Improving GP Classifier Generalization Using a Cluster Separation Metric

Ashley George, Malcolm I. Heywood
Dalhousie University, Faculty of Computer Science
6050 University Avenue, Halifax, Nova Scotia, Canada, B3H 1W5
ageorge@cs.dal.ca, mheywood@cs.dal.ca

ABSTRACT

Genetic Programming offers freedom in the definition of the cost function that is unparalleled among supervised learning algorithms. However, this freedom goes largely unexploited in previous work. Here, we revisit the design of fitness functions for genetic programming by explicitly considering the contribution of the wrapper and cost function. Within the context of supervised learning, as applied to classification problems, a clustering methodology is introduced using cost functions which encourage maximization of separation between in and out of class exemplars. Through a series of empirical investigations of the nature of these functions, we demonstrate that classifier performance is much more dependable than previously the case under the genetic programming paradigm.

Categories and Subject Descriptors

I.2.2 [**Artificial Intelligence**]: Automatic Programming

General Terms

Algorithms, Experimentation, Performance

Keywords

genetic programming, clustering, classification, evaluation

1. INTRODUCTION

One of the purported advantages of Genetic Programming (GP) relative to other supervised learning algorithms is that there is much more freedom in how the fitness (cost) function is expressed. For example, neural networks typically require a cost function that is smooth and therefore differentiable [1], whereas no such requirement exists for GP [3]. To date, however, GP fitness functions do not necessarily build on this freedom in a manner designed to encourage the identification of robust solutions [2]. In this work the design of fitness functions for classification problems is revisited by explicitly considering the contributions made by wrapper and cost function. Specifically, the GP wrapper is used to transform the 'raw' GP output (gp_{out}), a value limited only by the numerical range of the computing platform, to an interval appropriate for distinguishing class (y). Here binary classification problems are considered, thus typical ranges would be [0, 1] or [-1, 1].

GECCO'06, July 8–12, 2006, Seattle, Washington, USA.
ACM 1-59593-186-4/06/0007.

Table 1: Wrapper-Distance Metrics

Label	Wrapper	Error Metric
Hits	$y = \begin{cases} 0\ if\ (gp_{out} \leq 0) \\ 1\ otherwise \end{cases}$	$1 - (d_i \oplus y_i)$
Square	$y = 2 \times (1 + exp(-gp_{out}))^{-1} - 1$	$(d_i - y_i)^2$

In the case of a switching wrapper, the ensuing fitness (cost) function then merely counts the number of misclassified training exemplars (hits). The hypothesis of this work is that such an approach to designing a wrapper-cost function combination results in an inefficient search process, adversely affecting the generalization of the resulting classifier. Instead we suggest to 'bypass' the wrapper (i.e. the wrapper is the identity function) and instead express the problem of GP classification as finding a mapping such that exemplars for each class are mapped to different clusters on the 'raw' GP output. The objective is now to maximize the inter-class separation whilst minimizing the intra-class variance. This corresponds to maximizing the cluster separation distance [4].

2. FITNESS FUNCTIONS AND WRAPPERS

Since Koza popularized Genetic Programming [3], the wrapper for classification problems has frequently taken the form of a switching function. Such a wrapper limits the fitness function to a count of the number of correctly classified exemplars, or hits (a binary distance metric). Conversely, an activation function that is smooth (and monotonically increasing) provides the basis for exemplar errors that increase as the transition point of the activation function is approached, as well as penalizing exemplars that are explicitly misclassified. Moreover, as each error distance is now real valued, we are also free to build a fitness (cost) function that penalizes or weights errors in different ways. In this work we will consider fitness functions based on a squared error penalty in addition to the switching type wrapper. Table 1 summarizes the association between wrapper and error metric. In all cases the fitness function is merely the sum of error taken across all training exemplars for a given wrapper / error distance metric combination.

2.1 A Fitness Function based on Cluster Separation

As indicated above, for a 'robust' classifier or good gener-

alization properties, we expect to bias the classifier toward a mapping that maximizes the distance between points on the raw GP output axis (gp_{out}) representing in and out of class exemplars. Moreover, we also take the view that by minimizing the variance associated with in and out of class exemplars, the resulting mapping should be more sensitive to cases that differ from that established for the majority of cases. In effect we have a requirement for a cluster separation metric [4], thus, inter-cluster separability is maximized, maximizing the distance in mean values for in and out of class exemplars; and intra-cluster variance is minimized, minimizing the variance for the clusters representing in and out of class exemplars.

All the properties are measured with respect to the 'raw' GP output. Given these objectives, we can now state the corresponding distance metric, $D_{0/1}$, for maximization,

$$D_{0/1} = \frac{abs\left(\mu_0 - \mu_1\right)}{\sqrt{\sigma_0^2 + \sigma_1^2}}$$

where, μ_0 and μ_1 are the mean of class 0 and 1 exemplar clusters, as mapped to points on the 1-dimensional GP output axis; and σ_0 and σ_1 are the corresponding estimates for variance.

3. RESULTS

The emphasis of this work is naturally on the contribution of the wrapper, thus any GP model is applicable. We used a fixed length linear representation in the ensuing results. The only difference between experiments is therefore due to the wrapper-fitness function combination, where we consider a total of four cases: hits, square error, cluster separation. In order to present results in a comparative manor, a count of the number of correctly classified exemplars is used (i.e. this is only used post training). Utilization of a percent correctly classified reporting scheme implies that a methodology is required for expressing the wrapper output in terms of a (binary) classification. In the case of the tansig wrapper, labels are associated with which side of the tansig transition point for which the corresponding gp_{out} lies i.e. if $gp_{out} < 0$ then class 0 else class 1. In the case of the cluster separation distance, labels are defined by which cluster mean represents the nearest neighbour to gp_{out} (where the cluster means for class 1 and 0 are established over training data alone).

A total of 3 benchmark classification problems of increasing difficulty were considered: Breast, C-Heart, and Liver. All are taken from the UCI repository. Each dataset was split into training and test partitions. The training partition contained 75% of the exemplars, with the remaining 25% falling in the test set. Partitions were generated using uniform selection, with the constraint that the ratio of in- to out-of-class exemplars of the original dataset be maintained.

Table 2 details the quartile (hits) accuracy of each wrapper-fitness function on the three datasets, where a total of 50 initializations were made per wrapper - dataset combination. It is immediately apparent that the hits based wrapper returns the widest variation in results, with typically high third quartile results, but poor median and first quartiles. We might characterize this in terms of sensitivity to initial population and an emphasis on exploration at the expense of exploitation during credit assignment. The tansig wrapper provides less variation in the results, but at the expense of median and third quartile results. The proposed cluster separation metric provides much more dependable performance in terms of both spread and accuracy. Thus, there is a good correlation between training and test performance as well as ultimate performance.

Table 2: Quartile (1st, median, 3rd) Classification Accuracy

(<- Train) Hits (Test ->)					
Breast	Heart	Liver	Breast	Heart	Liver
344	105	109	114	34	36
423	122	130	145	40	45
510	167	187	167	56	54
Square Error					
202	106	109	70	34	36
291	123	111	90	41	37
356	123	150	122	41	50
Cluster Separation					
383	147	147	126	43	44
442	159	150	145	51	50
462	168	151	154	56	50

4. CONCLUSIONS

A wrapperless methodology has been introduced for providing more meaningful feedback to GP on binary classification problems. The basic motivation has been to encourage GP to find a mapping from the original multidimensional input space to a one dimensional (GP) output space such that (a) clusters represent classes, and (b) the distance or independence between the two clusters is maximized. The cluster separation metric explicitly supported this goal, with the ensuing classifiers demonstrating less sensitivity to the initial parametrisation and very good generalization accuracy. Additional work considers the case of a local membership function in which Gaussian membership functions are used to denote class membership [5]. Moreover, by using such an approach the case of one class classification using GP models is facilitated, where this is appropriate for applications in novelty detection.

5. REFERENCES

[1] S. Haykin Neural Networks: A Comprehensive Foundation. 2nd Ed. Prentice Hall, 1999.

[2] I. Kushchu Genetic Programming and Evolutionary Generalization. IEEE Transactions on Evolutionary Computation. 6(5) 2002 pp 431-442.

[3] J.R. Koza. Genetic Programming as a Means for Programming Computers by Natural Selection. Statistics and Computing. Vol. 4, 1994, pp 87-112.

[4] K. R. Castleman. Digital Image Processing. Prentice Hall, Englewood Cliffs, NJ. USA, 1996.

[5] A. George. Local versus Global Wrapper Functions in Genetic Programming. Master Thesis. Dalhousie University, Faculty of Computer Science. Canada, April 2006.