Available online at www.isas.org.in/jisas
JOURNAL OF THE INDIAN SOCIETY OF
AGRICULTURAL STATISTICS 73(2) 2019 167–174

Online Classification and Visualization using the C4.5 Decision Tree Algorithm

Shashi Dahiya, Suvajit Das and Anshu Bharadwaj
ICAR-Indian Agricultural Statistics Research Institute, New Delhi

Received 22 January 2019; Revised 26 March 2019; Accepted 28 March 2019

SUMMARY

Classification is the data mining task of assigning the objects to one of the several predefined categories. It is a predictive modelling task in which a model is built for the target variable as a function of the explanatory variables. It is also called the supervised learning since the training dataset has records with predefined labelled classes. These labelled training records supervise the learning of the classification model. The various class labels can be represented by discrete values where the ordering among the values has no meaning. There are many well established techniques for classification, out of which decision tree technique is a very important and popular technique from the machine learning domain. C4.5 is a well-known decision tree algorithm used for classifying datasets which is available in all data mining software. Since it is an important algorithm for inducing the decision trees and generating the rules precisely from the datasets, it is highly used by the data mining and machine learning community. To provide an online platform to the users for applying the algorithm on their datasets without installing any data mining software, a web based software for rule generation and decision tree induction using C4.5 algorithm is developed. The visualization in the form of tree structure enhances the understanding of the generated rules.

Keywords: Classification, Data mining, Predictive modelling, Decision tree, Visualization.

1. INTRODUCTION

Knowledge Discovery in Databases (KDD) is the process of knowledge extraction from big masses of data with the goal of obtaining meaning and consequently understanding of the data, as well as to acquire new knowledge. It consists of a technology composed of a group of mathematical and technical models of software that are used to find patterns and regularities in the data. The KDD process is interactive and iterative; involving the basic steps of data selection, pre-processing, data transformation, data mining and pattern evaluation and ultimately the knowledge is discovered (Azevedo and Santos, 2008). Data mining is an essential step of the KDD process, where intelligent methods are applied in order to extract the data patterns.

Data Mining (DM) is "the nontrivial extraction of implicit, previously unknown, and potentially useful information from data" (Fayyad *et al.*, 1996). It is a fast growing field born through the osmosis of existing disciplines like – Database Systems, Statistics, Machine Learning, Visualization, Algorithms and Other Disciplines. The overall process of DM is shown in Figure 1. Initially the problem is identified and the data related to that problem is collected and prepared using the pre-processing tools for analysis and modelling phases (Dahiya, S. 2017). The data is modelled and analysed using the various data mining tools and techniques relevant to the task such as Classification, Clustering or Association Rule Mining. After Model Building the model is applied to the real situation and validated.

Classification is the DM task of assigning objects to one of the several predefined categories (Han, Kamber & Pei 2012). It is a two-step process, which consists of a learning step (in which a model is constructed) and a classification step (in which the model is used to predict the class labels for the given data). Dunham, 2008 defined the classification problem as follows:

Corresponding author: Shashi Dahiya
E-mail address: shashi.dahiya@icar.gov.in

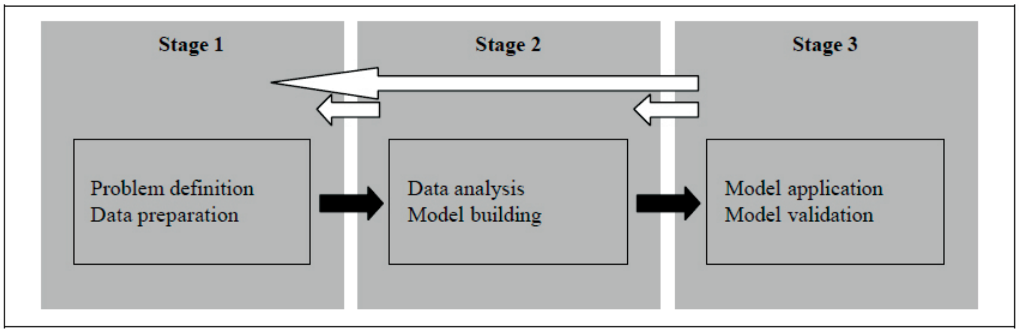


Fig. 1. The overall process of Data Mining

Given a dataset D = {r1, r2, …. , rn} of n records (items, tuples, instances) and a set of classes C = {c1, c2, …, cn}. The classification problem is to define a mapping f: D → C, where each ri is assigned to one class. A class Cj contains precisely those records mapped to it; that is, Cj = {ri | f (ri) = Cj, 1≤ i≤ n}.

This definition views classification as a mapping from dataset to the set of classes. The classes are predefined and distinct and partition the entire dataset. Each record in the dataset is assigned to exactly one class. The classes that exists for a classification problem are indeed equivalence classes. The model construction and prediction phases of a Classifier are depicted in Fig. 2:

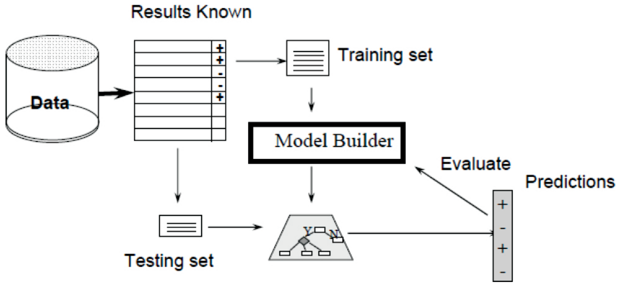


Fig. 2. The Model Construction and Prediction phases of a Classifier

Machine Learning (ML) is concerned with the design and development of algorithms and techniques that allow computers to "learn" and evolve behavior including aspects of intelligence based on empirical data (Ethem, 2010). It is a main research area for computer programs to automatically learn to recognize complex patterns and make intelligent decisions based on data (Mohri *et al.*, 2012). The major focus of machine learning is to extract information from data automatically, by computational and statistical methods. For classification task, ML research often focuses on the accuracy of the model along with the efficiency and scalability of mining methods on large datasets (Han, Kamber & Pei 2012).

Some of the ML classification techniques are Decision Tree (DT), Support Vector Machine (SVM), Genetic Algorithm (GA), Artificial Neural Networks (ANN), Logistic Regression (LR) and Naïve Bayes (NB).

DT is one of the popular and widely used machine learning technique for classification. DT is commonly built by recursive partitioning (Quinlan 1993). A univariate (single attribute) split is chosen for the root of the tree using some attribute selection measure (e.g., mutual information, gain ratio, gini index). The data is then divided according to the test, and the process repeats recursively for each child. The decision node is an attribute test with each branch being a possible value of the attribute (Peng *et al.*, 2009). The resulting decision tree can be validated using test data instances. Some of decision tree algorithms are ID3, C4.5, NBTree, CART, CHAID, QUEST and others. Different Decision tree algorithms use different attribute selection measures to calculate the information contained in each attribute to select the root node of the decision tree.

1.1 C4.5 Algorithm

C4.5 algorithm uses decision tree as a classifier for classifying the datasets. It is Ross Quinlan's enhancement of his own ID3 algorithm for decision tree classification.

The major enhancements in C4.5 algorithm are:

- Handling both continuous and discrete attributes - In order to handle continuous attributes, C4.5 creates a threshold and then splits the list into those whose attribute value is above the threshold and those that are less than or equal to it.
- Handling training data with missing attribute values.
- C4.5 uses gain ratio as attribute selection measure while ID3 uses simple information gain which is biased towards attributes with large number of values. If an attribute in the dataset is an id then it would be chosen. This is because each branch would produce a leaf, which would cause Info id (D) = 0. Thus, information gain would be maximal because Gain (A) = Info(D). To avoid this biasness, gain ratio is used to select the root node of the decision tree in C4.5.

C4.5 performs very well in classifying the dataset and it is often referred to as a statistical classifier. It uses a statistical measure called entropy from information Theory (Ghahramani, 2000) and builds the tree from the top down, with no backtracking from a fixed set of examples. The concept learning process in C4.5 is illustrated in Fig. 3.

STEPS:

1. Check for base cases

2. For each attribute *a*

1. Find the normalized information gain from splitting on *a*

3. Let *a_best* be the attribute with the highest normalized information gain

4. Create a decision *node* that splits on *a_best*

5. Recurse on the sublists obtained by splitting on *a_best*, and add those nodes as children of *node*

Fig. 2. Concept learning in C4.5 algorithm

Fig. 3. The Concept Learning Process in C4.5

The implementation of C4.5 algorithm is available in standalone open source data mining software such as WEKA (http://www.cs.waikato.ac.nz/) and RapidMiner (http://rapidminer.com) which needs to be downloaded first and then use the algorithm. It is available in licensed software such as Statistica and Clementine which needs to be purchased for using the algorithm. There is no online implementation of the algorithm which could provide its usage without the hurdle of downloading or purchasing of the software. Moreover, the complete tree visualization with generated rules and accuracy measures is not available in most of the standalone software. Hence an online software for classification and visualization of data using C4.5 algorithm is highly required by the researchers, students and faculty working in the area of data mining, machine learning, visualization and statistical analysis.

2. METHODOLOGY

The modified waterfall development model (Munassar and Govardhan, 2010) is used for the development of Online Decision Tree Classification software (ODTC). It is a web based application based on the standard three-layer client server architecture. It can be accessed from any internet connected device.

The following three layers of the application perform different functionalities:

1. Client Side Interface Layer (CSIL)
2. Server Side Application Layer (SSAL)
3. Data Base Layer (DBL)

CSIL is the presentation layer of the application. It ensures that the system is user friendly. It consists of data forms for accepting information from the user and validating those forms using JavaScript. This is the most important layer because it acts as an interface between the user and the software. It is implemented using HTML, Cascading Style Sheet (CSS) and JavaScript embedded within ASP.NET pages.

SSAL is the business layer of the application. It consists of application logic on the web server which can mediate the communication between the client side and the database layers. It is implemented using Visual C# language in ASP.NET framework. ASP.NET is a powerful and flexible technology for creating dynamic web pages (Chris, 2010). It is a convergence of two major Microsoft technologies, Active Server Pages (ASP) and the .NET framework (Walther, 2011). This layer encapsulates the entire interaction with the database and hides the details from the presentation layer.

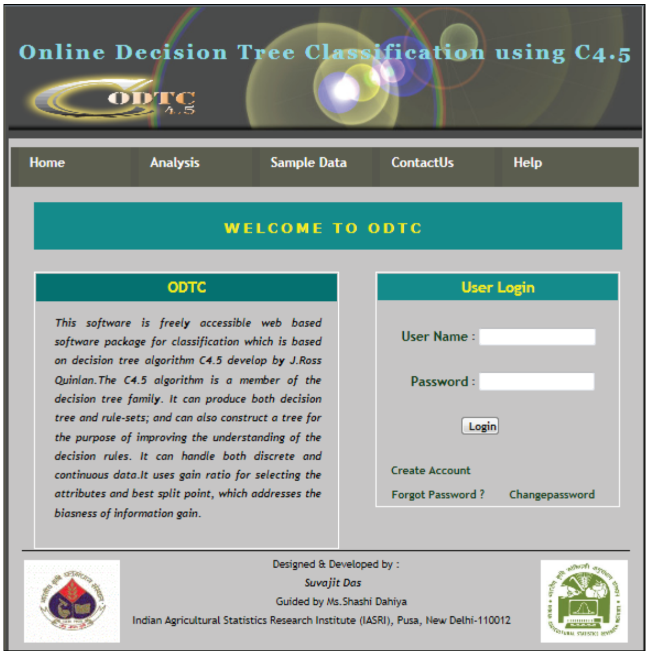
DBL has been implemented using Microsoft SQL Server (Varga *et al.*, 2016) which is a relational database management system. The purpose of the DBL is to store the data in an organized and structured way and to enable the retrieval of specified data by multiple, simultaneous users. ADO.NET has been used for the connectivity of the database with the application. It is the data access technology built into the .NET framework (Esposito, 2005). The database has been used to record the Login credentials and other related details about each user registering on the software website and for storing the decision rules generated from of the input data.

3. SOFTWARE DESCRIPTION AND FEATURES

Online Decision Tree Classification (ODTC) is web based software freely accessible to the registered users. Security is ensured to ODTC as only authenticated users can access the software. The data management part of the software provides the facility to enter the data online in a grid structure similar to a Microsoft Excel spread sheet. The data entry task is very easy in this grid structure as each data entity is entered separately in a cell which is a combination of a row and a column. This data can be saved and analysed later using the Analysis option present on the home page. It also provides the facility to import the data from xls, csv or txt file formats. If the data is incomplete or has some missing values, the software provides the provision to fill up the missing attribute values using imputation. The imputation method is different for numerical and categorical data. ODTC is a decision rule generation software that generates the rules using the If, Then and Else keywords. These rules classify the records and assign a class label to it, thereby providing a clear-cut decision on the record. These rules can also be exported and saved in an excel format. It also facilitates the researchers to formulate rules from their findings that can enhance decision making and further analytical activities. It also provides the visualization facility for the generated rules in the form of a decision tree which enhances the understanding of the generated rules.

3.1 User Management

A new user needs to get registered and creates its login credentials by filling up the registration form available on the Create New Account Link available on the home page of the software. A registered user has to login into the system by filling up the login form available on the home page. If the user forgets his password, ODTC provides an option to reset the password using the Forgot Password functionality. The registered user can also change its password using the Change Password functionality (Fig. 4).



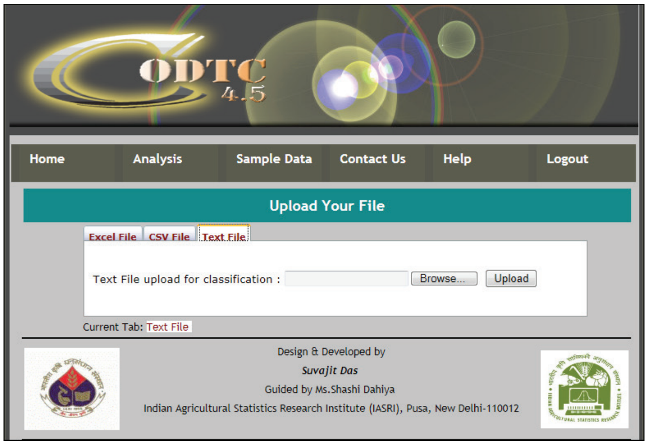
This software is freely accessible web based software package for classification which is based on decision tree algorithm C4.5 develop by J.Ross Quinlan.The C4.5 algorithm is a member of the decision tree family. It can produce both decision tree and rule-sets; and can also construct a tree for the purpose of improving the understanding of the decision rules. It can handle both discrete and continuous data.It uses gain ratio for selecting the attributes and best split point, which addresses the biasness of information gain.

Designed & Developed by :
Suvajit Das
Guided by Ms.Shashi Dahiya
Indian Agricultural Statistics Research Institute (IASRI), Pusa, New Delhi-110012

Fig. 4. User Management

3.2 Input Data Management

After the successful login, the user can upload his data file available in any of the three file formats supported by the software. The three file formats supported by the system are the Excel, CSV and Text file formats (Fig. 5).



Text File upload for classification :

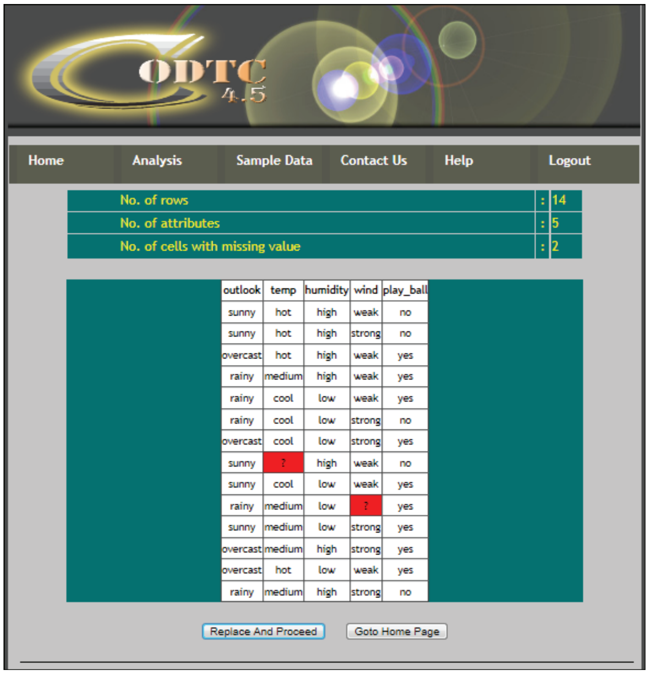
Current Tab: Text File

Design & Developed by
Suvajit Das
Guided by Ms.Shashi Dahiya
Indian Agricultural Statistics Research Institute (IASRI), Pusa, New Delhi-110012

Fig. 5. File upload

3.3 Missing Value Handling

For classifying a data set with high accuracy, the dataset should not have missing values. ODTC provides the facility to detect and impute the missing cells in the uploaded dataset. All the missing values in the dataset are replaced by a red colour cell with a question mark ("?"). In case of categorical attributes, the missing values will be imputed by the *MODE* (highest frequency value) of all distinct values of the respective attribute. In case of numerical attributes, the missing values will be imputed by the *ARITHMATIC MEAN* of all those values of the respective attribute, which belongs to the same class in which the instance containing the missing value belongs. The system also represents the total number of rows, number of attributes and number of cells with the missing values on the top of the data grid (Fig. 6).



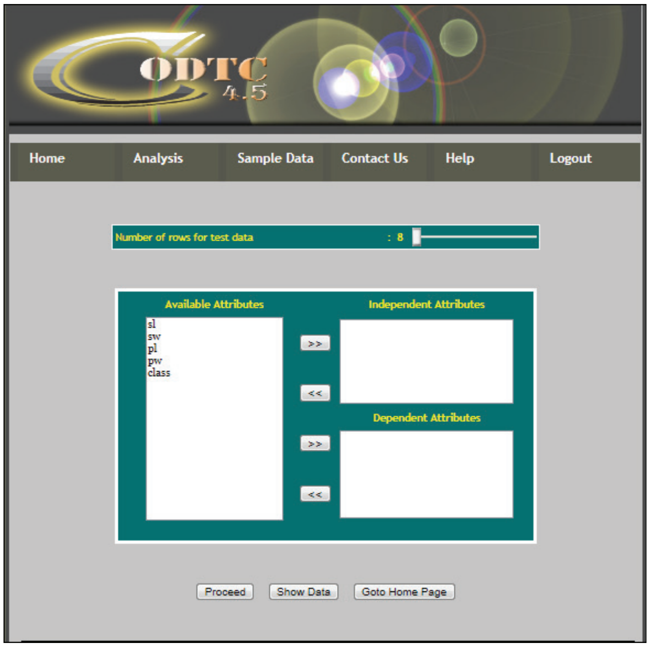
No. of rows	:	14
No. of attributes	:	5
No. of cells with missing value	:	2

outlook	temp	humidity	wind	play_ball
sunny	hot	high	weak	no
sunny	hot	high	strong	no
overcast	hot	high	weak	yes
rainy	medium	high	weak	yes
rainy	cool	low	weak	yes
rainy	cool	low	strong	no
overcast	cool	low	strong	yes
sunny	?	high	weak	no
sunny	cool	low	weak	yes
rainy	medium	low	?	yes
sunny	medium	low	strong	yes
overcast	medium	high	strong	yes
overcast	hot	low	weak	yes
rainy	medium	high	strong	no

Fig. 6. Missing value handling by ODTC

3.4 Dataset Partition and Attribute Selection

For classification of the dataset, ODTC provides the facility to partition the dataset randomly into training set and the test set according to the number of test instances chosen by the user. The test data partition can be selected by the user between a minimum of 5% and maximum of 40% of the total number of data instances. For example, if the user has uploaded a dataset of 1000 instances then the number of test data instances can be selected between 50 (5% of 1000) to 400 (40% of 1000). The remaining part of the dataset will be considered as the training dataset. The user has to specify the desired number of test instances by scrolling the sliding bar provided at the top of the screen (Fig. 7). For demonstration, the software uses the IRIS dataset available online as part of the machine learning repository (http://archive.ics.uci.edu). This dataset is also available under the Sample Data tab on the home page of the software.



Number of rows for test data : 8

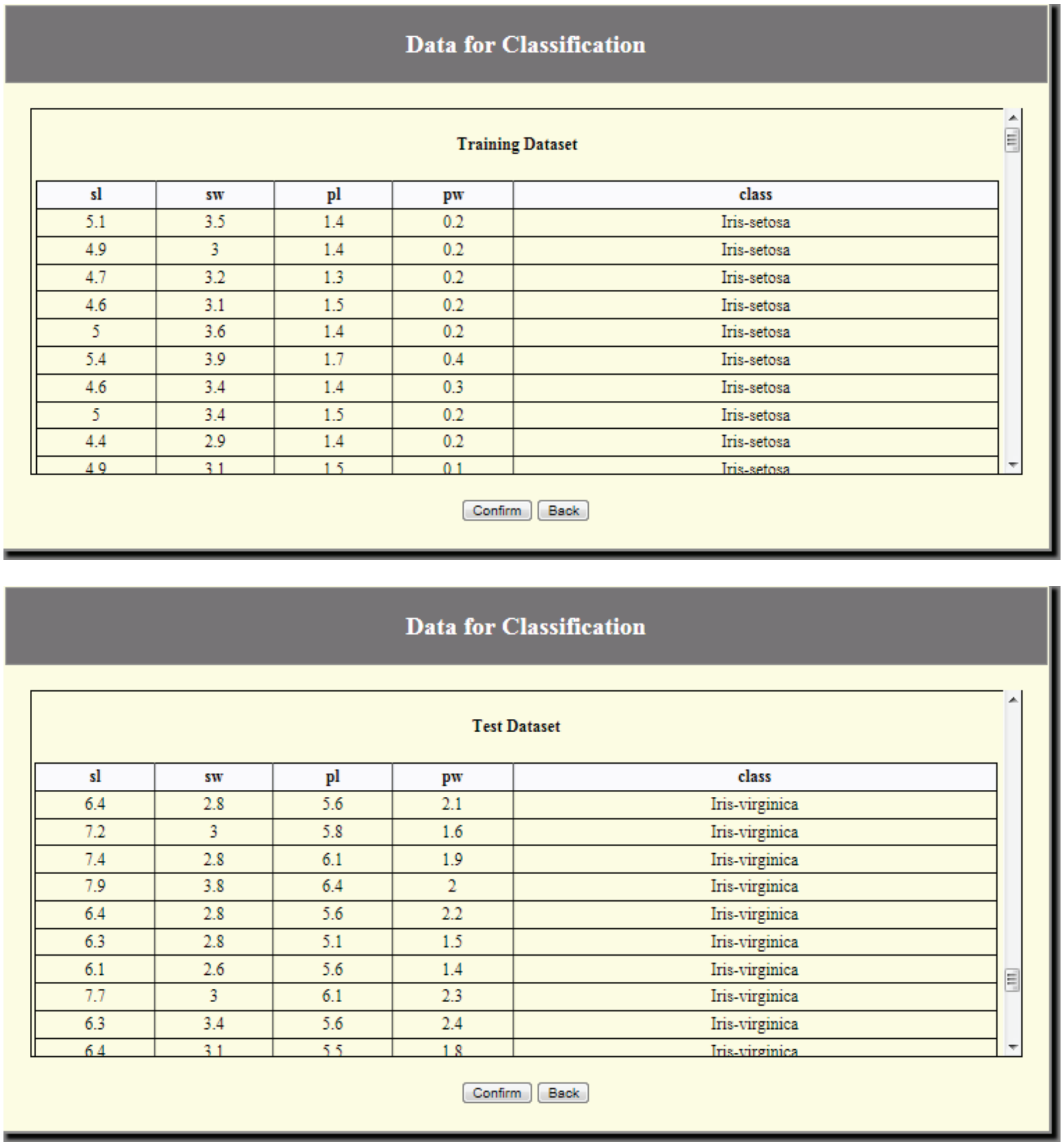
sl
sw
pl
pw
class

Fig. 7. Attribute Selection in ODTC

After completing the selection of test data instances, the dependent and independent attributes for classification are selected by the user one by one using the forward arrows provided on the same screen (Fig. 3.4). After clicking on the proceed button, user can see the prepared dataset for classification as per the independent and dependent attributes selected and the test data set instances specified by him. The dependent attribute will be depicted as the Class attribute in the prepared dataset for classification (Fig. 8).

3.5 Rule Generation

The user can confirm the training dataset and test data set partitions or can go back for changing anything in the selection process. After confirming the dataset for classification, the software uses the C4.5 algorithm at the backend to learn the training data. It then generates the decision rules in the form



Data for Classification

Training Dataset

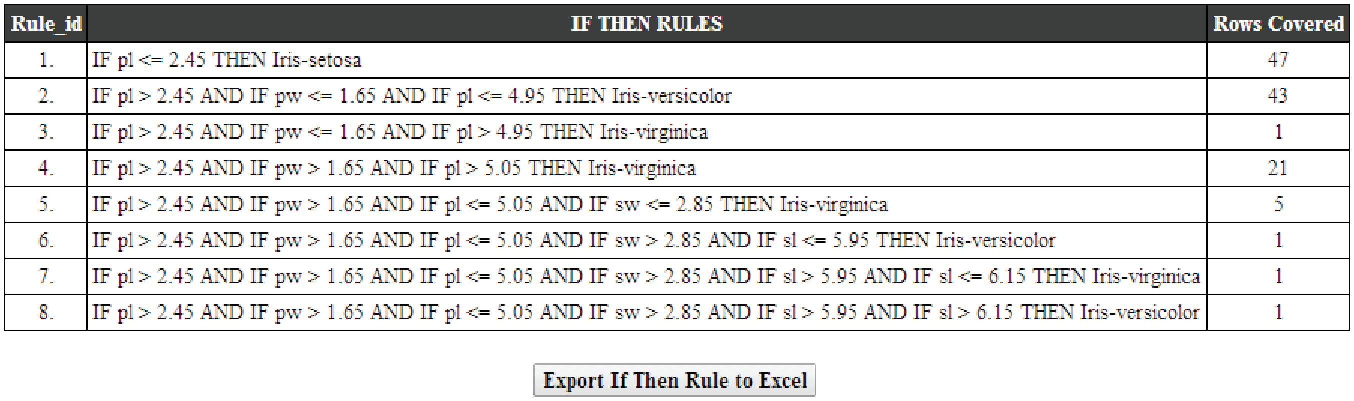
sl	sw	pl	pw	class
5.1	3.5	1.4	0.2	Iris-setosa
4.9	3	1.4	0.2	Iris-setosa
4.7	3.2	1.3	0.2	Iris-setosa
4.6	3.1	1.5	0.2	Iris-setosa
5	3.6	1.4	0.2	Iris-setosa
5.4	3.9	1.7	0.4	Iris-setosa
4.6	3.4	1.4	0.3	Iris-setosa
5	3.4	1.5	0.2	Iris-setosa
4.4	2.9	1.4	0.2	Iris-setosa
4.9	3.1	1.5	0.1	Iris-setosa

Data for Classification

Test Dataset

sl	sw	pl	pw	class
6.4	2.8	5.6	2.1	Iris-virginica
7.2	3	5.8	1.6	Iris-virginica
7.4	2.8	6.1	1.9	Iris-virginica
7.9	3.8	6.4	2	Iris-virginica
6.4	2.8	5.6	2.2	Iris-virginica
6.3	2.8	5.1	1.5	Iris-virginica
6.1	2.6	5.6	1.4	Iris-virginica
7.7	3	6.1	2.3	Iris-virginica
6.3	3.4	5.6	2.4	Iris-virginica
6.4	3.1	5.5	1.8	Iris-virginica

Fig. 8. Training and Test Data Set partitions in ODTC



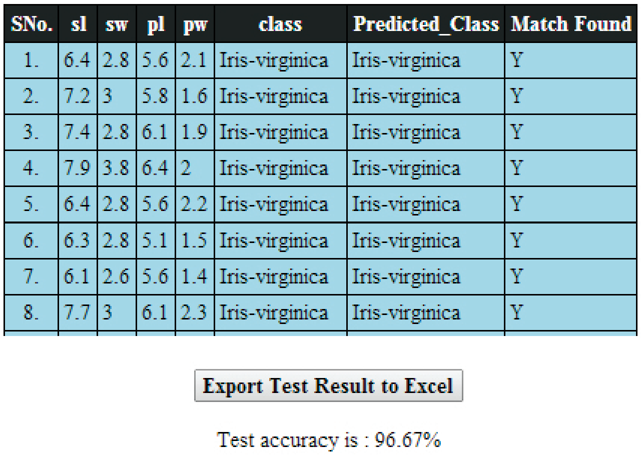
Rule_id	IF THEN RULES	Rows Covered
1.	IF pl <= 2.45 THEN Iris-setosa	47
2.	IF pl > 2.45 AND IF pw <= 1.65 AND IF pl <= 4.95 THEN Iris-versicolor	43
3.	IF pl > 2.45 AND IF pw <= 1.65 AND IF pl > 4.95 THEN Iris-virginica	1
4.	IF pl > 2.45 AND IF pw > 1.65 AND IF pl > 5.05 THEN Iris-virginica	21
5.	IF pl > 2.45 AND IF pw > 1.65 AND IF pl <= 5.05 AND IF sw <= 2.85 THEN Iris-virginica	5
6.	IF pl > 2.45 AND IF pw > 1.65 AND IF pl <= 5.05 AND IF sw > 2.85 AND IF sl <= 5.95 THEN Iris-versicolor	1
7.	IF pl > 2.45 AND IF pw > 1.65 AND IF pl <= 5.05 AND IF sw > 2.85 AND IF sl > 5.95 AND IF sl <= 6.15 THEN Iris-virginica	1
8.	IF pl > 2.45 AND IF pw > 1.65 AND IF pl <= 5.05 AND IF sw > 2.85 AND IF sl > 5.95 AND IF sl > 6.15 THEN Iris-versicolor	1

Fig. 9. Decision Rules generated by ODTC using C4.5 Algorithm

of If then statements from the training data set. Along with the rules user can get information about the "Rule coverage" (Number of instances covered by a particular rule) (Fig. 9). The decision rules can be exported to Microsoft Excel for analysis.

3.6 Evaluation of Results

After learning the training dataset using C4.5 algorithm, the software classifies the test dataset by predicting the class value for each test instance. The match found is 'Y' if the Actual Class and the Predicted class are same. The Results can be exported to Microsoft Excel for analysis. The "Test accuracy" which is the percentage of data instances correctly classified by the algorithm is also depicted in the result screen (Fig. 10). User has the facility to export all these results into excel format.

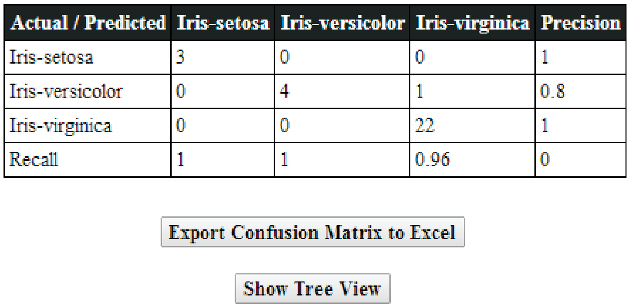


SNo.	sl	sw	pl	pw	class	Predicted_Class	Match Found
1.	6.4	2.8	5.6	2.1	Iris-virginica	Iris-virginica	Y
2.	7.2	3	5.8	1.6	Iris-virginica	Iris-virginica	Y
3.	7.4	2.8	6.1	1.9	Iris-virginica	Iris-virginica	Y
4.	7.9	3.8	6.4	2	Iris-virginica	Iris-virginica	Y
5.	6.4	2.8	5.6	2.2	Iris-virginica	Iris-virginica	Y
6.	6.3	2.8	5.1	1.5	Iris-virginica	Iris-virginica	Y
7.	6.1	2.6	5.6	1.4	Iris-virginica	Iris-virginica	Y
8.	7.7	3	6.1	2.3	Iris-virginica	Iris-virginica	Y

Test accuracy is : 96.67%

Fig. 10. Result of Classification

The "Confusion Matrix" of test dataset is also produced by the software depicting the Actual vs. Predicted values of all class values along with the Precision and Recall values as shown in Fig. 11.



Actual / Predicted	Iris-setosa	Iris-versicolor	Iris-virginica	Precision
Iris-setosa	3	0	0	1
Iris-versicolor	0	4	1	0.8
Iris-virginica	0	0	22	1
Recall	1	1	0.96	0

Fig. 11. Confusion Matrix formed by ODTC for Iris Dataset

3.7 Decision Tree Visualization

Using the "Show Tree View" button user can go for visualization of the decision tree which is formed by using the C4.5 algorithm. It is a pictorial representation of finding out the class value of a test data instance by following the generated decision rules using the algorithm. It enhances the understanding of the generated rules. The decision tree formed by the software for the Iris dataset is depicted in Fig. 12. The print option for printing the decision tree is also provided by the software on the top of the Decision Tree.

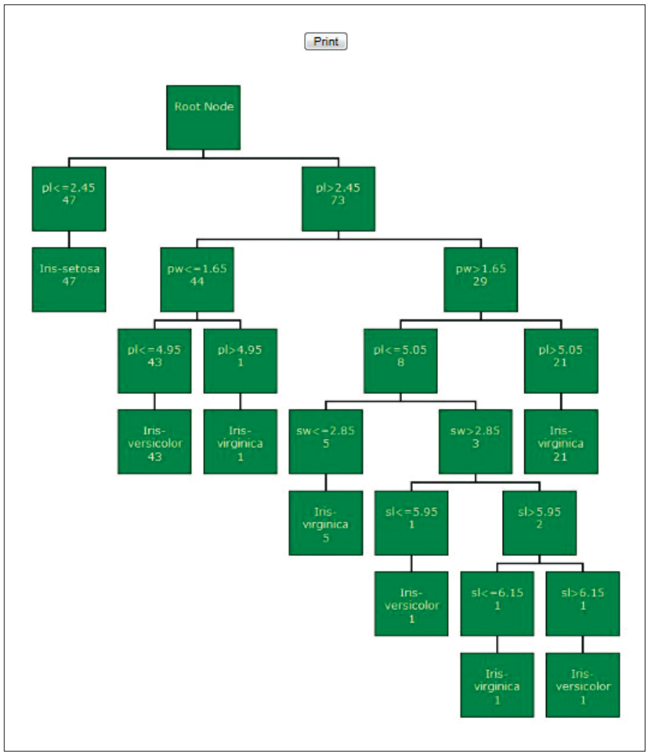


Fig. 12. Decision Tree Visualization by ODTC (Iris Dataset)

The software was validated using the weather dataset and the bench mark Iris dataset provided in the sample data module.

4. CONCLUSION

The results obtained from ODTC were comparable and in some cases found better when tested with WEKA. ODTC is an online facility of generation of decision rules along with their evaluation measures. The software is user friendly and does not demand expertise of computer programming. Using ODTC, user can register, login, generate rules, visualize the results and generate a decision tree easily using the C4.5 algorithm. The system can be used for classifying and extracting the hidden patterns in huge datasets using the C4.5 algorithm online. There is no need to install or purchase a standalone software for using the algorithm. This can be used as a knowledge generation software which can extract and impart knowledge about a particular dataset in the form of decision rules and decision tree online. This generated knowledge can be used by researchers, academicians and students working in the area of data mining, machine learning, statistical analysis and visualization. Online software for other important data mining algorithms can be developed.

REFERENCES

Azevedo, A. and Santos, M.F. (2008). KDD, SEMMA and CRISP-DM: a parallel overview Archived 2013-01-09 at the Wayback Machine. *In Proceedings of the IADIS European Conference on Data Mining*, 182–185.

Chris, L. (2010). ASP.NET 3.5 Website Programming: Problem - Design - Solution. Wrox Press Ltd. Birmingham, UK,

Dahiya, S. (2017). A Hybrid Intelligent Data Mining Approach for Credit Risk Management, Ph.D Thesis, MRIU, India.

Dunham, M.H. (2008). Data Mining Introductory and Advanced Topics. Pearson Education.

Esposito, D. (2005). Programming Microsoft ASP.NET 2.0. WP Publishers & Distributers Pvt. Limited, Bangalore, India.

Ethem, A. (2010). Introduction to Machine Learning. London: The MIT Press.

Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (1996). The KDD process for extracting useful knowledge from volumes of data. *Communications of the ACM*, **39(11)**, 27-34.

Ghahramani, Z. (2000). Information Theory. Macmillian Reference Ltd, UK.

Han, J., Kamber, M., Pei, J. (2012). Data Mining: Concepts and Techniques. Morgan Kaufmann, Elsevier, USA.

Mohri, M., Rostamizadeh, A., Talwalkar, A. (2012). Foundations of Machine Learning. USA, Massachusetts: MIT Press. ISBN 9780262018258.

Munassar, N.M.A. and Govardhan, A. (2010). A Comparison Between Five Models of Software Engineering. IJCSI International Journal of Computer Science, **7(5)**, 94-101.

Peng, W., Chen, J. and Zhou, H. (2009). An Implementation of ID3-Decision Tree Learning Algorithm. School of Computer Science & Engineering, Australia.

Quinlan, R. (1993). C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers, San Mateo, CA.

Varga, S., Cherry, D., D' Antony, J. (2016). Introducing Microsoft SQL Server. Microsoft Publication.

Walther, S. (2011). ASP.NET 3.5 Unleashed. Sams Publishers, ISBN: 9780768680256.

WEB REFERENCES

http://archive.ics.uci.edu (A Machine Learning Repository, Center for Machine Learning and Intelligent Systems)

http: //www.cs.waikato.ac.nz/ (The WEKA open Source Software)

http://www.rapidminer.com (The RapidMiner open source Software)