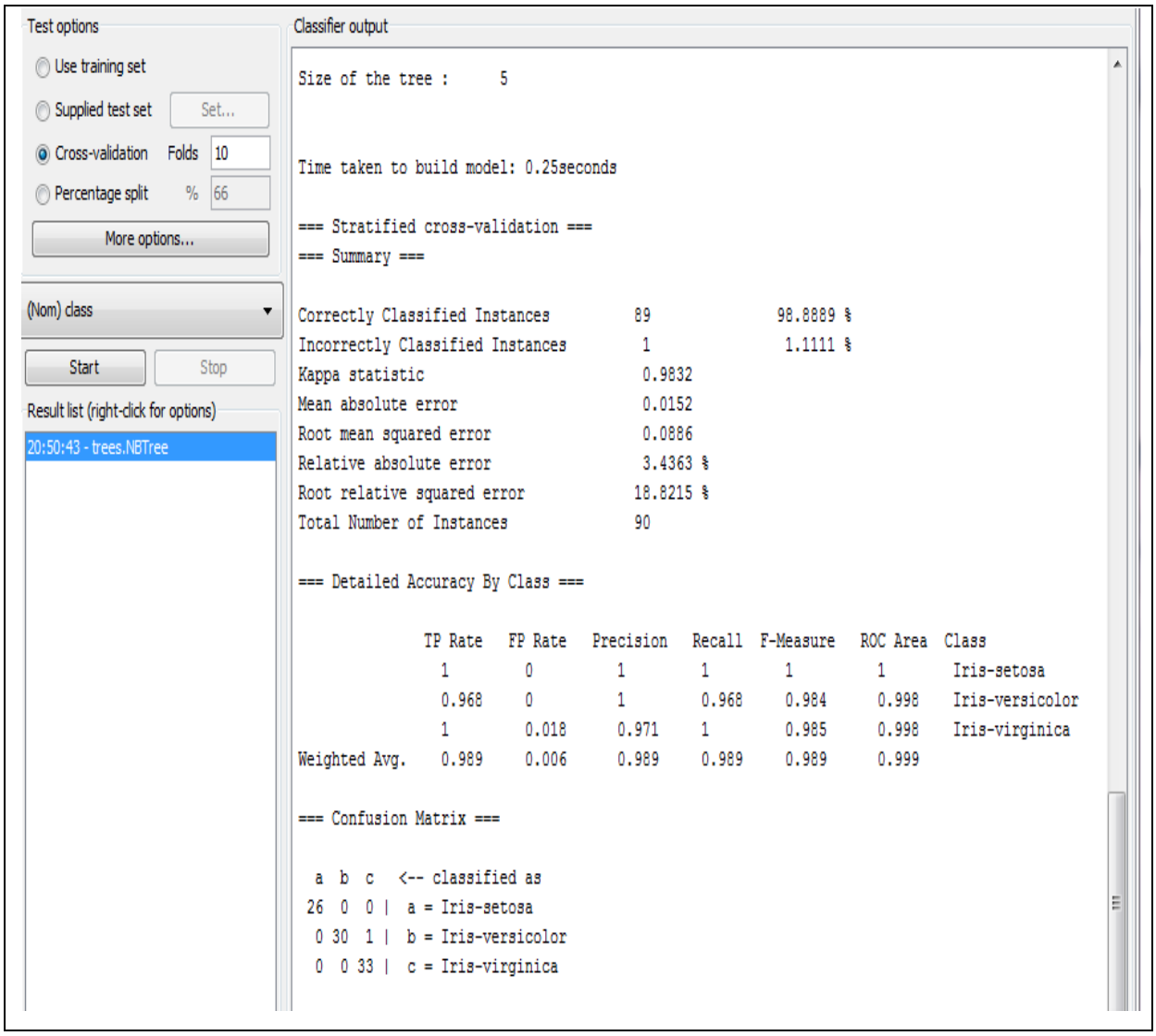


Figure 26. NBTree and Confusion Matrix in WEKA

Generated NBTree is used for applying the test data. Performance of classification model and correctly classified instances are used to decide model is good for this dataset or not. When model is applying to test data, test option is supplied test set. In this section, test data is imported. Generated NBTree for test data is shown in Figure 27. Accuracy of test data is 92.5%. In this study, iris-versicolor target is correctly predicted 90.91%. Iris-setosa is correctly predicted 100%. Iris-virginica is correctly predicted 80%.

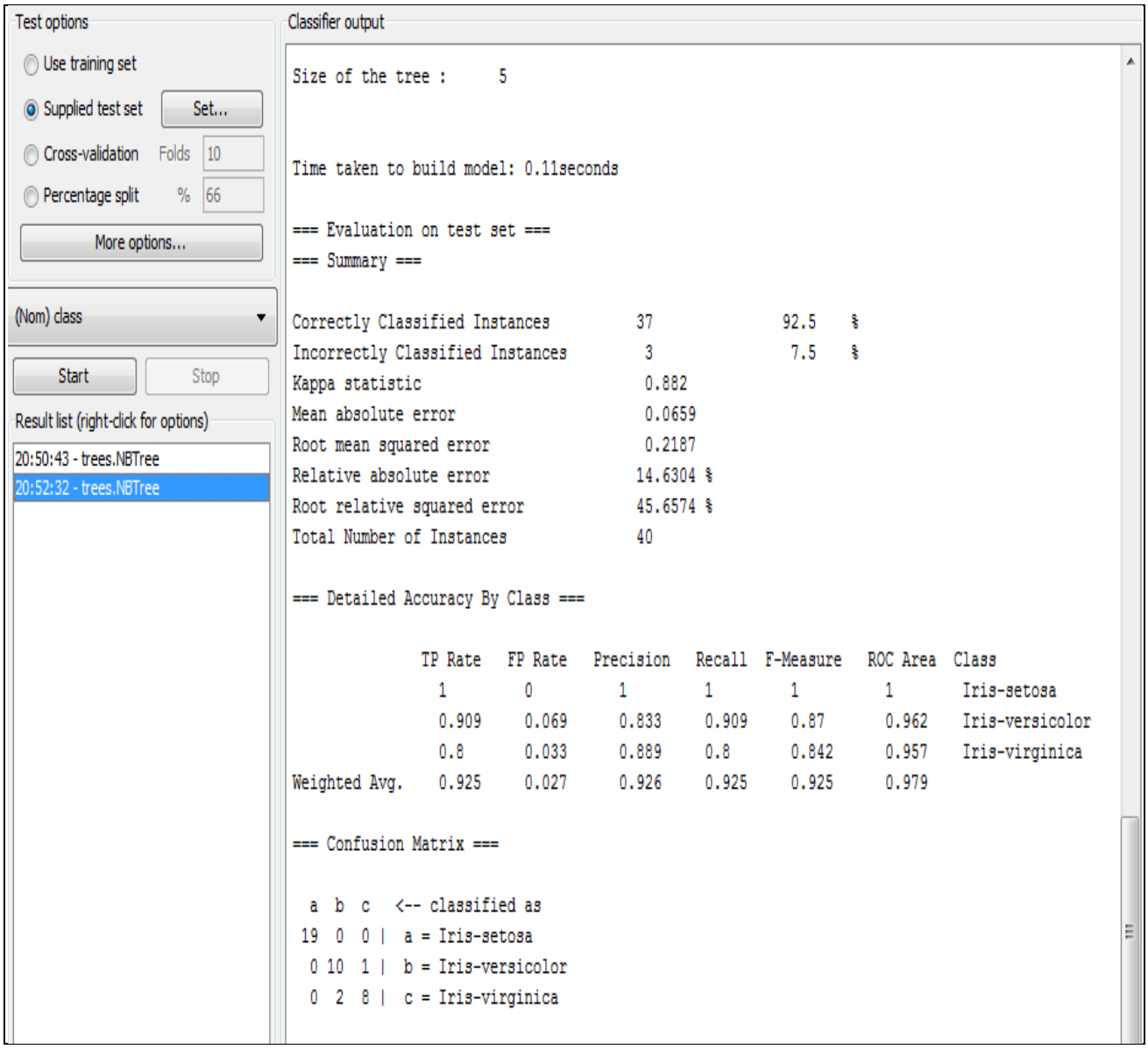


Figure 27. Evaluation of NBTree on Test Data

In Figure 28 the applied data and predictions of them are shown.

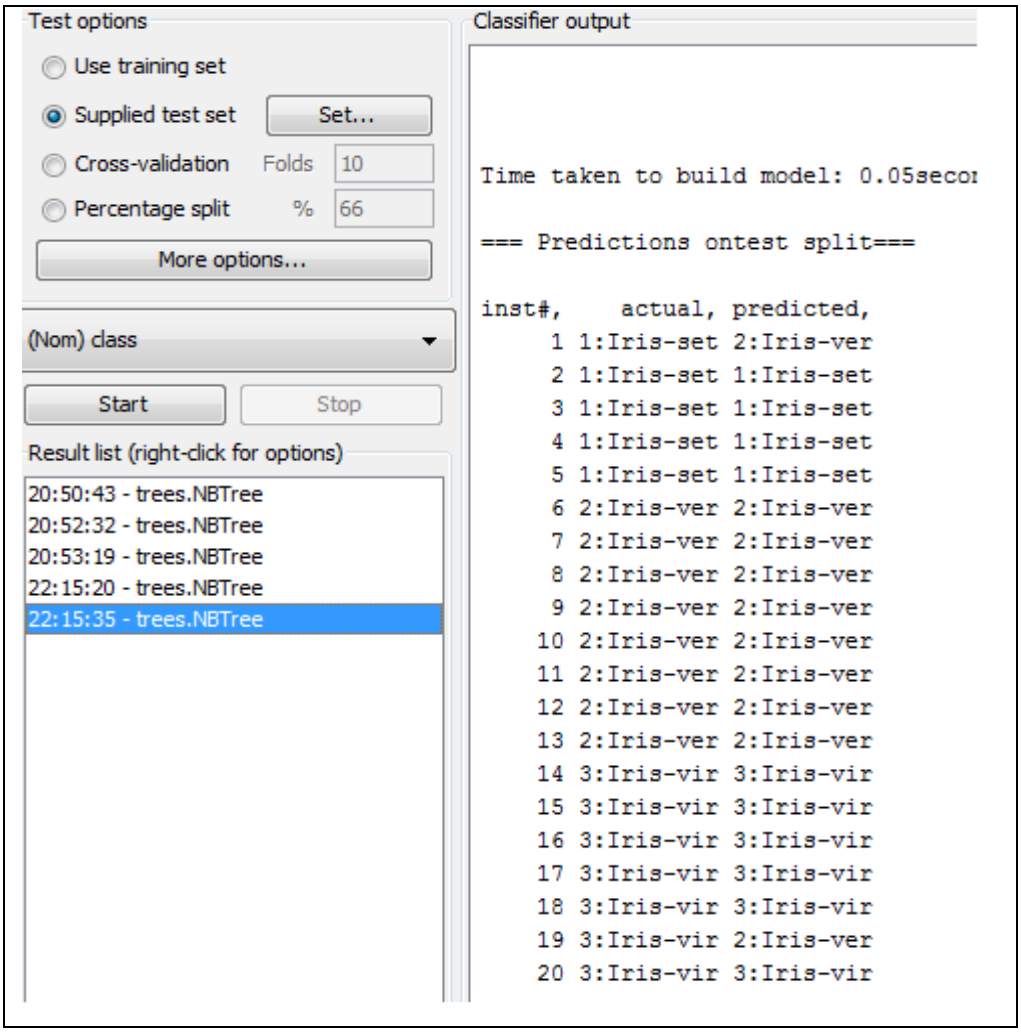


Figure 28. Predictions of NBTree Model

In WEKA, there are many tree algorithms such as BFTree, J48, Random Tree and RepTree. Iris dataset is also applied to these algorithms and gained some results. These results' screen shots are given in Appendix A.

When BFTree algorithm is applied to training data set, correctly prediction is 96.66%. In test data 87.5% and apply data is correctly classified 95%.

When J48 algorithm is applied to training data set, correctly prediction is 97.77%. In test data 87.5% and apply data is correctly classified 95%.

When Random Tree algorithm is applied to training data set, correctly prediction is 98.88%. In test data 87.5% and apply data is correctly classified 95%.

When RepTree algorithm is applied to training data set, correctly prediction is 97.77%. In test data 87.5% and apply data is correctly classified 95%.

To sum up, using different algorithms for classification in WEKA shows that also algorithms show changes among them. NBTree algorithm is provided the best result among for this data.

6.4 Application with R

R has a console and commands are written here. When data is imported, reading data file is shown in Figure 29. “training” is the name of dataset.

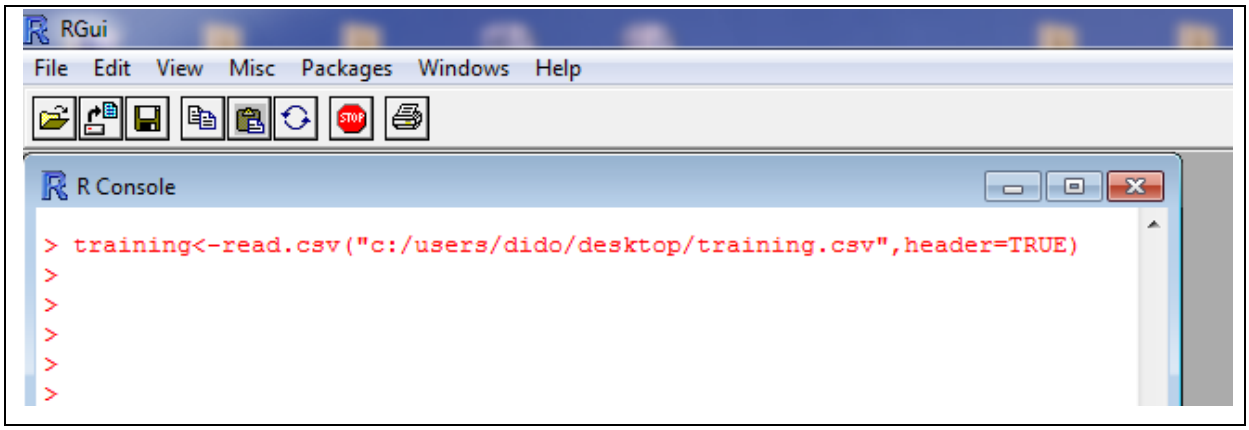


Figure 29. Import Data to R

R has some data mining packages. Party is the one of them and it includes classification tree (ctree) algorithm. Package is used like command which is shown in Figure 30.

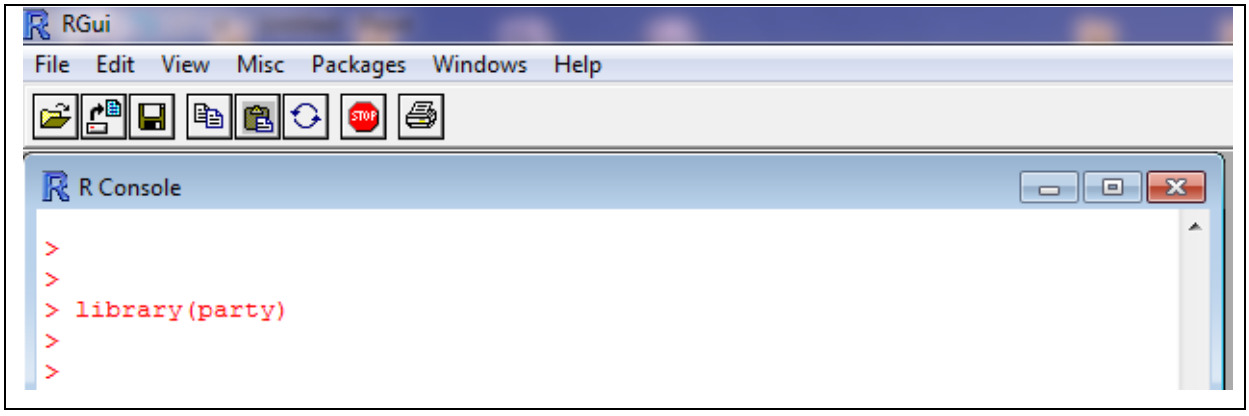


Figure 30. Using Data Mining Packages for R

After initialize package, creating ctree is shown in Figure 31. Plot command is shown the graphics of the generated tree.

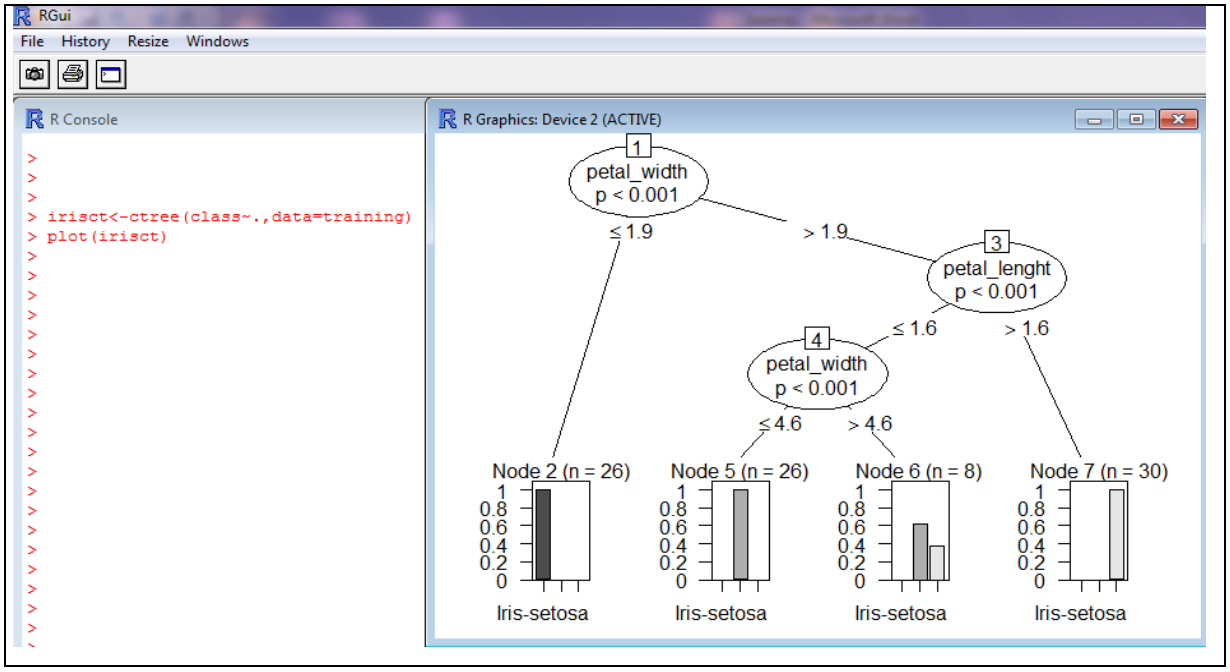


Figure 31. Ctree for R

Confusion matrix for ctree algorithm in R is created with table command is shown in Figure 32. In this confusion matrix is said that accuracy of this tree is 96.67%.

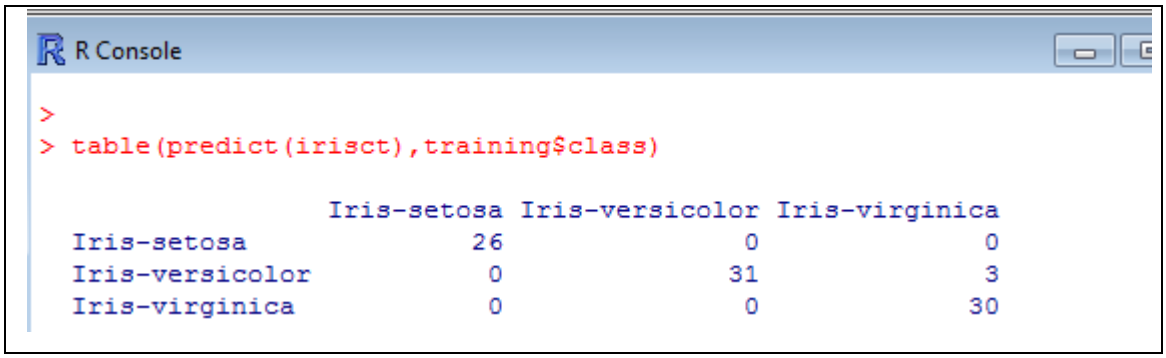


Figure 32. Confusion Matrix of ctree

Figure 33 shows that the classification tree of training dataset. Weights means how many data related to that leaves. In this tree again petal_width and petal_length are the attributes of tree.

```
> print(irisct)

         Conditional inference tree with 4 terminal nodes

Response:  class
Inputs:  sepal_width, sepal_lenght, petal_width, petal_lenght
Number of observations:  90

1) petal_width <= 1.9; criterion = 1, statistic = 84.447
  2)*  weights = 26
1) petal_width > 1.9
  3) petal_lenght <= 1.6; criterion = 1, statistic = 43.918
    4) petal_width <= 4.6; criterion = 0.999, statistic = 14.425
      5)*  weights = 26
    4) petal_width > 4.6
      6)*  weights = 8
  3) petal_lenght > 1.6
    7)*  weights = 30
```

Figure 33. Ctree for Training Dataset

While test data is applied to classification tree, predicted table is generated as shown in Figure 34. In this confusion matrix is said that accuracy of this tree is 92.5%.

```
> table(predicted,test$class)

predicted         Iris-setosa Iris-versicolor Iris-virginica
  Iris-setosa              19               0              0
  Iris-versicolor           0               9              1
  Iris-virginica            0               2              9
```

Figure 34. Confusion Matrix of Test Dataset

6.5 Application with RapidMiner

RapidMiner is imported data using repositories tab and choosing import which file type you have. Figure 33 shows the importing data to RapidMiner. While importing

data, how the data should be parsed and how columns are separated is choosing. In iris data columns are separated with comma. Also should define the one attribute for class label is shown in Figure 34.

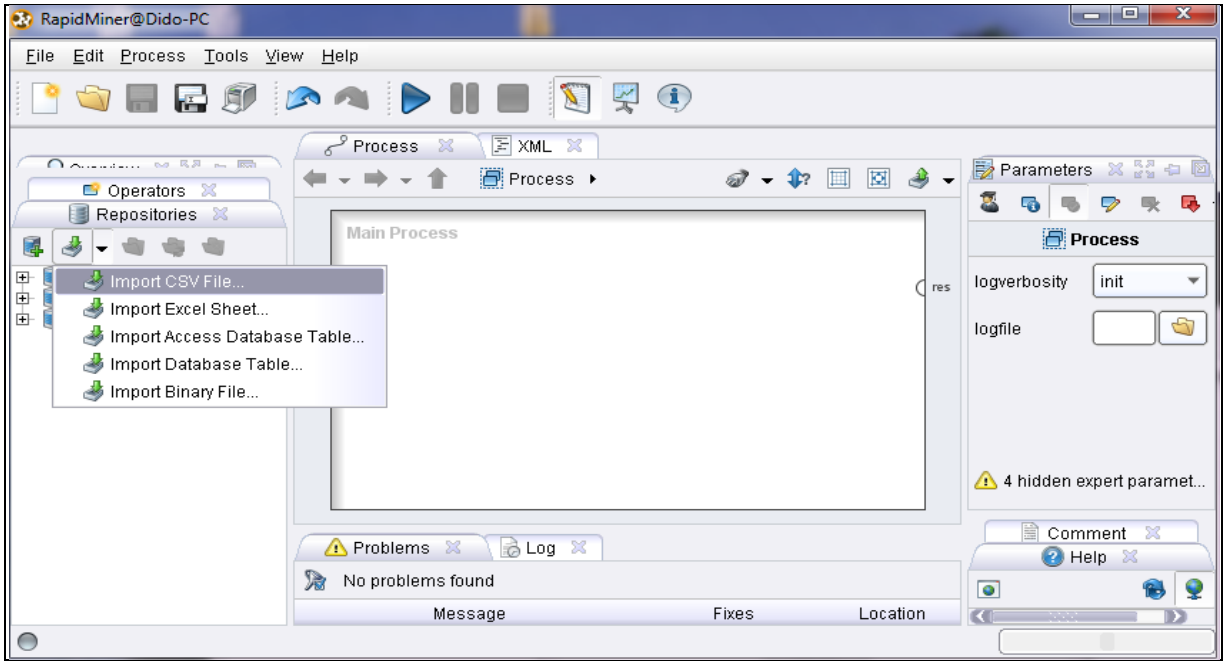


Figure 35. Import Data to RapidMiner

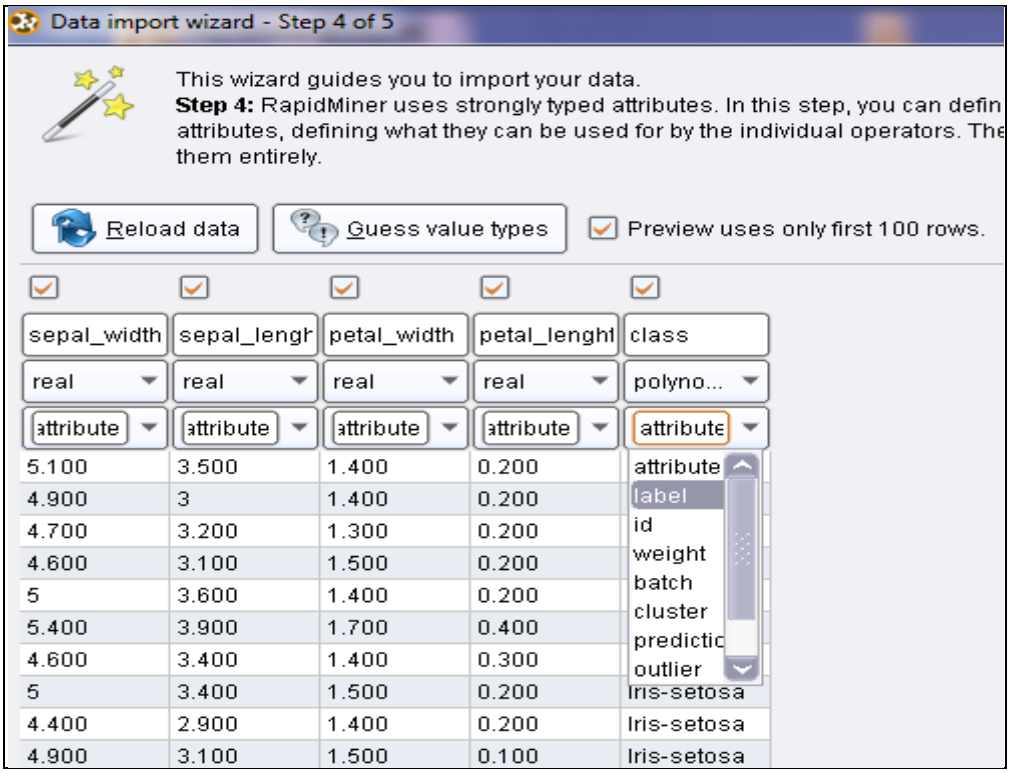


Figure 36. Select Which Attributes is Class

RapidMiner provides some tree induction algorithms and decision tree is one of them. Iris data is applied to decision tree algorithm in RapidMiner is shown in Figure 35. In this decision tree classification algorithm settings are chosen such calculation metric gain ratio, minimal size for split 4, minimal leaf size 2, minimal gain 0.1, maximal depth 20 and confidence is 0.25. Generated decision tree model is shown in Figure 36.

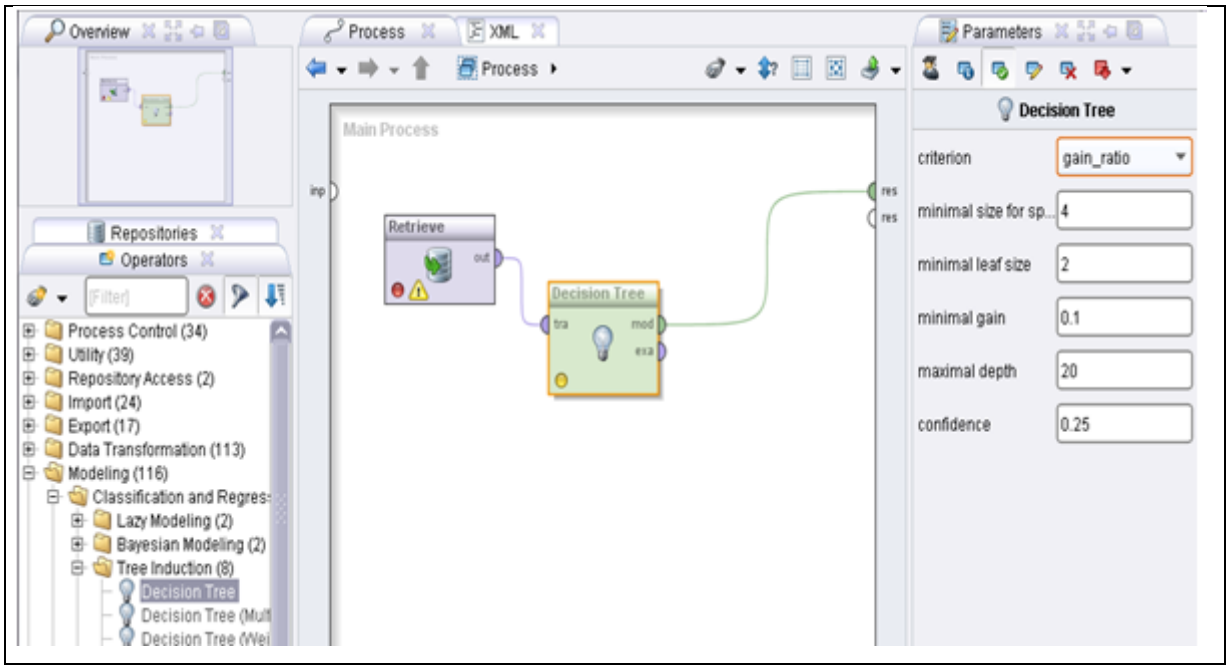


Figure 37. Decision Tree for RapidMiner

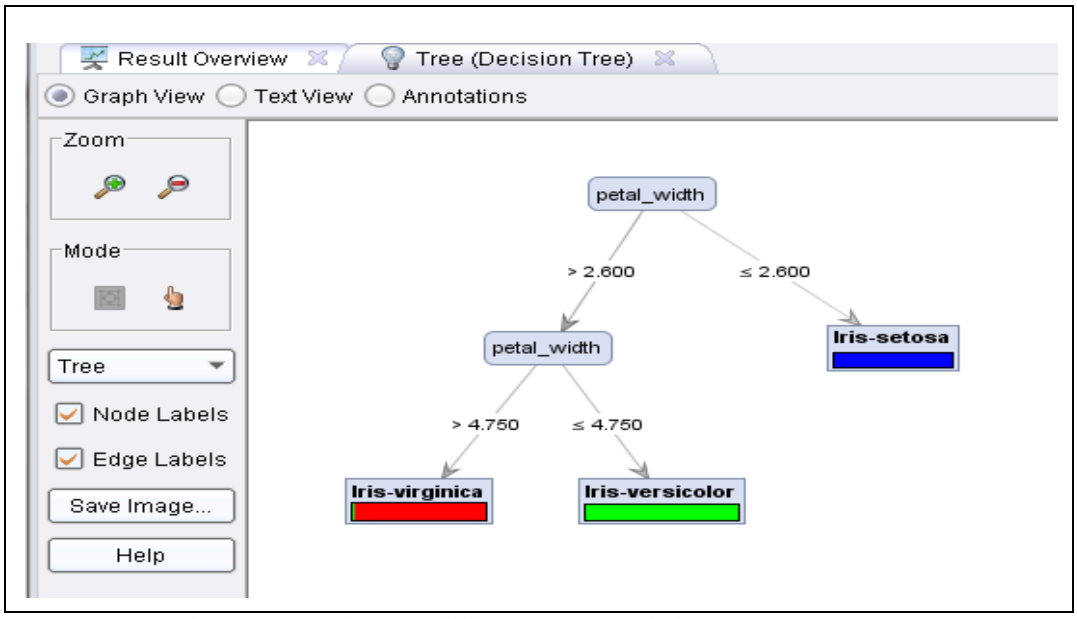


Figure 38. The Decision Tree Model of IRIS Dataset

To show performance of the data in RapidMiner, apply model and the performance are linked to generated decision tree. In Figure 37 shows the relation with each other.

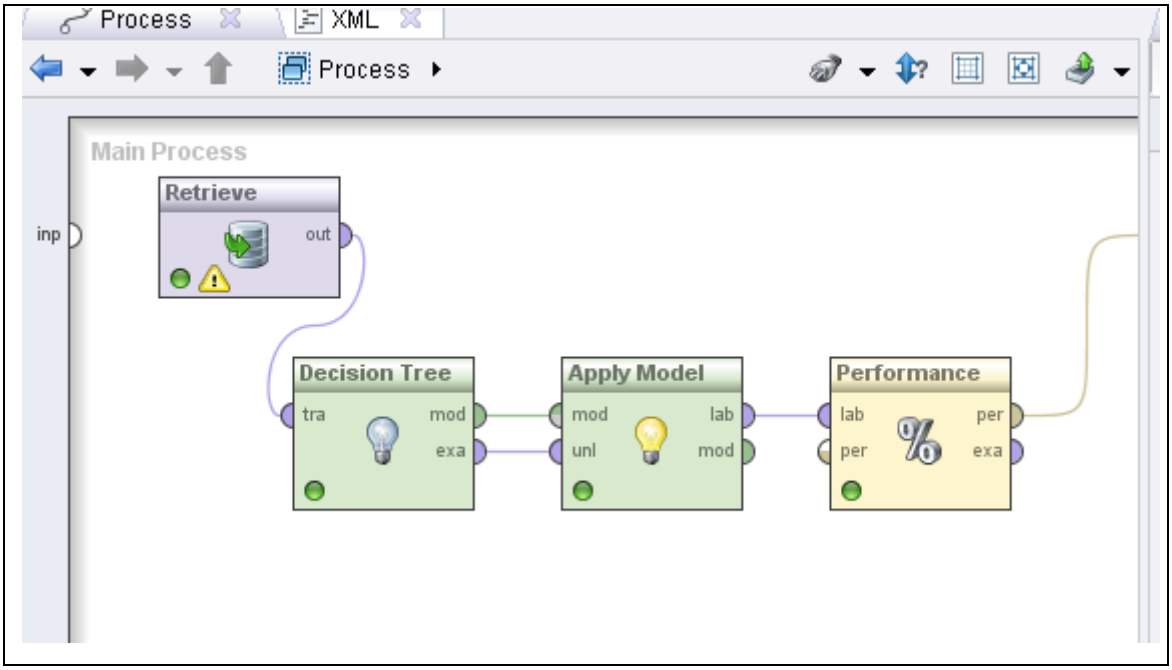
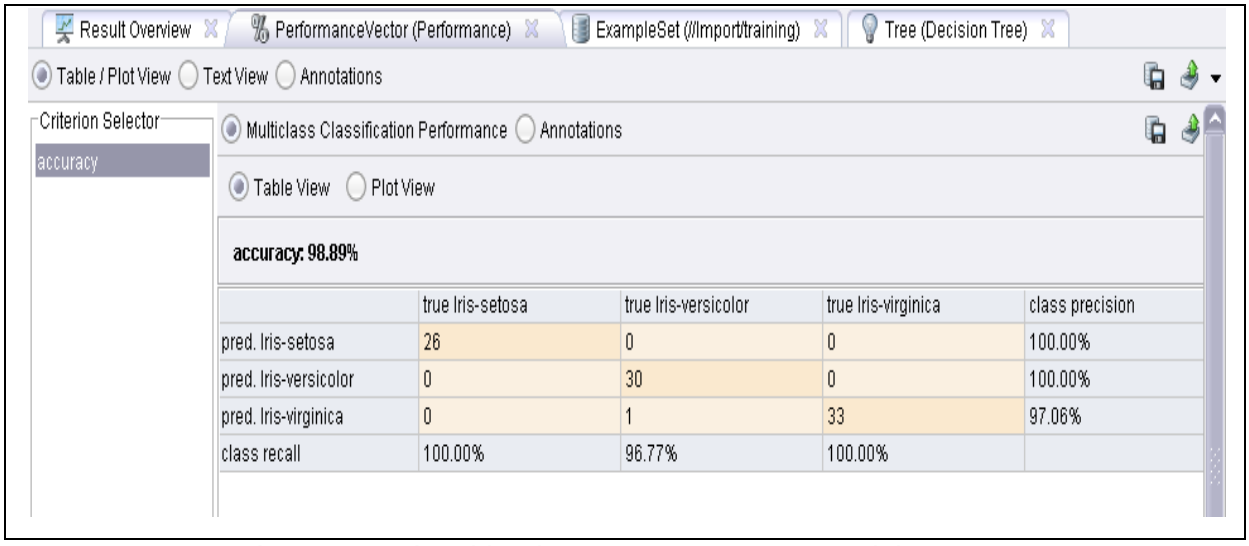


Figure 39. Visualization of Main Process of Decision Tree

In matrix rows are predicted values, columns present actual values in Figure 38. When looking the iris-virginica case, model predict 33 of them true but one is not. It predicts them like iris-versicolor. Accuracy of model is 98.89%.



	true Iris-setosa	true Iris-versicolor	true Iris-virginica	class precision
pred. Iris-setosa	26	0	0	100.00%
pred. Iris-versicolor	0	30	0	100.00%
pred. Iris-virginica	0	1	33	97.06%
class recall	100.00%	96.77%	100.00%	

Figure 40. Classification Performance of Decision Tree

Generated decision tree is used for applying the test data. Performance of classification model and correctly classified instances are used to decide model is good for this dataset or not. When model is applying to test data, test set is linked to apply model. In this section, test data is imported. Generated decision tree for test data is shown in Figure 39. Accuracy of test data is 87.5% is shown in Figure 40. In this study, iris- versicolour target is correctly predicted 87.5%. Iris-setosa is correctly predicted 100%. Iris- virginica is correctly predicted 69.23%.

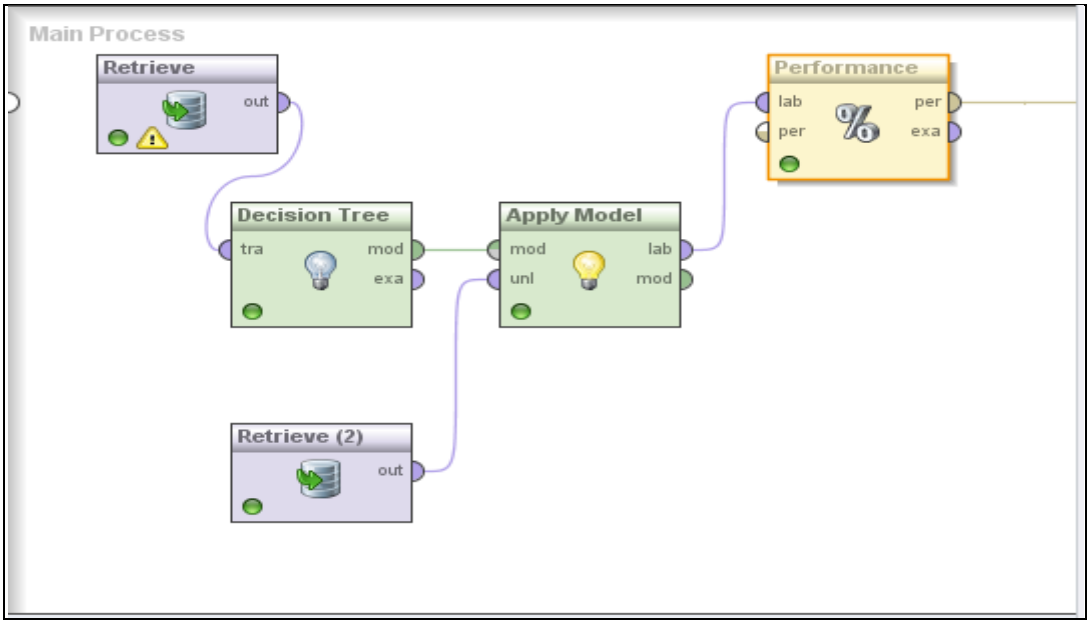


Figure 41. Applying Decision Tree to Test Data

Multiclass Classification Performance Annotations

Table View Plot View

accuracy: 87.50%

	true Iris-setosa	true Iris-versicolor	true Iris-virginica	class precision
pred. Iris-setosa	19	0	0	100.00%
pred. Iris-versicolor	0	7	1	87.50%
pred. Iris-virginica	0	4	9	69.23%
class recall	100.00%	63.64%	90.00%	

Figure 42. Classification Performance for Test Data

6.6 Comparison of Data Mining Techniques

Data mining techniques may be compared with sensitivity, specificity and classification accuracy.

Sensitivity measures the proportion of actual positives which are correctly identified. Specificity measures the proportion of negatives which are correctly identified. To calculate these measures, some terms also have. TP means true positive items that correctly classified. TN means true negative items correctly classified. FN means false negative items correctly classified and FP means false positive items correctly classified.

$$Sensitivity = \frac{TP}{TP+FN} \quad (6.1)$$

$$Specificity = \frac{TN}{TN+FP} \quad (6.2)$$

The classification accuracy depends on the number of samples correctly classified.

$$Accuracy = \frac{TP+TN}{total\ \ number\ \ of\ \ items} * 100 \quad (6.3)$$

7. DISCUSSION OF THE RESULTS

In the thesis, Oracle Data Miner, WEKA, R, RapidMiner and ToolDiag tools have been used for comparison. In Oracle Data Miner, decision tree with gini metric is applied. In WEKA, NBTree algorithm is applied. In R, ctree algorithm is applied. In RapidMiner, decision tree with gain ratio metric is applied. All tools give different results for same dataset. In Table 2 shows that their sensitivity, specificity and accuracy of models.

Table 2 – Comparison of Data Mining Software Tools

Software Tools	**Method**	**Training Data Specificity**	**Test Data Specificity**	**Training Data Sensitivity**	**Test Data Sensitivity**	**Accuracy of Test Data**
ORACLE	Decision Tree (gini)	96.4%	97.3%	91.6%	90.3%	92.5%
WEKA	NBTree	99.4%	97.3%	98.92%	90.3%	92.5%
R	Ctree	98.3%	96.6%	96.9%	90.6%	92.5%
RapidMiner	Decision Tree (gain ratio)	99.4%	94.4%	98.9%	85.5%	87.5%

There are a few points to be added for the sake of discussion.

First of all, the decision tree algorithms of software tools are not exactly the same. Thus results may have some effects.

The accuracy drop that appears in RapidMiner might be eliminated or reduced by playing with the default values of the gain ratio.

The software tools might also be compared according to their usability, efficiency and GUI capabilities.

Even for decision tree algorithms the results may differ. In order to show this different WEKA algorithms are applied on the same data. The results are given as Appendix A for comparison.

In addition to this study, larger data set available about heart disease is also applied but it is not in thesis because of timeless and insufficient data information about it.

The last but not the least point is to make the comparison using different training and test data instances to see the differences.

8. SUMMARY AND CONCLUSIONS

8.1 Summary

In this study, knowledge discovery and data mining topics are elaborated and their interrelationships are defined. Classification that is a predictive technique of data mining is used to compare the data mining tools and their algorithms. There are many tools for data mining and many algorithms for them. ORACLE Data Miner, WEKA, R, RapidMiner and ToolDiag are applied on the IRIS data set and predictions are gained from decision trees.

8.2 Conclusions

The IRIS dataset has 4 attributes that contains sepal and petal details for types of iris plant. By applying classification techniques on this dataset some predictions can be gained for IRIS plant. IRIS dataset is available free and it is commonly used for comparison.

8.3 Extension of the Study

In the thesis a sample dataset known as IRIS dataset is treated using the several data mining tools to be able to get some experience in using professional software on the topic and making comparison among them.

As an extension of the thesis, these tools will be applied on a larger data set available about heart disease which is also taken from UC Irvine Machine Learning Repository. It has 303 instances named “Cleveland Dataset” and has 14 attributes.

The real objective of this study is after testing other algorithms, to apply the result on the real heart disease data that may be obtained in the Başkent University Hospitals.

REFERENCES

[1] AKTAŞ, A.Z., Structured Analysis & Design of Information Systems, Prentice Hall, 1987.

[2] BECERRA-FERNANDEZ, I., GONZALEZ, A., and SABHERWAL, R., Knowledge Management, Pearson, 2004.

[3] AWAD, E.M., and GHAZIRI, H.M., Knowledge Management, Prentice Hall, 2004.

[4] LUTZ, H., Knowledge Discovery with Support Vector Machines, John Wiley& Sons, 2009.

[5] BANDYOPADHYAY, S., MAULIK, U., HOLDER L. B., and COOK D. J., Advanced Methods for Knowledge discovery from Complex Data, Springer, 2005.

[6] MAIMON, O., and ROKACH, L., The Data Mining and Knowledge Discovery Handbook, Springer, 2005.

[7] MAIMON, O., and ROKACH, L., Decomposition Methodology for Knowledge Discovery and Data Mining: Theory and Applications, World Scientific, 2005.

[8] FAYYAD, U., PIATETSKY-SHAPIRO, G., and SMYTH, P., From Data Mining to Knowledge Discovery in Databases, AI Magazine 17(3): Fall 1996, 37-54.

[9] GOEBEL, M., and GRUENWALD, L., A Survey of Data Mining and Knowledge Discovery Software Tools, ACM SIGKDD, Vol1., No.1, pp.20-33, June 1999.

[10] FAYYAD, U., PIATETSKY-SHAPIRO, G., and SMYTH, P., The KDD Process for Extracting Useful Knowledge from Volumes of Data, Communication of ACM: November 1996, Vol. 39,No. 11.

[11] KOUTSONIKOLA, V. A., PETRIDOU, S. G., VAKALI, A. I.,and PAPADIMITRIOU G.I., A new approach to web users clustering and validation: a divergence-based scheme, International Journal of Web Information Systems, Vol. 5 Iss: 3, p.348 – 371, 2009.

[12] TAN, P-N., STEINBACH, M., and KUMAR, V., Introduction to Data Mining, Addison-Wesley, 2006.

[13] DE, F., JOSEPH A., and BARNARD, W., URAN Institute's Six Sigma Breakthrough and Beyond - Quality Performance Breakthrough Methods, McGraw-Hill, 2005.

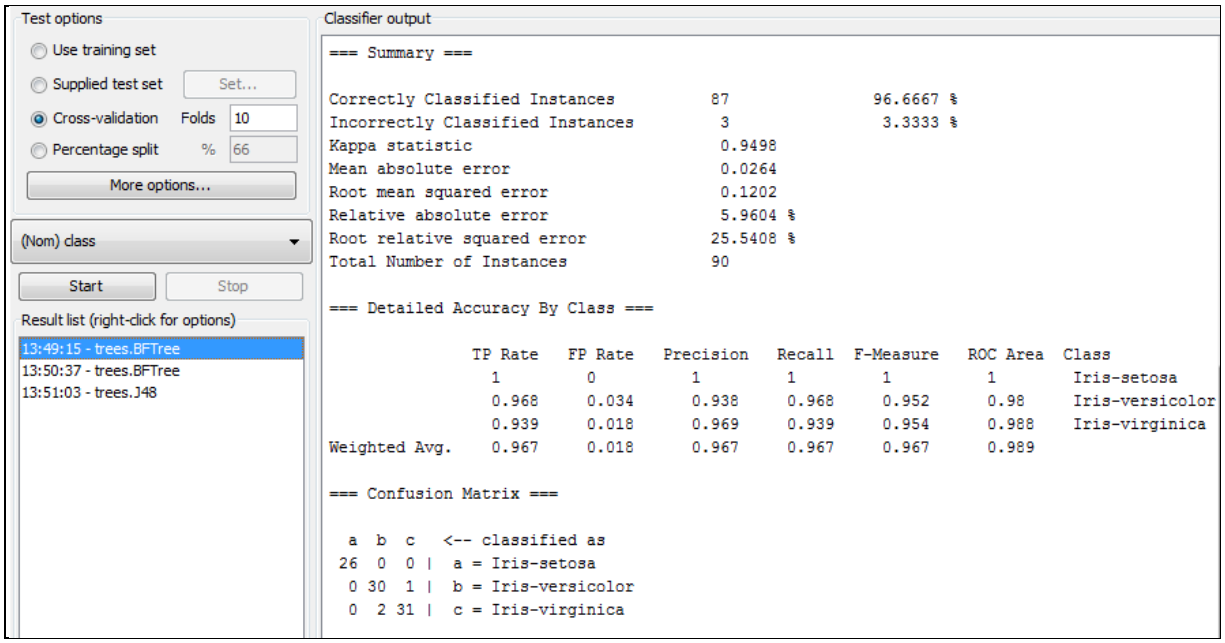
[14] AZEVEDO, A. and SANTOS, M.F., KDD, SEMMA and CRISP-DM: A Parallel Overview, IADIS European Conference, p. 182- 185, 2008.

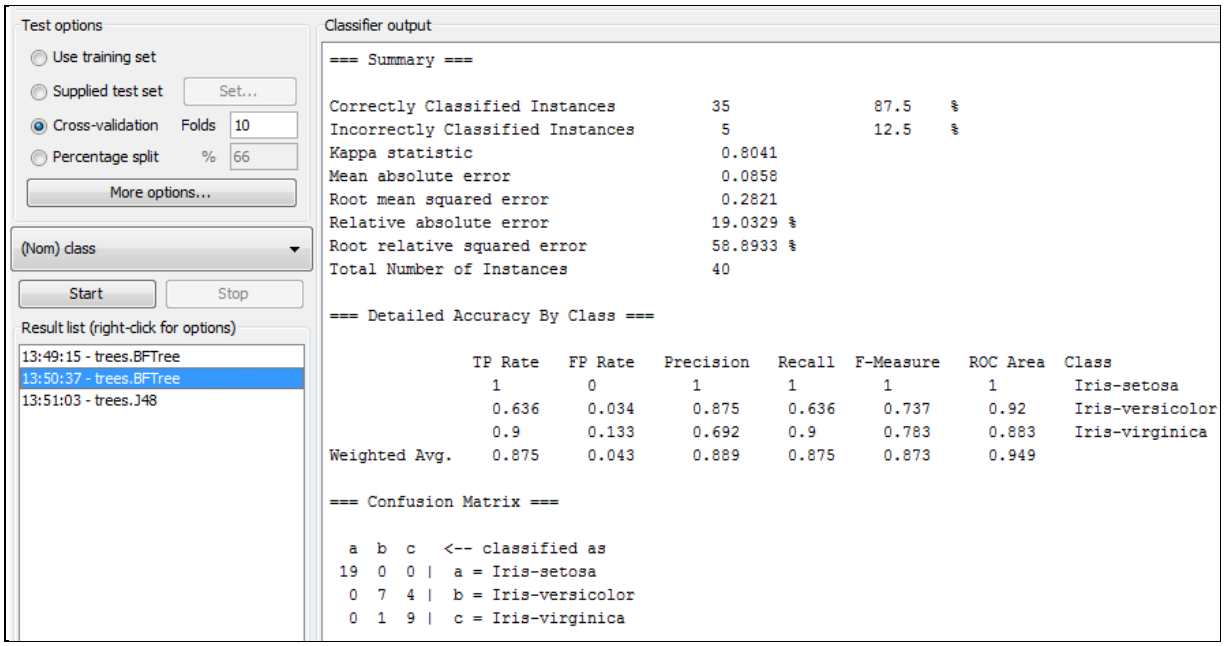
[15] HOLMES, G., DONKIN, A. and WITTEN, I.H., WEKA: A machine learning workbench, In Proc Second Australia and New Zealand Conference on Intelligent Information Systems, Brisbane, Australia, 1994.

APPENDIX A

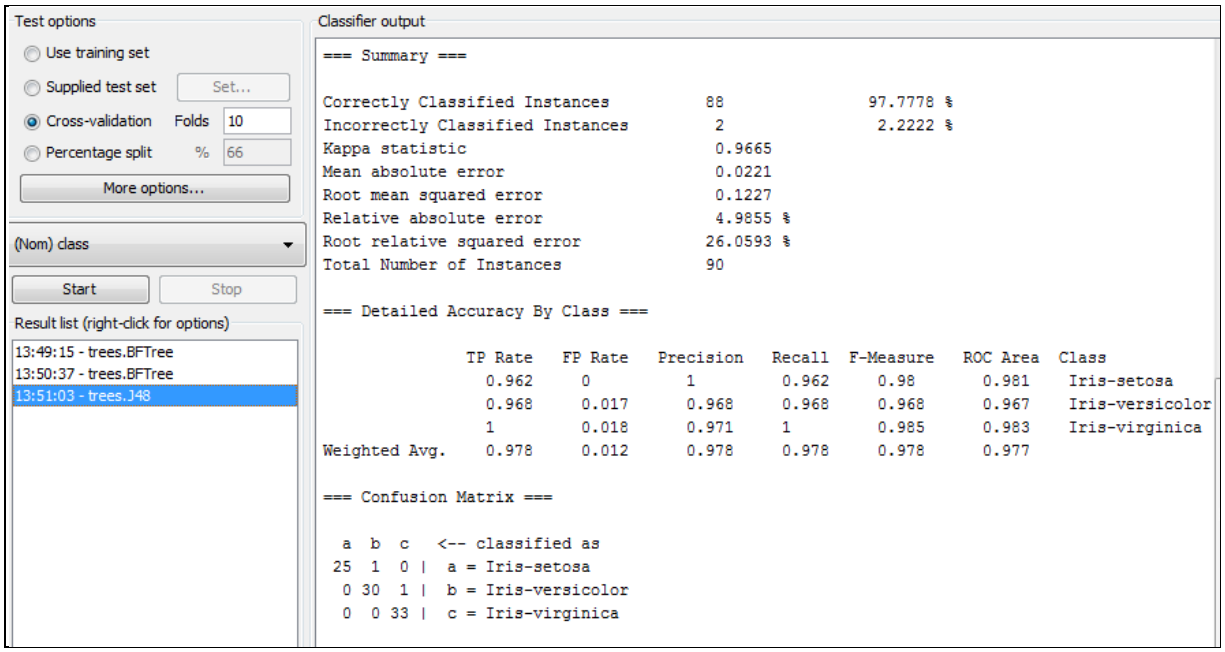
Different WEKA Algorithm Applications

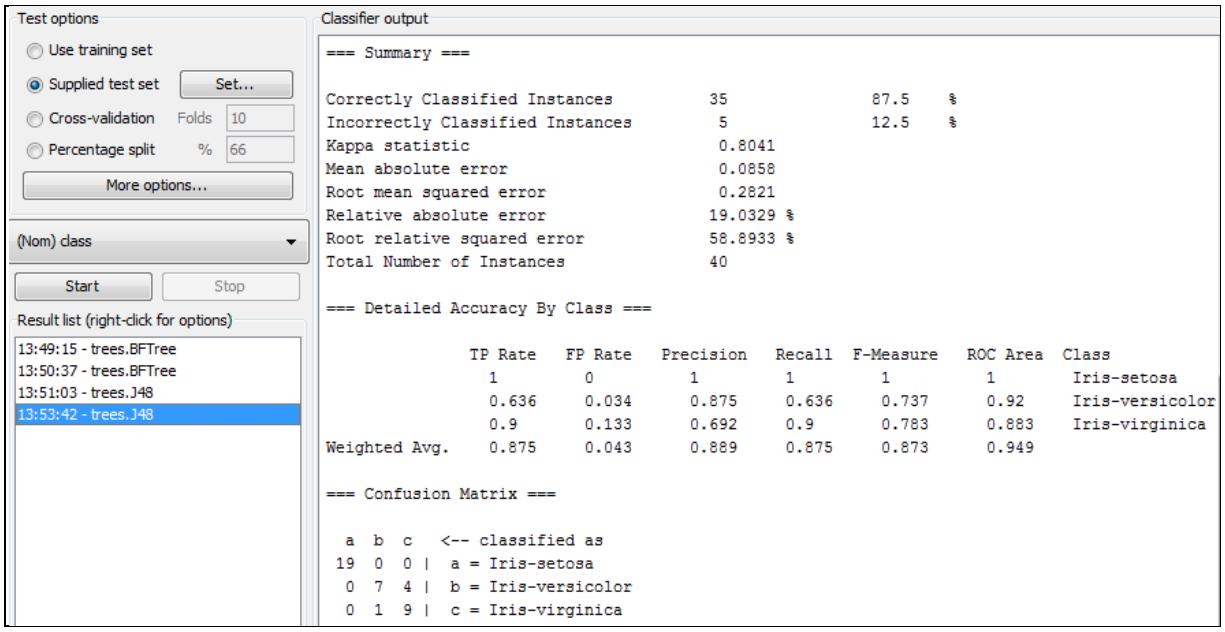
Results for BFTree



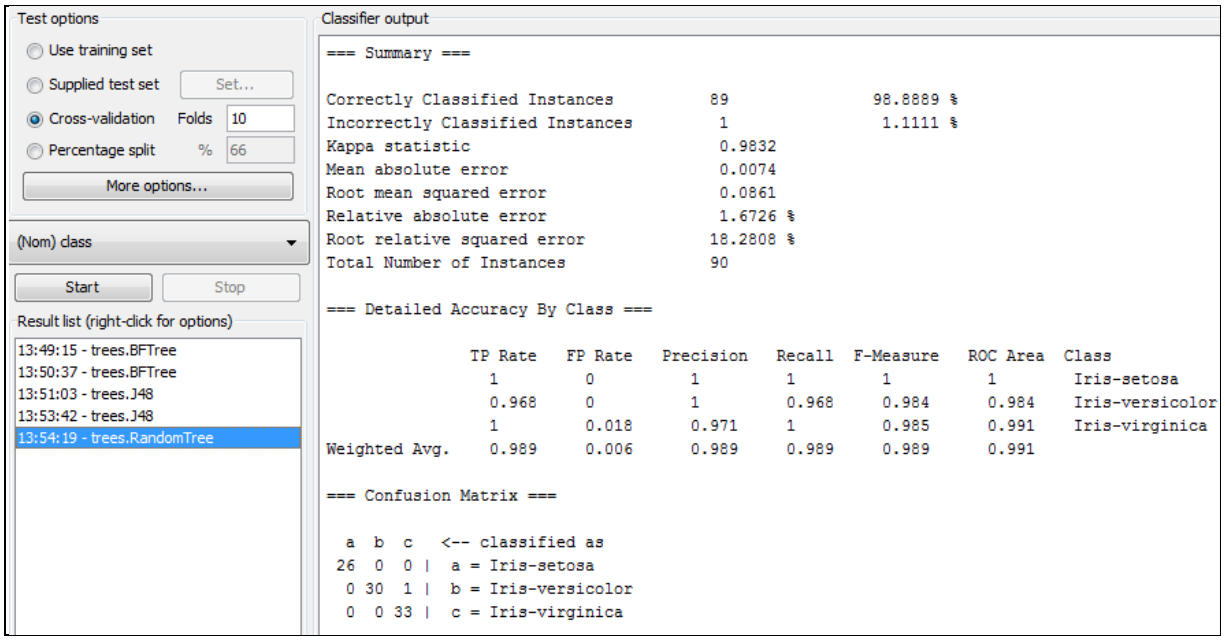


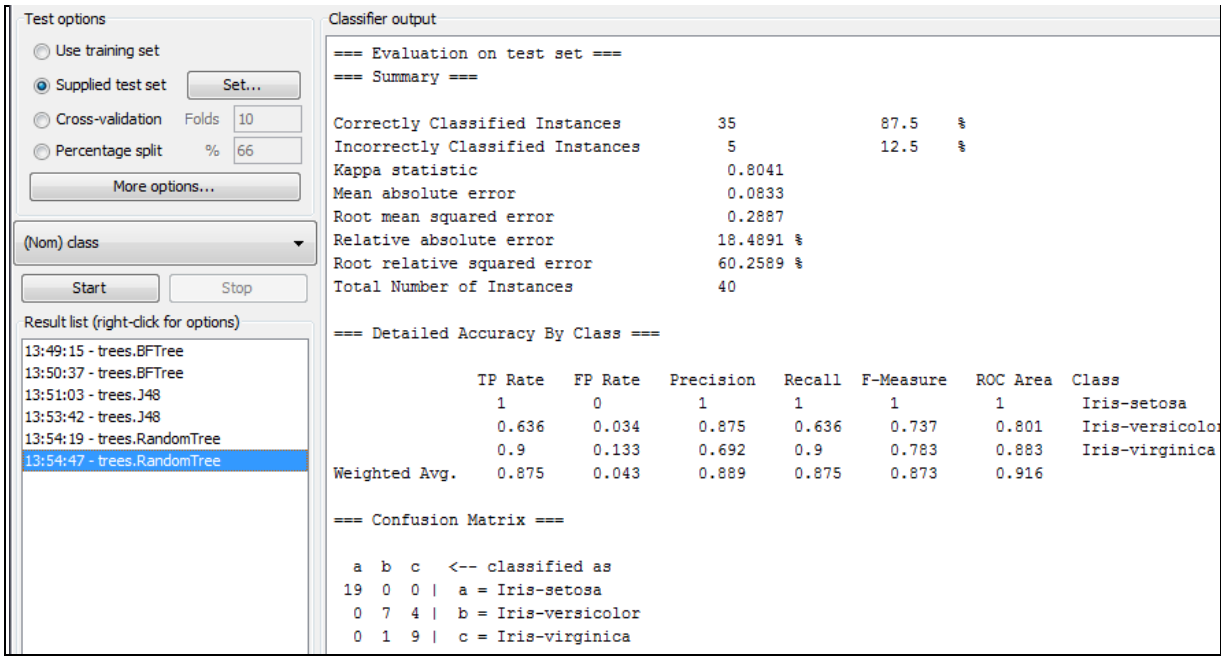
Results for J48





Results for RandomTree





Results for RepTree