Comparative Analysis of Classification Algorithms on Different Datasets using WEKA

Rohit Arora
M.Tech. CSE Deptt.
Hindu College of Engineering
Sonepat, Haryana, India

Suman
Asstt. Prof. CSE Deptt.
Hindu College of Engineering
Sonepat, Haryana, India

ABSTRACT
Data mining is the upcoming research area to solve various problems and classification is one of main problem in the field of data mining. In this paper, we use two classification algorithms J48 (which is java implementation of C4.5 algorithm) and multilayer perceptron alias MLP (which is a modification of the standard linear perceptron) of the Weka interface. It can be used for testing several datasets. The performance of J48 and Multilayer Perceptron have been analysed so as to choose the better algorithm based on the conditions of the datasets. The datasets have been chosen from UCI Machine Learning Repository. Algorithm J48 is based on C4.5 decision based learning and algorithm Multilayer Perceptron uses the multilayer feed forward neural network approach for classification of datasets. When comparing the performance of both algorithms we found Multilayer Perceptron is better algorithm in most of the cases.

Keywords
Classification, Data Mining Techniques, Decision Tree, Multilayer Perceptron

1. INTRODUCTION
Data mining is the process to pull out patterns from large datasets by joining methods from statistics and artificial intelligence with database management. It is an upcoming field in today world in much discipline. It has been accepted as technology growth and the need for efficient data analysis is required. The plan of data mining is not to give tight rules by analysing the data set, it is used to guess with some certainty while only analysing a small set of the data.

In recent times, data mining has been obtained a great attention in the knowledge and information industry due to the vast availability of large amounts of data and the forthcoming need for converting such data into meaningful information and knowledge. The data mining technology is one comprehensive application of technology item relying on the database technology, statistical analysis, artificial intelligence, and it has shown great commercial value and gradually to other profession penetration in the retail, insurance, telecommunication, power industries use [1].

The major components of the architecture for a typical data mining system are shown in Fig 1 [2].

Good system architecture will make possible the data mining system to make best use of the software environment. It achieves data mining tasks in an effective and proper way to exchange information with other systems which is adaptable to users with diverse requirements and change with time.

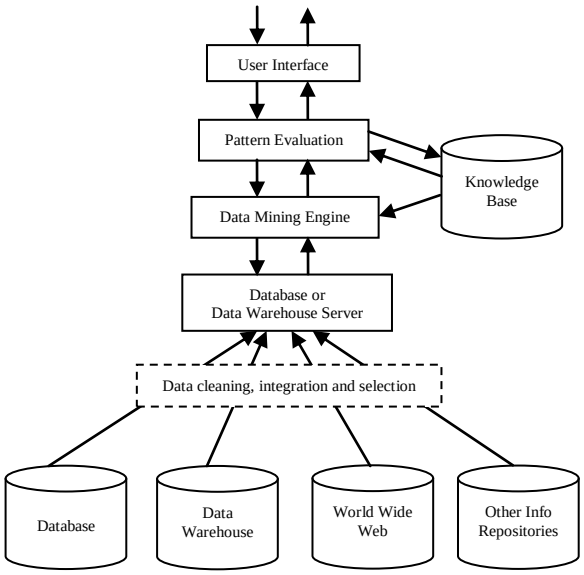

Fig 1: Architecture of a Typical Data Mining System

2. RELATED WORK
Recently studies have been done on various performance of decision tree and on backpropagation.

Classification is a classical problem in machine learning and data mining [3].

Decision trees are popular because they are practical and easy to understand. Rules can also be extracted from decision trees easily. Many algorithms, such as ID3 [4] and C4.5 [5], have been devised for decision tree construction.

In [6] neural networks are suitable in data-rich environments and are typically used for extracting embedded knowledge in the form of rules, quantitative evaluation of these rules, clustering, self-organization, classification and regression. They have an advantage, over other types of machine learning algorithms, for scaling.

The use of neural networks in classification is not uncommon in machine learning community [7]. In some cases, neural networks give a lower classification error rate than the decision trees but require longer learning time [8], [9]. A decision tree can be converted to a set of (mutually exclusive) rules, each one corresponding to a tree branch. Algorithms have been proposed to learn directly sets of rules (that may not be representable by a tree) [10] or to simplify the set of rules corresponding to a decision tree [5].

The alternating decision tree method [11] is a classification algorithm that tries to combine the interpretability of decision trees with the accuracy improvement obtained by boosting.

3. METHODOLOGY

3.1 Datasets

There are five datasets we have used in our paper taken from UCI Machine Learning Repository [12]. The details of each datasets are shown in Table 1.

Table 1: Details of 5 datasets

Datasets	Instances	Attributes	No. of Classes	Type
balance-scale	625	5	3	Numeric
diabetes	768	9	2	Numeric
glass	214	10	7	Numeric
lymphography	148	19	4	Nominal
vehicle	946	19	4	Numeric

The first dataset balance-scale [12] was generated to model psychological experimental results. The attributes are the left weight, the left distance, the right weight, and the right distance. The correct way to find the class is the greater of (left-distance * left-weight) and (right-distance * right-weight). If they are equal, it is balanced.

In the diabetes dataset [12] several constraints were placed on the selection of instances from a larger database. In particular, all patients here are females at least 21 years old of Pima Indian heritage.

The glass dataset [12] is used to determine whether the glass was a type of "float" glass or not.

In the lymphography dataset [12] there is one of three domains provided by the Oncology Institute that has repeatedly appeared in the machine learning literature.

The datastet vehicle [12] is used to classify a given outline as one of four types of vehicle, using a set of features extracted from the profile. The vehicle may be viewed from one of many different angles.

3.2 Weka interface

Weka (Waikato Environment for Knowledge Analysis) is a popular suite of machine learning software written in Java, developed at the University of Waikato, New Zealand [13]. The Weka suite 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.

The original non-Java version of Weka was TCL/TK front-end software used to model algorithms implemented in other programming languages, plus data preprocessing utilities in C, and a Makefile-based system for running machine learning experiments.

This Java-based version (Weka 3) is used in many different application areas, in particular for educational purposes and research. There are various advantages of Weka:

- It is freely available under the GNU General Public License
- It is portable, since it is fully implemented in the Java programming language and thus runs on almost any architecture
- It is a huge collection of data preprocessing and modeling techniques
- It is easy to use due to its graphical user interface

Weka supports several standard data mining tasks, more specifically, data preprocessing, clustering, classification, regression, visualization, and feature selection. All techniques of Weka's software are predicated on the assumption that the data is available as a single flat file or relation, where each data point is described by a fixed number of attributes (normally, numeric or nominal attributes, but some other attribute types are also supported).

3.3 Classification algorithm J48

J48 algorithm of Weka software is a popular machine learning algorithm based upon J.R. Quilan C4.5 algorithm. All data to be examined will be of the categorical type and therefore continuous data will not be examined at this stage. The algorithm will however leave room for adaption to include this capability. The algorithm will be tested against C4.5 for verification purposes [5].

In Weka, the implementation of a particular learning algorithm is encapsulated in a class, and it may depend on other classes for some of its functionality. J48 class builds a C4.5 decision tree. Each time the Java virtual machine executes J48, it creates an instance of this class by allocating memory for building and storing a decision tree classifier. The algorithm, the classifier it builds, and a procedure for outputting the classifier is all part of that instantiation of the J48 class.

Larger programs are usually split into more than one class. The J48 class does not actually contain any code for building a decision tree. It includes references to instances of other classes that do most of the work. When there are a number of classes as in Weka software they become difficult to comprehend and navigate [14].

3.4 Classification function Multilayer Perceptron

Multilayer Perceptron classifier is based upon backpropagation algorithm to classify instances. The network is created by an MLP algorithm. The network can also be monitored and modified during training time. The nodes in this network are all sigmoid (except for when the class is numeric in which case the output nodes become unthresholded linear units).

The backpropagation neural network is essentially a network of simple processing elements working together to produce a complex output. The backpropagation algorithm performs learning on a multilayer feed-forward neural network. It iteratively learns a set of weights for prediction of the class label of tuples. A multilayer feed-forward neural network consists of an input layer, one or more hidden layers, and an output layer. An example of a multilayer feed-forward network is shown in Fig 2 [2].

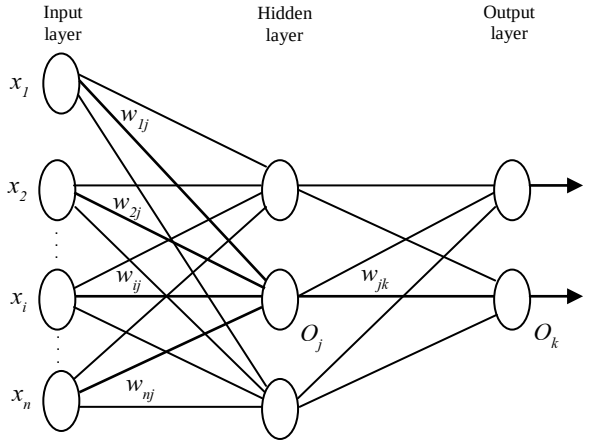

Fig 2: A multilayer feed-forward neural network

Each layer is made up of units. The inputs to the network correspond to the attributes measured for each training tuple. The inputs are fed simultaneously into the units making up the input layer. These inputs pass through the input layer and are then weighted and fed simultaneously to a second layer of "neuronlike" units, known as a hidden layer. The outputs of the hidden layer units can be input to another hidden layer, and so on. The number of hidden layers is arbitrary, although in practice, usually only one is used [2]. At the core, backpropagation is simply an efficient and exact method for calculating all the derivatives of a single target quantity (such as pattern classification error) with respect to a large set of input quantities (such as the parameters or weights in a classification rule) [15]. To improve the classification accuracy we should reduce the training time of neural network and reduce the number of input units of the network [16].

4. RESULTS

For evaluating a classifier quality we can use confusion matrix. Consider the algorithm J48 running on balance-scale dataset in WEKA, for this dataset we obtain three classes then we have 3x3 confusion matrix. The number of correctly classified instances is the sum of diagonals in the matrix; all others are incorrectly classified. Let TP_A be the number of true positives of class A, TP_B be the number of true positives of class B and TP_C be the number of true positives of class C. Then, TP_A refers to the positive tuples that were correctly labeled by the classifier in first row-first column i.e. 235. Similarly, TP_B refer to the positive tuples that were correctly labeled by the classifier in second row-second column i.e. 0. And, TP_C refer to the positive tuples that were correctly labeled by the classifier in third row-third column i.e. 244 shown in Table 2.

Table 2. Confusion matrix of three classes of balance-scale

		Predicted class			
		A	B	C	Total
Actual class	A	235	10	43	288
	B	32	0	17	49
	C	32	12	244	288
	Total				625

Accuracy = $(TP_A+TP_B + TP_C)/$(Total number of classification)

i.e. **Accuracy = (235+0+244)/625 = 76.64**

The confusion matrix helps us to find the various evaluation measures like Accuracy, Recall, Precision etc.

Table 3. Accuracy on balance-scale

S.N.	Parameters	J48	MLP
1	TP Rate	0.77	0.91
2	FP Rate	0.17	0.04
3	Precision	0.73	0.92
4	Recall	0.77	0.91
5	F-Measure	0.75	0.91
6	ROC Area	0.81	0.98

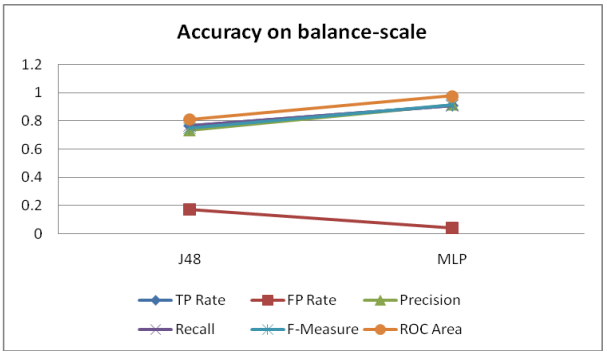

Fig 3: Accuracy chart on balance-scale

In balance-scale dataset accuracy parameters have shown in Table 3 and Fig 3. Algorithm J48 having lower value than MLP. So MLP is better method for balance-scale dataset.

Table 4. Accuracy on diabetes

S.N.	Parameters	J48	MLP
1	TP Rate	0.74	0.75
2	FP Rate	0.33	0.31
3	Precision	0.74	0.75
4	Recall	0.74	0.75
5	F-Measure	0.74	0.75
6	ROC Area	0.75	0.79

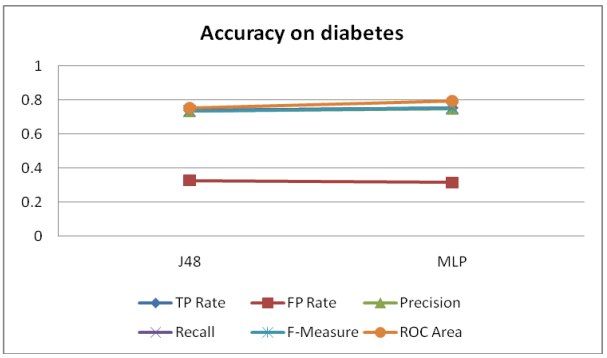

Fig 4: Accuracy chart on diabetes

In diabetes dataset the accuracy parameters have shown in Table 4 and Fig 4. The above chart shows that it have almost equal accuracy measures except ROC Area measure in which MLP has higher accuracy on the diabetes dataset. So, MLP is better method for diabetes.

Table 5. Accuracy on glass

S.N.	Parameters	J48	MLP
1	TP Rate	0.67	0.68
2	FP Rate	0.13	0.14
3	Precision	0.67	0.67
4	Recall	0.67	0.68
5	F-Measure	0.67	0.66
6	ROC Area	0.81	0.85

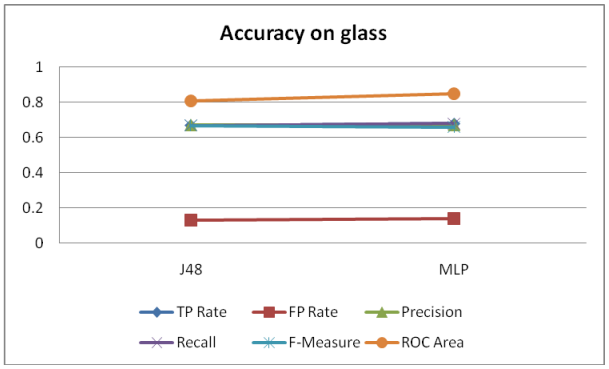

Fig 5: Accuracy chart on glass

In glass dataset accuracy parameters have shown in Table 5 and Fig 5. The above chart shows that it have almost equal accuracy measures except ROC Area measure in which MLP has higher accuracy on the glass dataset. So, MLP is better method for glass dataset.

Table 6. Accuracy on lymphography

S.N.	Parameters	J48	MLP
1	TP Rate	0.77	0.85
2	FP Rate	0.19	0.16
3	Precision	0.78	0.84
4	Recall	0.77	0.85
5	F-Measure	0.77	0.83
6	ROC Area	0.79	0.92

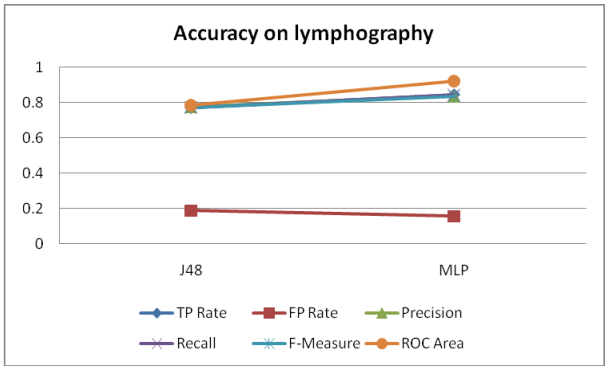

Fig 6: Accuracy chart on lymphography

In lymphography dataset accuracy parameters have shown in Table 6 and Fig 6. MLP has better accuracy measures except FP rate. So, MLP is better method for lymphography dataset.

Table 7. Accuracy on vehicle

S.N.	Parameters	J48	MLP
1	TP Rate	0.73	0.82
2	FP Rate	0.09	0.06
3	Precision	0.72	0.81
4	Recall	0.73	0.82
5	F-Measure	0.72	0.82
6	ROC Area	0.86	0.95

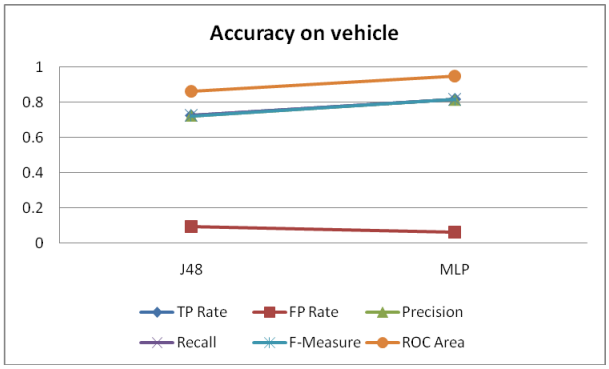

Fig 7: Accuracy chart on vehicle

In vehicle dataset accuracy parameters have shown in Table 7 and Fig 7. Algorithm MLP has better accuracy measure except FP rate. So MLP is better method for vehicle dataset.

Table 8. Accuracy measure of J48 and MLP

S.N.	Datasets	J48	MLP
1	balance-scale	76.64	90.72
2	diabetes	73.828	75.391
3	glass	66.822	67.757
4	lymphography	77.027	84.46
5	vehicle	72.459	81.679

From the values of Table 8 and the chart shown in Fig 8, the accuracy measures is calculated on J48 and MLP algorithms.

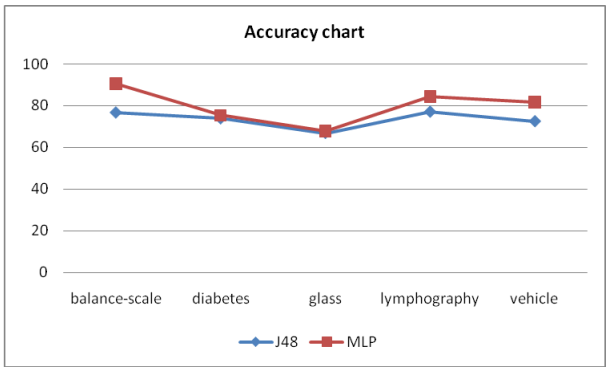

Fig 8: Accuracy chart of J48 and MLP

The J48 and MLP classification algorithm applies on all the datasets for accuracy measure. From the above chart in Fig 8 it is clear that MLP gives better results for almost 4 datasets and approximate equal accuracy for glass dataset. Hence we can clearly say that MLP is better algorithm than J48 for the given 5 datasets.

5. CONCLUSION

In this paper, we evaluate the performance in terms of classification accuracy of J48 and Multilayer Perceptron algorithms using various accuracy measures like TP rate, FP rate, Precision, Recall, F-measure and ROC Area. Accuracy has been measured on each datasets. On balance-scale, lymphography and vehicle datasets Multilayer Perceptron is clearly better algorithm. On diabetes and glass datasets accuracy is almost equal and Multilayer Perceptron is slightly better algorithm. Thus we found that Multilayer Perceptron is better algorithm in most of the cases. Generally neural networks have not been suited for data mining but from the above results we conclude that algorithm based on neural network has better learning capability hence suited for classification problems if learned properly.

6. FUTURE SCOPE

For the future work more algorithms from classification can be incorporated and much more datasets should be taken or try to get the real dataset from the industry to have the actual impact of the performance of algorithms taken into consideration. Moreover, in Multilayer Perceptron algorithm speed of learning with respect to number of attributes and the number of instances can be taken into consideration for the performance.

7. REFERENCES

[1] Z. Haiyang, "A Short Introduction to Data Mining and Its Applications", IEEE, 2011

[2] J. Han and M. Kamber, "Data Mining: Concepts and Techniques", Morgan Kaufmann, 2nd , 2006

[3] R. Agrawal, T. Imielinski, and A.N. Swami, "Database Mining: A Performance Perspective," IEEE Trans. Knowledge and Data Engineering, vol. 5, no. 6, pp. 914-925, Dec. 1993.

[4] J.R. Quinlan, "Induction of Decision Trees," Machine Learning, vol. 1, no. 1, pp. 81-106, 1986.

[5] J.R. Quinlan, C4.5: Programs for Machine Learning. Morgan Kaufmann, 1993.

[6] Y. Bengio, J. M. Buhmann, M. Embrechts, and J. M. Zurada, "Introduction to the special issue on neural networks for data mining and knowledge discovery," IEEE Trans. Neural Networks, vol. 11, pp. 545-549, 2000.

[7] D. Michie, D.J. Spiegelhalter, and C.C. Taylor, "Machine Learning, Neural and Statistical Classification", Ellis Horwood Series in Artificial Intelligence, 1994.

[8] J.R. Quinlan, "Comparing Connectionist and Symbolic Learning Methods," S.J. Hanson, G.A. Drastall, and R.L. Rivest, eds., Computational Learning Theory and Natural Learning Systems, vol. 1, pp. 445-456. A Bradford Book, MIT Press, 1994.

[9] J.W. Shavlik, R.J. Mooney, and G.G. Towell, "Symbolic and Neural Learning Algorithms: An Experimental Comparison," Machine Learning, vol. 6, no. 2, pp. 111-143, 1991.

[10] P. Clark and T. Niblett, "The CN2 induction algorithm. Machine learning", 3(4):261-283, 1989.

[11] Y. Freund and L. Mason. The alternating decision tree algorithm. In Proceedings of the 16th International Conference on Machine Learning, pages 124-133, 1999.

[12] UCI Machine Learning Repository: http://archive.ics.uci.edu/ml/datasets.html

[13] Weka: http://www.cs.waikato.ac.nz/~ml/weka/

[14] I. H. Witten, E. Frank, and M. A. Hall, Data Mining: Practical Machine Learning Tools and Techniques, 3rd ed. Morgan Kaufmann, 2011

[15] P. J. Werbos, "Backpropagation Through Time: What It Does and How to Do It", IEEE, 1990

[16] H. Lu, R. Setiono, and H. Liu, "Effective Data Mining Using Neural Networks", IEEE, 1996