Stat 243 Class Project: Building a Genetic Algorithm Based Variable Selection Algorithm

Joy Hou, Kevin Li, Greta Olesen

December 10, 2014

Introduction

Genetic Algorithms are search heursitics that mimic the process of Darwinian natural selection. As Givens and Hoeting(2005) stated, candidate solutions to a maximization/minimization problem are considered as biological organisms represented by the genetic code which specifies the model attributes. Genetic algorithms are especially useful in large scale combinatorial and nonlinear optimization when traditional optimization techniques become untractable. Rather than exhaustively searching for the global optimal solution, the Genetic Algorithm utilizes heuristic-based search methods returns a solution close to the global optimal solution.

Variable selecton problems, when presented with a large set of potential predictors, become increasingly computationally expensive. As a result, practitioners and academics have turned to search heuristics such as the Genetic Algorithm.

The basic genetic algorithm is as follows:

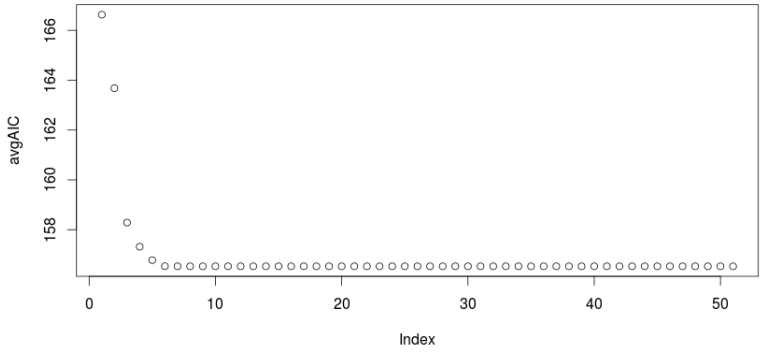
1. Initialize the first generation of models by generating a random population of n chromosomes
2. Evaluate the fitness/performance f(m) of each chromosome/model m in the first generation
3. Create a new population by repeating the following steps until the termination condition is met:
   (a) Select n/2 pairs of parents chromosomes/models from the previous generation according to the fitness (fitter chromosomes have a greater chance to be selected)
   (b) Crossover is carried out with a crossover probability. If crossover was performed, the 2 parent chromosomes are crossed over to produce 2 children. If no crossover was performed, 2 copies of the original chromosomes/models will be kept.

(c) Mutation is carried out with a mutation probability. When mutation is carried out, the new offspring is mutated at each locus.

(d) Accept the children/offsprings as the next population

4. If the terminal condition is satisfied, stop, and return the best solution in the current population. Otherwise, return to step 2

The genetic algorithm by no means returns the optimal solution, but users can generally expect an acceptable solution with rapid convergence that often resembles that presented figure below:



Code Structure

Testing

Contributions