Heuristic Search Techniques (Genetic Algorithms)

Thomas R. Harvey

July 14, 2023

This is the first problem sheet on topic four of the 2023 Oxford Summer School, titled "Machine Learning in Mathematics & Theoretical Physics". This tutorial focuses on Genetic Algorithms (GA), and follows on from the first lecture given by Steve Abel.

To assist you in solving the problems, we have provided an example Jupyter notebook showcasing the application of GA to taxicab numbers (https://en.wikipedia.org/wiki/Taxicab_number). You can access the notebook at https://www.tinyurl.com/ga-ox-taxi. Please refer to this notebook for the upcoming problem-solving exercises.

Problem 1

a (In class)

Modify the code above so that GA searches for the minima of a given multi-variable function, where the values of the variables lie in a given interval. This does not have to be the turning point of the function, it can also live on the boundary of the intervals.

b

Modify you answer to the last question such that GA searches for local maxima and minima of a given function. You will likely need to have several genetic runs to find them all.

Problem 2

a (In class)

Modify the code to solve a Knapsack problem (https://en.wikipedia.org/wiki/Knapsack_problem). For this problem, we have a a list of items with weights and values along with a maximum weight that our bag can hold. The aim is to maximise the total value in the bag, while being constrained by the maximum allowed weight.

b

Now, allow for the possibility of multiple bags and ignore the value of the items. Use GA that minimise the total number of bags needed to carry all items.

c

Consider the possibility that a single bag could go missing. Rewrite your fitness function to minimise the value lost if a bag was to go missing, while still using the minimum number of bags to carry all items.

Problem 3

Write your own version of GA in a programming language of your choice.

Problem 4

A magic square (https://en.wikipedia.org/wiki/Magic_square) is a 3x3 grid of unique integers, such that all rows, columns and diagonals sum to the same value. Use GA to search for a magic square of squares. That is, a magic square where all entries are unique square numbers.

This problem is for your own amusement only, as there is no known example of a magic square of squares. In fact there is evidence that no such example can exist.