Deep Learning

Ash Bellett

Contents

1 Feed-Forward Network

Input: X
Rows are samples, columns are features.
Single row is fed into network.
Input into network: x

Network weights: W
Weights of layer i: w_i

Output of layer i: o_i
The initial output o_0 is the input:

$$o_0 = x$$

The prediction $\hat{y}$ is the output o_n of the final layer n:

$$\hat{y} = o_n$$

Activation a_i of each layer:

$$a_i = w_i \cdot o_{i-1}$$

Relationship between input and output o_i of layer i:

$$o_i = g(a_i)$$

$$o_i = g(w_i \cdot o_{i-1})$$

Activation function can be sigmoid, hyperbolic tangent or rectified linear unit.

The network loss E is a function of the predicted and desired output where N is ideally a large number of samples:

$$E = \frac{1}{N} \sum_{i=0}^{N} f(\hat{y_i}, y_i)$$

Loss function f can be mean squared error or cross entropy.

Update network weights using gradient descent:

$$w_{t+1} = w_t - \alpha \frac{\partial E_t}{\partial w_t}$$

The derivative of the error function with respect to the network weights $\frac{\partial E}{\partial w}$ expressed in terms of the network layers:

$$\frac{\partial E}{\partial w} = \frac{1}{n} \sum_{i=0}^{n} \frac{\partial E}{\partial w_i}$$

The derivative of the error function with respect to the network weights $\frac{\partial E}{\partial w}$ can be broken into other partial derivatives using the chain rule:

$$\frac{\partial E}{\partial w} = \frac{\partial E}{\partial a}\frac{\partial a}{\partial w}$$

The derivative of the activations with respect to the weights is:

$$\frac{\partial a_i}{\partial w_i} = \frac{\partial}{\partial w_i}(w_i \cdot o_{i-1}) = o_{i-1}$$

The derivative of the loss with respect to the activations $\frac{\partial E}{\partial a}$ is dependent on which loss function and activation function is used. Using the sigmoid function and mean squared error:

$$E = \frac{1}{2}(\hat{y} - y)^2 = \frac{1}{2}(o_n - y)^2$$

$$g(x) = \sigma(x) = \frac{1}{1 + e^{-x}}$$

For the output layer, the activations are not passed through the activation function:

$$o_n = a_n$$

$$\frac{\partial E}{\partial a_n} = \frac{\partial}{\partial a_n}\left(\frac{1}{2}(a_n - y)^2\right) = a_n - y = \hat{y} - y$$

Combining these derivatives:

$$\frac{\partial E}{\partial w_n} = (\hat{y} - y)o_{n-1}$$

For the hidden layers, the activations are passed through the activation function:

$$o_i = \sigma(a_i)$$

The derivative of the loss with respect to the activations can be broken into other partial derivatives using the chain rule:

$$\frac{\partial E}{\partial a_i} = \frac{\partial E}{\partial a_{i+1}}\frac{\partial a_{i+1}}{\partial a_i}$$

$$a_{i+1} = w_{i+1} \cdot o_i = w_{i+1} \cdot \sigma(a_i)$$

$$\frac{\partial a_{i+1}}{\partial a_i} = \frac{\partial}{\partial a_i}\left(w_{i+1} \cdot \sigma(a_i)\right) = w_{i+1} \cdot \left(\sigma(a_i)\left(1 - \sigma(a_i)\right)\right) = w_{i+1}\left(o_i(1 - o_i)\right)$$

$$\frac{\partial E}{\partial a_i} = \frac{\partial E}{\partial a_{i+1}} \cdot w_{i+1}\left(o_i(1 - o_i)\right)$$

$$\frac{\partial E}{\partial w_i} = \frac{\partial E}{\partial a_{i+1}} \cdot w_{i+1}\left(o_i(1 - o_i)\right)o_{i-1}$$