Learning Neural Networks

Alice Gao

Lecture 11

Readings: R & N 18.7

Based on work by K. Leyton-Brown, K. Larson, and P. van Beek

Outline

Learning Goals

By the end of the lecture, you should be able to

- Represent simple logical functions (e.g. AND, OR, and NOT) using a perceptron.
- Describe the function represented by a simple multi-layer feed-forward neural network.
- Trace the execution of the back-propagation algorithm.

CQ: What does this perceptron compute?

CQ: What does h_1 compute?

$$h_1 = f(x_1 + x_2 - 0.5)$$

where $f(x) = 1$ if $x > 0$ and $f(x) = 0$ if $x \leq 0$.

(A) $(x_1 \vee x_2)$

(B) $(x_1 \wedge x_2)$

(C) $(\neg(x_1 \vee x_2))$

(D) $(\neg(x_1 \wedge x_2))$

CQ: What does this perceptron compute?

CQ: What does h_2 compute?

$$h_2 = f(-x_1 - x_2 + 1.5)$$

where $f(x) = 1$ if $x > 0$ and $f(x) = 0$ if $x \leq 0$.

(A) $(x_1 \vee x_2)$

(B) $(x_1 \wedge x_2)$

(C) $(\neg(x_1 \vee x_2))$

(D) $(\neg(x_1 \wedge x_2))$

CQ: What does this perceptron compute?

CQ: What does o_1 compute?

$$o_1 = f(h_1 + h_2 - 1.5)$$

where $f(x) = 1$ if $x > 0$ and $f(x) = 0$ if $x \leq 0$.

(A) $(h_1 \vee h_2)$

(B) $(h_1 \wedge h_2)$

(C) $(\neg(h_1 \vee h_2))$

(D) $(\neg(h_1 \wedge h_2))$

CQ: Gradient descent

CQ: Suppose that our goal is to minimize a given function $f(x)$. After calculating the gradient (derivative) of f with respect to x at a point $x = x_0$, how should we change the value of x_0 to minimize f?

(A) If the gradient is positive, we should decrease x. If the gradient is negative, we should increase x.

(B) If the gradient is positive, we should increase x. If the gradient is negative, we should decrease x.

(C) We should always increase x regardless of the sign of the gradient.

(D) We should always decrease x regardless of the sign of the gradient.

Revisiting the Learning Goals

By the end of the lecture, you should be able to

- Represent simple logical functions (e.g. AND, OR, and NOT) using a perceptron.
- Describe the function represented by a simple multi-layer feed-forward neural network.
- Trace the execution of the back-propagation algorithm.