Lecture Note - 07: Neural Network

Dihui Lai

March 17, 2020

Contents

1 A Single Neuron

An artificial neuron is the basic computing unit in an artificial neural network. There are different way to define a neuron. The most common one is shown in Figure 1.

Input: A neuron receives multiple inputs x_1, x_2, ..., x_n. The signals are summed up after modulated by a set of weights w_1, w_2, ... w_n. Let us denote the weighted sum z

$$z = \sum_{i=1}^{n} w_i x_i \tag{1}$$

Usually a biased term w_0 is added to the summation and we have

$$z = \sum_{i=1}^{n} w_i x_i + w_0 \tag{2}$$

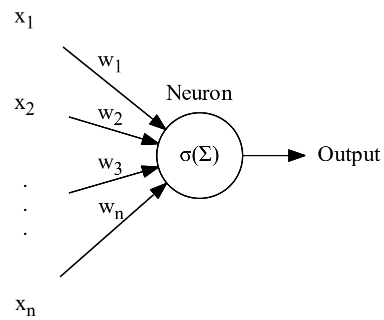

Figure 1: Schema: a single neuron with activation function σ

Output: The weighted sum is further transferred via an activation function σ and becomes the final output of the neuron

$$a = \sigma(z) = \sigma(\sum_{i=1}^{n} w_i x_i + w_0) \tag{3}$$

Activation: The activation function can of different types. Below is a list of common activation functions. Almost all activation functions have an S-shape except for the ReLu function.

Name	Definition
Step Function	$\sigma(z) = \begin{cases} 0 & \text{for } z < 0 \\ 1 & \text{for } z \geq 0 \end{cases}$
Logistic or sigmoid	$\sigma(z) = \frac{1}{1+e^{-z}}$
hyperbolic tangent	$\sigma(z) = \frac{(e^{z}-e^{-z})}{(e^{z}+e^{-z})}$
ReLU	$\sigma(z) = \begin{cases} 0 & \text{for } z \leq 0 \\ z & \text{for } z > 0 \end{cases}$

2 Neural Network: Forward Path

Each neuron at layer l receives inputs from all neuron from the previous layer $l-1$,

$$z_k^l = \sum_j w_{kj}^{l-1} a_j^{l-1} \tag{4}$$

Here, a_j^{l-1} is the input from jth neuron from $l-1$ layer. The neurons transfer the input signal z_k^l via a transfer function σ and send as input to to the next layer

$$a_k^l = \sigma(z_k^l) \tag{5}$$

Inserting equation (4) to (5), we have

$$z_k^l = \sum_j w_{kj}^{l-1} \sigma(z_j^{l-1}) = \sum_j w_{kj}^{l-1} a_j^{l-1} \tag{6}$$

3 The Derivatives of Neural Network

In order to calculate the neural network properly we need to understand a few derivatives.

- The partial derivatives of z_k^l against z_j^{l-1}

$$\frac{\partial z_k^l}{\partial z_j^{l-1}} = w_{kj}^{l-1} \sigma'(z_j^{l-1}) \tag{7}$$

- The partial derivatives of z_k^l against a weight element w_{kj}^{l-1}

$$\frac{\partial z_k^l}{\partial w_{kj}^{l-1}} = \sigma(z_j^{l-1}) \tag{8}$$

- Given a function of $f(z_j^{l-1})$, we can explicitly express it in terms of z^ls,

$$f(z_j^{l-1}) = f(z_1^l(z_j^{l-1}), z_2^l(z_j^{l-1}) ... z_m^l(z_j^{l-1}) ...)$$

  its derivative w.r.t z_j^{l-1} is

$$\frac{\partial f}{\partial z_j^{l-1}} = \sum_m \frac{\partial f}{\partial z_m^l} \frac{\partial z_m^l}{\partial z_j^{l-1}} = \sum_m \frac{\partial f}{\partial z_m^l} w_{mj}^{l-1} \sigma'(z_j^{l-1}) \tag{9}$$

4 SGD and Backpropagation

Consider neural network that has N layers, the cost function is dependent on all the zs of neurons in all layers

$$C\left(\vec{z}^N(\vec{z}^{N-1}(...\vec{z}^l(\vec{z}^{l-1})...)...\vec{z}^1)\right) \tag{10}$$

1. Update the weights by changing it along the gradient to reduce the cost function
2. do one data point at a time

$$w_{kj}^L \leftarrow w_{kj}^L - \eta \frac{\partial C}{\partial w_{kj}^L}, L = 1, 2, ...l-1, l, ...N \tag{11}$$

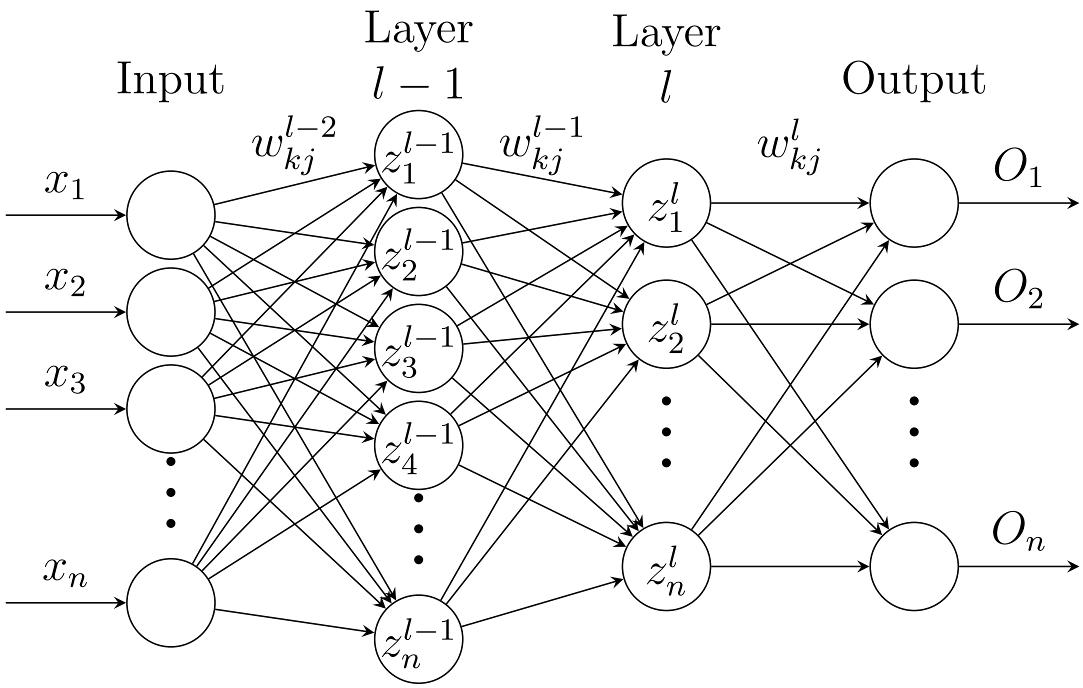

Figure 2: A neural network of multiple layer structures. The hidden layers before layer $l-1$ and after layer l are not shown.

Now let us figure out what $\frac{\partial C}{\partial w^L_{kj}}$ is. Without losing generality, let us consider how the derivative looks like when we consider how signals passes from layer $l-1$ to layer l. By using the chain rule, we have

$$\frac{\partial C(...(z^l_k(w^{l-1}_{kj}, ...)))}{\partial w^{l-1}_{kj}} = \frac{\partial C}{\partial z^l_k}\frac{\partial z^l_k}{\partial w^{l-1}_{kj}} \tag{12}$$

There are two terms that we need to understand from the R.H.S. of equation (8).

- The second term is simply the output from jth neuron in layer $l-1$ since we have from equation (6)

$$\frac{\partial z^l_k}{\partial w^{l-1}_{kj}} = \sigma(z^{l-1}_j) = a^{l-1}_j \tag{13}$$

- Using equation (8), the first term becomes

$$\frac{\partial C}{\partial z^l_k} = \sum_m \frac{\partial C}{\partial z^{l+1}_m} w^l_{mk}\sigma'(z^l_k)$$

  Use notation $\delta^l_k = \frac{\partial C}{\partial z^l_k}$ for short, we have

$$\delta^l_k = \sum_m \delta^{l+1}_m w^l_{mk}\sigma'(z^l_k) \tag{14}$$

Finally, we get our iteration methods for calculating the gradient of the cost function i.e.

$$
\begin{cases}
\dfrac{\partial C}{\partial w_{kj}^{l-1}} = \delta_k^l a_j^{l-1} \\
\delta_k^l = \displaystyle\sum_m \delta_m^{l+1} w_{mk}^l \sigma'(z_k^l)
\end{cases}
\tag{15}
$$