

# General feed forward neural networks

Kristian Wichmann

July 20, 2017

## 1 Formalism and nomenclature

Consider a feed forward neural net with  $L + 1$  layers, in the sense that layer zero is the input layer and layer  $L$  the output layer. Each non-input layer has its own activation function  $\sigma_l$ .

The size of layer  $l$  we will denote  $S_l$ . So layer  $l$  has a  $S_l \times S_{l-1}$  matrix of weights  $W^{(l)}$  and a bias vector  $b^{(l)}$  with dimension  $S_l$ . So, given an input vector  $x$  (with dimension  $S_{l-1}$ ), the pre-activation and the activation of layer  $l$  can be expressed as:

$$z^{(l)} = W^{(l)}x + b^{(l)}, \quad a^{(l)} = \sigma_l(z^{(l)}) = \sigma_l(W^{(l)}x + b^{(l)}) \quad (1.1)$$

We will consider  $N$  data points, each with a feature vector of dimension  $S_0$ . We group these into an  $S_0 \times N$  matrix  $X$ :

$$X = \begin{pmatrix} | & \cdots & | \\ x_1 & \cdots & x_N \\ | & \cdots & | \end{pmatrix} \quad (1.2)$$

Note that this is the transpose of the usual "data frame" structure.

We can now write the pre-activations and activations of the first layer as:

$$Z^{(1)} = W^{(1)}X + b^{(1)}, \quad A^{(1)} = \sigma_1(Z^{(1)}) = \sigma_1(W^{(1)}X + b^{(1)}) \quad (1.3)$$

Here, we've used " $+b^{(1)}$ " as a shorthand for adding the vector  $b^{(1)}$  to every column. We could write this as " $+b^{(1)}J_N^t$ " if we wanted to be accurate. ( $J_N$  is a column vector of  $N$  ones).

Similarly, we may generally write the pre-activations and activations of layer  $l$  as:

$$Z^{(l)} = W^{(l)}A^{(l-1)} + b^{(l)}, \quad A^{(l)} = \sigma_l(Z^{(l)}) = \sigma_l(W^{(l)}A^{(l-1)} + b^{(l)}) \quad (1.4)$$

We will also identify  $X$  with the activations of "layer zero":  $A^{(0)} = X$ .

Finally, we have an error function  $J$  which measures the distance to some target data  $T \in \mathbb{R}^{S_L \times N}$ :

$$T = \begin{pmatrix} | & \cdots & | \\ t_1 & \cdots & t_N \\ | & \cdots & | \end{pmatrix} \quad (1.5)$$

We will assume the error function is of the form  $J = J(T, A^{(L)})$ , taking on real values. I.e. it only depends on the targets and the activations of the output layer.

## 2 Backpropagation - Output layer

Forward propagation through the network is described by equation 1.1. The procedure is assumed to be done before we look at how to determine partial derivatives of  $J$  through backpropagation.

### 2.1 Weights

The derivatives with respect to the output layer weights and biases can be found through the chain rule:

$$\frac{\partial J}{\partial W_{ij}^{(L)}} = \sum_{k=1}^{S_L} \sum_{n=1}^N \frac{\partial J}{\partial A_{kn}^{(L)}} \frac{\partial A_{kn}^{(L)}}{\partial W_{ij}^{(L)}}, \quad \frac{\partial J}{\partial b_i^{(L)}} = \sum_{k=1}^{S_L} \sum_{n=1}^N \frac{\partial J}{\partial A_{kn}^{(L)}} \frac{\partial A_{kn}^{(L)}}{\partial b_i^{(L)}} \quad (2.1)$$

We may find the derivatives of  $A^{(L)}$  with respect to the weights and biases:

$$\frac{\partial A_{kn}^{(L)}}{\partial W_{ij}^{(L)}} = \frac{\partial A_{kn}^{(L)}}{\partial \sigma_L} \frac{\partial \sigma_L}{\partial Z_{kn}^{(L)}} \frac{\partial Z_{kn}^{(L)}}{\partial W_{ij}^{(L)}} \quad (2.2)$$

But since  $\frac{\partial A_{kn}^{(L)}}{\partial \sigma_L}$  is simply one, this reduces to:

$$\frac{\partial A_{kn}^{(L)}}{\partial W_{ij}^{(L)}} = \frac{\partial \sigma_L}{\partial Z_{kn}^{(L)}} \frac{\partial Z_{kn}^{(L)}}{\partial W_{ij}^{(L)}} \quad (2.3)$$

This also means that when writing the  $J$ -derivative in matrix form, there will be a Hadamard product between the first two terms instead of ordinary matrix multiplication:

$$\frac{\partial J}{\partial W^{(L)}} = \frac{\partial J}{\partial A^{(L)}} \odot \frac{\partial \sigma_L}{\partial Z^{(L)}} \frac{\partial Z^{(L)}}{\partial W^{(L)}} \quad (2.4)$$

Finally, we may calculate the derivatives of  $Z^{(L)}$  with respect to the weights:

$$\frac{\partial Z_{kn}^{(L)}}{\partial W_{ij}^{(L)}} = \frac{\partial}{\partial W_{ij}^{(L)}} (W^{(L)} A^{(L-1)} + b^{(L)})_{kn} = \frac{\partial}{\partial W_{ij}^{(L)}} \left( \sum_{l=1}^{S_{L-1}} W_{kl}^{(L)} A_{ln}^{(L-1)} + b_k^{(L)} \right) \quad (2.5)$$

Differentiating  $W^{(L)}$  with respect to  $W^{(L)}$  yields two Kronecker deltas:

$$\frac{\partial Z_{kn}^{(L)}}{\partial W_{ij}^{(L)}} = \sum_{l=1}^{S_{L-1}} \delta_{ik} \delta_{jl} A_{ln}^{(L-1)} = \delta_{ik} A_{jn}^{(L-1)} \quad (2.6)$$

Now, we may insert equations 2.3 and 2.6 into 2.1:

$$\frac{\partial J}{\partial W_{ij}^{(L)}} = \sum_{k=1}^{S_L} \sum_{n=1}^N \frac{\partial J}{\partial A_{kn}^{(L)}} \frac{\partial \sigma_L}{\partial Z_{kn}^{(L)}} \delta_{ik} A_{jn}^{(L-1)} = \sum_{n=1}^N \frac{\partial J}{\partial A_{in}^{(L)}} \frac{\partial \sigma_L}{\partial Z_{in}^{(L)}} A_{jn}^{(L-1)} \quad (2.7)$$

We can rewrite this using the Hadamard product between the two derivatives and swapping the indices of  $A^{(L-1)}$ , turning into a transpose:

$$\sum_{n=1}^N \left[ \frac{\partial J}{\partial A^{(L)}} \odot \frac{\partial \sigma_L}{\partial Z^{(L)}} \right]_{in} (A^{(L-1)})_{nj}^t \quad (2.8)$$

Or in matrix form:

$$\frac{\partial J}{\partial W^{(L)}} = \left[ \frac{\partial J}{\partial A^{(L)}} \odot \frac{\partial \sigma_L}{\partial Z^{(L)}} \right] (A^{(L-1)})^t = H^{(L)} (A^{(L-1)})^t \quad (2.9)$$

Here we've introduced  $H^{(L)}$ , the matrix of the Hadamard product, for notational ease:

$$H^{(L)} = \frac{\partial J}{\partial A^{(L)}} \odot \frac{\partial \sigma_L}{\partial Z^{(L)}} \quad (2.10)$$

## 2.2 Biases

The procedure for the biases is the same until we get to the  $Z^{(L)}$  derivative:

$$\frac{\partial Z_{kn}^{(L)}}{\partial b_i^{(L)}} = \frac{\partial}{\partial b_i^{(L)}} \left( \sum_{l=1}^{S_{L-1}} W_{kl}^{(L)} A_{ln}^{(L-1)} + b_k^{(L)} \right) = \delta_{ik} \quad (2.11)$$

Reinserting all the way back to equation 2.1 we get:

$$\frac{\partial J}{\partial b_i^{(L)}} = \sum_{k=1}^{S_L} \sum_{n=1}^N \frac{\partial J}{\partial A_{kn}^{(L)}} \frac{\partial \sigma_L}{\partial Z_{kn}^{(L)}} \delta_{ik} = \sum_{n=1}^N \frac{\partial J}{\partial A_{in}^{(L)}} \frac{\partial \sigma_L}{\partial Z_{in}^{(L)}} \quad (2.12)$$

Again, we can write this in matrix notation using the Hadamard product:

$$\frac{\partial J}{\partial b^{(L)}} = \left[ \frac{\partial J}{\partial A^{(L)}} \odot \frac{\partial \sigma_L}{\partial Z^{(L)}} \right] J_N^t = H^{(L)} J_N^t \quad (2.13)$$

### 2.3 Output layer "error"

We will call the quantity  $\frac{\partial J}{\partial A^{(L)}}$  the output layer "error":

$$\Delta^{(L)} = \frac{\partial J}{\partial A^{(L)}} \quad (2.14)$$

The quotes are used, because there's no need for this to be equal/proportional to what we usually call errors, i.e. distance between the output  $A^{(L)}$  and  $T$ . However, often this is the case (or rather,  $J$  is specifically chosen to make this so - see below). At any rate, we may now write:

$$H^{(L)} = \Delta^{(L)} \odot \frac{\partial \sigma_L}{\partial Z^{(L)}} \quad (2.15)$$

$$\frac{\partial J}{\partial W^{(L)}} = H^{(L)} (A^{(L-1)})^t = \left[ \Delta^{(L)} \odot \frac{\partial \sigma_L}{\partial Z^{(L)}} \right] (A^{(L-1)})^t \quad (2.16)$$

$$\frac{\partial J}{\partial b^{(L)}} = H^{(L)} J_N^t = \left[ \Delta^{(L)} \odot \frac{\partial \sigma_L}{\partial Z^{(L)}} \right] J_N^t \quad (2.17)$$

### 2.4 Example error function: Euclidean distance

A common form of error function is:

$$J(T, A^{(L)}) = \frac{1}{2N} \sum_{m=1}^N \|a_m^{(L)} - t_m\|^2 \quad (2.18)$$

Here,  $a_m^{(L)}$  is the  $m$ 'th column of  $A^{(L)}$  and the double dashes is the usual Euclidean norm in  $S_L$  dimensions. So we may rewrite:

$$J(T, A^{(L)}) = \frac{1}{2N} \sum_{m=1}^N \sum_{j=1}^{S_L} (A_{jm}^{(L)} - T_{jm})^2 \quad (2.19)$$

Now we can find the derivative:

$$\Delta_{in}^{(L)} = \frac{\partial J}{\partial A_{in}^{(L)}} = \frac{1}{2N} \sum_{m=1}^N \sum_{j=1}^{S_L} \frac{\partial}{\partial A_{in}^{(L)}} (A_{jm}^{(L)} - T_{jm})^2 \quad (2.20)$$

Again, we get two Kronecker deltas:

$$\frac{1}{2N} \sum_{m=1}^N \sum_{j=1}^{S_L} 2(A_{jm}^{(L)} - T_{jm}) \delta_{ij} \delta_{nm} = \frac{1}{N} (A_{in}^{(L)} - T_{in}) \quad (2.21)$$

The quantity in parenthesis is the  $ij$  element of the error term matrix for the output layer. So indeed we get:

$$\Delta^{(L)} = \frac{1}{N} \left( A_{in}^{(L)} - T_{in} \right) \quad (2.22)$$

Or in matrix form:

$$\Delta^{(L)} = \frac{1}{N} \left( A^{(L)} - T \right) \quad (2.23)$$

### 3 Backpropagation - Last hidden layer

#### 3.1 Weights

Now, let's consider derivatives with respect to weights in layer  $L - 1$ . Simply applying the chain rule, we get:

$$\frac{\partial J}{\partial W_{ij}^{(L-1)}} = \sum_{k=1}^{S_L} \sum_{n=1}^N \sum_{k'=1}^{S_{L-1}} \sum_{n'=1}^N \frac{\partial J}{\partial A_{kn}^{(L)}} \frac{\partial \sigma_L}{\partial Z_{kn}^{(L)}} \frac{\partial Z_{kn}^{(L)}}{\partial A_{k'n'}^{(L-1)}} \frac{\partial \sigma_{L-1}}{\partial Z_{k'n'}^{(L-1)}} \frac{\partial Z_{k'n'}^{(L-1)}}{\partial W_{ij}^{(L-1)}} \quad (3.1)$$

Now, this will obviously be very similar to the calculations above, but to be certain, let's proceed carefully. The only term in the above we have not calculated yet is:

$$\frac{\partial Z_{kn}^{(L)}}{\partial A_{k'n'}^{(L-1)}} = \frac{\partial}{\partial A_{k'n'}^{(L-1)}} \left( W^{(L)} A^{(L-1)} + b^{(L)} \right)_{kn} = \quad (3.2)$$

$$\frac{\partial}{\partial A_{k'n'}^{(L-1)}} \left( \sum_{l=1}^{S_{L-1}} W_{kl}^{(L)} A_{ln}^{(L-1)} + b_k^{(L)} \right) = \quad (3.3)$$

$$\sum_{l=1}^{S_{L-1}} W_{kl}^{(L)} \delta_{lk'} \delta_{nn'} = W_{kk'}^{(L)} \delta_{nn'} \quad (3.4)$$

Now we're ready to insert into equation 3.1:

$$\frac{\partial J}{\partial W_{ij}^{(L-1)}} = \sum_{k=1}^{S_L} \sum_{n=1}^N \sum_{k'=1}^{S_{L-1}} \sum_{n'=1}^N \Delta_{kn}^{(L)} \frac{\partial \sigma_L}{\partial Z_{kn}^{(L)}} W_{kk'}^{(L)} \delta_{nn'} \frac{\partial \sigma_{L-1}}{\partial Z_{k'n'}^{(L-1)}} \delta_{ik'} A_{jn'}^{(L-2)} = \quad (3.5)$$

$$\sum_{k=1}^{S_L} \sum_{n=1}^N H_{kn}^{(L)} W_{ki}^{(L)} \frac{\partial \sigma_{L-1}}{\partial Z_{in}^{(L-1)}} A_{jn}^{(L-2)} \quad (3.6)$$

Use the trick of rearranging terms and swapping indices:

$$\frac{\partial J}{\partial W_{ij}^{(L-1)}} = \sum_{k=1}^{S_L} \sum_{n=1}^N (W^{(L)})_{ik}^t H_{kn}^{(L)} \frac{\partial \sigma_{L-1}}{\partial Z_{in}^{(L-1)}} (A^{(L-2)})_{nj}^t = \quad (3.7)$$

$$\sum_{n=1}^N \left[ (W^{(L)})^t H^{(L)} \right]_{in} \frac{\partial \sigma_{L-1}}{\partial Z_{in}^{(L-1)}} (A^{(L-2)})_{nj}^t \quad (3.8)$$

Once again, we can collect the first two terms into a Hadamard product:

$$\frac{\partial J}{\partial W_{ij}^{(L-1)}} = \sum_{n=1}^N \left[ (W^{(L)})^t H^{(L)} \odot \frac{\partial \sigma_{L-1}}{\partial Z^{(L-1)}} \right]_{in} (A^{(L-2)})_{nj}^t = \quad (3.9)$$

$$\left[ (W^{(L)})^t H^{(L)} \odot \frac{\partial \sigma_{L-1}}{\partial Z^{(L-1)}} (A^{(L-2)})^t \right]_{ij} \quad (3.10)$$

Or in matrix form:

$$\frac{\partial J}{\partial W^{(L-1)}} = \left[ \underbrace{(W^{(L)})^t H^{(L)}}_{\Delta^{(L-1)}} \odot \frac{\partial \sigma_{L-1}}{\partial Z^{(L-1)}} \right] (A^{(L-2)})^t \quad (3.11)$$

Here, we've defined the underbraced part to be the "error" for layer  $L-1$ . The formula now takes a form very similar to equation 2.15:

$$\frac{\partial J}{\partial W^{(L-1)}} = \left[ \Delta^{(L-1)} \odot \frac{\partial \sigma_{L-1}}{\partial Z^{(L-1)}} \right] (A^{(L-2)})^t = H^{(L-1)} (A^{(L-2)})^t \quad (3.12)$$

Here, we've defined the  $H$  for layer  $L-1$  as:

$$H^{(L-1)} = \Delta^{(L-1)} \odot \frac{\partial \sigma_{L-1}}{\partial Z^{(L-1)}} \quad (3.13)$$

### 3.2 Biases

This will be very similar to the weights case. The only thing that changes, is that the last term in the chain rule decomposition is:

$$\frac{\partial Z_{k'n'}^{(L-1)}}{\partial b_i^{(L-1)}} = \delta_{ik'} \quad (3.14)$$

So all of the calculations play out the same way as above, except there's no multiplication by  $A^{(L-2)}$ . Instead we get:

$$\frac{\partial J}{\partial b_i^{(L-1)}} = \sum_{n=1}^N \left[ (W^{(L)})^t H^{(L)} \odot \frac{\partial \sigma_{L-1}}{\partial Z^{(L-1)}} \right]_{in} = \sum_{n=1}^N H_{in}^{(L-1)} \quad (3.15)$$

Or in matrix form:

$$\frac{\partial J}{\partial b^{(L-1)}} = H^{(L-1)} J_N^t \quad (3.16)$$

## 4 Backpropagation - General layer

Here, we wish to prove that in general, the formula for derivatives of  $J$  with respect to weights and biases from any layer  $l$  can be written:

$$\frac{\partial J}{\partial W^{(l)}} = H^{(l)} (A^{(l-1)})^t, \quad \frac{\partial J}{\partial b^{(l)}} = H^{(l)} J_N^t \quad (4.1)$$

Here,  $H^{(l)}$  is defined recursively:

$$H^{(l)} = \Delta^{(l)} \odot \frac{\partial \sigma_l}{\partial Z^{(l)}}, \quad \Delta^{(l)} = (W^{(l+1)})^t H^{(l+1)}, \quad \Delta^{(L)} = \frac{\partial J}{\partial A^{(L)}} \quad (4.2)$$

### 4.1 A useful lemma

To show this it turns out to be useful to start by proving the following:

$$\frac{\partial J}{\partial A^{(l)}} = \Delta^{(l)} \quad (4.3)$$

This is done through induction, although backwards from  $l = L$  down to  $l = 1$ .

#### 4.1.1 Induction start

This corresponds to  $l = L$ . Here, this is true by definition:

$$\frac{\partial J}{\partial A^{(L)}} = \Delta^{(L)} \quad (4.4)$$

#### 4.1.2 Induction step

So we need to prove  $(l) \Rightarrow (l-1)$ . Notice the following:

$$\frac{\partial J}{\partial A^{(l)}} = \frac{\partial J}{\partial A^{(L)}} \odot \frac{\partial \sigma_L}{\partial Z^{(L)}} \frac{\partial Z^{(L)}}{\partial A^{(L-1)}} \odot \dots \odot \frac{\partial \sigma_{l+1}}{\partial Z^{(l+1)}} \frac{\partial Z^{(l+1)}}{\partial A^{(l)}} \quad (4.5)$$

$$\frac{\partial J}{\partial A^{(l-1)}} = \underbrace{\frac{\partial J}{\partial A^{(L)}} \odot \frac{\partial \sigma_L}{\partial Z^{(L)}} \frac{\partial Z^{(L)}}{\partial A^{(L-1)}} \odot \dots \odot \frac{\partial \sigma_{l+1}}{\partial Z^{(l+1)}} \frac{\partial Z^{(l+1)}}{\partial A^{(l)}}}_{\frac{\partial J}{\partial A^{(l)}}} \odot \frac{\partial \sigma_l}{\partial Z^{(l)}} \frac{\partial Z^{(l)}}{\partial A^{(l-1)}} \quad (4.6)$$

The Hadamard products follow from the same logic that led to equation 2.4. By the induction assumption,  $\frac{\partial J}{\partial A^{(l)}} = \Delta^{(l)}$ , so we need to calculate:

$$\frac{\partial J}{\partial A^{(l-1)}} = \Delta^{(l)} \odot \frac{\partial \sigma_l}{\partial Z^{(l)}} \frac{\partial Z^{(l)}}{\partial A^{(l-1)}} \quad (4.7)$$

In coordinate form:

$$\left[ \frac{\partial J}{\partial A^{(l-1)}} \right]_{in} = \sum_{j=1}^{S_l} \sum_{n'=1}^N \left( \Delta_{jn'}^{(l)} \frac{\partial \sigma_l}{\partial Z_{jn'}^{(l)}} \right) \frac{\partial Z_{jn'}^{(l)}}{\partial A_{in}^{(l-1)}} \quad (4.8)$$

But we've already calculated the last derivative:

$$\sum_{j=1}^{S_l} \sum_{n'=1}^N \left( \Delta_{jn'}^{(l)} \frac{\partial \sigma_l}{\partial Z_{jn'}^{(l)}} \right) W_{ji}^{(l)} \delta_{nn'} = \sum_{j=1}^{S_l} \left( \Delta_{jn}^{(l)} \frac{\partial \sigma_l}{\partial Z_{jn}^{(l)}} \right) W_{ji}^{(l)} \quad (4.9)$$

Now use the usual trick of swapping indices:

$$\left[ \frac{\partial J}{\partial A^{(l-1)}} \right]_{in} = \sum_{j=1}^{S_l} (W^{(l)})_{ij}^t \left( \Delta_{jn}^{(l)} \frac{\partial \sigma_l}{\partial Z_{jn}^{(l)}} \right) = \left[ (W^{(l)})^t H^{(l)} \right]_{ij} \quad (4.10)$$

This is exactly the  $ij$ 'th element of  $\Delta^{(l-1)}$ , as desired.

## 4.2 Weights

The lemma makes it easy to derive the formula for weights in the  $l$ 'th layer:

$$\frac{\partial J}{\partial W^{(l)}} = \underbrace{\frac{\partial J}{\partial A^{(L)}} \odot \frac{\partial \sigma_L}{\partial Z^{(L)}} \frac{\partial Z^{(L)}}{\partial A^{(L-1)}} \odot \cdots \odot \frac{\partial \sigma_{l+1}}{\partial Z^{(l+1)}} \frac{\partial Z^{(l+1)}}{\partial A^{(l)}}}_{\frac{\partial J}{\partial A^{(l)}}} \odot \frac{\partial \sigma_l}{\partial Z^{(l)}} \frac{\partial Z^{(l)}}{\partial W^{(l)}} = \quad (4.11)$$

$$\Delta^{(l)} \odot \frac{\partial \sigma_l}{\partial Z^{(l)}} \frac{\partial Z^{(l)}}{\partial W^{(l)}} \quad (4.12)$$

Element-wise, using all the (by now) usual tricks:

$$\frac{\partial J}{\partial W_{ij}^{(l)}} = \sum_{k=1}^{S_l} \sum_{n=1}^N \Delta_{kn}^{(l)} \frac{\partial \sigma_l}{\partial Z_{kn}^{(l)}} \frac{\partial Z_{kn}^{(l)}}{\partial W_{ij}^{(l)}} = \sum_{k=1}^{S_l} \sum_{n=1}^N \Delta_{kn}^{(l)} \frac{\partial \sigma_l}{\partial Z_{kn}^{(l)}} A_{jn}^{(l)} \delta_{ki} = \quad (4.13)$$

$$\sum_{n=1}^N \Delta_{in}^{(l)} \frac{\partial \sigma_l}{\partial Z_{in}^{(l)}} A_{jn}^{(l)} = \sum_{n=1}^N \Delta_{in}^{(l)} \frac{\partial \sigma_l}{\partial Z_{in}^{(l)}} (A^{(l)})_{nj}^t = \quad (4.14)$$

$$\sum_{n=1}^N \left[ \Delta^{(l)} \odot \frac{\partial \sigma_l}{\partial Z^{(l)}} \right]_{in} (A^{(l)})_{nj}^t \quad (4.15)$$

Back in matrix form:

$$\frac{\partial J}{\partial W^{(l)}} = H^{(l)} (A^{(l)})^t \quad (4.16)$$



### 4.3 Biases

This is almost the same as for the weights:

$$\frac{\partial J}{\partial b^{(l)}} = \underbrace{\frac{\partial J}{\partial A^{(L)}} \odot \frac{\partial \sigma_L}{\partial Z^{(L)}} \frac{\partial Z^{(L)}}{\partial A^{(L-1)}} \odot \cdots \odot \frac{\partial \sigma_{l+1}}{\partial Z^{(l+1)}} \frac{\partial Z^{(l+1)}}{\partial A^{(l)}}}_{\frac{\partial J}{\partial A^{(l)}}} \odot \frac{\partial \sigma_l}{\partial Z^{(l)}} \frac{\partial Z^{(l)}}{\partial b^{(l)}} = \quad (4.17)$$

$$\Delta^{(l)} \odot \frac{\partial \sigma_l}{\partial Z^{(l)}} \frac{\partial Z^{(l)}}{\partial b^{(l)}} \quad (4.18)$$

Elementwise:

$$\left[ \frac{\partial J}{\partial b^{(l)}} \right]_i = \sum_{j=1}^{S_l} \sum_{n=1}^N \Delta_{jn}^{(l)} \frac{\partial \sigma_l}{\partial Z_{jn}^{(l)}} \delta_{ji} = \sum_{n=1}^N \Delta_{in}^{(l)} \frac{\partial \sigma_l}{\partial Z_{in}^{(l)}} = \sum_{n=1}^N H_{in}^{(l)} \quad (4.19)$$

In matrix form:

$$\frac{\partial J}{\partial b^{(l)}} = H^{(l)} J_N^t \quad (4.20)$$