GENETIC ALGORITHMS

By Joy Reistad

Overview

- What are genetic algorithms?
- History
- Methodology
  - Initialization
  - Selection
  - Crossover
  - Mutation
- Examples

What are genetic algorithms?

- Type of search used in artificial intelligence
- Based on the principles of natural selection
  - Charles Darwin
  - Survival of the fittest
- There are three main principles of natural selection

Main Principles of Natural Selection

- A population produces more offspring than can survive
- Those offspring that survive go on to reproduce
- Variation exists within a population

Ideas behind genetic algorithm

- Most fit members of a population will have the highest chance of being chosen to reproduce, just as in nature
- Overtime solutions will become better

Ideas behind genetic algorithm

- In nature individuals in a population must compete for both resources and mates
- Genes from more fit individuals will propagate through a population
- Successive generations become more suited to their environment

Uses of Genetic Algorithms

- Type of search technique used to find
  - Approximate solutions to optimization and search problems
- Many variations of genetic algorithms

History

- 1950: Alan Turing proposed a type of learning machine that would use the principles of evolution
- ~1954: Nils Aall Barricelli, Alex Frazer, Hans-Joachim Bremermann, and others began computer simulations modeled after evolution

History

- 1960s-1970s: Ingo Rechenberg and Hans-Paul Schwefel began using Searches using principles of evolution as a method for solving optimization problems
- 1970s-1980s computer scientists began applying genetic algorithms to a wide variety of subjects

History

- John Holland is known as the father of genetic algorithms.
- His work during the 1960's and 1970s laid the foundation for genetic algorithms and drew increasing attention to their use.
- In 1975 published Adaptation in Natural and Artificial Systems.

History

- Late 1980's products for desktop computer and industrial use were developed.
- John Koza coined the term genetic programming for the use of genetic algorithms in evolving programs to perform certain tasks

Methodology

- Initialization
- Selection
- Crossover
- Mutation
- Repeat with new generations

until condition is reached

Methodology: Initialization

- A genetic algorithm population begins with a population of n randomly generated individuals.
- Randomly generating the initial population allows your algorithm to encompass the entire range of possible solutions.

Methodology: Initialization

- Individuals of a population make up a generation
- Each individual is an attempted solution to a problem.
- Solutions may not be very good in the beginning, however they become better with each generation

Methodology: Selection

- For each individual in a population, the fitness of the individual is determined using some sort of fitness function.
- Fitness function is
  - Defined for entire range of possible solutions
  - Problem specific
  - Used to measure the quality of the solution

Methodology: Selection

- Each individual in the population is given a certain chance of being selected for reproduction based on their fitness.
- More fit individuals are more likely to be selected

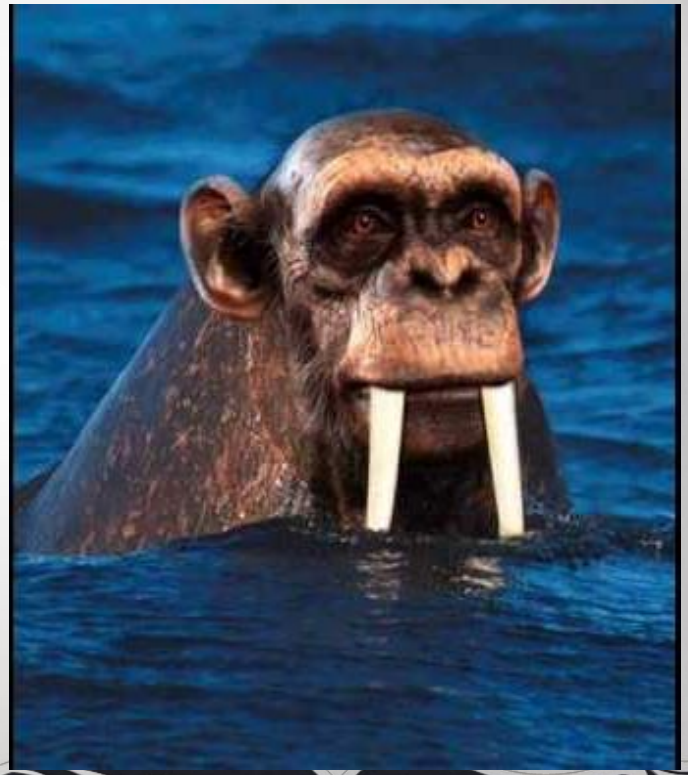
Methodology: Selection

- Once all elements' fitness has been evaluated, a pool of parents is chosen using the probability of each individual being selected
- The probability is found by the following equation

$$P(i\ is\ selected) = \frac{fitness\ of\ i}{\sum_{j=1}^{n}(\text{fitness of j})}$$

Methodology: Crossover

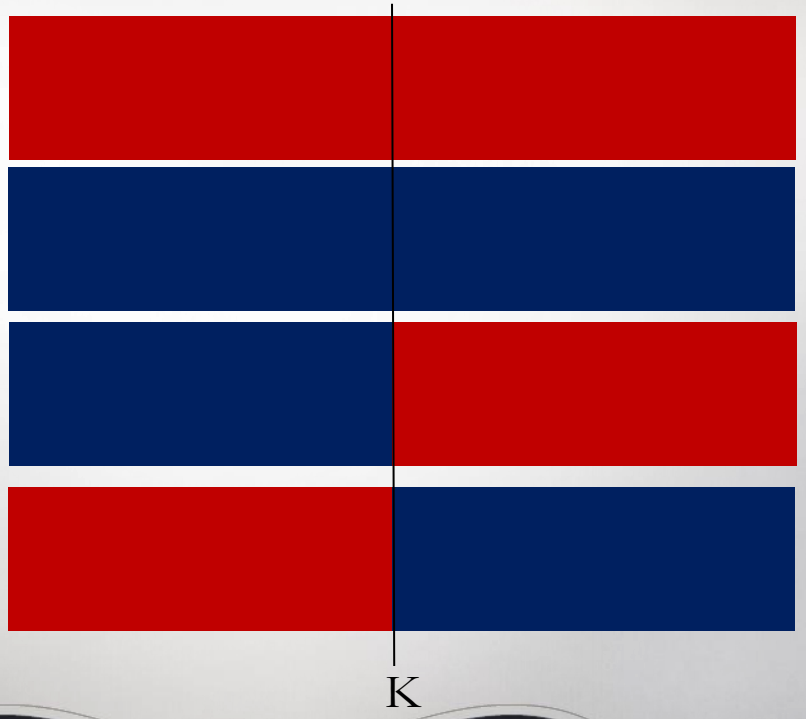
- Individuals in the mating pool are paired up for reproduction events
- A point, k ,between 1 and length - 1 is randomly chosen
- The values of the two parents are exchanged around this point.



Child 1

Child 2

Methodology: Crossover



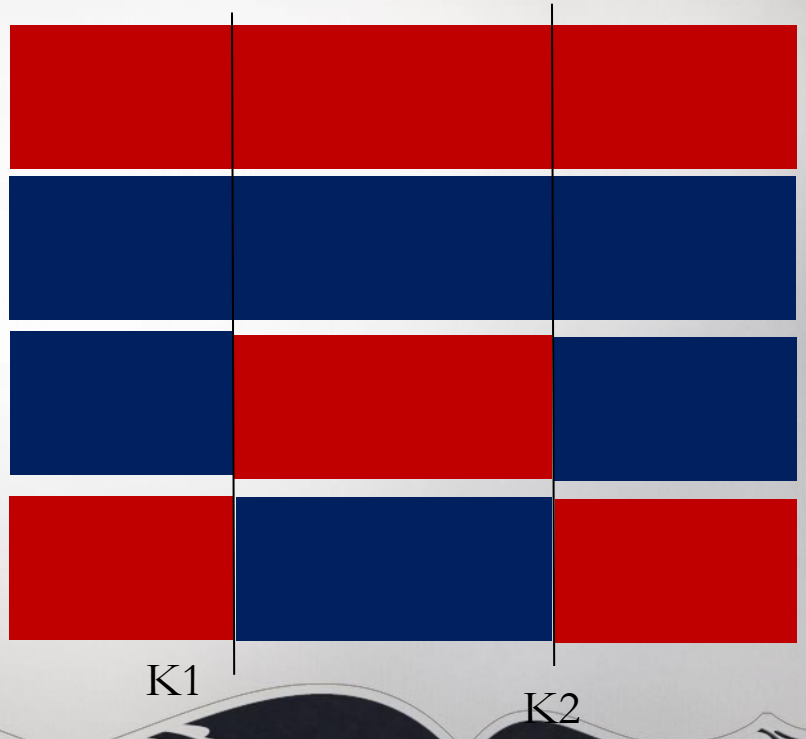
Methodology: Crossover

❖ There are other ways of performing crossovers in genetic algorithms

- Two-point crossover method
- Cut and splice method

Methodology: Crossover

- Two Point Crossover Method
- Two random points are chosen between 1 and length -1



Methodology: Crossover

- Cut and Splice Crossover Method
- Each parent has separate crossover point chosen
- Children get opposite sides of parents genes

Methodology: Crossover

- Some researchers suggest that using more than two parents will provide greater genetic diversity and will generate better solutions
- Most genetic algorithms still use the idea of two parents
- However what if it turned into something horrible

Methodology: Crossover

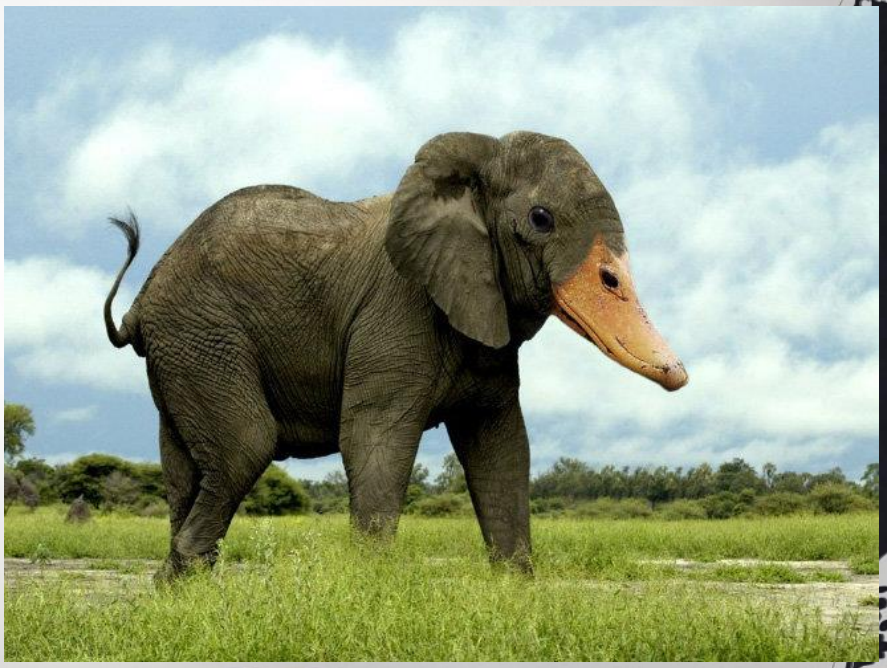
- Dunt Dunt Dun!!!!!!

Methodology: Mutation

- With the creation of offspring comes a very small chance of mutation
- Its purpose is to maintain diversity within the population and inhibit premature convergence
- Keeps genes that may normally be lost and bring new genes into the population

Methodology: Mutation

- A mutation rate that is too low can cause genetic drift
- May cause the genetic diversity of a population to be lost prematurely
- Can cause a genetic algorithm to converge to a less good solution



Methodology: Mutation

- A too high rate of mutation can also cause problems
- Premature convergence
- Loss of good solutions

Methodology: New Generation

- Once a generation of n individuals is formed the process begins again with the selection of a new parent pool
- This cycle continues until end condition is reached

Methodology: End Condition

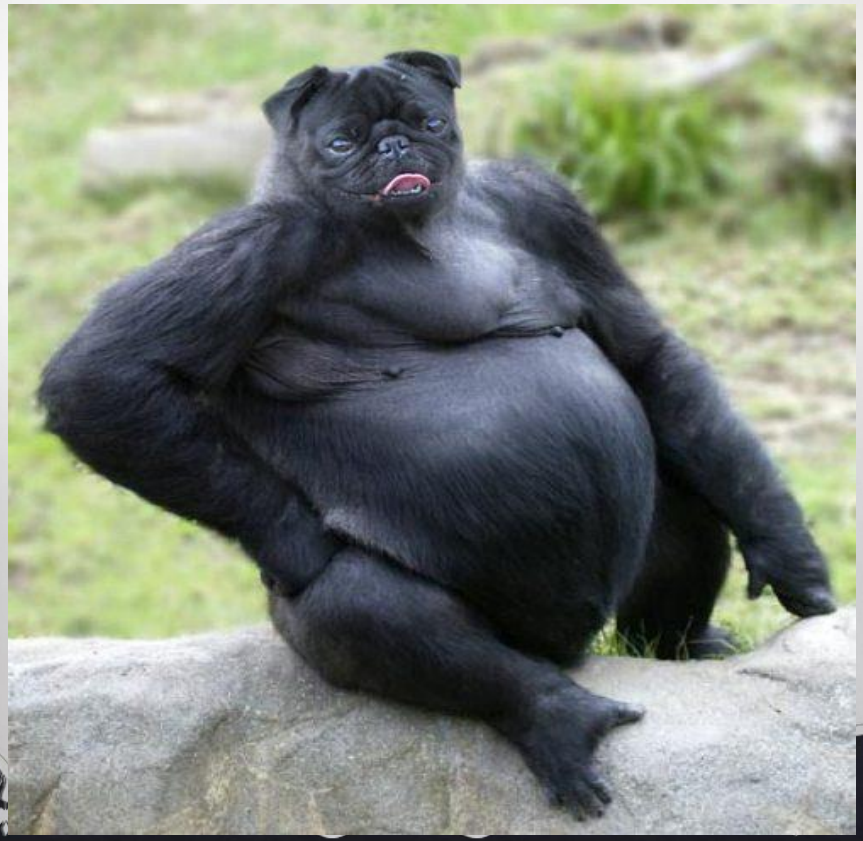
- The end condition can be several different things
- Certain number of reproductive events have been completed
- A satisfactory fitness level has been reached
  - Does not guarantee convergence

Example

Conclusion

- Genetic algorithms are a very useful tool in approximating solutions to optimization and search problems
- Begin with a randomized initial population of n individuals
- Select the best for mating, cross the parents genes to form two children
- Small chance of mutation, keeps variety
- Once new generation of n individuals is formed repeat selection and mating process until end condition is met.

Any Questions



References

- Machine Learning, 1988, Volume 3, Number 2-3, Page 95
- David E. Goldberg, John H. Holland
-
- Genetic algorithm - Wikipedia, the free encyclopedia. (n.d.). Retrieved November 4, 2015, from https://en.wikipedia.org/wiki/Genetic_algorithm
- MITCHELL, M. 1998. An Introduction to Genetic Algorithms. MIT
- Bar-Joseph, Z., & Navlakha, S. (n.d.). *Algorithms in nature* [PDF]. Retrieved from http://www.cs.cmu.edu/~02317/slides/lec_9.pdf
- Introduction to Genetic Algorithms. (n.d.). Retrieved from http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol1/hmw/article1.html
- Lande, R.. (1976). Natural Selection and Random Genetic Drift in Phenotypic Evolution. *Evolution*, *30*(2), 314–334. http://doi.org/10.2307/2407703
- Van Veldhuizen, D. A., & Lamont, G. B. (1998). Multi-objective Evolutionary Algorithm Research: A History and Analysis. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.35.8924&rep=rep1&type=pdf