Derivation on Backpropagation for Neural Networks

Jianxiong Xiao

February 15, 2014

Neuron: $\mathbf{z}^{(l+1)} = \mathbf{W}^{(l)}\mathbf{x}^{(l)} + \mathbf{b}^{(l)}$.
Activation funciton: $\mathbf{x}^{(l)} = f\left(\mathbf{z}^{(l)}\right)$.
Let's assume we only have one training point to a N-layer Neural Network $\left\langle \mathbf{x}^{(1)}, \mathbf{y}^{(N)} \right\rangle$ (For more than one data points, just sum their individual gradients).
Objective function for training is

$$\min_{\mathbf{W},\mathbf{b}} J\left(\mathbf{W},\mathbf{b}\right) = \min_{\mathbf{W},\mathbf{b}} \frac{1}{2}\left(\mathbf{y}^{(\mathbf{N})} - \mathbf{x}^{(N)}\right)^2$$

We optimize this by gradient descent (a.k.a. Back Propagation):
For each training iteration, given all current values of $\mathbf{W}^{(l)}, b^{(l)}$, we want to compute update rules

$$\mathbf{W}^{(l)} \leftarrow \mathbf{W}^{(l)} - \alpha \frac{\partial}{\partial \mathbf{W}^{(l)}} J\left(\mathbf{W},\mathbf{b}\right)$$

$$\mathbf{b}^{(l)} \leftarrow \mathbf{b}^{(l)} - \alpha \frac{\partial}{\partial \mathbf{b}^{(l)}} J\left(\mathbf{W},\mathbf{b}\right)$$

So the whole problem for training a NN is to compute gradient $\frac{\partial}{\partial \mathbf{W}^{(l)}} J\left(\mathbf{W},\mathbf{b}\right)$ and $\frac{\partial}{\partial b^{(l)}} J\left(\mathbf{W},\mathbf{b}\right)$ using the current $\mathbf{W}$ and $\mathbf{b}$.
So for the last layer (the N-th layer), we have

$$\frac{\partial J\left(\mathbf{W},\mathbf{b}\right)}{\partial \mathbf{W}^{(N-1)}} = \frac{\partial \frac{1}{2}\left(\mathbf{y}^{(N)} - \mathbf{x}^{(N)}\right)^2}{\partial \mathbf{W}^{(N-1)}} = \frac{\partial \mathbf{x}^{(N)}}{\partial \mathbf{W}^{(N-1)}} \frac{\partial \frac{1}{2}\left(\mathbf{y}^{(N)} - \mathbf{x}^{(N)}\right)^2}{\partial \mathbf{x}^{(N)}} = \frac{\partial \mathbf{z}^{(N)}}{\partial \mathbf{W}^{(N-1)}} \frac{\partial f\left(\mathbf{z}^{(N)}\right)}{\partial \mathbf{z}^{(N)}} \left(\mathbf{x}^{(N)} - \mathbf{y}^{(N)}\right) = \mathbf{x}^{(N)} \frac{\partial f\left(\mathbf{z}^{(N)}\right)}{\partial \mathbf{z}^{(N)}} \left(\mathbf{x}^{(N)} - \mathbf{y}^{(N)}\right) = \mathbf{x}^{(N)} f^{'}\left(\mathbf{z}^{(N)}\right)\left(\mathbf{x}^{(N)} - \mathbf{y}^{(N)}\right)$$

$$\frac{\partial J\left(\mathbf{W},\mathbf{b}\right)}{\partial \mathbf{b}^{(N-1)}} = \frac{\partial \frac{1}{2}\left(\mathbf{y}^{(N)} - \mathbf{x}^{(N)}\right)^2}{\partial \mathbf{b}^{(N-1)}} = \frac{\partial \mathbf{x}^{(N)}}{\partial \mathbf{b}^{(N-1)}} \frac{\partial \frac{1}{2}\left(\mathbf{y}^{(N)} - \mathbf{x}^{(N)}\right)^2}{\partial \mathbf{x}^{(N)}} = \frac{\partial \mathbf{z}^{(N)}}{\partial \mathbf{b}^{(N-1)}} \frac{\partial f\left(\mathbf{z}^{(N)}\right)}{\partial \mathbf{z}^{(N)}} \left(\mathbf{x}^{(N)} - \mathbf{y}^{(N)}\right) = \frac{\partial f\left(\mathbf{z}^{(N)}\right)}{\partial \mathbf{z}^{(N)}} \left(\mathbf{x}^{(N)} - \mathbf{y}^{(N)}\right) = f^{'}\left(\mathbf{z}^{(N)}\right)\left(\mathbf{x}^{(N)} - \mathbf{y}^{(N)}\right)$$

For the second last layer (the $(N-1)$-th layer), a change in $\mathbf{z}^{(l)}$ can affect all the nodes which are connected to $\mathbf{x}^{(l)}$'s output

$$\begin{aligned}
\frac{\partial J\left(\mathbf{W},\mathbf{b}\right)}{\partial \mathbf{b}^{(N-2)}} &= \frac{\partial \frac{1}{2}\left(\mathbf{y}^{(N)} - \mathbf{x}^{(N)}\right)^2}{\partial \mathbf{b}^{(N-2)}} = \frac{\partial \mathbf{x}^{(N)}}{\partial \mathbf{b}^{(N-2)}} \frac{\partial \frac{1}{2}\left(\mathbf{y}^{(N)} - \mathbf{x}^{(N)}\right)^2}{\partial \mathbf{x}^{(N)}} = \frac{\partial \mathbf{z}^{(N)}}{\partial \mathbf{b}^{(N-2)}} \frac{\partial f\left(\mathbf{z}^{(N)}\right)}{\partial \mathbf{z}^{(N)}} \left(\mathbf{x}^{(N)} - \mathbf{y}^{(N)}\right) = \frac{\partial \mathbf{z}^{(N)}}{\partial \mathbf{b}^{(N-2)}} f^{'}\left(\mathbf{z}^{(N)}\right)\left(\mathbf{x}^{(N)} - \mathbf{y}^{(N)}\right) \\
&= \frac{\partial (\mathbf{W}^{(N-1)}\mathbf{x}^{(N-1)} + \mathbf{b}^{(N-1)})}{\partial \mathbf{b}^{(N-2)}} f^{'}\left(\mathbf{z}^{(N)}\right)\left(\mathbf{x}^{(N)} - \mathbf{y}^{(N)}\right) = \frac{\partial \mathbf{x}^{(N-1)}}{\partial \mathbf{b}^{(N-2)}} \frac{\partial (\mathbf{W}^{(N-1)}\mathbf{x}^{(N-1)} + \mathbf{b}^{(N-1)})}{\partial \mathbf{x}^{(N-1)}} f^{'}\left(\mathbf{z}^{(N)}\right)\left(\mathbf{x}^{(N)} - \mathbf{y}^{(N)}\right) \\
&= \frac{\partial \mathbf{z}^{(N-1)}}{\partial \mathbf{b}^{(N-2)}} \frac{\partial \mathbf{x}^{(N-1)}}{\partial \mathbf{z}^{(N-1)}} \mathbf{W}^{(N-1)} f^{'}\left(\mathbf{z}^{(N)}\right)\left(\mathbf{x}^{(N)} - \mathbf{y}^{(N)}\right) = \frac{\partial \left(\mathbf{W}^{(N-2)}\mathbf{x}^{(N-2)} + \mathbf{b}^{(N-2)}\right)}{\partial \mathbf{b}^{(N-2)}} f^{'}\left(\mathbf{z}^{(N-1)}\right)\mathbf{W}^{(N-1)} f^{'}\left(\mathbf{z}^{(N)}\right)\left(\mathbf{x}^{(N)} - \mathbf{y}^{(N)}\right) \\
&= f^{'}\left(\mathbf{z}^{(N-1)}\right)\mathbf{W}^{(N-1)} f^{'}\left(\mathbf{z}^{(N)}\right)\left(\mathbf{x}^{(N)} - \mathbf{y}^{(N)}\right)
\end{aligned}$$

Same for $\mathbf{W}^{(N-2)}$,

$$\frac{\partial J\left(\mathbf{W},\mathbf{b}\right)}{\partial \mathbf{W}^{(N-2)}} = \frac{\partial \left(\mathbf{W}^{(N-2)}\mathbf{x}^{(N-2)} + \mathbf{b}^{(N-2)}\right)}{\partial \mathbf{W}^{(N-2)}} f^{'}\left(\mathbf{z}^{(N-1)}\right)\mathbf{W}^{(N-1)} f^{'}\left(\mathbf{z}^{(N)}\right)\left(\mathbf{x}^{(N)} - \mathbf{y}^{(N)}\right) = \mathbf{x}^{(N-2)} f^{'}\left(\mathbf{z}^{(N-1)}\right)\mathbf{W}^{(N-1)} f^{'}\left(\mathbf{z}^{(N)}\right)\left(\mathbf{x}^{(N)} - \mathbf{y}^{(N)}\right)$$

To simplify notation, we define the error vector (sensitivity)

$$\mathbf{e}^{N} = f^{'}\left(\mathbf{z}^{(N)}\right)\left(\mathbf{x}^{(N)} - \mathbf{y}^{(N)}\right) \quad \text{and} \quad \mathbf{e}^{l} = f^{'}\left(\mathbf{z}^{(l)}\right)\mathbf{W}^{(l)}\mathbf{e}^{l+1}$$

This means that in order to compute the sensitivity for a unit, we should first sum over the next layer's sensitivies corresponding to units that are connected to this unit, and multiply each of those connections by the associated weights defined at next layer. We then multiply this quantity by the derivative of the activation function evaluated at the current layer's pre-activation inputs $\mathbf{z}^{(l)}$.

Then

$$\frac{\partial J\left(\mathbf{W},\mathbf{b}\right)}{\partial \mathbf{W}^{(N-1)}} = \mathbf{x}^{(N)}\mathbf{e}^{N-1+1} \quad \text{and} \quad \frac{\partial J\left(\mathbf{W},\mathbf{b}\right)}{\partial \mathbf{b}^{(N-1)}} = \mathbf{e}^{N-1+1}$$

and

$$\frac{\partial J\left(\mathbf{W},\mathbf{b}\right)}{\partial \mathbf{W}^{(N-2)}} = \mathbf{x}^{(N-2)}\mathbf{e}^{N-2+1} \quad \text{and} \quad \frac{\partial J\left(\mathbf{W},\mathbf{b}\right)}{\partial \mathbf{b}^{(N-2)}} = \mathbf{e}^{N-2+1}$$

therefore the generic rule for computing the gradient is

$$\frac{\partial J\left(\mathbf{W},\mathbf{b}\right)}{\partial \mathbf{W}^{(l-1)}} = \mathbf{x}^{(l)}\mathbf{e}^{l} \quad \text{and} \quad \frac{\partial J\left(\mathbf{W},\mathbf{b}\right)}{\partial \mathbf{b}^{(l-1)}} = \mathbf{e}^{l}$$

Therefore, as Sebastian Seung pointed out, the "error" $\mathbf{e}^{l}$ which we propagate backwards through the network can be thought of as "sensitivities" of each unit with respect to perturbations of the bias $\mathbf{b}$.