GP Classification under Imbalanced Data sets: Active Sub-sampling and AUC Approximation*

John Doucette and Malcolm I. Heywood†

January 5, 2009

Abstract

The problem of evolving binary classification models under increasingly unbalanced data sets is approached by proposing a strategy consisting of two components: Sub-sampling and 'robust' fitness function design. In particular, recent work in the wider machine learning literature has recognized that maintaining the original distribution of exemplars during training is often not appropriate for designing classifiers that are robust to degenerate classifier behavior. To this end we propose a 'Simple Active Learning Heuristic' (SALH) in which a subset of exemplars is sampled with uniform probability under a class balance enforcing rule for fitness evaluation. In addition, an efficient estimator for the Area Under the Curve (AUC) performance metric is assumed in the form of a modified Wilcoxon-Mann-Whitney (WMW) statistic. Performance is evaluated in terms of six representative UCI data sets and benchmarked against: canonical GP, SALH based GP, SALH and the modified WMW statistic, and deterministic classifiers (Naive Bayes and C4.5). The resulting SALH-WMW model is demonstrated to be both efficient and effective at providing solutions maximizing performance assessed in terms of AUC.

1 Introduction

Genetic Programming (GP) provides many unique opportunities for posing solutions to the basic Machine Learning design questions of representation, cost function, and credit assignment. In this work we are specifically interested in the topic of cost function design under the classification domain of supervised learning. Classically, an equally weighted cost function is assumed, such as 'hits' [11] or sum square error [2]. Such a design choice might be natural under balanced binary classification problems where each class carries an equal risk, but is questionable in the wider context of real world data sets that are frequently

*Published in the Proceedings of the European Conference on Genetic Programming (EuroGP), 2008. Lecture Notes in Computer Science, Vol. 4971. Copyright Springer-Verlag.

†J. Doucette and M.I. Heywood are with the Faculty of Computer Science, Dalhousie University, 6050 University Av., Halifax, NS, B3H 1W5, Canada.

unbalanced. At the very least, as the class distribution becomes increasingly unbalanced, the likelihood of evolving degenerate classifier behavior will increase [6], [19]. Addressing the class imbalance problem has at least two related perspectives: identification of an appropriate cost (fitness) function, and sampling the original distribution of training exemplars such that the learning algorithm adapts under a different distribution than the original data set.

In the case of sampling algorithms, several paradigms have appeared, including: (1) boosting and bagging algorithms that tend to result in multiple individuals being built relative to static resampling of the original training data, and; (2) active learning or sub-sampling algorithms that may identify a sub-sample of exemplars from the larger training data set at each training cycle. The later case is of interest in this work. In particular we begin with the observation made from Weiss and Provost (under decision tree induction) [20]; that is, robust classifiers may be built relative to the post training performance metric of Area Under the Curve (AUC) if sub-samples are built stochastically using a uniform sampling heuristic that simultaneously enforces class balance in the sub-sample.

In this work we assume the balanced stochastic sub-sampling model as our base line model for scaling GP to larger (and therefore more interesting) data sets than would typically be the case without a hardware speedup; hereafter denoted the Simple Active Learning Heuristic (SALH). Next we investigate the utility of a fitness function capable of approximating the properties of the AUC metric. Specifically, AUC represents a rank based performance metric that explicitly measures performance in terms of two typically 'conflicting' performance goals at multiple performance points. As such, the model is encouraged to, for example, maximize recall while simultaneously minimizing false positive rate, thus explicitly penalizing degenerate behaviors that might dominate models trained from unbalanced distributions of exemplars. One drawback associated with the wider utility of AUC as a cost function in Machine Learning has been the computational cost of first estimating the Receiver Operating Characteristic (ROC) curve and then deriving the associated AUC. Naturally, by assuming a sub-sampling model we decouple the evolutionary cycle from the original dimension of the data set. However, even under such conditions a significant overhead still exists in the inner loop if we attempt to estimate the AUC directly. The final component of the model investigated in this work is therefore to make use of the Wilcoxon-Mann-Whitney (WMW) statistic where this provides a direct estimator for the AUC metric [9], [21]. To this end, we detail modifications necessary to focus the ensuing GP classifiers, such that 'robust' performance under the WMW metric was generalized to corresponding behavior under test conditions.

The proposed model of WMW fitness function estimated over exemplar subsets identified under SALH, is benchmarked over six unbalanced data sets from the UCI repository [16]. Comparisons are made against both deterministic classifiers (C4.5 and Naive Bayes), canonical GP, and GP under SALH (both of the latter assume 'hits' based fitness). The WMW model is the most successful in maximizing the area under the curve performance statistic on test data,

bettering C4.5 on five of the six data sets, and significantly better than either alternative GP paradigm.

2 Related Work

As indicated in the introduction we approach the problem of designing a 'robust' classifier using two inter-related concepts: establishing a suitable training exemplar sampling algorithm, and establishing an appropriate cost (fitness) function. Within the context of Genetic Programming in particular, several works have proposed approaches to the training exemplar sub-sampling problem. The work of Gathercole and Ross in particular demonstrated that not all exemplars are equally relevant to the training task at any point in time [8]. Two heuristics, denoted exemplar 'age' and 'difficulty' were used to bias the selection of exemplars to appear in the current fitness evaluation (training epoch). Such a model provides a considerable speedup relative to fitness evaluation over all training exemplars and was demonstrated to result in individuals performing no worse. Recent research has considered the utility of competitive coevolutionary models as the basis for an alternative model of active learning. In particular the host-parasite model of Hillis demonstrated that such a model could provide the basis for biasing the selection of a subset of training exemplars at fitness evaluation [10]. The host-parasite model does however suffer from the problem of establishing the relevant problem dependent 'virulence factor' to ensure that the exemplars selected (parasite) do not dominate the ability of the learners (host) [3], [15]. More recently, the competitive coevolutionary model for rewarding the ability of exemplars to 'distinguish' between learners under a Pareto model of coevolution has received a lot of interest [7], [17], [5]. Attempts to make use of the Pareto competitive coevolutionary paradigm under the GP classification domain have utilized a two population model, with one population representing the subset of training exemplars on which fitness evaluation is conducted, and a second population in which classifiers are evolved. Under such an architecture, Pareto competitive frameworks to date concentrate on establishing archiving strategies that posses desirable properties (such as monotonic progress) [4]. However, the indexing of exemplars by the 'point' population does not hold any implicit structure to guide the definition of appropriate variation operators. As such the most successful Pareto Competitive models, under the GP classification domain, have relied on the uniform selection of exemplar indexes and a class balancing heuristic to create the point population [13].

With respect to the utility of performance metrics that explicitly reward the evolution of 'robust' as opposed to naive classifier behaviors, many authors have considered cost functions which make use of fixed penalty functions [19], [18]. Adaptive cost functions have also been proposed, for example Eggermont *et al.* developed a scheme for periodically re-weighting the error associated with training exemplars during training [6]. This is naturally related to the 'difficulty' heuristic devised by Gathercole, but without attempting to use this as an exemplar selection bias under an active learning paradigm. Langdon and

Buxton considers the problem of AUC optimization given two previous classifiers with different ROC profiles [12]. However, the problem addressed is naturally distinct from designing the initial classifiers such that ROC profiles are suitably distinct.

The two themes central to the method adopted in this work result from the findings of Weiss and Provost on training decision tree induction classifiers under unbalanced data sets [20], and a successful attempt at constructing an AUC type cost function for training a neural network classifier on a very unbalanced data set [21]. In the case of Weiss and Provost, a systematic study is performed on the impact of training subset class balance under the C4.5 algorithm. A clear statistically significant preference was demonstrated for uniform exemplar selection while enforcing equal representation of major and minor classes. Such a heuristic was central to establishing effective operation of point population sampling algorithms for Pareto competitive coevolution of classifiers, and in some cases may out perform this model [14]. The work of Yan *et al.*, began by formally demonstrating that minimizing metrics such as mean square error or cross-entropy is not sufficient for maximizing AUC [21]. They then make use of the WMW estimator for AUC and derive an alternative, back-propagation compatible version of the metric, thus enabling them to train a multi-layer perceptron to maximize the AUC performance metric directly, and demonstrate the utility of such an algorithm under a very unbalanced 'churn' prediction problem.

3 Methodology

The basic goal of this work is to provide a generic model for the evolution of GP classifiers under unbalanced data sets through a combined approach of class balanced stochastic sub-sampling and a modified WMW cost function. The combined approach is necessary to: (1) actively bias the distribution of exemplars over which learning is conducted; (2) establish a 'robust' cost function, and; (3) address the computational cost of fitness evaluation. In the following we will define the sub-sampling based active learning model of GP classification, and WMW cost function and associated modification for the case of GP.

3.1 Simple Active Learning Model

In order to decouple the cost of fitness evaluation from the size of the training data set, an active learning model is assumed. The Dynamic Subset Selection (DSS) model of Gathercole and Ross has been widely used in the GP domain. However, in this work we assume a simpler model. Specifically, exemplars are selected with uniform probability from the original training partition, such that major and minor class provide an exemplar subset of fixed size with equal representation of both classes. The DSS algorithm was originally compared against subsets formed from exemplars sampled with purely uniform probability, but without the requirement for equal class balance [8]. This may naturally result

in subsets being formed that represent major and minor classes with the same distribution as in the entire training partition. However, as the distribution of major to minor class increases, the likelihood of building 'degenerate' subsets increases, see for example the comparison of DSS and canonical GP in [13]. The study of Weiss and Provost establishes that such a scheme for building subsamples will result in optimizing for an accuracy based performance metric, but relative to a more informative performance metric such as AUC, will result in very low scores. Thus, this study adopts a class balance enforcing subsampling model that selects exemplars with uniform probability from major and minor class, until the equal class constraint is satisfied.

Aside from the active learning model for defining a new subset of exemplars at each generation, the GP classifier takes the form of canonical tree structured GP. Specifically, in the case of the wrapper operator, a sigmoid is employed where this has the desirable property of encouraging exemplars to move away from the switching point of the class boundary.

$$y(x) = \frac{2}{(1 + \exp(-GPout(x)))} - 1 \tag{1}$$

where *GPout(x)* denotes the 'raw' scalar value returned by the root node of the phenotype following evaluation of the program on input vector 'x', and $y(x)$ denotes class membership over the interval [-1, 1] with respect to exemplar 'x'. Naturally, values tending towards '-1' indicate out of class and values tending towards '1' are indicative of in class membership.

3.2 Wilcoxon-Mann-Whitney Fitness Function

The area under the curve (AUC) metric expresses classifier performance in terms of the area under the 'receiver operating characteristic' or ROC. Such curves typically characterize classifier performance in terms of true positive rate versus false positive rate [1], [9]. Unlike most widely utilized performance metrics, such as accuracy or precision and recall, the ROC curve does not rely on a single performance point to characterize classifier behavior. That is to say, both true positive rate and false positive rate are estimated at multiple performance points for each exemplar; where the performance points are derived, for example, from cuts taken across the class membership function of (1). Needless to say, the more thresholds utilized, the more accurate the characterization, but the more expensive the evaluation. Specifically, estimating the ROC curve requires the re-evaluation of true and false positive rates for a sufficient number of performance points to provide an accurate rendition of the curve. Only with this complete can estimate of the AUC. All of this takes place within the inner loop of GP. Thus in this work, we do not estimate the AUC or ROC, but make use of the Wilcoxon-Mann-Whitney (WMW) statistic, where this is already known to be an equivalent estimator for AUC without building the ROC [9]. The WMW statistic has the form,

$$WMW(I,P,N) = \sum_{i=0}^{|P|} \sum_{j=0}^{|N|} C(y, P_i, N_j) \tag{2}$$

where

$$C(y,a,b) = \begin{cases} 1 & if\, y(a) > y(b) \\ 0 & otherwise \end{cases} \tag{3}$$

and $y(a)$ is the class membership returned by the wrapper operator of equation (1) under the exemplar represented by input vector a, P is the set of all majority class exemplars, and N is the set of all minority class exemplars. Thus, $P_i(N_j)$ is the ith (jth) element of $P(N)$.

Naturally, equation (2) conducts a series of pairwise comparisons between in and out of class exemplars, only rewarding cases in which the class membership function for in class exemplars exceeds that for the out of class cases as per (3). However, when used in combination with a continuous valued (as opposed to binary) membership operator it is also necessary to explicitly reward class membership values that fall on the relevant side of the origin. We denote the resulting WMW based fitness function $WMW_{fitness}$, expressed as follows,

$$WMW_{fitness}(y,P,N) = f(y,P) \cdot f(y,N) + WMW(y,P,N) \tag{4}$$

where

$$f(y,S) = \sum_{i=0}^{|S|} \begin{cases} 1 & if\, d(S_i) = y(S_i) \\ 0 & otherwise \end{cases} \tag{5}$$

and $d(S_i)$ is the desired class label of exemplar i, and $I(S_i)$ is the corresponding binary class label suggested by the GP classifier (i.e. thresholding the wrapper about the origin).

The first component of the right hand side of (4) contributes a 'point' for every pair of correctly labeled exemplars; whereas the second component takes the form of the original WMW metric. On an exemplar by exemplar basis, the WMW contribution is satisfied first; thus, evolution will first find individuals with good AUC properties and then normalize the pairwise dominance property relative to the origin of the activation function, equation (1).

4 Results

4.1 Canonical GP

The empirical evaluation is conducted utilizing a common canonical tree structured model of GP [11] using the 1.1 distribution of lilgp [22], although the GP representation itself has no impact on the algorithm proposed. The selection operator takes the form of Koza's 'overselection', thus the top thirty two (bottom sixty eight) percent of the population account for eighty (twenty) percent of the parents. Such a model naturally has a higher take-over rate than would be the

Table 1: Data Set Characterization

Name	Size	% Minority Class	# Attributes
Abalone	4,177	8.7	8
Sick Thyroid	3,163	9.3	25
Opt. Digits	5,620	9.9	64
Solar Flare	1,389	15.7	10
Adult	48,842	23.9	14
Liver	345	42	6

case for fitness-proportionate selection alone. The terminal set was limited to indexing the features of the problem domain, whereas the function set took the form of the four arithmetic operators, four higher order operators with a single argument ($sine(a)$, $cos(a)$, e^a, $\sqrt{a}$) and the standard conditional statement with four arguments (c if $a < b$, d otherwise). The remaining GP parameters take the form: Population size (800), Max tree nodes (256), Half-half initialization (2-6 node depth), Crossover (0.7), Mutation (0.3), Internal versus leaf node likelihood (0.9/0.1). In no cases was any attempt made to optimize these values.

4.2 Data sets

A total of six data sets were employed in the evaluation, five corresponding to a subset of those used in the study of Weiss and Provost [20], and the sixth corresponding to the widely utilized BUPA liver diagnosis data set. All data sets are available through the UCI repository [16] and have been selected on account of the resulting varied ratio of major to minor class distributions. As per the the Weiss and Provost study, we make the multiclass data sets binary by defining the minor class as in class and the remaining classes as the out of class exemplars. In each case the data set is stratified, with twenty five percent of the data set being withheld for the purpose of establishing a test set and the remainder of the data representing the training set. Table 1 establishes the basic properties each data set as a whole.

4.3 Evaluation

Evaluation is conducted in the form of three separate comparisons. In the first case we compare canonical GP with the Simple Active Learning Heuristic (SALH) of Section 3.1 over the four smaller data sets (too computationally expensive to apply canonical GP to the Adult data set). The only difference between the two models is the set of exemplars utilized for fitness evaluation. Experiment two compares GP classifiers evolved using the SALH and a count based fitness function, versus the same active learning heuristic, but with fitness evaluated over the modified WMW metric of (4). Our last comparison

compares GP models evolved under the WMW metric with those trained under deterministic machine learning algorithms.

All post training evaluation will be performed in terms of test set performance as measured by the AUC metric derived from the trapezoidal integration algorithm [1]. That is to say, the AUC metric expresses the area under the curve as estimated from the receiver operating characteristic (ROC). The ROC is constructed from the performance of each classifier under true positive and false positive rates taken from twenty two points representing thresholds taken uniformly across the interval of the wrapper operator, equation (1). This results in a scalar characterization of performance, with values in the range of zero (no better than guessing) to a half (perfect classification of both minority and major classes).

4.3.1 Canonical GP versus SALH

In order to establish whether the baseline canonical model of classification, that is fitness evaluation over the entire set of training exemplars, produces classifiers that are any more robust than fitness evaluation over the balanced uniform model of subset selection, we compare post training AUC performance over fifty runs of each model on all but the 'Adult' data set from Table 1. In no cases was a statistically significant difference recorded at a Confidence interval of ninety five percent (Student T-test). Thus no negative impact is attributed to fitness evaluation conducted over exemplar subsets identified under SALH versus all training exemplars.

4.3.2 Combining SALH with WMW fitness

The next test establishes the significance of introducing the WMW fitness function in combination with SALH. In effect we how have an efficient mechanism for evolving individuals under a 'robust' estimator of fitness, albeit only over subsets selected stochastically under the balance enforcing heuristic. Figures 1 and 2 summarize AUC returned on each test data set as first quartile, median, and third quartile (statistic collected over fifty runs). The WMW fitness function yields solutions with statistically significantly better AUC values under five of the six data sets at a confidence of ninety nine percent, and at ninety five percent in the case of the liver data set. The amount of variation in results returned in models trained using the WMW based fitness function are also much lower than that under the hits based fitness function. In short, even when the natural distribution of the original data set tends to equal representation of both classes, the WMW based fitness function is much more effective at directing the credit assignment process.

4.3.3 Comparison with Deterministic models

Our final test compares the test set performance of models trained using Naive Bayes and C4.5 to those evolved under the WMW fitness function. This immediately presents the problem of establishing a framework for making the

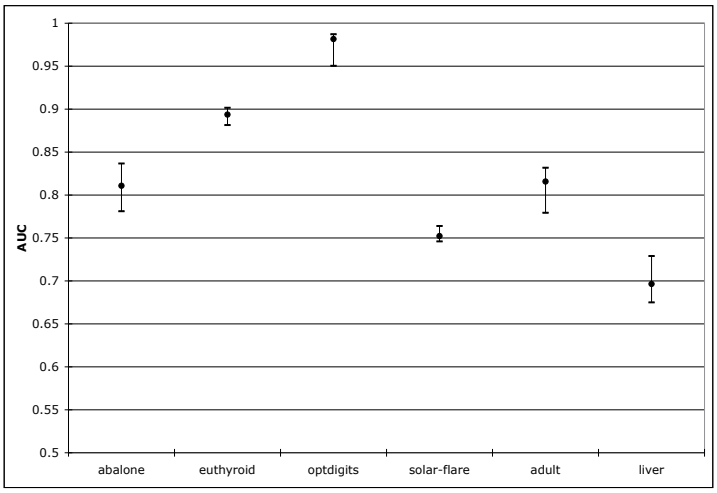

Figure 1: SALH with WMW fitness function: Post training AUC performance under test partition for each data set.

comparison. In particular, the deterministic models are trained following a single pass through the entire training data set, whereas evolutionary models are evolved over multiple runs. Making multiple folds of the original partition does not alter this relationship. Each fold would still require multiple runs of the evolutionary method. In effect GP requires the initialization of "free parameters" that are distinct from the learning parameters, whereas deterministic models such as C4.5 and Naive Bayes only have learning parameters. Thus, we adopt the following policy in which the deterministic model is used to establish a performance threshold against which we then ask what is the likelihood of the evolutionary model matching or bettering this performance.

Figure 3 reports the likelihood of the GP classifier initializations matching or bettering the performance of the Naive Bayes and C4.5 deterministic classifiers. Larger bars imply more of the GP solutions matched or bettered the base line established by the deterministic model. Conversely, no bar implies that all GP solutions were worse than the deterministic base line. The data set that returned no benefit from the GP model was the largest data set, Adult, where this might be an indicator for evolving over more generations (the common training limit of fifty generations implies that only seven percent of the Adult training data is sampled). Both Abalone and Euthyroid, the two data sets with the largest degree of class imbalance were most likely to result in the GP model improving

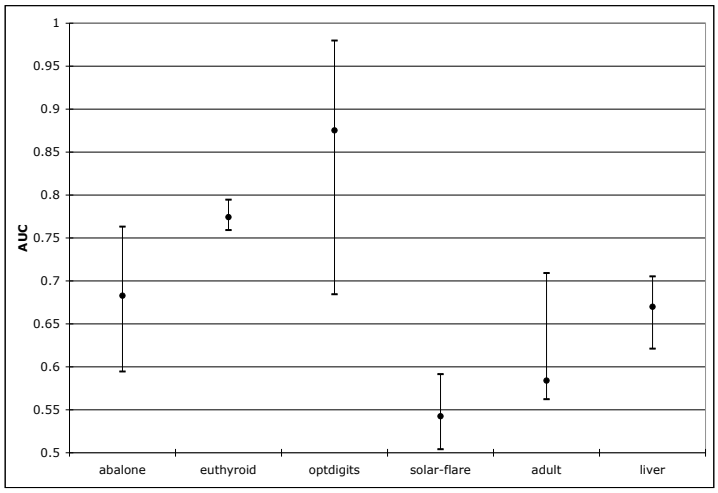

Figure 2: SALH with 'hits' fitness function: Post training AUC performance under test partition for each data set.

on the deterministic classifier base line. Interestingly, C4.5 found the Solar flare data set particularly difficult, whereas Naive Bayes did not perform as well on the Liver data set (the most balanced data set considered). However, the Naive Bayes classifier bettered both C4.5 and GP on the Optical Character recognition problem. In short the GP solutions were better than the Naive Bayes classifier in a minimum of sixty percent of the cases in four of the six data sets, and unable to better Naive Bayes on the other two data sets. Under the performance target set by C4.5, GP was better at least fifty percent of the time under four of the six data sets, and returned results that were better in at least twenty five percent of the initializations on the fifth data set.

5 Conclusion

The problem of training GP classifiers under large unbalanced binary data sets is addressed through the dual approach of training exemplar selection and appropriate fitness function design. We begin by utilizing a class balance heuristic under an active learning paradigm, for the evolution of GP classifiers. The ensuing Simple Active Learning Heuristic is shown to perform at least as well as canonical GP evolved over all training exemplars. The second part of our

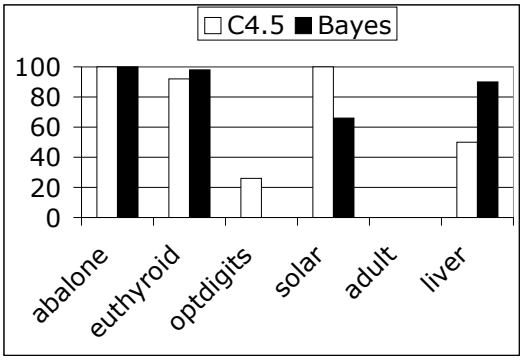

Figure 3: Percent of SALH-WMW solutions matching or bettering the deterministic classifier baseline under post training AUC performance statistic evaluation of test partition.

approach begins with the WMW estimator for the AUC metric. As is, this metric rewards pairwise dominance behaviour as measured between minor and major class exemplars. However, we are also interested in maximizing the separation between the two sets of behaviors as mapped to 'GPout' and resolved in terms of a smooth wrapper operator, a sigmoid. To this end, we introduce a second factor into the fitness function, such that individuals that both establish the pairwise dominance property and enforce class membership relative to the wrapper operator origin receive more reward than those establishing the dominance property alone. Benchmarking on six data sets from the UCI repository with minor class representations in the range of five to forty percent of the data set demonstrates that the proposed approach is significantly better than classifiers evolved under the same active learning heuristic and typically better than C4.5 or Naive Bayes under five and four of the six data sets respectively.

Future work will continue to investigate the significance of fitness functions in GP classifier design. In particular, recent work in machine learning has demonstrated a bias between classes of cost function and classifier operation. We anticipate there being equivalent relationships between function set design and classes of fitness function.

Acknowledgments

The authors gratefully acknowledge the support of NSERC USRA and Discovery Grant programs, and CFI New Opportunities infrastructure program.

References

[1] A. P. Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159, 1997.

[2] M. Brameier and W. Banzhaf. A comparison of linear Genetic Programming and Neural Networks in medical data mining. *IEEE Transactions on Evolutionary Computation*, 5(1):17–26, 2001.

[3] J. Cartlidge and S. Bullock. Learning lessons from the common cold: How reducing parasite virulence improves coevolutionary optimization. In *IEEE Congress on Evolutionary Computation*, pages 1420–1425, 2002.

[4] E. D. de Jong. A monotonic archive for pareto-coevolution. *Evolutionary Computation*, 15(1):61–94, 2007.

[5] E. D. de Jong and J. B. Pollack. Ideal evaluation from coevolution. *Evolutionary Computation*, 12(2):159–192, 2004.

[6] J. Eggermont, A. E. Eiben, and J. I. van Hermert. Adapting the fitness function in GP for data mining. In *Proceedings of the European Conference on Genetic Programming (EuroGP)*, volume 1598 of *LNCS*, pages 193–202, 1999.

[7] S. G. Ficici and J. B. Pollock. Pareto optimality in coevolutionary learning. In *European Conference on Artificial Life*, pages 316–325, 2001.

[8] C. Gathercole and P. Ross. Dynamic training subset selection for supervised learning in genetic programming. In *Parallel Problem Solving in Nature III*, volume 866 of *LNCS*, pages 312–321, 1994.

[9] D. J. Hand. *Construction and Assessment of Classification Rules*. John Wiley, 1997.

[10] W. D. Hillis. Co-evolving parasites improve simulated evolution as an optimization procedure. In *Artificial Life II*, volume X of *Santa Fe Institute Studies in the Sciences of Complexity*, pages 313–324, 1990.

[11] J. R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.

[12] W. B. Langdon and B. F. Buxton. Evolving receiver operating characteristics for data fusion. In *Proceedings of the European Conference on Genetic Programming (EuroGP)*, volume 2038 of *LNCS*, pages 87–96, 2001.

[13] M. Lemczyk and M. I. Heywood. Training binary GP classifiers efficiently: a Pareto-coevolutionary approach. In *Proceedings of the European Conference on Genetic Programming (EuroGP)*, volume 4445 of *LNCS*, pages 229–240, 2007.

[14] P. Lichodzijewski and M. I. Heywood. Pareto-coevolutionary Genetic Programming for problem decomposition in multi-class classification. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, volume 1, pages 464–471, 2007.

[15] A. R. McIntyre and M. I. Heywood. Toward co-evolutionary training of a multi-class classifier. In *Proceedings of the Congress on Evolutionary Computation (CEC)*, volume 3, pages 2130–2137, 2005.

[16] D. J. Newman, S. Hettich, C. L. Blake, and C. J. Merz. UCI repository of machine learning databases. [http://www.ics.uci.edu/∼mlearn/mlrepository.html], 1998.

[17] J. Noble and R. A. Watson. Pareto coevolution: Using performance against coevolved opponents in a game as dimensions for Pareto selection. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, pages 493–500, 2001.

[18] G. Patterson and M. Zhang. Fitness functions in Genetic Programming for classification with unbalanced data. In *Proceedings of the Australian Joint Conference on Artificial Intelligence*, volume 4830 of *LNAI*, pages 464–471, 2007.

[19] D. Song, M. I. Heywood, and A. N. Zincir-Heywood. Training Genetic Programming on half a million patterns: An example from anomaly detection. *IEEE Transactions on Evolutionary Computation*, 9(3):225–239, 2005.

[20] G. M. Weiss and R. Provost. Learning when training data are costly: The effect of class distribution on tree induction. *Journal of Artificial Intelligence Research*, 19:315–354, 2003.

[21] L. Yan, R. Dodier, M. C. Mozer, and R. Wolniewicz. Optimizing classifier performance via the Wilcoxon-Mann-Whitney statistic. In *Proceedings of the International Conference on Machine Learning*, pages 848–855, 2003.

[22] D. Zoungker and B. Punch. lilgp genetic programming system. version 1.1. [http://garage.cse.msu.edu/], 1998.