Analyzing the influence of Selection on Genetic Programming's Generalization ability in Symbolic Regression:

A comparison of ϵ-Lexicase Selection and Tournament Selection

---

Student: Roman Höhn

Student ID: 2712497

Supervisor: David Wittenberg

03.996.3299 Seminar Information Systems

Chair of Business Administration and Computer Science

Johannes Gutenberg University Mainz

Summerterm 2022

Date of Submission: 2022-05-26

Contents

1 Exposé

1.1 Introduction

Genetic programming (GP), a subfield of evolutionary computation (EC), is a metaheuristic that is used to automatically evolve computer programs by simulating the process of darwinian evolution. The basic principle of GP is to gradually evolve solutions by repeatedly selecting parent solutions from a randomized population of computer programs based on a fitness metric. Then, we use genetic operators on the selected solutions to genereate new offspring candidate solutions. By repeating this process over many generations, GP acts as a guided search for high fitness solutions throughout the decision space.

A unique feature of GP among other evolutionary optimization procedures is the possibility to evolve solutions of variable length, this feature makes GP especially well suited for solving problems in the domain of symbolic regression where little to no a priori knowledge about the optimal form and structure of the target function is available (Paris, Robilliard and Fonlupt, 2004, p. 795). The goal of symbolic regression is to find a mathematical model for an observed set of datapoints (Paris, Robilliard and Fonlupt, 2004, p. 794). Symbolic Regression has been one of the first GP applications and to this day is an actively studied and highly relevant area of research (Poli, Langdon and McPhee, 2008, p. 114).

The overall performance of a GP system can depend strongly on the choice of its underlying operators, one crucial component in this is the parent selection operator. The aim of this research is to study the effect of GP systems ability to generalize based on the usage of the two parent selection operators tournament selection and ϵ-Lexicase selection.

Tournament Selection is a commonly used selection operator in EC and is the most used operator in GP systems (Fang and Li, 2010, p. 181). A parent solution is selected by randomly sampling k individuals from the current population into a tournament pool and then the solution with the highest fitness value from the tournament pool is selected (Fang and Li, 2010, p. 182). In real world applications with numerous test cases, the same principle is usually applied to an aggregate fitness value among all test cases.

Lexicase selection on the other side has been suggested as an alternative to tournament selection that is not based on aggregating fitness scores. It samples n test cases in random order and then eliminates solutions from the selection pool on a per test case basis if they are not performing on an elite level [1](Helmuth, Spector and Matheson, 2015, p. 1). Since regular Lexicase selection has been shown to perform suboptimal on continous-valued optimization problems, a modified variation called ϵ-Lexicase selection has been suggested for symbolic regression applications by La Cava et. al (2016). ϵ-Lexicase selection has shown itself to outperform both standard lexicase selection as well as tournament selection in overall performance while showing only negligible computational overhead (La Cava, Spector and Danai, 2016, p. 747).

An important quality of all supervised machine learning applications, including GP, is the ability to produce models that can generalize learned patterns from the test cases it was

[1] a more in depth description of the algorithm is given in Chapter 2

trained on to new, previously unseen cases. If a model is extensivley optimized on initial training data, overfitting to the specific training data sample can lead to a decrease in the model's ability to generalize it's learned knowledge. Besides the level of noise inside the training data, another supposed key contributers for overfitting in GP systems is the overall complexity/size of candidate solutions that are bred. Larger programs have a higher tendency to specialize on difficult or unusual test cases which lead to a lower ability to generalize on other cases (Paris, Robilliard and Fonlupt, 2004, p. 268).

1.2 Current State of Research

Lexicase selection has been developed specifically for the purpose of solving problems with GP that require the output program to perform optimal on a wide range of different test cases (Helmuth, Spector and Matheson, 2015, p. 1).

More recent research suggests ϵ-Lexicase selection as an alternative selection operator that can improves overall performance of GP system for continous-valued problems in comparison to the traditionally used selection methods tournament selection and standard lexicase selection (La Cava, Spector and Danai, 2016, p. 741). In comparison to other selection operators, populations that are evolved using variations of the lexicase selection operator show a very high degree of genetic diversity which might be a key contributer to the improved performance (Helmuth, Spector and Matheson, 2015, p. 1) (La Cava, Spector and Danai, 2016, p. 745).

The performance increase of ϵ-Lexicase selection for symbolic regression problems has been demonstrated and reported for many benchmark problems which led to widespread adaption of it in symbolic regression applications (La Cava, Spector and Danai, 2016, pp. 744–745).

To the best of my knowledge no published research exists that adresses the question if the usage of ϵ-Lexicase selection has an influence on the output model's generalization ability if compared to traditional tournament selection. Answering this question is the primary motivation of this research project.

1.3 Methodology

To study the effect of selection operator choice on generalization in GP for symbolic regression I selected a well studied dataset about energy efficiency in buildings that is part of the UC Irvine Machine Learning Repository (Dua and Graff, 2017), it contains eight individual building attributes that map to two different outcomes, heating and cooling load, for N=768 cases (Tsanas and Xifara, 2012).

Because of the limited scope of this research, I will focus on the two selection operators tournament and ϵ-lexicase selection with the aim to study their effects on overall generalization ability of the resulting model. Using two otherwise identical GP systems[2], one deploying tournament selection and the other ϵ-lexicase selection, the task is to find a mathematical model that best fits the observed data points from the energy heating dataset as measured by the mean absolute error (MAE).

[2]The specific parameters for both GP system will be detailed in chapter 2

To measure each output model's generalization ability the dataset will be randomly split into a training and testing dataset. Each model will be evolved using only the training dataset and afterwards the MAE will be measured for the previously unseen testing dataset. Each model's MAE on the unseen dataset will be interpreted as the primary measurement for the ability of a model to generalize (Gonçalves, 2016, p. 43). To answer my research questions I will perform a test of statistical significance on the difference of the resulting testing error using a Mann-Whitney U test with Bonferroni correction as suggested by Gonçalves (2016, p.42).

The experiment will be done using the python programming language in conjunction with DEAP, a framework for distributed evolutionary algorithms that implements various tools and algorithms for genetic programming (Fortin *et al.*, 2012).

1.4 Theoretical Foundations

1.4.1 Reuse

In GP, candidate solution are computer programs that consist of terminals and functions which are commonly represented as nodes and leaves inside a tree structure.

References

Dua, D. and Graff, C. (2017) "UCI machine learning repository." University of California, Irvine, School of Information; Computer Sciences. Available at: http://archive.ics.uci.edu/ml.

Fang, Y. and Li, J. (2010) "A review of tournament selection in genetic programming," in Cai, Z. et al. (eds.) *Advances in computation and intelligence*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 181–192.

Fortin, F.-A. *et al.* (2012) "DEAP: Evolutionary algorithms made easy," *Journal of Machine Learning Research*, 13, pp. 2171–2175.

Gonçalves, I. (2016) "An exploration of generalization and overfitting in genetic programming: Standard and geometric semantic approaches," in.

Helmuth, T., Spector, L. and Matheson, J. (2015) "Solving uncompromising problems with lexicase selection," *IEEE Transactions on Evolutionary Computation*, 19(5), pp. 630–643. doi:10.1109/TEVC.2014.2362729.

La Cava, W., Spector, L. and Danai, K. (2016) "Epsilon-lexicase selection for regression," in *Proceedings of the genetic and evolutionary computation conference 2016*. New York, NY, USA: Association for Computing Machinery (GECCO '16), pp. 741–748. doi:10.1145/2908812.2908898.

Paris, G., Robilliard, D. and Fonlupt, C. (2004) "Exploring overfitting in genetic programming," in Liardet, P. et al. (eds.) *Artificial evolution*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 267–277.

Poli, R., Langdon, W.B. and McPhee, N.F. (2008) *A field guide to genetic programming*. Published via `http://lulu.com`; freely available at `http://www.gp-field-guide.org.uk`. Available at: https://digitalcommons.morris.umn.edu/cgi/viewcontent.cgi?article=1001&context=cs_facpubs.

Tsanas, A. and Xifara, A. (2012) "Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools," *Energy and buildings*, 49, pp. 560–567. doi:10.1016/j.enbuild.2012.03.003.