# Backprop on Fully Connected Feedforward Network
## 1 Hidden Layer

By the end of a forward pass:

$$Z_1 = XW_1 + b_1 \longrightarrow A_1 = ReLU(Z_1)$$
$$Z_2 = A_1 W_2 + b_2 \longrightarrow A_2 = Softmax(Z_2)$$

$$loss = -\frac{1}{N}\sum_i \sum_j y_{ij} \log(A_{2_{ij}})$$

How does changing the weights and biases affect the loss?

We want gradient of loss wrt weights and biases, $\frac{\partial L}{\partial W}, \frac{\partial L}{\partial b}$

We reach through chain rule of calculus —

Loss per sample: $L = -\sum_j y_j \log(A_{2_j})$

remember log amplifies differences near zero and acts on the probability like results of softmax - multiplying by $y_j$ uses one hot encoding to select the probability assigned to the 'right' class and zero out the others. We then negate b/c $\log(x)$, $x \in (0,1)$ will be negative.

We do the derivatives backwards based on dependencies

loss depends on $A_2$
$A_2$ depends on $Z_2 = A_1 W_2 + b_2$
$Z_2$ depends on $A_1$
$A_1$ depends on $Z_1 = XW_1 + b_1$
$Z_1$ depends on $W_1, b_1$ & input $X$

Its easiest to see how the output $A_2$ changes the loss, and then we propagate that change backward thru each layer using chain rule
- avoids recomputing derivatives repeatedly

First, do $\frac{\partial L}{\partial A_{2_j}}$ for one sample: $\frac{\partial L}{\partial A_2} = \frac{\partial}{\partial A_2}(-y_j \log(A_2))$

$$\frac{\partial L}{\partial A_2} = -\frac{y_j}{A_{2_j}}$$ note: log here is ln, $\frac{d}{dx}(\ln x) = \frac{1}{x}$

Next we know $\frac{\partial L}{\partial Z_2} = \frac{\partial L}{\partial A_2} \cdot \frac{\partial A_2}{\partial Z_2}$ from the chain rule

$Z_2 = A_1 W_2 + b_2$, $L = -y_j \log(A_2)$

Softmax: $A_{2_j} = \frac{e^{Z_{2_j}}}{\sum_k e^{Z_{2_k}}}$ where the denominator is normalizing

$$\frac{\partial A_2}{\partial Z_2} = \begin{cases} A_{2_j}(1 - A_{2_j}) & j = k \\ A_{2_j} A_{2_k} & j \neq k \end{cases}$$ how changing one z affects all probabilities

messy, so we trust instead of compute

$$\longrightarrow \frac{\partial L}{\partial Z_2} = \sum_{j=1}^{C} \frac{\partial L}{\partial A_{2_j}} \frac{\partial A_{2_j}}{\partial Z_{2_k}}$$ the chain rule allows sum across all paths

If you do this algebra, all the cross terms cancel ($i \neq j$) and you get

$$\frac{\partial L}{\partial Z_2} = A_k - y_k$$

and that's why cross entropy loss + softmax is a popular combo.

Mental Model:
- $Z_2 \rightarrow$ softmax $\rightarrow A_2 \rightarrow$ loss as flowchart
- each knob $Z_{2_k}$ affects all A outputs, each A output affects L
- arrows from all $Z_{2_k}$ to all $A_j$ each weighted by $\frac{\partial A_j}{\partial Z_{2_k}}$
- arrows from all $A_j$ to L, weighted by $\frac{\partial L}{\partial A_j}$
- chain rule = multiply along each arrow, sum contributions into $\frac{\partial L}{\partial Z_{2_k}}$

Next in line is $\frac{\partial L}{\partial W_2}$ & $\frac{\partial L}{\partial b_2}$:

by chain rule: $\frac{\partial L}{\partial W_2} = \frac{\partial L}{\partial Z_2} \cdot \frac{\partial Z_2}{\partial W_2}$ $\quad Z_2^{(i,j)} = \sum A_1^{(i,i)} W_2^{(k,j)} + b_2^{(j)}$

$$= (A_k - y_k) A_1 \qquad \frac{\partial L}{\partial W_2} = A_1^T \cdot \frac{\partial L}{\partial Z_2}$$

and to make that multiplication work in the matrix world, we transpose $A_1$ so the inner dims of $A_1$ & $\frac{\partial L}{\partial Z_2}$ match

by chain rule: $\frac{\partial L}{\partial b_2} = \frac{\partial L}{\partial Z_2} \cdot \frac{\partial Z_2}{\partial b_2}$ & $\frac{\partial Z_2}{\partial b_2} = 1$

$$\longrightarrow \frac{\partial L}{\partial b_2} = \frac{\partial L}{\partial Z_2} \text{ or } \frac{\partial L}{\partial b_2} = \sum_{\text{samples}} \frac{\partial L}{\partial Z_2}$$

Next we want $\frac{\partial L}{\partial Z_1} = \frac{\partial L}{\partial A_1} \cdot \frac{\partial A_1}{\partial Z_1} = \left(\frac{\partial L}{\partial Z_2} \cdot \frac{\partial Z_2}{\partial A_1}\right) \frac{\partial A_1}{\partial Z_1}$

where $\frac{\partial Z_2}{\partial A_1} = W_2$, how output pre-activ depends on hidden layer activation

$\frac{\partial A_1}{\partial Z_1}$ = derivative of activation function

thus $\frac{\partial L}{\partial Z_1} = \frac{\partial L}{\partial Z_2} \cdot W_2^T \odot \sigma'(Z_1)$ where $\odot$ is element wise mult & $\sigma'(Z_1)$ is deriv of ReLU

then by chain rule $\frac{\partial L}{\partial W_1} = \frac{\partial L}{\partial Z_1} \cdot \frac{\partial Z_1}{\partial W_1}$ but $Z_1 = XW_1 + b_1$, $\frac{\partial Z_1}{\partial W_1} = X$

so $\frac{\partial L}{\partial W_1} = X^T \cdot \frac{\partial L}{\partial Z_1}$

and $\frac{\partial L}{\partial b_1} = \sum_{\text{samples}} \frac{\partial L}{\partial Z_{1_k}}$