Mathematics for Deep Learning

Partial Derivatives for Backpropagation

Inputs and Output

Let x denote an input vector. For the moment we'll leave its dimension and how we index its components unspecified.

Similarly, let y denote the output vector corresponding to the input vector, x. Its dimension and how we index its components will for a moment also remain unspecified, but note that the dimensions of x and y are generally completely different.

Neural Nets and Layers

We will conceive of a neural net as consisting of L layers. Each layer takes an input vector and produces an output vector. Then

$$y = x^{(L)} \leftarrow x^{(L-1)} \leftarrow x^{(L-2)} \leftarrow \ldots \leftarrow x^{(2)} \leftarrow x^{(1)} \leftarrow x^{(0)} = x$$

The parenthesized superscripts denote layer numbers, not exponents. A net with one layer ($L = 1$) has two vectors, consisting of one input vector, one output vector, and no vectors in between. An $L = 4$ neural net has five vectors $x^{(0)}$ to $x^{(4)}$.

We still haven't said what the dimensions of the vectors are or how we index their components, but again note that the dimensions of the various $x^{(l)},\ l = 0,\ \ldots,\ l$ are in general unrelated.

Functions to Represent Layer Operations

The lth function representing the layer l operation transforms $x^{(l)}$ to $x^{(l+1)}$. For clarity — not compactness — we'll write this operation as $x^{(l+1)} = f^{(l+1\leftarrow l)}\left(x^{(l)}\right)$.

In a neural net with L layers, there are L functions f. The one that operates on the input vector is $f^{(1\leftarrow 0)}\left(x^{(l)}\right)$, and the one producing the output vector is $f^{(L\leftarrow L-1)}\left(x^{(L-1)}\right)$.

Parameters