ProICET – A Cost-Sensitive System for the Medical Domain

Rodica Potolea, Camelia Vidrighin, Cristina Savin
Technical University of Cluj-Napoca, Computer Science Department
Rodica.Potolea@cs.utcluj.ro

Abstract

In recent years, data mining has started to receive increasing interest as a method of complementing domain specific expertise in various spheres of human activity. Apart from data specific issues, a key particularity of many real world problems, such as medical diagnosis, are the costs involved, the most important being the test and the misclassification costs. This paper evaluates ProICET, a new system built around the ICET algorithm. The system has been previously benchmarked on classical medical data sets. Here, we use a real medical dataset to test the current version of our system. The comparative analysis confirms that ProICET is the best at cost minimization out of several successful classifiers, while keeping a good accuracy rate.

1. Introduction

In recent years, data driven analytical research has started to complement domain specific methods in various areas, such as loan decision, oil-slick detection, medical field, or sales and marketing. In the medical domain, the data driven approach has proved of valuable support especially in the diagnosis process. This is an area in which traditional methods can profit from the solutions provided by machine learning because, despite the boost in biomedical technology, the accuracy of diagnosis and prognosis remains in many cases rather low. The causes of this situation are multiple.

First of all, it is known that medical diagnosis is subjective; it is influenced by the physician making the diagnosis (his experience, intuition and biases, or his psycho-physiological condition). Moreover, the amount of data that should be considered in order to make an accurate prediction is usually huge. Machine learning can be used to automatically infer diagnostic rules from descriptions of past, successfully treated patients, and, consequently, help specialists make the diagnostic process more objective and more reliable. Records of previous patients are gathered into hospital archives and can be made available through machine learning techniques; the classifier derived from this data provides support for future diagnosis and treatment and can help improve the physician's speed, accuracy and reliability in establishing a new diagnosis. Also, it may offer very useful support in the training of students and for non-specialists.

Additionally to complementing the clinical diagnosis process, data mining tools offer the possibility of extracting useful information from huge amounts of data, a task impossible for any doctor. This provides the possibility to generate new medical knowledge, which can then drive further the data mining approaches.

2. Cost-Sensitive Medical Data Mining

Whether a healthy patient is diagnosed as ill or the other way around has very different implications in real life. However, a typical classification algorithm would make no difference between the two. This is because such algorithms are only concerned with error reduction, i.e. minimizing the number of errors. As the previous example suggested, in real world problems, the cost of different errors is seldom the same. Consequently, for a classifier to be of practical use it must consider a more complex form for the function it needs to minimize.

A special category of methods that address this problem are cost-sensitive learners, which are directed towards the reduction of the total cost, instead of just minimizing the number of misclassification errors.

Turney [11] provides a general taxonomy of costs involved in inductive concept learning, the most important of which being *misclassification costs* and *test costs*. The first category encompasses the costs which are conventionally considered by most cost-sensitive classifiers; however, several solutions address the second category also. A brief survey of the most important cost-sensitive classifiers, as described in the literature, will be provided in the following.

A first naive approach to reducing misclassification costs is *stratification*, which changes the distribution of instances for each class, by including proportionally more examples of the classes with high misclassification costs. Although it has the advantage of being very simple, the approach has also a serious limitation: it restricts the

dimension or the form of the misclassification cost matrix, being applicable only to two-class problems or to problems where the cost is independent of the predicted class. More complex techniques, which overcome these limitations, usually involve meta-learning algorithms, which typically are applicable to a range of base classifiers. In this category we include algorithms based on various *ensemble* methods, such as *AdaBoost.M1* [4], *AdaCost* [3], or *MetaCost* [2] and those which take an *evolutionary* approach, the best-known being *ICET* [10].

Introduced by Freund and Schapire, *AdaBoost.M1* [4] combines several weak classifiers through voting, such as to obtain a composite classifier with higher predictive accuracy than any of its components. Each distinct model is built, during several boosting steps, through the same learning mechanism, by varying the distribution of examples in the training set. After each boosting phase, the weights of the misclassified examples are increased, while those for the correctly classified examples are decreased. It has been mathematically proved that the error rate for the composite classifier on the un-weighted training examples approaches zero exponentially with an increasing number of boosting steps [4], [7]. Moreover, various experiments report that the reduction in error is maintained for unseen examples as well.

Another solution for reducing misclassification costs is *MetaCost* [2]. The algorithm is based on the Bayes optimal prediction principle, which minimizes the conditional risk of predicting that an example belongs to class *i*, given its attributes *x*. The solution requires accurate estimates for the class probabilities of examples in the training set. This distribution is obtained through an ensemble method, by uniform voting from individual classifiers. Once the conditional probabilities are estimated, the algorithm re-labels the examples in the training set, according to their optimal predictions and generates the final classifier, using the modified training set. *MetaCost* is applicable to a wide range of base classifiers. Moreover, it has the advantage of generating a single, understandable model, and it is efficient under changing costs (the conditional probabilities need to be computed only once, after which they can be used to generate models for various cost matrices).

The second main category of cost-sensitive learners is represented by those that tackle the problem of test costs. They typically involve some alteration of the information gain function, as to make it cost-sensitive. Various cost dependent functions have been proposed in the literature, such as *EG2*, *IDX* or *CS-ID3* [10].

Significantly less work has been done for aggregating several cost components. The most prominent approach in the literature is *ICET*, which combines a greedy search heuristic (decision tree) with a genetic search algorithm. Other possible solutions are explored in [5], [8] and [9].

Medical diagnosis is one field in which such an aggregated approach is of utmost importance. First of all, a doctor must always consider the potential consequences of a misdiagnosis. In this field, misclassification costs may not have a direct monetary quantification, but they represent a more general measure of the impact each particular misclassification may have on human life. These costs are non-uniform (diagnosing a sick patient as healthy carries a higher cost than diagnosing a healthy patient as sick). Another particularity of the medical diagnosis problem is that medical tests are usually costly. Moreover, collecting test results may be time-consuming; arguably time may not be a 'real' cost, but it does have some implication for the decision whether it is practical to take a certain test or not. In the real case, performing all possible tests in advance is unfeasible and only a relevant subset should be selected. The decision on performing or not a certain test should be based on the relation between its cost and potential benefits. When the cost of a test exceeds the penalty for a misclassification, further testing is no longer economically justified.

3. ProICET

One of the most prominent approaches to the classification problem is the decision tree learner. The classical algorithm for generating decision trees uses a greedy technique. As all hill climbing algorithms, it suffers from the horizon effect. A better solution would be to perform a heuristic search in the space of possible decision trees, through evolutionary means.

ICET (Inexpensive Classification with Expensive Tests), introduced by Peter Turney, is such a hybrid method. The theoretical grounds show the algorithm has potential, because it tackles the problem of cost-sensitive classification in a novel, yet sound manner: by combining a greedy search heuristic (decision tree) with a genetic algorithm [10].

ProICET is a new system, which has as starting point the *ICET* algorithm. A detailed description of the system is provided in Section 4.

The algorithm is developed around the following idea: the GA evolves a population of parameters, each individual corresponding to a decision tree. Standard mutation and crossover operators are applied to the tree population and, after a fixed number of iterations, the fittest individual is returned.

The decision tree algorithm is *Eg2* – a modified version of Quinlan's C4.5, which uses ICF (Information Cost Function) as attribute selection function [6].

For the i^{th} attribute, ICF may be defined as follows:

$$ICF_i = \frac{2^{\Delta I_i}}{(C_i + 1)^w}, \quad where \quad 0 \le w \le 1 \qquad (1)$$

Here, the ICF costs are used for encoding the individuals in the population and not for minimizing test costs directly, as in *Eg2*. The n costs, C_i, are not true costs, but bias parameters. They provide enough variation to prevent the decision tree learner from getting trapped in a local optimum, by overrating/ underrating the cost of certain tests based on past trials' performance. However, true test costs may be used when generating the initial population, as it has been shown to lead to some increase in performance [10].

The individuals are represented as a bit string of n + 2 numbers, represented in Gray code. The first n numbers represent the bias parameters, 'alleged' test costs in the ICF function. The last two stand for the algorithm's parameters CF and w; the first controls the level of pruning, as defined for C4.5, while w is needed by ICF.

The fitness function for an individual is computed by evaluating the average cost of classification of the corresponding tree, built by randomly dividing the training set in two subsets, the first used for the actual tree induction and the second for error estimation. The average cost of classification is the total cost, obtained by summing the test and misclassification costs, normalized to the training set size.

Test costs are specified as attribute - cost value pairs. If the same attribute is tested twice along the path (numeric attribute), the second time its cost is 0. The classification costs are defined by a cost matrix $(C_{ij})_{nxn}$, where C_{ij} is the cost of misclassifying an instance of class j as being of class i.

4. Implementation Issues

ProICET uses as a starting point the implementation of the C4.5 algorithm, revision 8, provided by *Weka* [13] (referred to as *J4.8*), and standard genetic mechanisms supplied by *GGAT* [1], both written in *java.*

The *Eg2* algorithm was developed from *J4.8* by modifying the information gain function to consider the cost associated to each attribute, as specified by equation (1), similarly to the implementation presented in [10].

We also had to consider a new evaluation procedure, sensitive to both test and misclassification costs. This was important for both the algorithm itself, in computing the fitness score, and for the methodology of evaluating the new system against well known classifiers, such as *MetaCost*, *AdaBoost*, *Eg2*, or *J4.8*.

The most important changes from the initial *ICET* algorithm [10] affected the evolutionary component, where several enhancements have been considered. We employed GGAT [1] – *General Genetic Algorithm Tool* – as a starting point to build the improved genetic component within *ProICET*.

Instead of generating new populations at each iteration, we employed the *single population* technique for evolving a new generation, which directly implements *elitism* (the best individuals of the current generation can survive unchanged in the next generation). Another prominent feature of *ProICET* is the use of *ranking* in the fitness function estimation. The individuals in the population are ordered according to their fitness value, after which probabilities of selection are distributed evenly, according to their rank in the ordered population. Ranking can be a very effective mechanism for avoiding the premature convergence of the population, which can occur if the initial pool has some individuals which dominate, having a significantly better fitness than the others.

For each individual, the *n+2* chromosomes were defined (*n* being the number of attributes in the data set, while the other two correspond to parameters *w* and *CF*); each chromosome is represented as a 14 bits binary string. The population size is 50 individuals. The *roulette wheel* technique was used for parent selection; as recombination techniques, we have employed *single point random mutation* with mutation rate 0.2, and *multipoint crossover*, with 4 randomly selected crossover points.

The number of the evaluation steps used in [10] is rather low. Therefore, in *ProICET*, the algorithm is run for 1000 fitness evaluation steps. Due to the fact that a new generation is evolved using single population, the final result yielded by the procedure is the best individual over the entire run, which makes the decision on when to stop the evolution less critical.

5. Evaluation on a Real Medical Problem

The work performed on the *ICET* algorithm was concerned with evaluating the hybrid approach against algorithms that are sensitive to test costs, therefore lacking both a comprehensive evaluation of the misclassification cost component, and a study of the behavior in medical problems with both types of costs involved. The work carried out in [12] tries to fill in this gap, by comparing the new system (*ProICET*) with some of the best-known classifiers – either cost sensitive, or very good at error reduction. The results obtained there illustrate that besides better costs, *ProICET* achieves very high accuracy rates (94-99%) on large medical benchmarks (Wisconsin breast cancer, Thyroid) [12], higher than those of the algorithms considered.

The results presented there have confirmed that the methodology introduced by the algorithm is promising, but did not tackle the problem of evaluating the system on real data. This is what we try to achieve here, by evaluating our new implementation on a dataset containing records of patients that have been diagnosed with prostate cancer. The dataset contains 16 attributes,

representing both preoperative and operative data. The class attribute is postoperative PSA, pre-discretized such as to obtain three possible values ('low' – for PSA < 0.1, 'medium' – for PSA between 0.1 and 1, and 'high' – for PSA > 1).

The medical question was to try and predict the value of postoperative PSA from preoperative and operative data (pre-op PSA, quality of life, nerve sparing, operation time, bleeding, operation type, technique, etc). A secondary issue was related to finding the best predictor attributes in this case.

In what the evaluation procedure is concerned, because the algorithm involves a large heuristic component, it assumes averaging the costs over 10 runs. Each run uses a pair of randomly generated training-testing sets, in the proportion 70% - 30%; the same proportion is used when separating the training set into a component used for training and one for evaluating each individual (in the fitness function).

One of the main issues when applying a cost-sensitive approach, especially in the medical field, is setting the costs. If the test costs are relatively easier to quantify (by limiting to their economical aspect), when building the misclassification cost matrices we come across a more serious matter: how can we measure the value of human life? This is still an open question, and many doctors are reluctant to set a value to different misclassification errors. Perhaps a good approach is to use several cost matrices, and compare the outcomes. Following this idea, we used two different values for the test costs - 0 and 0.1 - and four different cost matrices. The four matrices are shown in Table 1, where for each matrix, the line indices represent the predicted class, and the columns represent the actual class. The idea was to experiment on a few variants of the unbalance in the errors' cost, while keeping a reasonable ratio.

The various costs considered result in eight different batches; in each batch we evaluated five algorithms: *ProICET*, *Eg2*, *AdaBoost*, *J4.8* and *MetaCost*. The last three were provided by the Weka framework, and for *Eg2* we used the component we implemented for *ProICET*. The average total costs obtained are shown in Table 2.

Table 1 - Cost Matrices (M - Matrix)

M 1	lo	med	hi	**M 2**	lo	med	hi
lo	0.0	0.5	1.0	lo	0.0	0.5	1.0
med	1.5	0.0	0.7	med	3.0	0.0	0.7
hi	5.0	3.0	0.0	hi	10.0	6.0	0.0
M 3	lo	med	hi	**M 4**	lo	med	hi
lo	0.0	0.5	1.0	lo	0.0	0.5	1.0
med	0.75	0.0	0.7	med	3.0	0.0	0.5
hi	2.5	1.5	0.0	hi	5.0	3.0	0.0

Table 2 - Average Total Cost (TC - Test Cost; CM - Cost Matrix)

Average Total Cost	***Pro ICET***	**Ada Boost**	**Eg2**	**J4.8**	**Meta Cost**
C:0; TM:1	0.28	0.284	0.269	0.269	0.293
TC:0.1; TM:1	**0.414**	0.734	0.430	0.430	0.448
TC:0; TM:2	0.561	0.52	0.52	0.52	0.65
TC:0.1; TM:2	**0.678**	0.97	0.682	0.682	0.812
TC:0; TM:3	0.146	0.166	0.142	0.142	0.145
TC:0.1; TM:3	**0.252**	0.616	0.305	0.305	0.310
TC:0; TM:4	0.213	0.44	0.44	0.44	0.502
TC:0.1; TM:4	**0.575**	0.89	0.603	0.603	0.647

Table 3 - Average Accuracy (TC - Test Cost; CM - Cost Matrix)

Average Accuracy Rate (%)	***Pro ICET***	**Ada Boost**	**Eg2**	**J4.8**	**Meta Cost**
TC:0; TM:1	84.18	79.18	84.07	84.07	84.18
TC:0.1; TM:1	83.77				
TC:0; TM:2	83.87				83.26
TC:0.1; TM:2	84.07				
TC:0; TM:3	84.28				84.38
TC:0.1; TM:3	84.07				
TC:0; TM:4	84.07				83.36
TC:0.1; TM:4	83.77				

A first remark should be made about the fact that *ProICET* yields the lowest total cost in all the tests where both types of costs are considered. Moreover, Figure 1 shows that when the average cost over the eight different tests is considered, *ProICET* achieves again the smallest value.

In what the accuracy is concerned, we remark that the five algorithms attain similar rates (Table 3), with *AdaBoost* being the last. Since *AdaBoost* and *J4.8* are not cost-sensitive learners, their accuracy rate is not affected by the shift in misclassification costs, or test costs.

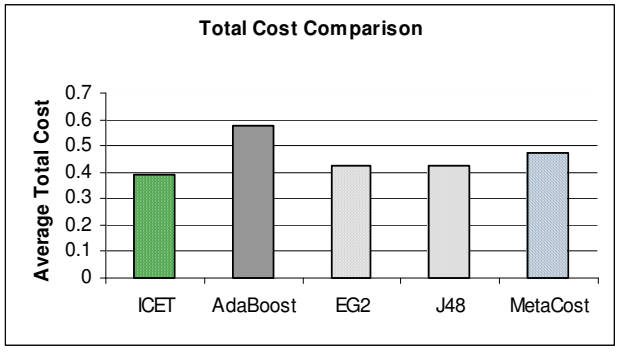

Figure 1 - Comparison of the total cost, averaged over the eight batches

Moreover, the balanced test costs have no influence on the accuracy rate of the *Eg2* algorithm (sensitive to test costs alone), therefore *Eg2* achieves the same accuracy rate as *J4.8*. Finally, since *MetaCost* is only sensible to misclassification costs, its accuracy rate is influenced only by the cost matrices.

The relatively low overall classification rate of our medical dataset is mostly owed to its small size. The comparative evaluation performed in [12] supports this conclusion, since the results obtained in our previous study show that *ProICET* yields lowest costs while maintaining high accuracy rates.

6. Conclusions and Future Development

In recent years, data driven analytic methods have started to gain increasing interest as complementing traditional methods in various applicative fields, such as oil slick detection, loan decisions, or medical diagnosis and prognosis.

Due to the particularities of the medical field, the cost-sensitive approach is particularly suited for medical data mining. One of the most prominent cost-sensitive algorithms that tackle both types of costs is *ICET*. By combining two well-known machine learning techniques, *ICET* manages to optimize overall costs by performing a genetic search.

Starting from the assumption that *ICET's* theoretical basis should provide a good starting point for a robust practical tool, we developed *ProICET,* a new system for medical diagnosis. *ProICET* implements the concept of performing a genetic search in the space of decision trees explored through greedy means. At the same time, it attempts to provide improved features in the genetic component, when compared to [10]. Due to our enhancements, *ProICET* managed to achieve lower costs than other powerful algorithms, such as *MetaCost*, *Eg2*, *AdaBoost*, or *J4.8*, when evaluated on classical medical benchmarks [12]. Moreover, the small values of the misclassification cost component achieved in [12] confirm very high accuracy rates.

Following the good results obtained by *ProICET* on classical medical datasets, we evaluated the system's performance on a real medical dataset, with records of patients that have been diagnosed with prostate cancer. The results obtained on this real medical data show that *ProICET* yields the lowest costs (out of the same systems we compared it with in [12]), while maintaining acceptable accuracy rates. The main reason why the accuracy rates are not at higher levels is rooted in the dataset: unbalanced, rare cases are very hard to learn because they are poorly represented. Moreover, since on larger medical problems the accuracy rates are higher [12], we estimate that, by increasing the size of the dataset, the system will achieve better results.

All in all, the results obtained so far confirm that *ProICET* is a robust system, achieving both low total costs and high accuracy rates.

Acknowledgements

This work has been supported by grant number 18CEEX-I03/2005, Intelligent System for Assisting the Therapeutical Decision at Patients with Prostate Cancer, of the Romanian Ministry for Education and Research.

References

[1] K. Derderian. "General Genetic Algorithm Tool". Tech. Rep.,www.karnig.co.uk/ga/ggat.html, 2002.

[2] P. Domingos. "Metacost: A general method for making classifiers cost-sensitive". Proceedings of the 5th International Conference on Knowledge Discovery and Data Mining, 1991.

[3] W. Fan, S. Stolfo, J. Zhang, and P. Chan. "Adacost: Misclassification cost-sensitive boosting". Proceedings of the 16th International Conference on Machine Learning, pages 97–105, 2000.

[4] Y. Freund and R. Schapire. "A decision-theoretic generalization of on-line learning and an application to boosting". Journal of Computer and System Sciences, 55(1):119–139, 1997.

[5] J. Li, X. Li, and X. Yao. "Cost-sensitive classification with genetic programming". Proceedings of the 2005 Congress on Evolutionary Computation, 3:2114–2121, 2005.

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

[7] J. Quinlan. "Boosting first-order learning". Proceedings of the 7th International Workshop on Algorithmic Learning Theory, 1160:143–155, 1996.

[8] S. Sheng and C. Ling. "Hybrid cost-sensitive decision tree". PKDD, pages 274–284, 2005.

[9] S. Sheng, C. Ling, and Q. Yang. "Simple test strategies for cost- sensitive decision trees". ECML, pages 365–376, 2005.

[10] P. Turney. "Cost sensitive classification: Empirical evaluation of a hybrid genetic decision tree induction algorithm". Journal of Artificial Intelligence Research, (2):369–409, 1995.

[11] P. Turney. "Types of cost in inductive concept learning". Proceedings of the Workshop on Cost- Sensitive Learning, 7th International Conference on Machine Learning, 2000.

[12] C. Vidrighin, C. Savin and R. Potolea, "A Hybrid Algorithm for Medical Diagnosis". Accepted at the IEEE Region 8 Eurocon 2007 Conference, April 2007.

[13] I. Witten and E. Frank. Data Mining: Practical machine learning tools and techniques, 2nd ed. Morgan Kaufmann,2005.