Particle Swarm Optimization

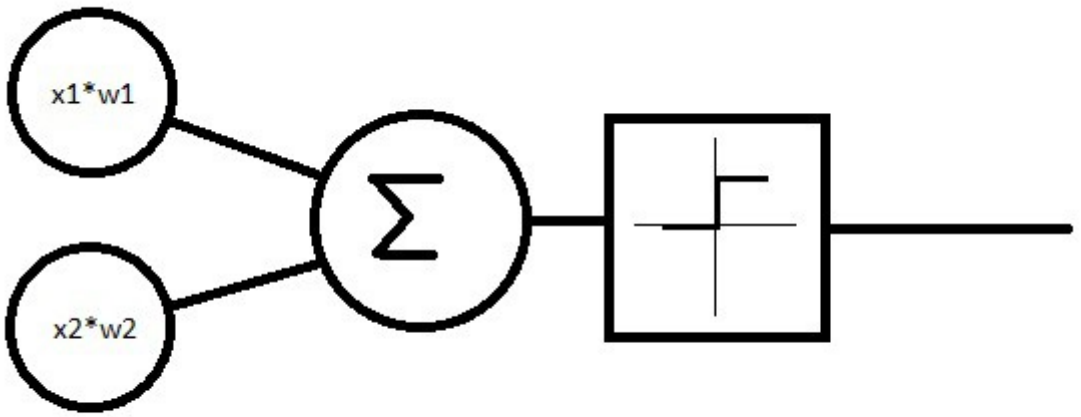
Introduction
Perceptrons
Overview of PSO
Demonstration
Walkthrough

Introduction

- Learning is the process of association.
- Machine learning is the use of machines to produce these associations.
- The mechanisms of association in machine learning are called classifiers.

Perceptrons

Perceptrons consist of inputs, weights, and an activation function.



Perceptrons

f([x1 , ... , xn]) = x1*w1 + ... + xn * wn

if (f([x1 , ... , xn]) > threshold)

    return 1

else

    return 0

Perceptrons

- (Supervised) training of a neural network involves adjusting the network's weights to reduce error in association tasks*.

  *Note to self: Mention XOR problem

- Training methods include, but are not limited to:
  - greedy random
  - hill-climbing
  - delta-rule/backpropagation

Greedy Random

- Randomly generate a new set of weights
- See if it does better on your training data than your best set
- If so, replace the best set with the new set
- Repeat until performance is satisfactory

(Stochastic) Hill-climbing

- Make a random change to your weights
- See if the neural net does better
- If so, keep the change.
- Otherwise, revert the change.
- Repeat until performance is satisfactory

Delta Rule/Backpropagation

- Compare the desired output of the neural net to the actual output.
- Use the calculated error to adjust the neural net's weights.
- Repeat until performance is satisfactory.

Note: The delta rule is a simple method useful only for single layer feedforward neural nets, such as perceptrons. Backpropagation is a more complex, generalized version useful for multi-layer feedforward neural nets.

Particle Swarm Optimization

- Particle swarm optimization (or PSO) is a method for optimizing a set of numerical parameters.
- It originally developed as simulations of social behavior in animals.

  https://www.youtube.com/watch?v=QbUPfMXXQIY

Particle Swarm Optimization

- In brief, swarming algorithms rely on the emergent behavior resulting from many individuals behaving in accordance with a simple set of rules.
- Algorithms which simulate the flocking behavior of birds need only three rules: separation, alignment, and cohesion.

Particle Swarm Optimization

- Each particle consists of a copy of a classifier. Each parameter has random value and a random velocity between 0 and 1 associated with it.
- We also allocate space to store the personal best set of parameters and the score associated with it.

Particle Swarm Optimization

- The particles exist in an n-dimensional space, where each dimension is one parameter in the classifier.
- Consequently, the position of each particle in this space is each given by the value of each parameter.

Particle Swarm Optimization

- Determine if any particle has a better set of parameters than the current best set.
- If so, store that set.
- For each particle
  - For each parameter
    - Adjust the parameter by the velocity
    - Adjust the velocities by the product of:
      - Some constant
      - Some random value
      - The difference between the particle's parameters and the current best set of parameters
- Repeat until performance is satisfactory.

https://github.com/raynorj/