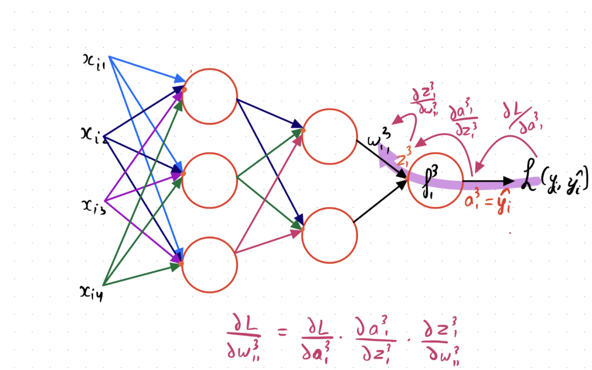
05 Neural Networks Layer- BackPropagation

How do we train this complex NN? What is Backpropagation?

- In order to update the parameters, we need to find their gradients. For this, we use the Backpropagation algorithm, where we traverse from right to left in the NN.
- It is based on the concept of chain rule of differentiation.



Gradient of w^{3}_{11}

We'll encounter a^3_1 while going from loss towards w^3_{11}

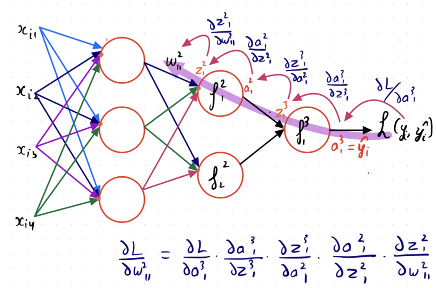
Therefore gradient: $\frac{\partial L}{\partial w^{3}_{11}} = \frac{\partial L}{\partial a^{3}_{1}} \cdot \frac{\partial a^{3}_{1}}{\partial z^{3}_{1}} \cdot \frac{\partial z^{3}_{1}}{\partial w^{3}_{11}}$

Gradient of w^2_{11}

We'll encounter a^3_1 and a^2_1 while going from loss towards w^2_{11}

Therefore gradient:

$$\frac{\partial L}{\partial w^{3}_{11}} = \frac{\partial L}{\partial a^{3}_{1}} \cdot \frac{\partial a^{3}_{1}}{\partial z^{3}_{1}} \cdot \frac{\partial z^{3}_{1}}{\partial a^{2}_{1}} \cdot \frac{\partial a^{2}_{1}}{\partial z^{2}_{1}} \cdot \frac{\partial z^{2}_{1}}{\partial w^{2}_{11}}$$



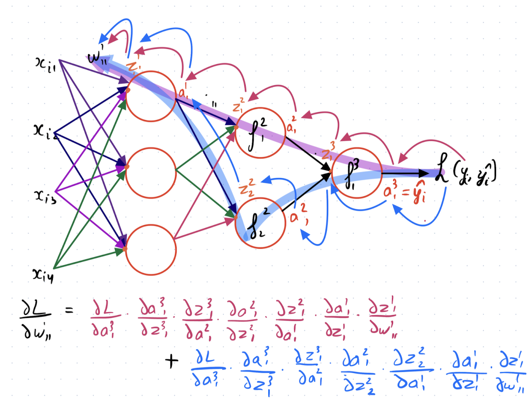
Gradient of w^1_{11}

There are 2 possible paths to reach $\mathbf{w^1_{11}}$

Path-1 : L -> a^3_1 -> a^2_1 -> a^1_1 -> w^1_{11}

Path-2 : L -> a^3_1 -> a^2_2 -> a^1_1 -> w^1_{11}

We need to combine the derivatives from path 1 and 2 by adding them up.



Therefore gradient:

$$\frac{\partial L}{\partial w^1_{11}} = \frac{\partial L}{\partial a^3_1} \cdot \frac{\partial a^3_1}{\partial z^3_1} \cdot \frac{\partial z^3_1}{\partial a^2_1} \cdot \frac{\partial a^2_1}{\partial z^2_1} \cdot \frac{\partial z^2_1}{\partial a^1_1} \cdot \frac{\partial a^1_1}{\partial z^1_1} \cdot \frac{\partial z^1_1}{\partial w^1_{11}} + \frac{\partial L}{\partial a^3_1} \cdot \frac{\partial a^3_1}{\partial z^3_2} \cdot \frac{\partial z^3_1}{\partial a^2_1} \cdot \frac{\partial a^2_2}{\partial z^2_2} \cdot \frac{\partial z^2_2}{\partial a^1_1} \cdot \frac{\partial a^1_1}{\partial z^1_1} \cdot \frac{\partial z^1_1}{\partial w^1_{11}}$$

What are the different activation functions?

- Sigmoid function
- Hyperbolic tan function: $\frac{e^z - e^{-z}}{e^z + e^{-z}}$
- ReLu: $ReLu(z) = max(z, 0)$
- Leaky ReLu: $Leaky\ ReLu(z) = max(z, \alpha z)$; α is a small gradient that we add

What is the vanishing gradient problem?

- The downside of both sigmoid and tanh is that their gradient is ~0, for most of the values of z
- This hampers the gradient descent process, as the calculated gradients become very small.
- For eg Suppose we wish to update weight w^1_{11}. its gradient is calculated as:

$$\frac{\partial L}{\partial w^1_{11}} = \frac{\partial L}{\partial a^3_1}\left[\frac{\partial a^3_1}{\partial a^2_{11}} \cdot \frac{\partial a^2_{11}}{\partial a^1_{11}} \cdot \frac{\partial a^1_{11}}{\partial w^1_{11}} + \frac{\partial a^3_1}{\partial a^2_{21}} \cdot \frac{\partial a^2_{21}}{\partial a^1_{12}} \cdot \frac{\partial a^1_{12}}{\partial w^1_{11}}\right]$$

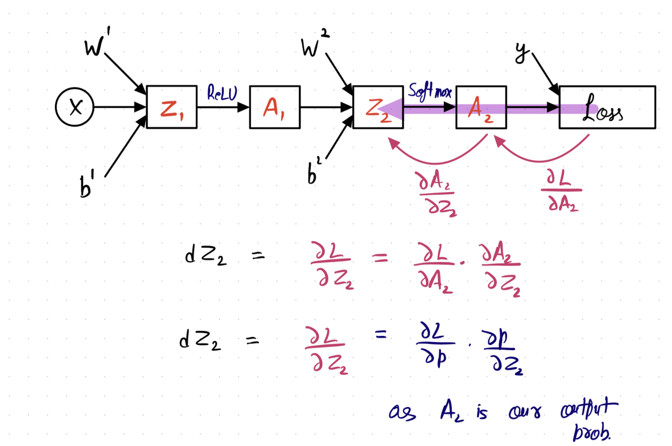
- So, the product of these terms inside the bracket will become very small.
- In fact, as the number of layers in the NN increase, this product will become smaller and smaller.

Backprop for MLP

- **Calculating dZ^2**

$dZ^2 = \frac{\partial L}{\partial Z^2} = \frac{\partial L}{\partial A^2} \cdot \frac{\partial A^2}{\partial Z^2}$

$dZ^2 = \frac{\partial L}{\partial p} \cdot \frac{\partial p}{\partial Z^2} = p_i - I(i == k)$

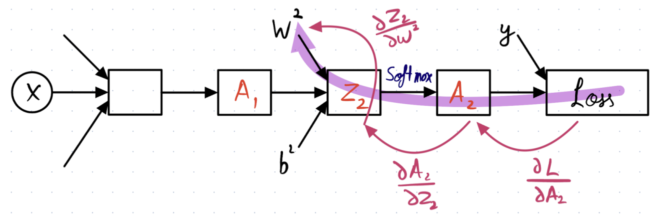


- **Calculating dW^2**

$$dW^2 = \frac{\partial L}{\partial W^2} = \frac{\partial L}{\partial A^2} \cdot \frac{\partial A^2}{\partial Z^2} \frac{\partial Z^2}{\partial W^2}$$

$$dW^2 = dZ^2 \cdot \frac{\partial Z^2}{\partial W^2}$$

$$dW^2 = dZ^2 \cdot A^1$$



$$\frac{\partial L}{\partial w^2} = \boxed{\frac{\partial L}{\partial A_2} \cdot \frac{\partial A_2}{\partial Z_2}} \frac{\partial Z_2}{\partial w^2} \qquad [\because Z_2 = w^2 A_1 + b^2]$$

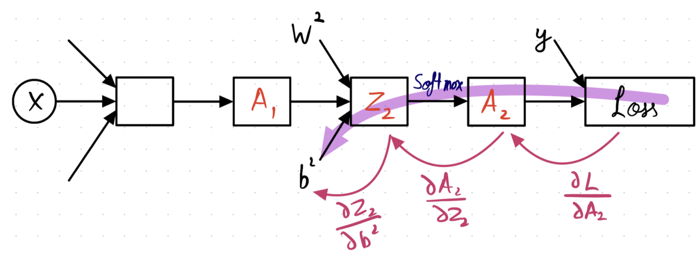
$$\frac{\partial (w^2 A_1 + b^2)}{\partial w^2}$$

$$\boxed{\frac{\partial L}{\partial w^2} = dZ_2 \cdot A_1}$$

- **Calculating db^2**

$$db^2 = \frac{\partial L}{\partial b^2} = \frac{\partial L}{\partial A^2} \cdot \frac{\partial A^2}{\partial Z^2} \frac{\partial Z^2}{\partial b^2}$$

$$db^2 = dZ^2 \cdot \frac{\partial Z^2}{\partial b^2} = dZ^2$$

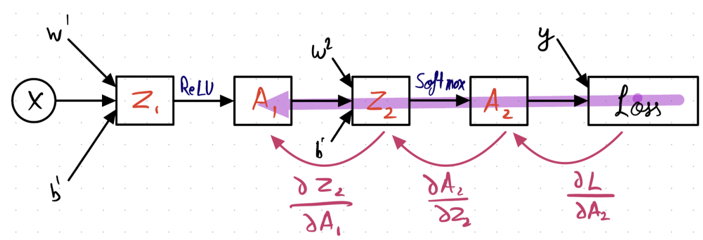


$$\frac{\partial Z_2}{\partial b^2} = \frac{\partial (w^2 A_1 + b^2)}{\partial b^2} = 1$$

- **Calculating dA^1**

$$dA^1 = \frac{\partial L}{\partial A^1} = \frac{\partial L}{\partial A^2} \cdot \frac{\partial A^2}{\partial Z^2} \frac{\partial Z^2}{\partial A^1}$$

$$dA^1 = dZ^2 \cdot \frac{\partial Z^2}{\partial A^1} = dZ^2 \cdot W^2$$

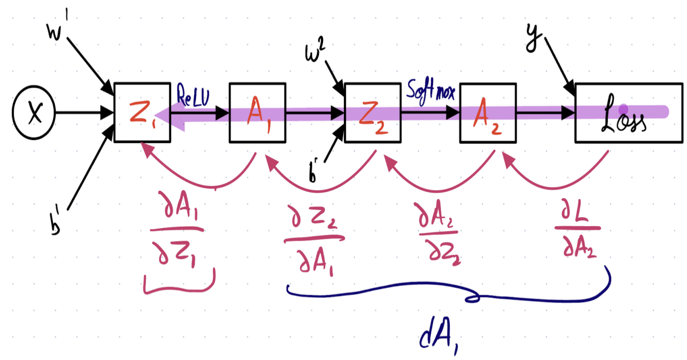


$$\frac{\partial Z_2}{\partial A_1} = \frac{\partial (w^2 A_1 + b^1)}{\partial A_1}$$

$$\boxed{\frac{\partial Z_2}{\partial A_1} = w^2}$$

- **Calculating dZ^1**

$$dZ^1 = \frac{\partial L}{\partial Z^1} = \frac{\partial L}{\partial A^2} \cdot \frac{\partial A^2}{\partial Z^2} \frac{\partial Z^2}{\partial A^1} \frac{\partial A^1}{\partial Z^1}$$



$$dZ^1 = dA^1 . \frac{\partial A^1}{\partial Z^1}$$

$$dZ^1 = dA^1 . \frac{\partial A^1}{\partial Z^1}$$

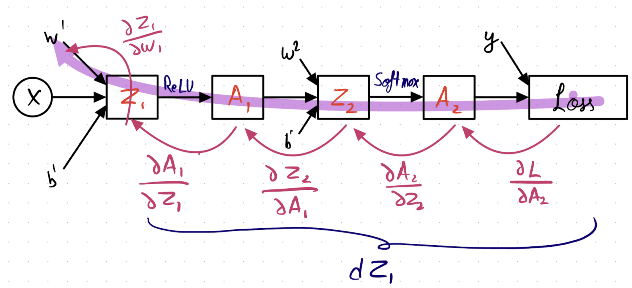
$$\frac{\partial L}{\partial Z_1} = \partial A_1 \cdot \frac{\partial A_1}{\partial Z_1}$$

Can be 0 or 1

$$= \begin{cases} \partial A_1 * 0 & \text{if} \quad Z_1 \leq 0 \\ \partial A_1 \times 1 & \text{if} \quad Z_1 > 0 \end{cases}$$

- **Calculating dW^1**

$$dW^1 = \frac{\partial L}{\partial W^1} = \frac{\partial L}{\partial A^2} \cdot \frac{\partial A^2}{\partial Z^2} \frac{\partial Z^2}{\partial A^1} \frac{\partial A^1}{\partial Z^1} \frac{\partial Z^1}{\partial W^1}$$



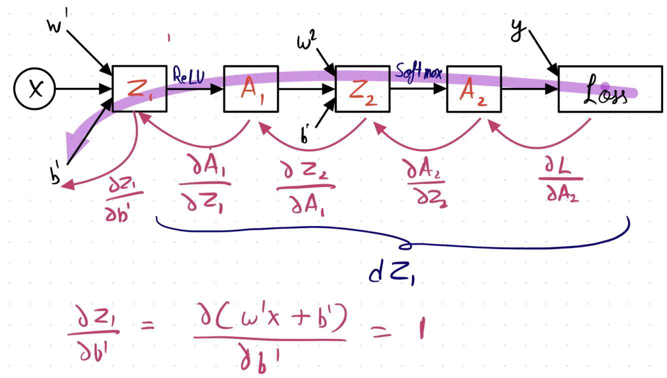
$$dW^1 = dZ^1 \frac{\partial Z^1}{\partial W^1} = dZ^1 . X$$

$$\frac{\partial Z_1}{\partial w'} = \frac{\partial (w'x + b')}{\partial w'} = x$$

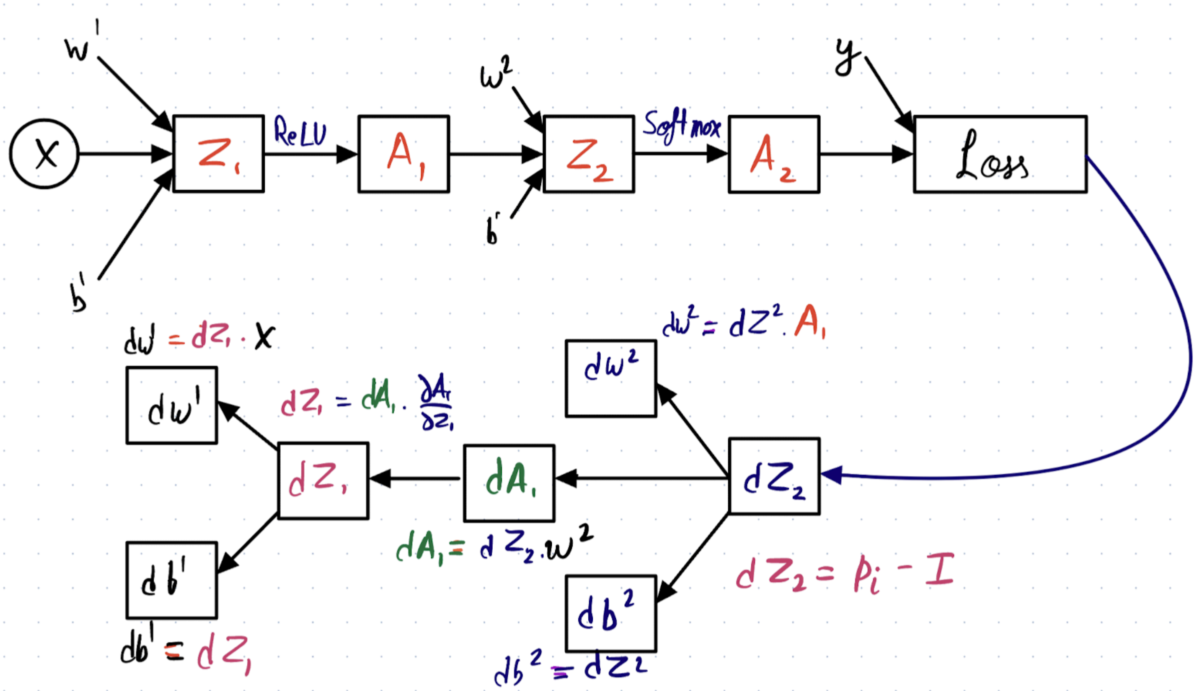
- **Calculating db^1**

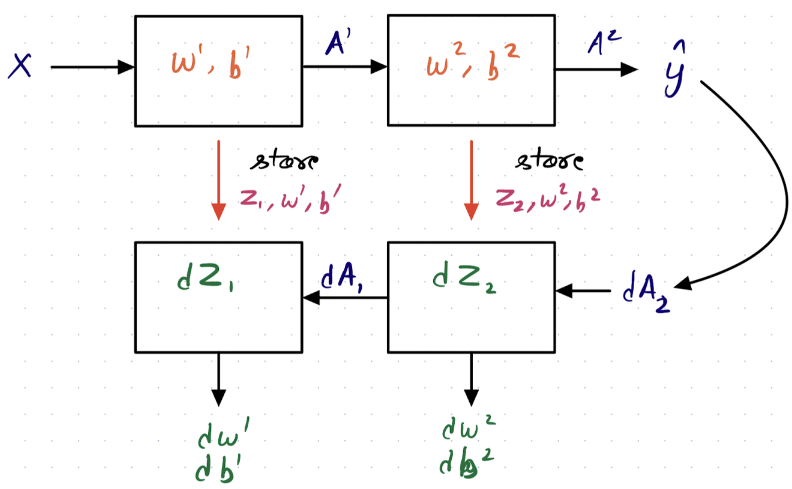
$$db^1 = \frac{\partial L}{\partial b^1} = \frac{\partial L}{\partial A^2} \cdot \frac{\partial A^2}{\partial Z^2} \frac{\partial Z^2}{\partial A^1} \frac{\partial A^1}{\partial Z^1} \frac{\partial Z^1}{\partial b^1}$$

$$db^1 = dZ^1 . \frac{\partial Z^1}{\partial b^1} = dZ^1 .1 = dZ^1$$



Summarizing forward and backward prop for MLP





While performing forward prop,

- we store/cache the value of Zj, Wj, bj to use them during back prop

Can we use Neural Networks for the Regression task?

- Yes, if the activation function for the output layer is a linear function, then NN will do regression.
- The activations for intermediate layers still need to be non-linear, otherwise, NN will not be able to map complex relationships.