Neural Networks

Project Schedule

Introduction Paper due 9/25/2014

Demonstration of the Equivalence of Various Perceptrons Networks done 2014-09-23

Overview of Neural Networks

A neural network is a system in which many similar functions, called *neurons*, are composed together in order to classify inputs, perform some computation, or approximate some function. Neural networks are used today in various machine-learning applications such as handwriting- or speech-recognition, and have several interesting mathematical properties.

Definition of a Neuron

A neuron is a function $f(x_1, x_2, \ldots, x_n) = \phi(\Sigma_{i=1}^{n} w_i x_i)$, where $\{w_1, w_2, \ldots w_n\}$ is a set of weights, with each weight w_i corresponding to an input x_i, and where ϕ is the activation function that determines the output of the neuron based on the sum of the products of the weights and inputs. For the sake of brevity we will also notate the weight and input vectors as $\vec{w}$ and $\vec{x}$, respectively.

A neuron could be also thought of as a partial application of ϕ and $\vec{w}$ over the factory function $F(\phi, \vec{w}, \vec{x})$.

There are two popular activation functions, the perceptron and the sigmoid function. The perceptron can be defined as the function $\phi_P(x) = \begin{cases} 0 & : x + b \leq 0 \\ 1 & : x + b > 0 \end{cases}$, where b is a bias that is specific to the neuron.

The sigmoid function can be defined as $\sigma(x) = \frac{1}{1+e^{-x}}$. Though bearing some resemblance to the perceptron, the sigmoid function has the advantage of being both smooth and differentiable. These properties make training a neural network much easier.

Demonstration of the Equivalence of Various Perceptron Networks to Certain Boolean Logic Functions

Statement and Proof of Correctness for a Training Algorithm for a Perceptron

Statement and Proof of the Turing-completeness of Neural Networks

Statement and Proof of the Universality Theorem