Solving the Traveling Salesman Problem with Genetic Algorithms

Daniel Kearney

March 19, 2018

Contents

0.1 Overview

0.1.1 History of the Genetic Algorithm

0.1.2 The Traveling Salesman Problem

0.2 Genetic Algorithm Function

The genetic algorithm works by mimicking the mechanism of chromosomes and genes in evolutionary biology. Individuals with various solutions to the problem compete with one another, reproducing into next generations, becoming fitter and fitter as the fittest solutions survive.

The algorithm works by initially creating a random ‘generation’ of solutions to seed the iterative process. Each individual solution – called a ‘chromosome’ – has a random solution to the problem at hand.

Once the seed generation is created, the loop begins. The invariant is as follows. First, certain individuals are selected into the next generation, with the fittest solutions being favored. Pairs of these parents are ‘crossed over’, a process where parts of each solution are absorbed into each other. Finally, a bit of randomness is added by ‘mutating’ some of the solutions with random adjustments.

The loop has no natural termination point. The loop can terminate after a certain number of generations, or when a desired fitness is reached.

0.3 Genetic Algorithm Implementation in Python

0.3.1 Selection

0.3.2 Crossing Over

0.3.3 Mutation

0.4 Complexity Analysis

0.5 Results

0.5.1 Small Dataset

0.5.2 Large Dataset

0.5.3 Large Real-world Dataset