COSC 4P82 Assignment 1

Brett Terpstra
bt19ex@brocku.ca - 692021

February 16, 2024

Contents

1 Introduction

2 Symbolic regression

2.1 Introduction

2.2 Parameter Table

Parameter	Value
Runs	10
Population Size	5000
Generations	50
Training Set	N/A
Testing Set	N/A
Crossover Operator	Subtree Crossover
Mutation Operator	Grow Tree, Max Depth 4
Crossover Rate	0.9 or 1.0*
Mutation Rate	0.1 or 1.0*
Elitism	Best 2 or 0 individuals Survive*
Selection	Fitness Proportionate
Function Set	*, /, +, -, exp, log, sin, cos
Terminal Set	X, Ephemeral Value
Tree Initialization	Half and Half, Max Depth 2-6
Max Tree Depth	17
Raw Fitness	See Fitness Evaluation
Standardized Fitness	= Raw Fitness

*4 Tests were run, 0.9 crossover, 0.9 mutation with 0 elitism and 2 elitism, and 1.0 crossover, 1.0 mutation with 0 elitism and 2 elitism.

2.3 Fitness Evaluation

Fitness is evaluated by taking the absolute value of the predicted y value minus the actual y value. If the difference is less than a user provided (default 1.e15) value cutoff it is added to the fitness value. If the difference value is less than the float epsilon value ($\doteq 0$) the number of hits is incremented. Lower fitness values are preferred.

2.4 Fitness Plots

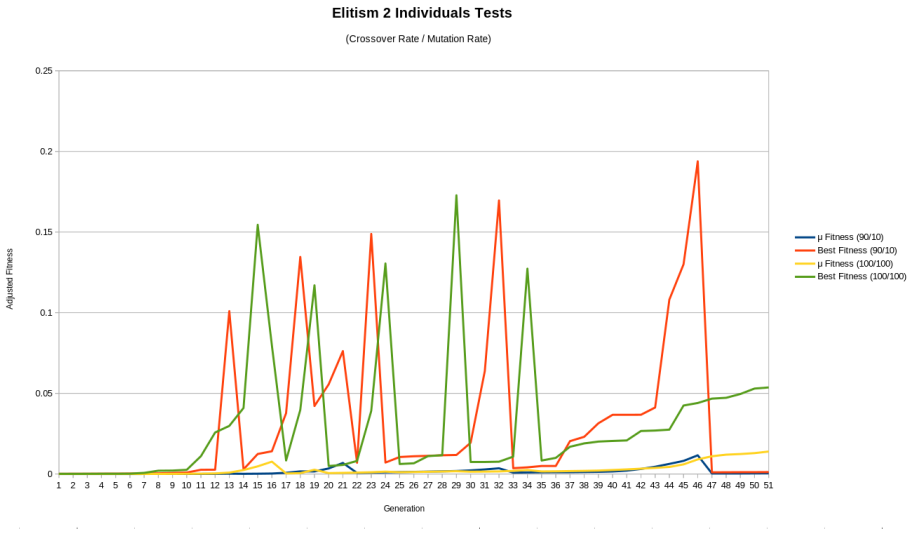

Figure 1: 2 Elites, 10 Runs Averaged

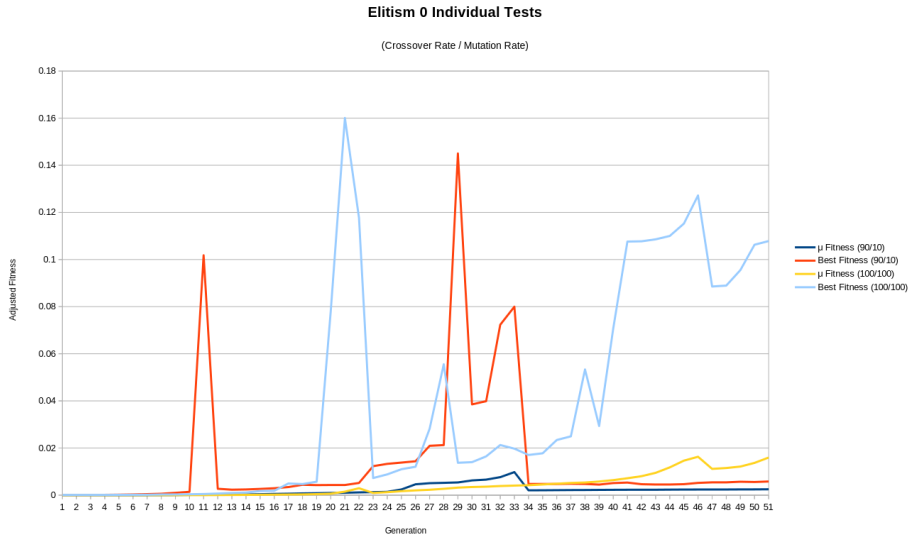

Figure 2: 0 Elites, 10 Runs Averaged

2.5 Analysis And Conclusion

The best average fitness of all the tests was 0.19384 using 0.9 crossover and 0.1 mutation.

3 Rice Classification

3.1 Introduction

3.2 Parameter Table

Parameter	Value
Runs	10
Population Size	5000
Generations	50
Training Set	N/A
Testing Set	N/A
Crossover Operator	Subtree Crossover
Mutation Operator	Grow Tree, Max Depth 4
Crossover Rate	0.9 or 1.0*
Mutation Rate	0.1 or 1.0*
Elitism	Best 2 or 0 individuals Survive*
Selection	Fitness Proportionate
Function Set	*, /, +, -, exp, log, sin, cos
Terminal Set	X, Ephemeral Value
Tree Initialization	Half and Half, Max Depth 2-6
Max Tree Depth	17
Raw Fitness	See Fitness Evaluation
Standardized Fitness	= Raw Fitness

4 Compiling / Executing

This assignment was made for linux using GCC 13.2.0, however any C++17 compliant compiler should work. The minimum GCC version appears to be 8.5, meaning this assignment can be built on sandcastle.

```
cd your_path_to_this_source/
mkdir build
cd build
cmake ../
make -j 32
```

The actual assignment executable is called `Assignment_1` while the automatic run system is called `Assignment_1_RUNNER`. `Assignment_1_RUNNER` has a help menu with options but the defaults will work assuming you run from the build directory and are using part b only. If you want to build for Part A run `cmake -DPART_B=OFF` and run `Assignment_1_RUNNER` with `-b`