# Comparing Ensemble Learning Approaches in Genetic Programming for Classification with Unbalanced Data

Urvesh Bhowan
School of Computer Science and Statistics, Knowledge and Data Engineering Group, Trinity College Dublin, Ireland
Urvesh.Bhowan@scss.tcd.ie

Mark Johnston
School of Mathematics, Statistics and Operations Research, Victoria University of Wellington, New Zealand
Mark.Johnston@msor.vuw.ac.nz

Mengjie Zhang
School of Engineering and Computer Science, Victoria University of Wellington, New Zealand
Mengjie.Zhang@ecs.vuw.ac.nz

## ABSTRACT

This paper compares three approaches to evolving ensembles in Genetic Programming (GP) for binary classification with unbalanced data. The first uses bagging with sampling, while the other two use Pareto-based multi-objective GP (MOGP) for the trade-off between the two (unequal) classes. In MOGP, two ways are compared to build the ensembles: using the evolved Pareto front alone, and using the whole evolved population of dominated and non-dominated individuals alike. Experiments on several benchmark (binary) unbalanced tasks find that smaller, more diverse ensembles chosen during ensemble selection perform best due to better generalisation, particularly when the combined knowledge of the whole evolved MOGP population forms the ensemble.

## Categories and Subject Descriptors

I.2.8 [**Problem Solving, Control Methods, and Search**]: Heuristic methods; I.5.2 [**Design Methodology**]: Classifier design and evaluation

## General Terms

Design

## Keywords

Genetic Programming, Multi-objective Optimisation, Classification, Class Imbalance.

## 1. GOALS

Machine learning algorithms can suffer a performance bias when at least one class has a small number of training examples (called the *minority class*) compared to the other(s) (called the *majority class*). Induced classifiers can have high accuracy on the majority class but poor accuracy on the important minority class. This paper compares three approaches for evolving ensembles using Genetic Programming (GP) which aim to achieve high and balanced accuracy rates on *both* classes when data in unbalanced. Multiple classifiers working together to predict the class labels are known to provide better generalisation than canonical "single-predictor" methods if these classifiers are accurate and *diverse* (do not make the same errors on the same inputs) [2][3].

Copyright is held by the author/owner(s).
*GECCO'13,* July 6-10, 2013, Amsterdam, The Netherlands.
ACM 978-1-4503-1964-5/13/07.

This paper has three main goals. The first compares two approaches to account for the unequal class sizes during ensemble learning: traditional bagging where data is sampled into smaller balanced subsets to train the individual members, and multi-objective GP (MOGP) where the minority and the majority classes are traded-off against each other during learning to evolve a *Pareto front* [2]. Unlike bagging, in MOGP the original (unbalanced) data set is used directly in training (without sampling), thereby reducing the risk of over-fitting (some sampling methods can exclude potentially useful data from learning. The second goal compares two approaches to build the ensembles in MOGP: using the evolved Pareto front alone, and using the whole evolved population of dominated and non-dominated individuals alike. We hypothesise that using the combined knowledge of the full evolved MOGP population can improve performances compared with using the Pareto front alone. The third goal uses offline evolutionary ensemble selection (off-EEL) [3] to find/choose only good individuals for the ensemble to improve ensemble performances.

## 2. GP ENSEMBLE APPROACHES

In bagging, the training set is sampled into $N$ balanced subsets (called bootstrap samples) by under-sampling the majority class with replacement, where $N$ is determined *a priori* (in the experiments, $N$ is 25). A GP classifier is then evolved using a given bootstrap training sample, and this process is repeated for all bootstrap samples. In the testing phase, a majority vote of the predicted class labels from all $N$ bagged GP classifiers determines the final ensemble output (i.e. class label) for each input. This overall process returns a single evolved ensemble.

In MOGP, a *Pareto front* of classifiers is simultaneously evolved along the objective trade-off surface in a single optimisation run. This is accomplished using Pareto dominance in fitness to rank the solutions in the population according to their objective performances. Pareto dominance asserts that one solution will *dominate* another solution if it is at least as good as the other solution on all the objectives and better on at least one [5]. The MOGP objective formulation incorporates a solution's accuracy and diversity on a *single* class, where the minority class objective is traded-off against the majority class objective. Diversity is measured using pairwise failure crediting (PFC) and aims to reduce the overlap of common errors between solutions (see [2] for details on the objectives and PFC). Once the solutions are

evaluated on the objectives, the widely-used SPEA2 [5] algorithm is used for Pareto dominance ranking; these values then represent the final MOGP fitness values used to identify the Pareto front (see [5] for details on SPEA2). This paper compares two approaches to building ensembles in MOGP: using the evolved Pareto front alone, and using the full evolved population. Similar to GP bagging, in the testing phase a majority vote from ensemble members determines the ensemble output (class label) for an input.

The offline evolutionary ensemble selection (off-EEL) algorithm [3] is used to choose good individuals for the ensemble. Given a pool of base classifiers, off-EEL sorts these classifiers by their fitness values. Each individual is then iteratively copied into the ensemble where, at each step, the ensemble is evaluated using a majority vote. Once all base classifiers are processed, the best-performing intermediate ensemble is chosen as the final ensemble.

A tree-based structure is used to represent the base GP solutions [4]. Each solution is a mathematical expression that outputs a (real) number for a given input which is mapped to the class labels using zero as the threshold. Four standard arithmetic operators, $+, -, \times$, and % (protected division), and the conditional operator `if` are used as functions.

## 3. EXPERIMENTAL FINDINGS

The experiments use eight benchmark binary classification problems. A 50/50% split is used to divide each data set into the *training* and *test* sets where both sets preserve the same class imbalance ratio. Seven tasks from the UCI Repository [1] are used: Ionosphere (Io), SPECT heart (Sp), Balance (Bal), Ecoli (using classes *im* and *pp* against the rest as $E_1$ and $E_2$, respectively), and Yeast (using classes *mit* and *me3* against the rest as $Y_1$ and $Y_2$, respectively). The eighth task, Pd, is an image classification task from `http://www.science.uva.nl/research/isla/downloads/pedestrians`. Io has a class imbalance ratio of 1:3; Sp, Pd and $E_1$ of 1:4; $E_2$ and $Y_1$ of 1:6; $Y_2$ of 1:9; and Bal of 1:12.

Table 1 shows the ensemble sizes and geometric mean accuracies with and without off-EEL [3] ensemble selection over 50 independant GP runs. Here *Full* and *PF* are the MOGP approaches using the full evolved population and Pareto front, respectively, and *Bag* uses bagging. Results in bold denote whether performance is statistically significantly better with or without ensemble selection, and the superscript identifies which ensemble approach in a given group has a significantly better performance (95% confidence).

Table 1 shows that all three GP approaches using off-EEL perform as well as, or significantly better than using no ensemble selection on tasks. The off-EEL ensembles are also much smaller in size compared to without, showing that choosing fewer, more diverse members for the ensemble improves generalisation. The relatively high geometric mean accuracies, particularly for off-EEL, show that balanced accuracy rates are achieved on both classes. *Bag* shows the most improvement in performance using off-EEL, suggesting that MOGP already has good diversity due to the PFC objective in fitness. Table 1 also shows that *Full* generally performs better than *PF* but only when off-EEL is also used (without ensemble selection, *Full* is not better than *PF* in any task). This suggests that as hypothesised, using the combined knowledge of the whole MOGP population of dominated and non-dominated individuals alike can improve performances compared to using the Pareto front alone. *Bag* and *Full* with off-EEL generally achieve competitive performances compared to each other (each shows no significant differences in performance in five tasks), but the former accomplishes this using smaller ensemble sizes than MOGP on these tasks.

**Table 1: Ensemble sizes and geometric mean accuracies for the GP approaches with and without off-EEL [3] ensemble selection over 50 runs. Bold text denotes whether performance is statistically significantly better with or without ensemble selection, and the superscript identifies which approach has significantly better performances (95% confidence).**

| Set | Appr. | No Ens. Sel. | | Off-EEL[3] | |
|---|---|---|---|---|---|
| | | Size | Geomean % | Size | Geomean % |
| Io | Full$^a$ | 487.3 | 90.1 ± 2.6 | 370.9 | **91.3 ± 2.6** |
| | PF$^b$ | 28.1 | 87.5 ± 3.8 | 22.6 | **90.0 ± 2.7** |
| | Bag$^c$ | 25.0 | 91.7 ± 2.3$^b$ | 17.0 | **92.4 ± 1.9**$^b$ |
| Sp | Full$^a$ | 430.7 | 68.1 ± 3.8 | 111.5 | **74.5 ± 2.7**$^b$ |
| | PF$^b$ | 27.3 | 68.3 ± 2.5 | 11.0 | 72.3 ± 3.1 |
| | Bag$^c$ | 25.0 | 72.7 ± 2.2$^a$ | 15.7 | **74.0 ± 2.0** |
| Pd | Full$^a$ | 491.7 | 89.5 ± 0.7$^c$ | 434.7 | 89.7 ± 0.6$^c$ |
| | PF$^b$ | 71.6 | 88.2 ± 1.3$^c$ | 31.2 | 88.6 ± 1.1$^c$ |
| | Bag$^c$ | 25.0 | 58.6 ± 7.7 | 11.9 | **74.1 ± 4.8** |
| $E_1$ | Full$^a$ | 349.4 | 77.9 ± 2.4$^c$ | 134.0 | 77.8 ± 3.4$^b$ |
| | PF$^b$ | 8.3 | 75.6 ± 4.2 | 5.9 | 75.0 ± 4.4 |
| | Bag$^c$ | 25.0 | 73.3 ± 5.4 | 16.8 | **77.4 ± 3.1**$^b$ |
| $E_2$ | Full$^a$ | 492.7 | 99.7 ± 0.5$^c$ | 483.2 | 99.9 ± 0.2$^c$ |
| | PF$^b$ | 15.4 | 98.9 ± 0.7$^c$ | 10.6 | **99.8 ± 0.2**$^c$ |
| | Bag$^c$ | 25.0 | 95.2 ± 1.5 | 8.4 | **97.0 ± 0.9** |
| $Y_1$ | Full$^a$ | 374.4 | 73.8 ± 1.3 | 263.7 | **74.8 ± 1.2** |
| | PF$^b$ | 39.7 | 73.3 ± 1.4 | 30.2 | **74.4 ± 1.1** |
| | Bag$^c$ | 25.0 | 74.0 ± 1.2 | 18.3 | 74.8 ± 1.1 |
| $Y_2$ | Full$^a$ | 374.4 | 90.9 ± 1.0 | 261.0 | **92.3 ± 0.7** |
| | PF$^b$ | 27.9 | 90.8 ± 1.4 | 17.2 | **91.9 ± 0.9** |
| | Bag$^c$ | 25.0 | 92.5 ± 0.8$^{a,b}$ | 19.8 | **93.1 ± 0.5**$^b$ |
| Bal | Full$^a$ | 361.9 | 83.6 ± 7.5 | 179.9 | 86.7 ± 5.9 |
| | PF$^b$ | 20.8 | 78.9 ± 13.4 | 10.4 | 84.1 ± 6.9 |
| | Bag$^c$ | 25.0 | 85.4 ± 7.1 | 10.3 | **92.3** ± 4.9$^{a,b}$ |

## 4. REFERENCES


[1] Asuncion, A., and Newman, D. UCI Machine Learning Repository, 2007. University of California, Irvine, School of Information and Computer Sciences.

[2] Bhowan, U., Johnston, M., Zhang, M., and Yao, X. Evolving diverse ensembles using genetic programming for classification with unbalanced data. *IEEE Transactions on Evolutionary Computation*. (Accepted, April 2012).

[3] Gagné, C., Sebag, M., Schoenauer, M., and Tomassini, M. Ensemble learning for free with evolutionary algorithms? In *Proceedings of Genetic and Evolutionary Computation Conference* (2007), ACM Press, pp. 1782–1789.

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

[5] Zitzler, E., Laumanns, M., and Thiele, L. SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization. Tech. rep., 2001. TIK-Report 103, Department of Electrical Engineering, Swiss Federal Institute of Technology.