Backpropagation notes by deeplizard

Definitions and Notation

We define

L = number of layers in the network
Layers are indexed as $l = 1, 2, \cdots, L-1, L$
Nodes in a given layer l are indexed as $j = 0, 1, \cdots, n-1$
Nodes in layer $l-1$ are indexed as $k = 0, 1, \cdots, n-1$

y_j = the desired value of node j in the output layer L for a single training sample

C_0 = loss function of the network for a single training sample (sum of squared errors)

$w_{jk}^{(l)}$ = the weight of the connection that connects node k in layer $l-1$ to node j in layer l

$w_j^{(l)}$ = the vector that contains all weights connected to node j in layer l by each node in layer $l-1$

$z_j^{(l)}$ = the input for node j in layer l

$g^{(l)}$ = the activation function used for layer l

$a_j^{(l)}$ = the activation output of node j in layer l

Observations

Loss C_0

Observe that the expression

$$\left(a_j^{(L)} - y_j\right)^2$$

is the squared difference of the activation output and the desired output for node j in the output layer L.

This can be interpreted as the loss for node j in layer L.

Therefore, to calculate the total loss, we should sum this squared difference for each node j in the output layer L.

This is expressed as

$$C_0 = \sum_{j=0}^{n-1}\left(a_j^{(L)} - y_j\right)^2.$$

Input $z_j^{(l)}$

We know that the input for node j in layer l is the weighted sum of the activation outputs from the previous layer $l-1$.

An individual term from the sum looks like this:

$$w_{jk}^{(l)} a_k^{(l-1)}$$

So, the input for a given node j in layer l is expressed as

$$z_j^{(l)} = \sum_{k=0}^{n-1} w_{jk}^{(l)} a_k^{(l-1)}.$$

Activation Output $a_j^{(l)}$

We know that the activation output of a given node j in layer l is the result of passing the input, $z_j^{(l)}$, to whatever activation function we choose to use $g^{(l)}$.

Therefore, the activation output of node j in layer l is expressed as

$$a_j^{(l)} = g^{(l)}\left(z_j^{(l)}\right).$$

Expressing C_0 as a composition of functions

Recall the definition of C_0,

$$C_0 = \sum_{j=0}^{n-1} \left(a_j^{(L)} - y_j\right)^2.$$

So the loss of a single node j in the output layer L can be expressed as

$$C_{0_j} = \left(a_j^{(L)} - y_j\right)^2.$$

We see that C_{0_j} is a function of the activation output of node j in layer L.

So, we can express C_{0_j} as a function of $a_j^{(L)}$ as

$$C_{0_j}\left(a_j^{(L)}\right).$$

Observe from the definition of C_{0_j} that C_{0_j} also depends on y_j. Since y_j is a constant, we only observe C_{0_j} as a function of $a_j^{(L)}$, and y_j as a parameter that helps define this function.

The activation output of node j in the output layer L is a function of the input for node j.

From an earlier observation, we know we can express this as

$$a_j^{(L)} = g^{(L)}\left(z_j^{(L)}\right).$$

The input for node j is a function of all the weights connected to node j.

So, we can express $z_j^{(L)}$ as a function of $w_j^{(L)}$ as

$$z_j^{(L)}\left(w_j^{(L)}\right).$$

Therefore,

$$C_{0_j} = C_{0_j}\left(a_j^{(L)}\left(z_j^{(L)}\left(w_j^{(L)}\right)\right)\right).$$

So, we can see that C_0 is a composition of functions.

We know that

$$C_0 = \sum_{j=0}^{n-1} C_{0_j},$$

so using the same logic, we observe that the total loss of the network for a single input is also a composition of functions.

This is useful in order to understand how to differentiate C_0.

To differentiate a composition of functions, we use the chain rule.

Calculations

Derivative of the loss with respect to weights

Let's look at a single weight that connects node *2* in layer $L-1$ to node *1* in layer L.

This weight is denoted as

$$w_{12}^{(L)}.$$

The derivative of the loss C_0 with respect to this particular weight $w_{12}^{(L)}$ is denoted as

$$\frac{\partial C_0}{\partial w_{12}^{(L)}}.$$

Since C_0 depends on $a_1^{(L)}$, and $a_1^{(L)}$ depends on $z_1^{(L)}$, and $z_1^{(L)}$ depends on $w_{12}^{(L)}$, then the chain rule tells us that to differentiate C_0 with respect to $w_{12}^{(L)}$ we take the product of the derivatives of the composed function.

This is expressed as

$$\frac{\partial C_0}{\partial w_{12}^{(L)}} = \left(\frac{\partial C_0}{\partial a_1^{(L)}}\right)\left(\frac{\partial a_1^{(L)}}{\partial z_1^{(L)}}\right)\left(\frac{\partial z_1^{(L)}}{\partial w_{12}^{(L)}}\right).$$

Let's break down each term from the expression on the right hand side of the above equation.

The first term: $\frac{\partial C_0}{\partial a_1^{(L)}}$

We know that

$$C_0 = \sum_{j=0}^{n-1}\left(a_j^{(L)} - y_j\right)^2.$$

Therefore,

$$\frac{\partial C_0}{\partial a_1^{(L)}} = \frac{\partial}{\partial a_1^{(L)}}\left(\sum_{j=0}^{n-1}\left(a_j^{(L)} - y_j\right)^2\right).$$

Expanding the sum, we see

$$\begin{aligned}\frac{\partial}{\partial a_1^{(L)}}\left(\sum_{j=0}^{n-1}\left(a_j^{(L)} - y_j\right)^2\right) &= \frac{\partial}{\partial a_1^{(L)}}\left(\left(a_0^{(L)} - y_0\right)^2 + \left(a_1^{(L)} - y_1\right)^2 + \left(a_2^{(L)} - y_2\right)^2 + \left(a_3^{(L)} - y_3\right)^2\right) \\ &= \frac{\partial}{\partial a_1^{(L)}}\left(\left(a_0^{(L)} - y_0\right)^2\right) + \frac{\partial}{\partial a_1^{(L)}}\left(\left(a_1^{(L)} - y_1\right)^2\right) + \frac{\partial}{\partial a_1^{(L)}}\left(\left(a_2^{(L)} - y_2\right)^2\right) + \frac{\partial}{\partial a_1^{(L)}}\left(\left(a\right.\right. \\ &= 2\left(a_1^{(L)} - y_1\right).\end{aligned}$$

So the loss from the network for a single input sample will respond to a small change in the activation output from node *1* in layer *L* by an amount equal to two times the difference of the activation output a_1 for node *1* and the desired output y_1 for node *1*.

The second term: $\frac{\partial a_1^{(L)}}{\partial z_1^{(L)}}$

We know that for each node *j* in the output layer *L*, we have

$$a_j^{(L)} = g^{(L)}\left(z_j^{(L)}\right),$$

and since $j = 1$, we have

$$a_1^{(L)} = g^{(L)}\left(z_1^{(L)}\right).$$

Therefore,

$$\begin{aligned}\frac{\partial a_1^{(L)}}{\partial z_1^{(L)}} &= \frac{\partial}{\partial z_1^{(L)}}\left(g^{(L)}\left(z_1^{(L)}\right)\right) \\ &= g'^{(L)}\left(z_1^{(L)}\right).\end{aligned}$$

So this is just the direct derivative of $a_1^{(L)}$ since $a_1^{(L)}$ is a direct function of $z_1^{(L)}$.

The third term: $\frac{\partial z_1^{(L)}}{\partial w_{12}^{(L)}}$

We know that, for each node *j* in the output layer *L*, we have

$$z_j^{(L)} = \sum_{k=0}^{n-1} w_{jk}^{(L)} a_k^{(L-1)}.$$

Since $j = 1$, we have

$$z_1^{(L)} = \sum_{k=0}^{n-1} w_{1k}^{(L)} a_k^{(L-1)}.$$

Therefore,

$$\frac{\partial z_1^{(L)}}{\partial w_{12}^{(L)}} = \frac{\partial}{\partial w_{12}^{(L)}}\left(\sum_{k=0}^{n-1} w_{1k}^{(L)} a_k^{(L-1)}\right).$$

Expanding the sum, we see

$$\frac{\partial}{\partial w_{12}^{(L)}}\left(\sum_{k=0}^{n-1} w_{1k}^{(L)} a_k^{(L-1)}\right) = \frac{\partial}{\partial w_{12}^{(L)}}\left(w_{10}^{(L)} a_0^{(L-1)} + w_{11}^{(L)} a_1^{(L-1)} + w_{12}^{(L)} a_2^{(L-1)} + \cdots + w_{15}^{(L)} a_5^{(L-1)}\right)$$
$$= \frac{\partial}{\partial w_{12}^{(L)}} w_{10}^{(L)} a_0^{(L-1)} + \frac{\partial}{\partial w_{12}^{(L)}} w_{11}^{(L)} a_1^{(L-1)} + \frac{\partial}{\partial w_{12}^{(L)}} w_{12}^{(L)} a_2^{(L-1)} + \cdots + \frac{\partial}{\partial w_{12}^{(L)}} w_{15}^{(L)} a_5^{(L-1)}$$
$$= a_2^{(L-1)}$$

So the input for node *1* in layer *L* will respond to a change in the weight $w_{12}^{(L)}$ by an amount equal to the activation output for node *2* in the previous layer, $L - 1$.

Combining terms

Combining all terms, we have

$$\frac{\partial C_0}{\partial w_{12}^{(L)}} = \left(\frac{\partial C_0}{\partial a_1^{(L)}}\right)\left(\frac{\partial a_1^{(L)}}{\partial z_1^{(L)}}\right)\left(\frac{\partial z_1^{(L)}}{\partial w_{12}^{(L)}}\right)$$
$$= 2\left(a_1^{(L)} - y_1\right)\left(g'^{(L)}\left(z_1^{(L)}\right)\right)\left(a_2^{(L-1)}\right)$$

Conclude

So now, we've seen how to calculate the derivative of the loss with respect to one individual weight for one individual training sample.

To calculate the derivative of the loss with respect to this same particular weight, w_{12}, for all n training samples, we calculate the average derviative of the loss function over all n training samples.

This can be expressed as

$$\frac{\partial C}{\partial w_{12}^{(L)}} = \frac{1}{n}\sum_{i=0}^{n-1} \frac{\partial C_i}{\partial w_{12}^{(L)}}.$$

We would then do this same process for each weight in the network to calculate the derivative of C with respect to each weight.

Derivative of the loss with respect to activation outputs

Motivation

We left off seeing how we can calculate the gradient
of the loss function with respect to any weight in the network.

Recall, the weight we chose to work with to explain this idea was $w_{12}^{(L)}$, and we saw that

$$\frac{\partial C_0}{\partial w_{12}^{(L)}} = \left(\frac{\partial C_0}{\partial a_1^{(L)}}\right)\left(\frac{\partial a_1^{(L)}}{\partial z_1^{(L)}}\right)\left(\frac{\partial z_1^{(L)}}{\partial w_{12}^{(L)}}\right).$$

Suppose we choose to work with a weight that is not in the output layer, like $w_{22}^{(L-1)}$.

Then the gradient of the loss with respect to this weight would be

$$\frac{\partial C_0}{\partial w_{22}^{(L-1)}} = \left(\frac{\partial C_0}{\partial a_2^{(L-1)}}\right)\left(\frac{\partial a_2^{(L-1)}}{\partial z_2^{(L-1)}}\right)\left(\frac{\partial z_2^{(L-1)}}{\partial w_{22}^{(L-1)}}\right).$$

The second and third terms on the right hand side would be calculated in the exact same way as we saw for $w_{12}^{(L)}$. The first term on the right hand side, $\frac{\partial C_0}{\partial a_2^{(L-1)}}$, will not be calculated in the same way as before.

We need to understand how to calculate this term in order to calculate the gradient of the loss function with respect to any weight that is *not* in the output layer.

The calculation of this term will be our focus.

Set up

We're going to show how we can calculate the derivative of the loss function with respect to the activation output for any node that is not in the output layer.

Let's look at a single activation output for node *2* in layer $L - 1$.

This is denoted as

$$a_2^{(L-1)}.$$

The derivative of the loss, C_0, with respect to this particular activation output $a_2^{(L-1)}$ is denoted as

$$\frac{\partial C_0}{\partial a_2^{(L-1)}}.$$

Observe that for each node j in L, the loss C_0 depends on on $a_j^{(L)}$, and $a_j^{(L)}$ depends on $z_j^{(L)}$. $z_j^{(L)}$ depends on all of the weights connected to node j from the previous layer, $L - 1$, as well as all the activation outputs from $L - 1$.
So, $z_j^{(L)}$ depends on $a_2^{(L-1)}$.

The chain rule tells us that to differentiate C_0 with respect to $a_2^{(L-1)}$, we take the product of the derivatives of the composed function. This derivative can be expressed as

$$\frac{\partial C_0}{\partial a_2^{(L-1)}} = \sum_{j=0}^{n-1}\left(\left(\frac{\partial C_0}{\partial a_j^{(L)}}\right)\left(\frac{\partial a_j^{(L)}}{\partial z_j^{(L)}}\right)\left(\frac{\partial z_j^{(L)}}{\partial a_2^{(L-1)}}\right)\right).$$

This equation looks almost identical to the equation we obtained for the derivative of the loss with respect to a given weight.
Recall that this previous derivative with respect to a given weight was expressed as

$$\frac{\partial C_0}{\partial w_{12}^{(L)}} = \left(\frac{\partial C_0}{\partial a_1^{(L)}}\right)\left(\frac{\partial a_1^{(L)}}{\partial z_1^{(L)}}\right)\left(\frac{\partial z_1^{(L)}}{\partial w_{12}^{(L)}}\right).$$

The two differences between the derivative of the loss with respect to an activation output, and the derivative of the loss with respect to a weight are:

1. The summation operation.
2. The last term on the right hand side.

The reason for the summation here is due to the fact that a change in one activation output in the previous layer is going to affect each node j in the following layer L, so we need to sum up these effects.

We can see that the first and second terms on the right hand side of the equation are the same as the first and second terms in the last equation with regards to the $w_{12}^{(L)}$ when $j = 1$. Since we've

already gone through the work to find how to calculate these two derivatives, we won't do it again here.

We'll only focus on breaking down the third term.

The third term: $\frac{\partial z_j^{(L)}}{\partial a_2^{(L-1)}}$

We know for each node j in layer L that

$$z_j^{(L)} = \sum_{k=0}^{n-1} w_{jk}^{(L)} a_k^{(L-1)}.$$

Therefore,

$$\frac{\partial z_j^{(L)}}{\partial a_2^{(L-1)}} = \frac{\partial}{\partial a_2^{(L-1)}} \sum_{k=0}^{n-1} w_{jk}^{(L)} a_k^{(L-1)}.$$

Expanding the sum, we have

$$\begin{aligned}
\frac{\partial}{\partial a_2^{(L-1)}} \sum_{k=0}^{n-1} w_{jk}^{(L)} a_k^{(L-1)} &= \frac{\partial}{\partial a_2^{(L-1)}} \left(w_{j0}^{(L)} a_0^{(L-1)} + w_{j1}^{(L)} a_1^{(L-1)} + w_{j2}^{(L)} a_2^{(L-1)} \cdots + w_{j5}^{(L)} a_5^{(L-1)} \right) \\
&= \frac{\partial}{\partial a_2^{(L-1)}} w_{j0}^{(L)} a_0^{(L-1)} + \frac{\partial}{\partial a_2^{(L-1)}} w_{j1}^{(L)} a_1^{(L-1)} + \frac{\partial}{\partial a_2^{(L-1)}} w_{j2}^{(L)} a_2^{(L-1)} \cdots + \frac{\partial}{\partial a_2^{(L-1)}} w_{j5}^{(L)} a_5^{(L-1)} \\
&= w_{j2}^{(L)}.
\end{aligned}$$

So the input for any node j in layer L will respond to a change in $a_2^{(L-1)}$ by an amount equal to the weight connecting node 2 in layer $L-1$ to node j in layer L.

Combining terms

Combining all terms, we have

$$\begin{aligned}
\frac{\partial C_0}{\partial a_2^{(L-1)}} &= \sum_{j=0}^{n-1} \left(\left(\frac{\partial C_0}{\partial a_j^{(L)}} \right) \left(\frac{\partial a_j^{(L)}}{\partial z_j^{(L)}} \right) \left(\frac{\partial z_j^{(L)}}{\partial a_2^{(L-1)}} \right) \right) \\
&= \sum_{j=0}^{n-1} \left(2 \left(a_j^{(L)} - y_j \right) \left(g'^{(L)} \left(z_j^{(L)} \right) \right) \left(w_{j2}^{(L)} \right) \right).
\end{aligned}$$

Now we can use this result to calculate the gradient of the loss with respect to any weight connected
to node 2 in layer $L-1$, like we saw for $w_{22}^{(L-1)}$, for example, with the following equation.

$$\frac{\partial C_0}{\partial w_{22}^{(L-1)}} = \left(\frac{\partial C_0}{\partial a_2^{(L-1)}} \right) \left(\frac{\partial a_2^{(L-1)}}{\partial z_2^{(L-1)}} \right) \left(\frac{\partial z_2^{(L-1)}}{\partial w_{22}^{(L-1)}} \right)$$

Note, to find the derivative of the loss function with respect to this same particular activation output, $a_2^{(L-1)}$, for all n training samples, we calculate the average derviative of the loss function over all n training samples. This can be expressed as

$$\frac{\partial C}{\partial a_2^{(L-1)}} = \frac{1}{n} \sum_{i=0}^{n-1} \frac{\partial C_i}{\partial a_2^{(L-1)}}.$$