Efficient learning of Arrhythmia data set with Multi class-cost sensitive classifiers

R.Karthikeyan[1], A.Kumaravel [2], V.Khanaa [3]

[1] Assistant Professor, Department of Computer Science and Engineering, Bharath University, Chennai, India,
[2] Professor and Dean, Department of Computer Science and Engineering, Bharath University, Chennai, India
[3] Professor, and Dean Department of Information Technology, Bharath University, Chennai, India

---***---

Abstract - *The cost models for measuring the loss due to wrong predictions are of interest and vary very much application to application. In this paper we illustrate a frame work for a dynamic cost model in the context of cancer data extensively. This has been compared to results obtained in cancer data set to study the cost effectiveness in different contexts.*

Key Words: *Heart disease data sets decision trees, decision rules, Meta classifiers, Bayes classifiers.*

1. INTRODUCTION:

Performance measure in the context like diagnosis of heart disease requires more attention by the data analytic community. Summative measures vary sometimes drastically due to variations is the distribution of instances in the underlying dataset. The typical example of the problem of cost-sensitive classification is medical diagnosis, where a doctor would like to balance the costs of various possible medical tests with the expected benefits of the tests for the patient. The words "cost", "expense", and "benefit" are used in this paper in the broadest sense, to include factors such as quality of life, in addition to economic or monetary cost. Cost is domain-specific and is quantified in arbitrary units. It is assumed here that the costs of tests are measured in the same units as the benefits of correct classification. The arrhythmia data set is reduced in such a way that 279 number of attributes made into final set of 25 attributes. The aim of this paper is to establish the relevance of the cost sensitiveness.

1.1 Cost-Sensitive Learning

The performance of a classifier for a two-class problem can be described by the confusion matrix described in Figure 1. Holding with the established practice, the minority class is designated the positive class and the majority class is designated the negative class.

	ACTUAL		
PREDICTED		Positive class	Negative class
	Positive class	True positive(TP)	False positive(FP)
	Negative class	False negative(FN)	True negative(TN)

Fig 1: A Confusion Matrix

Corresponding to a confusion matrix is a cost matrix. The cost matrix will provide the costs associated with the four outcomes shown in the confusion matrix, which we refer to as CTP, CFP, CFN, and CTN. As is often the case in cost-sensitive learning, we assign no costs to correct classifications, so CTP and CTN are set to 0. Since the positive (minority) class is often more interesting than the negative (majority) class, typically CFN > CFP (note that a false negative means that a positive example was misclassified). As discussed earlier, cost-sensitive learning can be implemented in a variety of ways, by using the cost information in the classifier-building process or by using a wrapper based method such as sampling. When misclassification costs are known the best metric for evaluating classifier performance is total cost. Total cost is the only evaluation metric used in this paper and is used to evaluate all three cost-sensitive learning methods. The formula for total cost is shown in equation below.

Total Cost = (FN × CFN) + (FP × CFP)

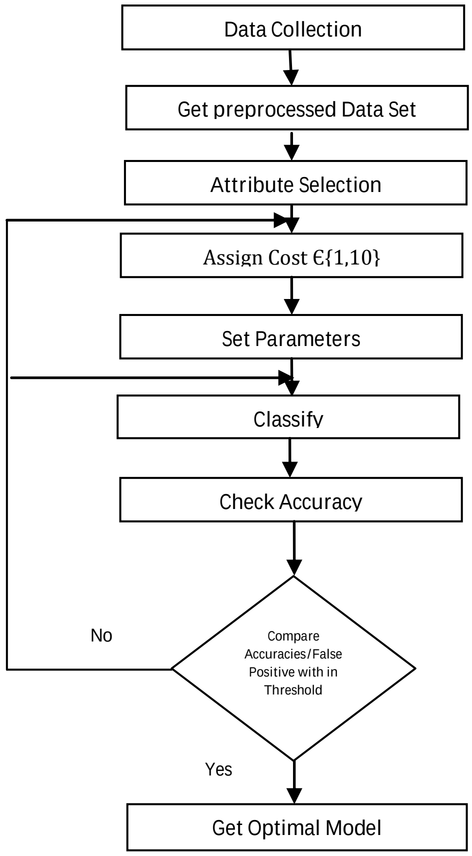

Fig.2 Iterative steps in Data Mining

2. DATASET COLLECTION AND DATA PREPARATION

In this section, we dwell the collection of data and format in which the data has to be presented for mining experiments following the iterative steps in Figure 2. We use java based implementation namely Weka tool from University of Waikato, New Zealand.
The datasets for these experiments are from [15].The original data format has been slightly modified and extended in order to get relational format.

2.1 Dataset description:

The database of diabetes describes a set of 279 attributes11 as shown in the below list 2.2. This dataset carries 16 categories of classes. The number of instances in this database is 452.

Class code	Class	Number of instances
01	Normal	245
02	Ischemic changes	15
03	Old Anterior Myocardial	15
04	Old Inferior Myocardial	15
05	Sinus tachycardy	13
06	Sinus bradycardy	25
07	Ventricular Premature (PVC)	3
08	Supraventricular Premature	07
09	Left bundle branch block	09
10	Right bundle branch block	50
11	1 degree AtrioVentricular block	0
12	2. degree AV block	0
13	degree AV block	0
14	Left ventricule hypertrophy	4
15	Atrial Fibrillation or Flutter	5
16	Others	22

3. METHODS DESCRIPTION

Here we select standard set of methods [13] for predicting from the data set described above. We consider three types of classifiers for our study such as tree based, Bayes approach based, and Meta level based classifiers. The following sections describe briefly the methods for classifier and results of such methods are tabulated further. Then final results are interpreted
Cost-sensitive meta-learning converts existing cost-insensitive classifiers into cost-sensitive ones without modifying them. Thus, it can be regarded as a middleware component that pre-processes the training data, or post-processes the output, from the cost-insensitive learning algorithms.
Cost-sensitive meta-learning can be based on thresholding, where

Threshold = FP/(FP+FN)

3.1 TREE CLASSIFIERS:

Supervised Learning is performed conducted using tree classifiers .We select four types of tree classifiers as shown below.

3.1.1 J48

The first number is the total number of instances (weight of instances) reaching the leaf. The second number is the number (weight) of those instances that are misclassified.
If your data has missing attribute values then you will end up with fractional instances at the leafs. When splitting on an attribute where some of the training instances have missing values, J48 will divide a training instance with a missing value for the split attribute up into fractional parts proportional to the frequencies of the observed non-missing values. This is discussed in the Witten & Frank Data Mining book as well as Ross Quinlan's original publications on C4.5.

3.1.3 Random Forest

Random forests are a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest. The generalization error for forests converges a.s. to a limit as the number of trees in the forest becomes large.

3.1.3 Random Tree

Class for constructing a tree that considers K randomly chosen attributes at each node. Performs no pruning. Also has an option to allow estimation of class probabilities based on a hold-out set back fitting.

3.1.4 LMT

Classifier for building 'logistic model trees', which are classification trees with logistic regression functions at the leaves. The algorithm can deal with binary and multi-class target variables, numeric and nominal attributes and missing values.

3.2 BAYES CLASSIFIERS

These types of classifiers includes probability measure for the class values and comes under supervised learning.

3.2.1 Bayes Net

Bayes Network learning using various search algorithms and quality measures. Base class for a Bayes Network classifier. Provides data structures and facilities common to Bayes Network learning algorithms like K2 and B.

3.2.2 Naïve Bayes

Class for a Naive Bayes classifier using estimator classes. Numeric estimator precision values are chosen based on analysis of the training data. For this reason, the classifier is not an Updateable Classifier you need the Updateable Classifier functionality, use the Naïve Bayes Updateable classifier.

3.2.3 Naive BayesUpdateable

Class for a Naive Bayes classifier using estimator classes. This is the updateable version of Naïve Bayes. This classifier will use a default precision of 0.1 for numeric attributes when build Classifier is called with zero training instances.

3.3 META CLASSIFIERS

Most of the time, the aggregation of more than one classifier has better performance. Such combinational methods are shown below.

3.3.1 Adaboost

Class for boosting a nominal class classifier using the Adaboost M1 method. Only nominal class problems can be tackled. Often dramatically improves performance, but sometimes over fits.

3.3.2 Bagging

Class for bagging a classifier to reduce variance. Can do classification and regression depending on the base learner. Generate B bootstrap samples of the training data: random sampling with replacement. Train a classifier or a regression function using each bootstrap sample for classification: majority vote on the classification results. For regression: average on the predicted values.

3.3.3 Dagging

This meta classifier creates a number of disjoint, stratified folds out of the data and feeds each chunk of data to a copy of the supplied base classifier. Predictions are made via averaging, since all the generated base classifiers are put into the Vote meta classifier. Useful for base classifiers that are quadratic or worse in time behavior, regarding number of instances in the training data.

3.3.4 Multi Boost

MultiBoosting is an extension to the highly successful AdaBoost technique for forming decision committees. It is able to harness both AdaBoost's high bias and variance reduction with wagging's superior variance reduction. Using C4.5 as the base learning algorithm, Multi-boosting is demonstrated to produce decision committees with lower error than either AdaBoost or wagging significantly more often than the reverse over a large representative cross-section of UCI data sets. It offers the further advantage over AdaBoost of suiting parallel execution.

3.4 RULES CLASSIFIERS

3.4.1 J-Rip

This class implements a propositional rule learner, Repeated Incremental Pruning to Produce Error Reduction (RIPPER), which was proposed by William W. Cohen as an optimized version of IREP.

3.4.2 One-R

Class for building and using a 1R classifier; in other words, uses the minimum-error attribute for prediction, discretizing numeric attributes.

3.4.3 PART

Class for generating a PART decision list. Uses separate-and-conquer. Builds a partial C4.5 decision tree in each iteration and makes the "best" leaf into a rule.

3.4.4 Zero-R

Class for building and using a 0-R classifier. Predicts the mean for a numeric classor the mode for a nominal class.

4.1 PERFORMANCE MEASURE:

Accuracy: probably the most widely used performance metric in Machine Learning. It is defined as the proportion of correct predictions the classifier makes relative to the size of the dataset.

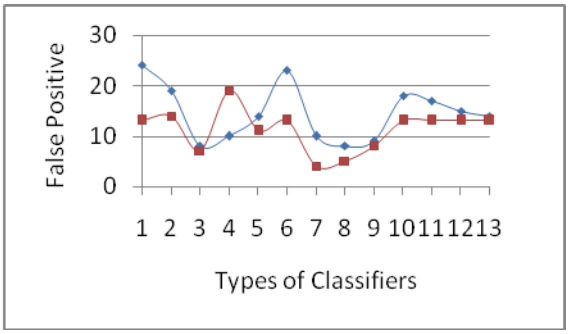

Chart -1: Variation of False Positive Vs Types of Classifiers

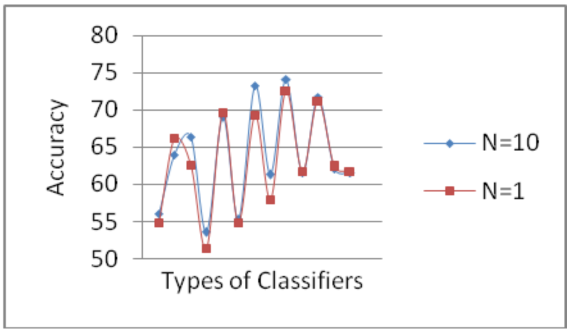

Chart -2:Variation of False Positive Vs Types of Accuracy

5.1 RESULTS:

We get two components in the main result for the above experiment following the framework as prescribed in the figure 2, namely variations in the classification accuracies and the variations in the false positives. Even though the accuracy variation is similar in both cases the reduction in false positive is clearly visible.

5.2 CONCLUSION:

The same set of experiments can be performed by varying the cost with more granularity and dynamics of classifiers accuracy can be studied.

ACKNOWLEDGEMENT

The authors would like to thank the management of Bharath University for the support and encouragement for this research work.

REFERENCES

[1]Peter D. Turney, Cost-Sensitive Classification: Empirical Evaluation of a Hybrid Genetic Decision Tree Induction Algorithm, Journal of Artificial Intelligence Research, 369-409, (1995)

[2] Optimizing F-Measures by Cost-Sensitive Classification, Shameem A. Puthiya Parambath, Nicolas Usunier, Yves Grandvalet

[3] Hanley JA, McNeil BJ. The meaning and use of the area under the Receiver Operating Characteristic (ROC) curve. Radiology 1982 143 29-36.

[4] Metz CE. Semin, Basic principles of ROC analysis, Nuclear Med, 1978, VIII(4) 283-298.

[5] Companion slides for the text by Dr..H.Dunham, Data Mining, Introductory and Advanced Topics, Prentice Hall, 2002.

[6] Uci reprository https://archive.ics.uci.edu/ml/datasets.html downloaded on 13th Aug 2014.

[7] A.Kumaravel, Pradeepa.R, Efficientmolecule reduction for drug design by intelligent search methods Int J Pharm Bio Sci 2013 Apr; 4(2): (B) 1023 – 1029

[8] Toward optimal feature selection using ranking methods and classification Algorithms Jasmina Novaković, Perica Strbac, Dusan Bulatović Yugoslav Journal of Operations Research 2011, Number 1, 119-135

[9] A.Gelman, Y. S. Su, M.Yajima, J. Hill, M.Pittau, J. Kerman, and T. Zheng, "arm:Data Analysis Using Regression and Multilevel/Hierarchical Models," R package version 1.5-02.://CRAN.Rproject.org/package=arm,2012.

[10] A.Kumaravel and D.Udhayakumarapandian, A Novel Subset Selection For Classification Of Diabetes Dataset By Iterative Methods Int J Pharm Bio Sci 2014 July ; 5 (3) : (B) 1 - 8

[11] https://archive.ics.uci.edu/ml/datasets/Arrhythmia (Accessed on 3rd july 2014)

[12] Dietterich, T. G., Jain, A., Lathrop, R.,Lozano-Perez, T. (1994). A comparison of dynamic reposing and tangent distance for drug activity prediction.Advances in Neural Information Processing Systems, 6. SanMateo, CA: Morgan Kaufmann. 216--223.

[13] A.Stensvand, T. Amundsen, L. Semb, D.M.Gadoury, and R.C. Seem. 1997.Ascospore release and infection of apple leaves by conidia and ascospores of Venturia inaequalis at low temperatures.Phytopathology 87:1046-1053.

[14]http://weka.sourceforge.net/doc/weka/ (Downloaded on 10th Aug 2014).

[15]Sourceboutwekahttp://www.cs.waikato.ac.nz/ml/weka/downloaded on 10 Aug 14.

BIOGRAPHIES

Mr.Karthikeyan completed his M.E Computer Science &Engineering and pursuing his Phd Research in data mining domain. He has experience in teaching for eight years and four years in IT

Dr.V.Khanaa has been working in Bharath University for thirty years and having more than hundred in National and International Journals and Conferences.

