VGP337 - Neural Network & Machine Learning

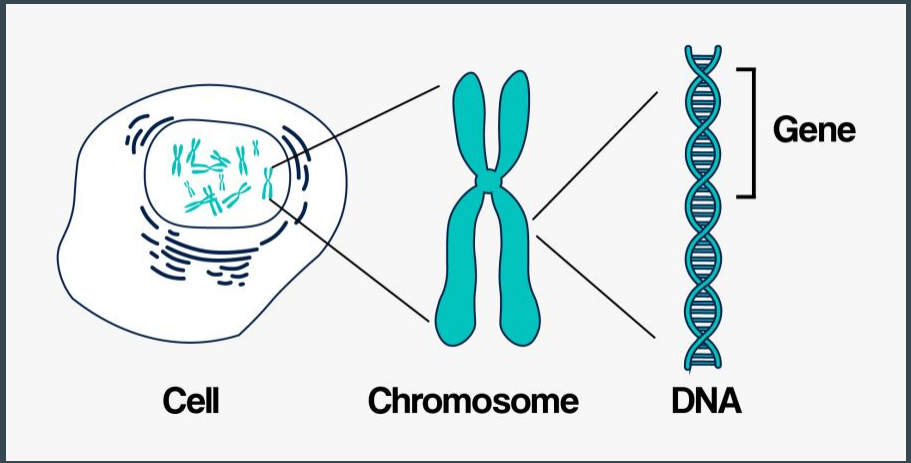
Instructor: Peter Chan

Genetic Algorithms

- A genetic algorithm (GA) is a stochastic search heuristic inspired by the process of natural selection
- ‘Stochastic’ refers to a randomly determined process
- ‘Natural selection’ means ‘Survival of the fittest’ and is one of the basic mechanisms of evolution
- GA is an example of a larger class of methods called Evolutionary Algorithms

The Double Helix of Life

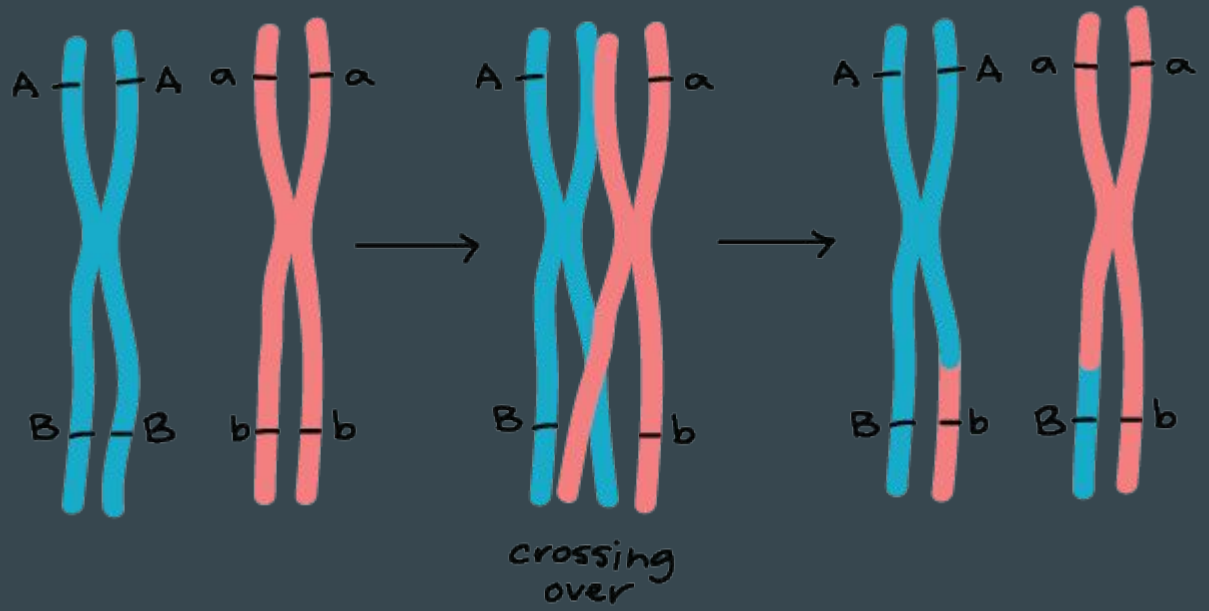
- All living organisms are essentially a large collection of cells
- Each cell contains the same set of strings of DNA, called chromosomes
- Individual chromosomes are built from smaller building blocks, called genes
- The set of chromosomes holds all the information necessary to reproduce that organism



Evolution

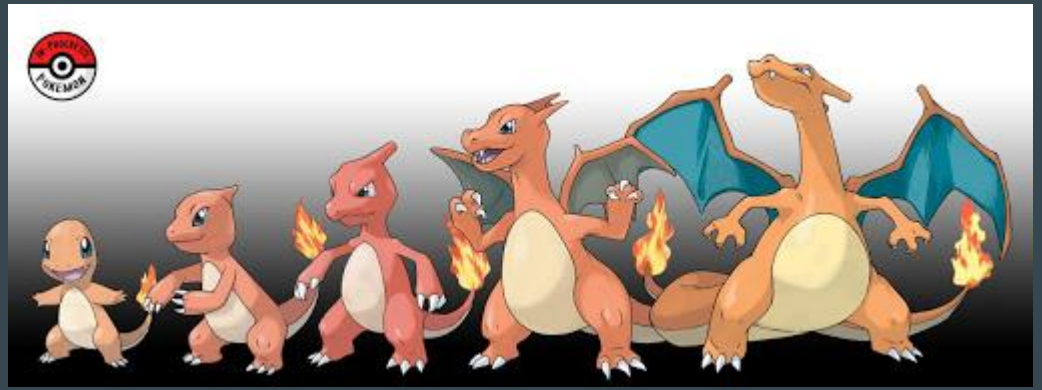
- Creatures evolve over many generations to become more successful at the tasks of survival and reproduction
- This works because when two organisms mate and reproduce, their chromosomes are mixed together to create an entirely new set of chromosomes
- This process is called crossover

Crossover and Recombination



Evolution

- The key is that the more fit the offspring are, the more likely they will go on to reproduce new generations
- Over time only the good genes will be pass down to future generations
- The measure of success of an organism to reproduce is called its fitness



Mutation

Mutation

- Base on this idea, it follows that after many generations, the new population will get genes that are only as good as the initial population
- Therefore, if the initial population does not have good enough genes to survive unforeseen environments, the species will eventually die off
- Fortunately, when genes are passed onto the offspring, there is a very small probability that the genes may be slightly changed
- This gives the organism the ability to develop new traits that are better than previous generations
- This process is known as mutation

Genetic Algorithm

- The way GA works is essentially mimicking evolution
- First, you need to find a way to encode a solution to your problem as a digital chromosome
- Then, you create a starting population of random chromosomes
- Next, compute the fitness of each chromosome in the population
- Apply random selection based on the fitness to crossover/mutate a new generation
- Repeat until a solution is found

Pseudocode

```
generate the initial population
compute fitness for each chromosome
while not converged
    randomly select parents based on fitness
    crossover + mutation
    compute fitness
    add offspring to new generation
```

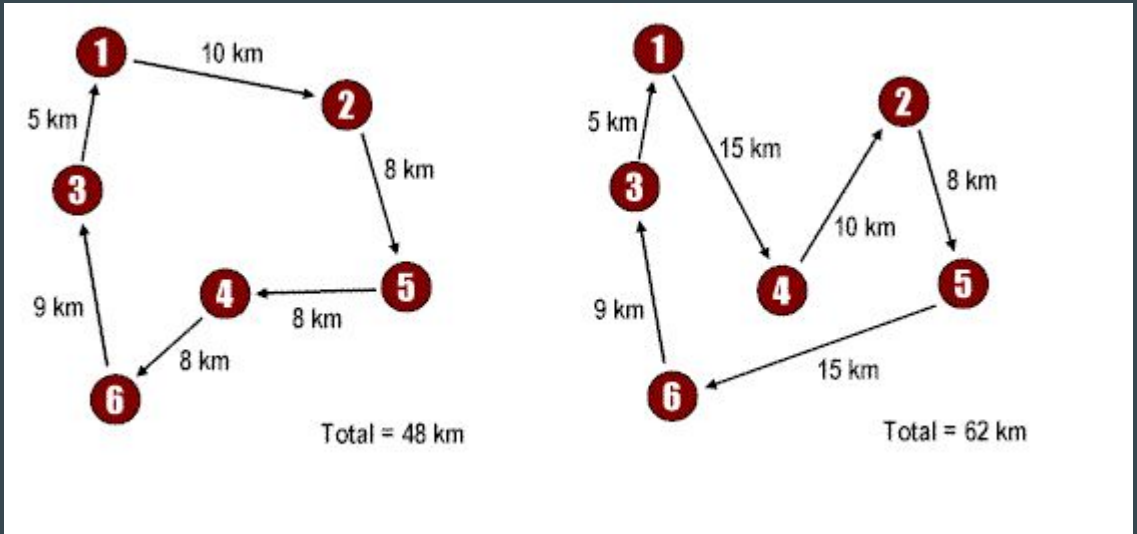
How Powerful is GA?

- Like any algorithm, the success of GA is highly dependent on the type of problems you need to solve
- Specifically, you need to consider how to encode a solution to your problem into a digital sequence, how crossover and mutation will work, and how to evaluate the fitness of a solution
- Some advantages of GA include:
  - It is gradient-free and can deal with a wide variety of problems (linear/non-linear, discrete or continuous, etc)
  - It is highly parallelizable
  - Mostly importantly, you don't need to know how to solve a problem!
- However, GA also does not guarantee a solution

The Traveling Salesman Problem

The Traveling Salesman Problem

Definition: Given a collection of cities, the traveling salesman must determine the shortest route that will enable him to visit each city precisely once and then return back to his starting point.



Objective: Find the order of the cities to visit such that the total distance travelled is minimized!

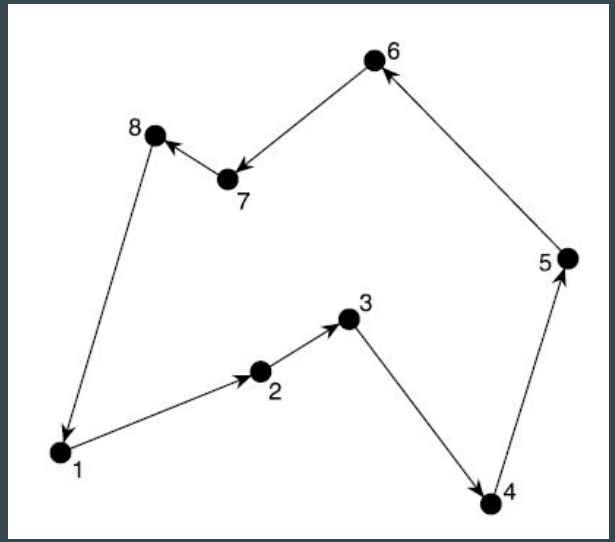
The Traveling Salesman Problem

- The problem may look deceptively simple
- However, TSP is actually an NP-hard problem
- This is a term used in computational complexity theory to classify problems that are suspected to have no polynomial time solution
- Finding the exact solution to such problems may take at least exponential time
- Fortunately, usually we can get reasonably good results using GA

Permutation Encoding

- In order to use GA on TSP, you need to be careful with how you encode a solution
- Consider the figure to the right
- Two possible tours may look like this:

  4, 5, 1, 8, 3, 6, 2, 7
  3, 6, 1, 4, 7, 2, 5, 8



Permutation Encoding

- If these are chosen as parents, after crossover you may get the follow:

$$4, 5, 1, 8, 3, 6, 2, 7 \rightarrow 4, 5, 1, 8, 3, 2, 5, 8$$
$$3, 6, 1, 4, 7, 2, 5, 8 \rightarrow 3, 6, 1, 4, 7, 6, 2, 7$$

- Which are no longer valid routes!

Permutation Encoding

- Instead, one way to solve this is to note the mapping, in this case:

$$6 \rightarrow 2$$
$$2 \rightarrow 5$$
$$7 \rightarrow 8$$

- Next, iterate through each gene and swap based on the mapping

$$4, 5, 1, 8, 3, 6, 2, 7 \rightarrow 4, 5, 1, 8, 3, 2, 6, 7$$
$$\rightarrow 4, 2, 1, 8, 3, 5, 6, 7$$
$$\rightarrow 4, 2, 1, 7, 3, 5, 6, 8$$

- You can apply the same for mutation as well

References

Introduction to Optimization with Genetic Algorithm

Introduction to Genetic Algorithms — Including Example Code

Chapter 3 in Ai Techniques for Game Programming by Mat Buckland

How do Genetic Algorithms Work? (Unity)

Genetic Algorithms

Travelling salesman problem