Function Regression

Assignment 4 CS472-F16 Due: Mon Nov 28, 2016 at 5pm PT

Points: 200

Reference Files: (all of them can be found in this tar:)

- rand.cpp A simple fast random number generator
- rand.h
- tree.cpp A tree library for GP
- tree.h
- opList.cpp A list of operators in a form for use with the tree library.
- opList.h
- testDataS15A6.tar Six test functions. I will reveal later what the functions were.

Given the amount of time left in the semester it is only reasonable to ask you do something smaller in scope. Write a program using Genetic Programming (GP) techniques to determine a model function f given a set of pairs of real values: $(x, f(x))$. I will supply six functions of various degrees of difficulty. Your program will give me a function that best fits the data.

Your function is composed of an expression using the binary operators: $+, -, *, /$ the unary operator: sin and the nullary operators: x and a constant.

The quality of the answer is determined by the distance between the function values computed from your derived function $f^*(x)$ and the values given in the table $f(x)$:

$$\text{error} \quad = \sum_i (f(x_i) - f^*(x_i))^2$$

Your GP should minimize the error represented as a sum of squares of differences or equivalently maximize the negative of the error.

Your program will be called func and takes no arguments but reads the list of known function value pairs. The format for input is number of pairs of points followed by pairs of real numbers x and $f(x)$ like:

```
3
1.3 666.0
-2  42.0
2.2 -3.141592653
```

Your program will be invoked like:

```
func < f1.dat
```

You get to choose parameters such as population size and maximum number of generations. Your program will have 5 seconds for each function data set.

It will output a single line describing you parameter choices:

```
MaxGen: 17   PopSize: 256   XoverProb: .3
```

You must at least have those parameters first and labeled as they are here. You may add as many other descriptive parameters on the same line to show what parameters you chose. The second line must be the error followed by expression you found. For example:

```
10.476313     ((x / sin((2.48899 * sin(1.16121)))) / ((2.74981 * x) / 0.467946))
```

To help you write this program reasonably quickly feel free to use the tree library above if you want. You can modify it if you like. Do not bring in other GP libraries!! If you use this library be sure to put it in your tar. You can bring in a tree library that does nothing but tree stuff if you like as long as it wasn't designed for GP and will build on the test machine. When in doubt, ask. Please include any files you need to do the compile in your tar.

My approach to solving the problem was to put in a mutate and xover operator into tree.cpp, get my GA code from the code breaking assignment and convert it to use the trees above. My individuals were pointers to trees. The trees object in the supplied library has a cached value at each node. I used this for the values of constants in the expression as well as for caching values. See comments in the code for some of the other details. Ignoring changing the types of things, more than 80% of the code was from other assignments we did or the tree library stuff.

1 Submission

Homework will be submitted as an uncompressed tar file to the homework submission page linked from the main class page. You can submit as many times as you like. The LAST file you submit BEFORE the deadline will be the one graded. For all submissions you will receive email giving you some automated feedback on the unpacking and compiling and running of code and possibly some other things that can be autotested. I will read the results of the runs and the reports you submit.

Have fun.