1 Back-propagation Derivation

This section derives the matrix form for the backpropagation algorithm. The dimensions of all vectors are given in Tables 1 and 2 below.

1.0.1 $\frac{dL}{dW_o}$, Gradient of the loss with respect to the outer layer weights. Here i use the notation consistent with lecture i.e. $\frac{dL}{dW_o} = \frac{dE}{dW^{(L)}}$

Loss of single training example is given be $e = -\sum_{k=1}^{K} y_k \log p_k$. Here, y_k is the true label and $p_k = x_k^{(L)}$ is the predicted output, then the derivative of the loss for a single training example, with respect to a single weight is given by

$$\frac{\partial e}{\partial w_{i,j}^{(L)}} = \frac{\partial e}{\partial s_j^{(L)}} \frac{\partial s_j^{(L)}}{\partial w_{i,j}^{(L)}} \tag{1}$$

Since, $s_j^{(L)} = b_j^{(L)} + \sum_{k=1}^{d^{(l-1)}} w_{k,j}^{(L)} x_k^{(L-1)}$

$$\frac{\partial s_j^{(L)}}{\partial w_{i,j}^{(L)}} = x_i^{(L-1)} \tag{2}$$

Now to find, $\frac{\partial e}{\partial s_j^{(L)}}$ we re-write e as follows (Keeping in mind that $x_k^{(L)}$ is simply the output from the soft-max function)

$$\begin{aligned}
e &= -\sum_{k=1}^{K} y_k \log(x_k^{(L)}) \\
&= -\sum_{k=1}^{K} y_k \log\left(\frac{e^{s_k^{(L)}}}{\sum_{k=1}^{k} e^{s_k^{(L)}}}\right) \\
&= -\sum_{k=1}^{K} y_k \left(s_k^{(L)} - \log(\sum_{k=1}^{K} e^{s_k^{(L)}})\right) \\
&= -\sum_{k=1}^{K} y_k s_k^{(L)} + \sum_{k=1}^{K} y_k \log(\sum_{k=1}^{K} e^{s_k^{(L)}})
\end{aligned} \tag{3}$$

Taking the derivative

$$\begin{aligned}
\frac{\partial e}{\partial s_j^{(L)}} &= -y_j + \sum_{k=1}^{K} y_k \frac{e^{s_j^{(L)}}}{\sum_{k=1}^{K} e^{s_k^{(L)}}} \\
&= -y_j + \sum_{k=1}^{K} y_k x_j^{(L)}
\end{aligned}$$

Since, $\sum_{k=1}^{K} y_k = 1$

$$\frac{\partial e}{\partial s_j^{(L)}} = x_j^{(L)} - y_j \tag{4}$$

Substituting (2) and (4) into (1), and introducing δ into notation, we get

$$\frac{\partial e}{\partial w_{i,j}^{(L)}} = x_i^{(L-1)} \delta_j^{(L)} \tag{5}$$

$$\delta_j^{(L)} = (x_j^{(L)} - y_j) \tag{6}$$

In matrix form this is for a single training example :

$$\frac{\partial e}{\partial \underline{W}^{(L)}} = \underline{x}^{(L-1)} \underline{\delta}^{(L)\top} \tag{7}$$

$$\underline{\delta}^{(L)} = (\underline{x}^{(L)} - \underline{y}) \tag{8}$$

For N training examples it becomes

$$\frac{\partial E}{\partial \underline{W}^{(L)}} = \frac{1}{N} \underline{X}^{(L-1)^\top} \underline{\Delta}^{(L)} \tag{9}$$

$$\underline{\Delta}^{(L)} = (\underline{X}^{(L)} - \underline{Y}) \tag{10}$$

The vector/matrix dimensions in the in this section, as well as all ensuing sections, are given below:

$\underline{x}^{(l)}$	$\underline{y}$	$\underline{X}^{(l)}$	$\underline{Y}$	$\underline{W}^{(l)}$	$\underline{s}^{(l)}$	$\underline{\delta}^{(l)}$	$\underline{S}^{(l)}$	$\underline{\Delta}^{(l)}$	$\vec{1}$
$d^{(l)} \times 1$	$d^{(l)} \times 1$	$N \times d^{(l)}$	$N \times K$	$d^{(l-1)} \times d^{(l)}$	$d^{(l)} \times 1$	$d^{(l)} \times 1$	$N \times d^{(l)}$	$N \times d^{(l)}$	1x N

Table 1: Dimensions of vectors

1.0.2 $\frac{dL}{dB_o}$, Gradient of the loss with respect to outer layer bias. To be consistent with lecture notation $\frac{dL}{dB_o} = \frac{dE}{dB^{(L)}}$

The loss with respect to the bias, for a single training example, in the L^{th} layer can be given by

$$\frac{\partial e}{\partial b_j^{(L)}} = \frac{\partial e}{\partial s_j^{(L)}} \frac{\partial s_j^{(L)}}{\partial b_j^{(L)}} \tag{11}$$

Since

$$s_j^{(L)} = b_j^{(L)} + \sum_{i=1}^{d^{(L-1)}} x_i^{(L-1)} w_{i,j}^{(L)}$$

$$\frac{\partial s_j^{(L)}}{\partial b_j^{(L)}} = 1 \tag{12}$$

and we already know from (4) that $\frac{\partial e}{\partial s_j^{(L)}} = x_j^{(L)} - y_j$ then (11) becomes

$$\frac{\partial e}{\partial b_j^{(L)}} = \delta_j^{(L)} \tag{13}$$

$$\delta_j^{(L)} = x_j^{(L)} - y_j \tag{14}$$

In matrix form, for a single training example, this is

$$\frac{\partial e}{\partial \underline{B}^{(L)}} = \underline{\delta}^{(L)} \tag{15}$$

$$\underline{\delta}^{(L)} = \underline{x}^{(L)} - \underline{y} \tag{16}$$

For N training examples it is given by the sum across axis=0 of (12) (i.e. sum all elements of columns), which is achieved with a matrix multiply by $\vec{1}$

$$\frac{\partial E}{\partial \underline{B}^{(L)}} = \frac{1}{N} (\vec{1}\, \underline{\Delta}^{(L)}) \tag{17}$$

$$\underline{\Delta}^{(L)} = \underline{X}^{(L)} - \underline{Y} \tag{18}$$

Where $\frac{\partial E}{\partial \underline{B}^{(L)}}$ is a $[1 \times d^{(L)}]$ column vector.

1.0.3 $\frac{dL}{dW_h}$, the gradeint with respect to a hidden layer weights. Once again, to stay consistent with lecture notation $\frac{dL}{dW_h} = \frac{dE}{dW^{(l)}}$

The loss with respect to a weight in a hidden layer i.e. $w_{i,j}^{(l)}$, for a single training example can be given by

$$\frac{\partial e}{\partial w_{i,j}^{(l)}} = \frac{\partial e}{\partial s_j^{(l)}} \frac{\partial s_j^{(l)}}{\partial w_{i,j}^{(l)}} \tag{19}$$

Since,

$$s_j^{(l)} = b_j^{(l)} + \sum_{i=1}^{d^{(l-1)}} x_i^{(l-1)} w_{i,j}^{(l)} \tag{20}$$

$$\frac{\partial s_j^{(l)}}{\partial w_{i,j}^{(l)}} = x_i^{(l-1)} \tag{21}$$

Now, we turn to finding $\frac{\partial e}{\partial s_j^{(l)}}$

$$\frac{\partial e}{\partial s_j^{(l)}} = \frac{\partial e}{\partial x_j^{(l)}} \frac{\partial x_j^{(l)}}{\partial s_j^{(l)}} \tag{22}$$

where the second term on the right is the derivative of the activation function (here, ReLU)

$$\frac{\partial x_j^{(l)}}{\partial s_j^{(l)}} = \theta'(s_j^{(l)}) = \begin{cases} 0 & s_j^{(l)} \leq 0 \\ 1 & s_j^{(l)} > 0 \end{cases} \tag{23}$$

Now, we turn our attention to the left term on the right side of equation (22)

$$\frac{\partial e}{\partial x_j^{(l)}} = \sum_{k=1}^{d^{(l+1)}} \frac{\partial e}{\partial s_k^{(l+1)}} \frac{\partial s_k^{(l+1)}}{\partial x_j^{(l)}} \tag{24}$$

Since

$$s_k^{(l+1)} = b_k^{(l+1)} + \sum_{i=1}^{d^{(l)}} x_i^{(l)} w_{i,k}^{(l+1)} \tag{25}$$

$$\frac{\partial s_k^{(l+1)}}{\partial x_j^{(l)}} = w_{j,k}^{(l+1)} \tag{26}$$

Finally, subbing (26) into (24) and this result into (22) and also subbing (23) into (22) we get

$$\frac{\partial e}{\partial s_j^{(l)}} = \theta'(s_j^{(l)}) \sum_{k=1}^{d^{(l+1)}} \frac{\partial e}{\partial s_k^{(l+1)}} \cdot w_{j,k}^{(l+1)} \tag{27}$$

Subbing this quantity and (21) both into (19) and we have defined the derivative of the error with respect to an arbitrary hidden layer weight

$$\frac{\partial e}{\partial w_{i,j}^{(l)}} = x_i^{(l-1)} \theta'(s_j^{(l)}) \sum_{k=1}^{d^{(l+1)}} \frac{\partial e}{\partial s_k^{(l+1)}} \cdot w_{j,k}^{(l+1)} \tag{28}$$

We can re-write this as

$$\frac{\partial e}{\partial w_{i,j}^{(l)}} = x_i^{(l-1)} \delta_j^{(l)} \tag{29}$$

$$\delta_j^{(l)} = \theta'(s_j^{(l)}) \sum_{k=1}^{d^{(l+1)}} \delta_k^{(l+1)} w_{j,k}^{(l+1)} \tag{30}$$

In matrix form, for a single training example, this becomes

$$\frac{\partial e}{\partial \underline{W}^{(l)}} = \underline{x}^{(l-1)} \underline{\delta}^{(l)\top} \tag{31}$$

$$\underline{\delta}^{(l)} = \theta'(\underline{s}^{(l)}) \otimes \underline{W}^{l+1} \underline{\delta}^{(l+1)} \tag{32}$$

Where $\otimes$ denotes elementwise multiplication, and $\theta'(\cdot)$ is applied on every element in vector $\underline{s}^l$ which is a $d^{(l)} \times 1$ vector (See table 1 for dimensions of vectors). Now to put this in matrix form, for multiple examples we extend the previous equation to be

$$\frac{\partial E}{\partial \underline{W}^{(l)}} = \frac{1}{N} \underline{X}^{(l-1)\top} \underline{\Delta}^{(l)} \tag{33}$$

$$\underline{\Delta}^{(l)} = [\theta'(\underline{S}^{(l)})^\top \otimes \underline{W}^{l+1} \underline{\Delta}^{(l+1)\top}]^\top \tag{34}$$

1.0.4 $\frac{dL}{dB_h}$, the gradient with respect to the hidden layer bias. Once again, to stay consistent with lecture notation $\frac{dL}{dB_h} = \frac{dE}{dB^{(l)}}$

$$\frac{\partial e}{\partial b_j^{(l)}} = \frac{\partial e}{\partial s_j^{(l)}} \frac{\partial s_j^{(l)}}{\partial b_j^{(l)}} \tag{35}$$

Since $s_j^{(l)} = b_j^{(l)} + \sum_{i=1}^{d^{(l-1)}} w_{i,j}^{(l)} x_i^{(l-1)}$

$$\frac{\partial s_j^{(l)}}{\partial b_j^{(l)}} = 1$$

From equation (27) we know

$$\frac{\partial e}{\partial s_j^{(l)}} = \delta_j^{(l)} = \theta'(s_j^{(l)}) \sum_{k=1}^{d^{(l+1)}} \delta_k^{(l+1)} \cdot w_{j,k}^{(l+1)} \tag{36}$$

Therefore our final equations are

$$\frac{\partial e}{\partial b_j^{(l)}} = \delta_j^{(l)} \tag{37}$$

$$\delta_j^{(l)} = \theta'(s_j^{(l)}) \sum_{k=1}^{d^{(l+1)}} \delta_k^{(l+1)} \cdot w_{j,k}^{(l+1)} \tag{38}$$

In matrix form, for a single training example, this becomes

$$\frac{\partial e}{\partial \underline{B}^{(l)}} = \underline{\delta}^{(l)} \tag{39}$$

$$\underline{\delta}^{(l)} = \theta'(\underline{s}^{(l)}) \otimes \underline{W}^{(l+1)} \underline{\delta}^{(l+1)} \tag{40}$$

Finally, for multiple training examples, we find this quantity and sum across its columns (i.e. sum across axis = 0)

$$\frac{\partial E}{\partial \underline{B}^{(l)}} = \frac{1}{N} \vec{1}\, \underline{\Delta}^{(l)} \tag{41}$$

$$\underline{\Delta}^{(l)} = [\theta'(\underline{S}^{(l)\top}) \otimes \underline{W}^{(l+1)} \underline{\Delta}^{(l+1)\top}]^\top \tag{42}$$