ISSN:0975-9646

A Critical Study of Selected Classification Algorithms for Liver Disease Diagnosis

Shapla Rani Ghosh[1] , Sajjad Waheed (PhD)[2]

[1]*MSc student (ICT),* [2]*Associate Professor (ICT)*
[1,2] *Department of Information and Communication Technology*
[1,2] *Mawlana Bhashani Science & Technology University, Bangladesh*

Abstract: **Patients with liver disease have been continuously increasing because of excessive consumption of alcohol, inhalation of harmful gases, intake of contaminated food, pickles and drugs. Automatic classification tools may reduce burden on doctors. This paper evaluates the selected classification algorithms for the classification of some liver patient datasets. Classification algorithms considered here are Naive Bayes classification (NBC), Bagging algorithm, Dagging algorithm, KStar algorithm, Logistic algorithm. These algorithms are evaluated based on four criteria: Accuracy, Precision, Sensitivity and Specificity. It was found that, KStar algorithm is best, because of high accuracy and low error. On the other hand, Naive Bayes had the minimum accuracy and maximum error.**

Keywords: **Classification Algorithms, Liver diagnosis.**

I. INTRODUCTION

Classification techniques are very popular in various automatic medical diagnoses tools. Problems with liver patients are not easily discovered in an early stage as it will be functioning normally even when it is partially damaged [1]. An early diagnosis of liver problems will increase patients survival rate. Liver disease can be diagnosed by analyzing the levels of enzymes in the blood [2]. Moreover, now a day's mobile devices are extensively used for monitoring humans' body conditions. Here also, automatic classification algorithms are needed. With the help of Automatic classification tools for liver diseases (probably mobile enabled or web enabled), one can reduce the patient queue at the liver experts such as endocrinologists.

II. WEKA

Waikato Environment for Knowledge Analysis (WEKA). The WEKA classifier package has its own version of C4.5 known as J48. The Weka J48 classifier chooses an attribute that best differentiates the output attribute values and creates a separate tree branch for each value of the attribute. WEKA is a collection of machine learning algorithms for Data Mining tasks. It contains tools for data preprocessing, classification, regression, clustering, association rules, and visualization [3]. For our purpose the classification tools were used. There was no preprocessing of the data.

A. Data sets

The data sets used for the tests come from the UCI Machine Learning repository [4]. We are dealing with classification tasks, thus we have selected sets of which the class values are nominal [5], [6]. Selection of the sets further depended on their size, larger data sets generally means higher confidence. Here, we choose six algorithms namely, Bagging algorithm, logistic model trees algorithm, REP tree algorithm, NaiveBayes algorithm, Dagging algorithm, KStar algorithm are used for comparison. A comparison is based on sensitivity, specificity and accuracy by true positive and false positive in confusion matrix.

B. Naive Bayes Algorithm:

Bayesian Classifiers are statistical classifiers based on bayes theorem. Bayesian classification is very simple. It works on one assumption that is the effect of an attribute value on a given class is independent of the values of the other attributes. This assumption is called class conditional independence [7]. Bayesian classification can predict class membership probabilities, such as probability that a given tuple belongs to a particular class [8]. The Naïve Bayesian classification predicts that the tuple *X* belongs to the class *Ci*. Using the formula-

$$P(C_i/X) = \frac{P(X/Ci)P(Ci)}{P(X)} \quad .$$

Where, $P(C_i/X)$ is maximum posteriori hypothesis for the class C_i. As $P(X)$ is constant for all classes, only $P(X/C_i)P(C_i)$ needed to be maximized.

If the class prior probabilities are not known, then it is commonly assumed that the classes are equally likely, that is-

$$P(C_1) = P(C_2) = \ldots\ldots = P(C_m).$$
$$P(C_i/X) = P(X_j/C_i).$$

C. KStar Algorithm

K* is an instance-based classifier, that is the class of a test instance is based upon the class of those training instances similar to it, as determined by some similarity function. It differs from other instance-based learners in that it uses an entropy-based distance function.

D. Lazy Classifier

Lazy learners store the training instances and do no real work until classification time. Lazy learning is a learning method in which generalization beyond the training data is delayed until a query is made to the system where the system tries to generalize the training data before receiving

queries. The main advantage gained in employing a lazy learning method is that the target function will be approximated locally such as in the k-nearest neighbour algorithm. Because the objective function is approximated locally for each query to the system, lazy learning systems can concurrently solve multiple problems and deal successfully with changes in the problem arena. [9][10] The disadvantages with lazy learning include the large space requirement to store the complete training dataset. Mostly noisy training data increases the case support unnecessarily, because no concept is made during the training phase and another disadvantage is that lazy learning methods are usually slower to evaluate.

E. Bagging

Bagging" stands for "bootstrap aggregating".

Let the original training data be L

- Repeat B times:
  - –Get a bootstrap sample L_k from L.
  - –Train a predictor using L_k.
- Combine B predictors by
  - –Voting (for classification problem)
  - –Averaging (for estimation problem)

F. Logistic Model tree regression

Logistic regression, also called a logit model, is used to model dichotomous outcome variables. In the logit model the log odds of the outcome is modeled as a linear combination of the predictor variables. Since the dependent variable of our problem is dichotomous, a logistic regression model can be built to predict the antigenic variety.

III. PROPOSED

A. Classification via Kstar Algorithm:

The example of KStar algorithm is applied on ILPD and the confusion matrix is generated for class having three possible values are shown in Fig 1.

```
=== Confusion Matrix ===
   a   b   <-- classified as
 441   0 |   a = Male
   0 141 |   b = Female
```

Fig 1. The confusion matrix of KStar algorithm

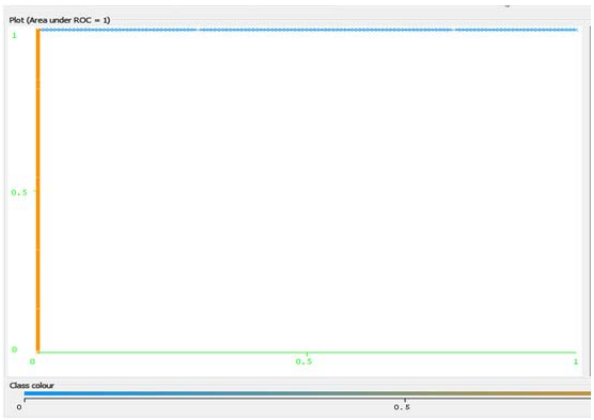


Fig 2. ROC plot for testing data set for Kstar Algorithm.

B. Classification via Bagging algorithm

The example of Bagging algorithm is applied on ILPD and the confusion matrix is generated for class having three possible values are shown in Fig 3.

```
=== Confusion Matrix ===
   a   b   <-- classified as
 435   6 |   a = Male
  64  77 |   b = Female
```

Fig 3. The confusion matrix of Bagging algorithm

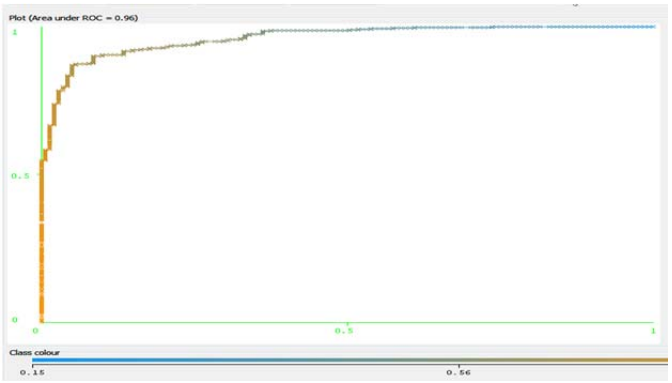


Fig 4. ROC plot for testing data set for Bagging algorithm.

C. Classification via REP Tree algorithm

The example of REP Tree algorithm is applied on ILPD and the confusion matrix is generated for class having three possible values are shown in Fig 5.

```
=== Confusion Matrix ===
   a   b   <-- classified as
 426  15 |   a = Male
 107  34 |   b = Female
```

Fig 5. The confusion matrix of REPTree algorithm

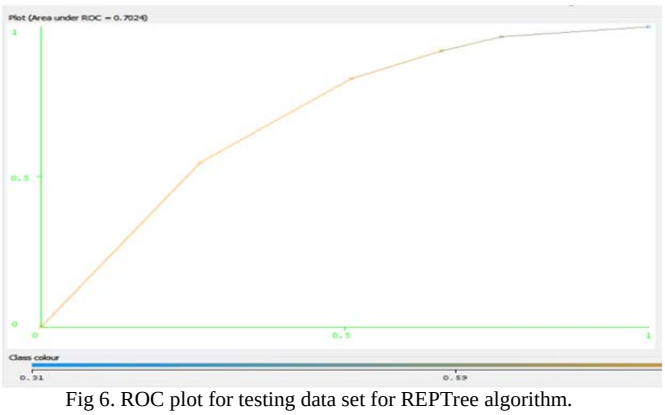


Fig 6. ROC plot for testing data set for REPTree algorithm.

D. Logistic Model tree Algorithm

The example of Logistic Model Tree algorithm is applied on ILPD and the confusion matrix is generated for class having three possible values are shown in Fig 7.

```
=== Confusion Matrix ===
   a   b   <-- classified as
 438   3 |   a = Male
 140   1 |   b = Female
```

Fig 7. The confusion matrix of Logistic model tree algorithm

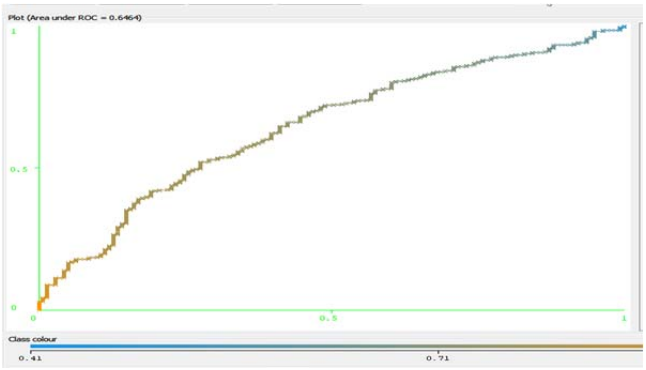


Fig 8. ROC plot for testing data set of Logictic algorithm.

E. Classification via Naive Bayes Algorithm:

The example of Naïve Bayes algorithm is applied on ILPD and the confusion matrix is generated for class having three possible values are shown in Fig9.

```
=== Confusion Matrix ===
   a   b   <-- classified as
 105 336 |   a = Male
  19 122 |   b = Female
```

Fig 9. The confusion matrix of Naïve Bayes algorithm.

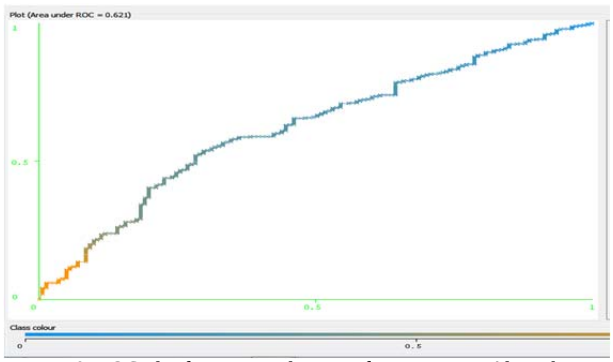


Fig10. ROC plot for Testing data set of Naive Bayes Algorithm

F. Classification via Dagging Algorithm

The example of Dagging algorithm is applied on ILPD and the confusion matrix is generated for class having three possible values are shown in Fig 11.

```
=== Confusion Matrix ===
   a   b   <-- classified as
 441   0 |   a = Male
 141   0 |   b = Female
```

Fig 11. The confusion matrix of Dagging algorithm.

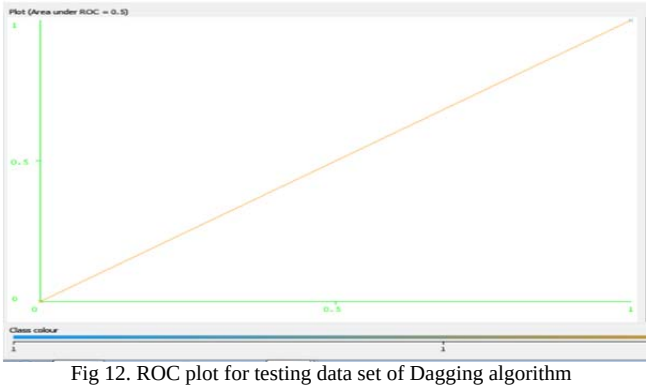


Fig 12. ROC plot for testing data set of Dagging algorithm

IV Results Analysis

We have performed classification using Bagging algorithm, logistic model trees algorithm, REP tree algorithm, KStar algorithm and NaïveBayes algorithm. The experimental results under the framework of WEKA (Version 3.6.10). The experimental results are partitioned into several sub item for easier analysis and evaluation. One the first part, sensitivity (SE), precision, accuracy (AC) and specificity(SP) will be partitioned in first table while the second part, we also show the relative mean absolute error (RMAE), root mean squared error (RMSE) and Mean absolute error (MAE).

A. Accuracy:

The accuracy of a classifier is the percentage of the test set tuples that are correctly classified by the classifier-

$$\text{Accuracy} = \frac{\text{no. of true positives} + \text{no. of true negatives}}{\text{no. of true positives} + \text{false positives} + \text{false negatives} + \text{true negative}}$$

B. Sensitivity:

Sensitivity is also referred as True positive rate i.e. the proportion of positive tuples that are correctly identified.

$$\text{Sensitivity} = \frac{\text{number of true positives}}{\text{number of true positives} + \text{number of false negatives}}$$

C. Precision:

Precision is defined as the proportion of the true positives against all the positive results (both true positives and false positives)

$$\text{Precision} = \frac{\text{number of true positives}}{\text{number of true positives} + \text{false positives}}$$

All measures can be calculated based on four values, namely True Positive, False Positive, False Negative, and False Positive [9]. These values are described below.

- True Positive (TP) is a number of correctly classified that an instances positive.
- False Positive (FP) is a number of incorrectly classified that an instance is positive.
- False Negative (FN) is a number of incorrectly classified that an instance is negative.
- True Negative (TN) is a number of correctly classified that an instance is negative.

D. Mean absolute error:

Mean absolute error, MAE, is the average of the difference between predicted and actual value in all test cases; it is the average prediction error, the formula for calculating MAE is given in equation shown below-

$$\frac{|a_1 - c_1| + |a_2 - c_2| + \ldots + |a_n - c_n|}{n}$$

Assuming that the actual output is a, expected output is c.

E. Root Mean-Squared Error:

RMSE is frequently used measure of differences between values predicted by a model or estimator and the values actually observed from the thing being modeled or

estimated. It is just the square root of the mean square error as shown in equation given below-

$$\sqrt{\frac{(a_1-c_1)^2+(a_2-c_2)^2+\ldots+(a_n-c_n)^2}{n}}$$

Assuming that the actual output is a, expected output is c.

The mean-squared error is one of the most commonly used measures of success for numeric prediction. This value is computed by taking the average of the squared differences between each computed value and its corresponding correct value. The root mean-squared error is simply the square root of the mean-squared-error. The root mean-squared error gives the error value the same dimensionality as the actual and predicted values.

Table I
Performance of classification result for using training set of ILPD

Algorithm name	Accuracy	Precision	Sensitivity	Specificity
Bagging	0.88	0.885	0.997	0.557
Dagging	0.758	0.574	0.948	0.35
KStar	1	1	1	1
NaiveBayes	0.39	0.706	0.788	0.099
Logistic	0.754	0.635	0.947	0.345
REPTree	0.79	0.774	0.956	0.393

Table II
Performance of error result for using training set of ILPD

Algorithm name	MAE	RMSE	RAE(%)
Bagging	0.238	0.303	64.837
Dagging	0.242	0.4922	65.9047
KStar	0	0.0002	0.0105
NaiveBayes	0.514	0.5813	139.716
Logistic	0.353	0.4199	95.9099
REPTree	0.316	0.3976	86.0198

TABLE III
Performance of classification result for using cross-validation set of ILPD

Algorithm name	Accuracy	Precision	Sensitivity	Specificity
Bagging	0.754	0.711	0.945	0.345
Dagging	0.758	0.574	0.948	0.35
KStar	0.706	0.705	0.933	0.292
NaiveBayes	0.402	0.686	0.402	0.104
Logistic	0.751	0.573	0.751	0.341
REPTree	0.741	0.677	0.943	0.329

Table IV
Performance of Error Result for using Cross-validation set of ILPD

Algorithm Name	MAE	RMSE	RAE(%)
Bagging	0.3305	0.4174	89.8944
Dagging	0.2433	0.4923	66.1755
KStar	0.3025	0.4757	82.2763
NaiveBayes	0.5219	0.5927	141.9405
Logistic	0.3603	0.4296	97.9955
REPTree	0.345	0.4358	93.8276

Table V
Performance of classification result for using percentage split set of ILPD

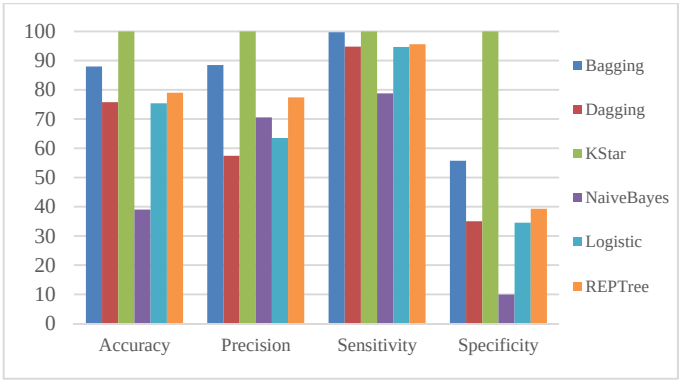
Algorithm name	Accuracy	Precision	Sensitivity	Specificity
Bagging	0.768	0.737	0.867	0.625
Dagging	0.783	0.613	0.877	0.645
KStar	0.742	0.758	0.851	0.593
NaiveBayes	0.434	0.731	0.603	0.279
Logistic	0.434	0.731	0.603	0.279
REPTree	0.707	0.702	0.827	0.549

Table VI
Performance of Error result for using percentage split set of ILPD

Algorithm name	MAE	RMSE	RAE(%)
Bagging	0.3438	0.4093	94.9077
Dagging	0.2323	0.4519	64.1335
KStar	0.2959	0.4604	81.692
NaiveBayes	0.4942	0.5423	136.421
Logistic	0.4942	0.5423	136.421
REPTree	0.3452	0.4516	95.3064

V. Conclusions

In this paper, we have compared the effectives of the classification algorithm namely, Naive Bayes classification (NBC), Bagging algorithm, Dagging algorithm, KStar algorithm, Logistic algorithm, and REP tree algorithm. Popular algorithms were considered for evaluating their classification performance in classifying Liver patient data set. Based on the above classifier and experimental results, we can clearly see that highest accuracy belong to the KStar algorithm. We observed that, KStar algorithm is best, because the accuracy rates are very high and error rates are low. The graphical representations of all the classification results is shown in figure 13.



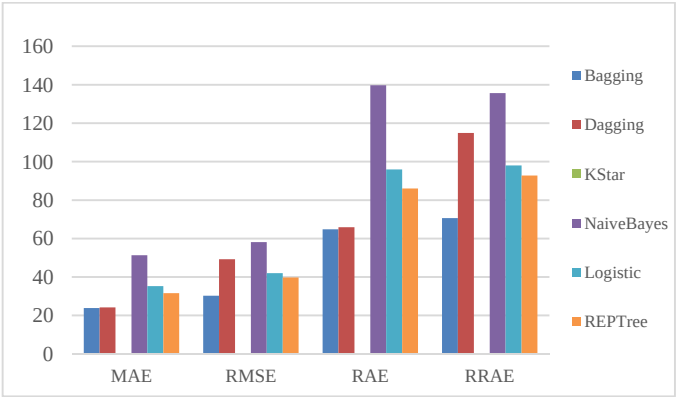


Fig 13. Classification result of all algorithms for training set

Acknowledgment

The authors would like express their sincere gratitude to Deepa Chakkamadathil Ramachandranthe for financing to publish this article and express heart felt thanks to Sushan Chowhan for manuscript preparation. The author is deeply indebted to all the staffs and workers of ICT department, MBSTU for their helpful cooperation to complete the research work.

REFERENCES

[1] R.H. Lin. “An intelligent model for liver disease diagnosis,” *Artificial Intelligence in Medicine.* vol.47, pp. 53-62, June 2009.

[2] E.R. Schiff, M.F. Sorell, W.C. Maddrey, *Diseases of the liver*, 10th ed., Philadelphia, PA19106, USA: Lippincott Williams and Wilkins, 2007.

[3] C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, 1995.

[4] C.L. Blake, D.J. Newman, S. Hettich and C.J. Merz. (2012) UCI machine learning repository databases. [Online]. Available: http://mlr.cs.umass.edu/ml/machine-learning-databases/00225/

[5] P. Ozer, "Data Mining Algorithms for Classification,"BSc Thesis, Radboud University Nijmegen, Netherland, Jan. 2008.

[6] Ian H. Witten and Eibe Frank. *Data Mining: Practical machine learning tools and techniques with java implementations*, Sanfrancisco, California: Morgan Kaufmann Publishers, ISBN 1-55860-552-5, 2000.

[7] B.V. Ramana1, M.S. P. Babu, N. B. Venkateswarlu. "A Critical Study of Selected Classification Algorithms for Liver Disease Diagnosis."International Journal of Database Management Systems, Vol.3. pp. 101-114, May 2011.

[8] P. Domingos and M. Pazzani, “On the optimality of the simple Bayesian classifier under zero-one loss,” *Machine Learning*, vol. 29, pp. 103-130. November 1997.

[9] G. Williams, R. Baxter, H. He, S. Hawkins and L. Gu, “A Comparative Study for RNN for Outlier Detection in Data Mining,” in Proceedings of the 2nd IEEE International Conference on Data Mining, December 2002, Maebashi city, Japan, p. 709.

[10] A.S. Achanta, J.G. Kowalski, C.T Rhodes. “Artificial neural net-works: implications for pharmaceutical sciences,” *Drug Dev. Ind. Pharm.*; vol. 21, pp. 119-155. January 1995.