A Comparative Study of an Evolvability Indicator and a Predictor of Expected Performance for Genetic Programming

Leonardo Trujillo, Yuliana Martínez
Instituto Tecnológico de Tijuana, México
leonardo.trujillo.ttl@gmail.com
ysaraimr@gmail.com

Edgar Galván-López
School of Computer Science and Electronic Engineering
University of Essex, United Kingdom
edgar.galvan@gmail.com

Pierrick Legrand
Université Victor Segalen Bordeaux 2
ALEA Team, INRIA Bordeaux
IMB, UMR CNRS, France
pierrick.legrand@u-bordeaux2.fr

ABSTRACT

An open question within Genetic Programming (GP) is how to characterize problem difficulty. The goal is to develop predictive tools that estimate how difficult a problem is for GP to solve. Here we consider two groups of methods. We call the first group Evolvability Indicators (EI), measures that capture how amendable the fitness landscape is to a GP search. Examples of EIs are Fitness Distance Correlation (FDC) and Negative Slope Coefficient (NSC). The second group are Predictors of Expected Performance (PEP), models that take as input a set of descriptive attributes of a problem and predict the expected performance of GP. This paper compares an EI, the NSC, and a PEP model for a GP classifier. Results suggest that the EI does not correlate with the performance of the GP classifiers. Conversely, the PEP models show a high correlation with GP performance.

Categories and Subject Descriptors

I.2.2 [**Artificial Intelligence**]: Automatic Programming—*program synthesis*

General Terms

Theory, Experimentation, Performance

Keywords

Genetic Programming, Performance prediction, Classification

1. INTRODUCTION

In the tenth anniversary issue of the Genetic Programming and Evolvable Machines Journal, O'Neill et al. [5] and Poli et al. [7] presented a comprehensive overview of the main theoretical and practical research problems within the field of Genetic Programming (GP). Among them, O'Neill et al. [5] described the open issue of *Fitness landscapes and problem difficulty in GP*. In their words, the problem is stated as follows: "*Identifying how hard a particular problem, or problem instance, will be for some GP system, enabling a practitioner to make informed choices before and during application.*"

GECCO'12 Companion, July 7–11, 2012, Philadelphia, PA, USA.
ACM 978-1-4503-1178-6/12/07.

Evolvability Indicators

The local and global structure of the fitness landscape describes the underlying difficulty of a search. Most metaheuristics work under the assumption that the the fitness of a candidate solution is positively correlated with the fitness of (some) of its neighbors. Such a property can be defined as the *evolvability* of a landscape [1, 5], characterizing whether or not the problem is amenable to an evolutionary search. In fact, two of the most successful measures of problem difficulty have focused on describing this property, the Fitness Distance Correlation (FDC) [4, 8] and the Negative Slope Coefficient (NSC) [6]. Thus, by considering all these elements, we refer to such measures as Evolvability Indicators (EIs).

Predictors of Expected Performance

Another way to characterize problem difficulty is to attempt to predict the expected performance that a GP search will achieve on a given problem instance [3, 9, 10]. This is a more pragmatic approach, in which the evolutionary search is taken as a black-box process and the performance of GP on a set of training problems is used to build predictors of the expected performance on unseen problems following a machine learning methodology. In what follows, we refer to such measures of problem difficulty as Predictors of Expected Performance (PEPs). Given a problem p, for which we want to compute a performance prediction, extract a feature vector $\beta = (\beta_1, \beta_2, ...\beta_N)$ of N distinct features that *describe the properties of* p. Then, a PEP P is given by a kernel function K, such that

$$P(\beta) \approx K(\beta) \; . \tag{1}$$

Notice that the form of K is not *a priori* restricted in any way. For instance, [3] use a linear function similar to the one proposed in [2]. However, [9] test more complex linear models and also non-linear models.

Classification with GP

In supervised classification a pattern $\mathbf{x} \in \mathbb{R}^P$ has to be classified as belonging to one of M distinct classes $\omega_1, ..., \omega_M$ using a training set $\mathcal{X}$ of P-dimensional patterns with a known clas-

sification. The idea is to build a mapping $g(\mathbf{x}) : \mathbb{R}^P \rightarrow M$, that assigns each pattern $\mathbf{x}$ to a corresponding class ω_i, where g is derived based on evidence provided by $\mathcal{X}$. GP can be used in different ways to solve such supervised classification tasks. However, in this work we only study the approach proposed in [11], which we denote as Probabilistic GP Classifier (PGPC).

2. COMPARATIVE ANALYSIS

The goal of the experimental work is to evaluate and compare the predictive accuracy of a state-of-the-art EI (NSC) and a PEP model for PGPC. To this end, we generate a set of 300 synthetic classification problems and apply PGPC to each of them, executing 30 independent runs on each problem, computing the average as our baseline estimate of the expected performance of PGPC. Then, we use NSC and a PEP to evaluate their predictive accuracy.

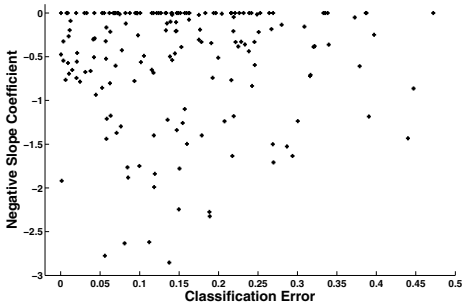

Figure 1: Scatter plot of the average classification error achieved by PGPC on each problem and the corresponding NSC value. Pearson's correlation coefficient ρ for the data is 0.02.

Evolvability for Classification Problems

Here, we reproduce the algorithm described in [6], with the same parameters except for the total amount of sampled individuals M. Whereas in [6] the authors used $M = 40,000$, here we use $M = 10,000$. Figure 1 presents a scatter plot where the horizontal axis is the average classification error and the vertical axis is the evolvability indicator provided by NSC. The results clearly suggest that the NSC does not correlate with performance.

Prediction of Classification Performance

The PEP is derived following the approach described in [9]. Therefore, the feature vector β for each problem is composed of: (1) The geometric mean ratio of the pooled standard deviations to standard deviations of the individual populations; (2) Volume of Overlap Region; (3) Feature efficiency; and (4) The Class Distance Ratio.

A linear PEP model is tested with quadratic terms (LQ-PEP) [9]. The set of classification problems is divided into a training set and a testing set, each with 50% of the problems, and 30 runs are executed with different random partitions.

3. CONCLUDING REMARKS

The key lessons we have learnt during our study are the following. Firstly, we have found that while EIs (in this work we experimentally study the Negative Slope Coefficient) can give a good estimation on the difficulty of the search problem, and also, it is not strongly correlated with expected performance; i.e, i does not correlate with the quality of the solution we can expect to find. Secondly, our results suggest that PEPs achieve a highly accurate prediction of GP performance. However, it is important to remember that both approaches have their particular advantages and shortcomings.

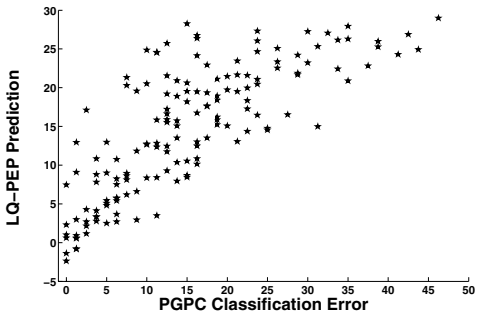

Figure 2: Scatter plot shows the average performance of PGPC (x-axis) and the predicted performance of the PEP model (y-axis). Pearson's correlation coefficient $\rho = 0.77$.

References

[1] L. Altenberg. *The evolution of evolvability in genetic programming*, pages 47–74. MIT Press, Cambridge, MA, USA, 1994.

[2] M. Graff and R. Poli. Practical performance models of algorithms in evolutionary program induction and other domains. *Artif. Intell.*, 174(15):1254–1276, October 2010.

[3] M. Graff and R. Poli. Performance models for evolutionary program induction algorithms based on problem difficulty indicators. In *Proceedings of the 14th European conference on Genetic programming*, EuroGP'11, pages 118–129, Berlin, Heidelberg, 2011. Springer-Verlag.

[4] T. Jones and S. Forrest. Fitness distance correlation as a measure of problem difficulty for genetic algorithms. In *Proceedings of the 6th International Conference on Genetic Algorithms*, pages 184–192, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc.

[5] M. O'Neill, L. Vanneschi, S. Gustafson, and W. Banzhaf. Open issues in genetic programming. *Genetic Programming and Evolvable Machines*, 11(3-4):339–363, September 2010.

[6] R. Poli and L. Vanneschi. Fitness-proportional negative slope coefficient as a hardness measure for genetic algorithms. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, GECCO '07, pages 1335–1342, New York, NY, USA, 2007. ACM.

[7] R. Poli, L. Vanneschi, W. B. Langdon, and N. F. Mcphee. Theoretical results in genetic programming: the next ten years? *Genetic Programming and Evolvable Machines*, 11(3-4):285–320, September 2010.

[8] M. Tomassini, L. Vanneschi, P. Collard, and M. Clergue. A study of fitness distance correlation as a difficulty measure in genetic programming. *Evol. Comput.*, 13(2):213–239, June 2005.

[9] L. Trujillo, Y. Martínez, E. Galván-López, and P. Legrand. Predicting problem difficulty for genetic programming applied to data classification. In *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, GECCO '11, pages 1355–1362, New York, NY, USA, 2011. ACM.

[10] L. Trujillo, Y. Martínez, and P. Melin. Estimating classifier performance with genetic programming. In *Proceedings of the 14th European conference on Genetic programming*, EuroGP'11, pages 274–285, Berlin, Heidelberg, 2011. Springer-Verlag.

[11] M. Zhang and W. Smart. Using gaussian distribution to construct fitness functions in genetic programming for multiclass object classification. *Pattern Recogn. Lett.*, 27(11):1266–1274, August 2006.