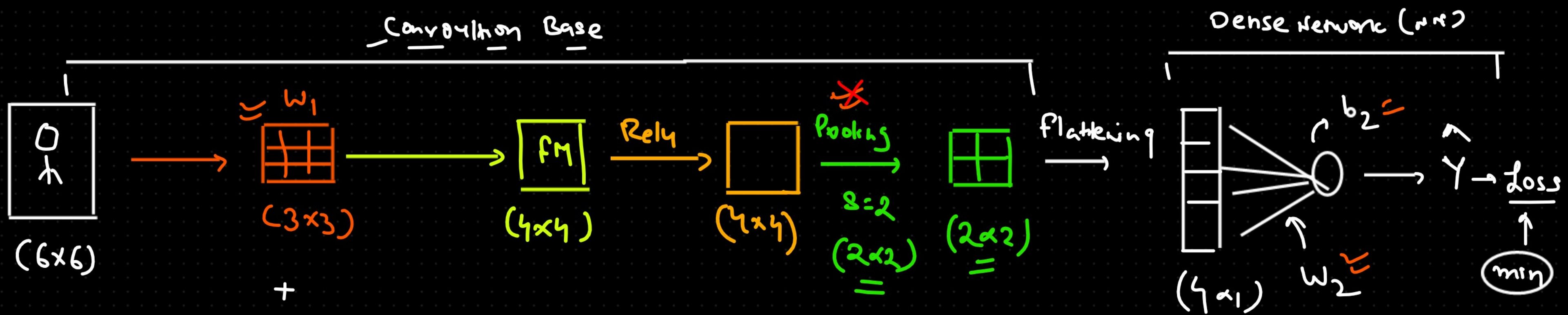


$+$

$\boxed{\text{bias}} \Rightarrow b_1$

Loss = binary classification

$$\boxed{L = -y_i \log(\hat{y}_i) - (1-y_i)\log(1-\hat{y}_i)}$$

Trainable Parameter

$w_1 = (3\times3) = 9 \qquad w_2 = 4 = 4$

$b_1 = (1\times1) = 1 \qquad b_2 = 1 = 1 \quad \Rightarrow 15$

$\downarrow$ trainable parameter

$$\Rightarrow \frac{n+2p-f}{s} + 1$$

$$\Rightarrow \frac{6+2\times0-3}{1} + 1$$

$$\Rightarrow 6-3+1 = \boxed{4}$$

$$I/P \rightarrow \circledast \rightarrow Z_1 \xrightarrow{\text{Relu}} A_1 \xrightarrow[\text{Pooling}]{\text{Max}} P_1 \xrightarrow{\text{Flatten}} F \rightarrow Z_2 \xrightarrow{\text{Sigmoid}} A_2 \rightarrow L$$

$\uparrow$

w_1, b_1

Forward Prop

$Z_1 = \text{Conv}(x, W_1) + b_1$

$A_1 = \text{Relu}(Z_1)$

$P_1 = \text{MaxPool}(A_1)$

$f = \text{flatten}(P_1)$

$Z_2 = W_2 f + b_2$

$A_2 = \sigma(Z_2)$

Loss = Binary cross entropy

$$\boxed{L = -y_i \log(\hat{y}_i) - (1-y_i)\log(1-\hat{y}_i)}$$

optimizer ⇒ GD ← Gradient Descent opt.

$$\boxed{W_N^1 = W_0^1 - \eta \frac{\partial L}{\partial W_1}} \qquad \boxed{b_1 = b_1 - \eta \frac{\partial L}{\partial b_1}}$$

$$\boxed{W_N^2 = W_0^2 - \eta \frac{\partial L}{\partial W_2}} \qquad \boxed{b_2 = b_2 - \eta \frac{\partial L}{\partial b_2}}$$

$$\boxed{\frac{\partial L}{\partial W_2}}$$

Partial derivation

$$\left\{ \frac{\Delta L}{\Delta W} \right\}$$

$\Delta \to 0$

rate of change

$$\left\{ \frac{dL}{dW} \right\}$$

rate of change at the particular point

Input $\rightarrow$ $*$ $\rightarrow$ Z_1 $\xrightarrow{\text{relu}}$ A_1 $\xrightarrow[\text{Pooling}]{\text{max}}$ P_1 $\xrightarrow{\text{flatten}}$ f $\xrightarrow{\boxed{W_2\, b_2}}$ Z_2 $\xrightarrow{\text{Sigmoid}}$ $A_2\ (\hat{y})$ $\longrightarrow$ $\boxed{\text{Loss}}$

$\uparrow$

$\boxed{W_1, b_1}$

$$\left\{ \frac{\partial L}{\partial W_2} = \frac{\partial L}{\partial A_2} \times \frac{\partial A_2}{\partial Z_2} \times \frac{\partial Z_2}{\partial W_2} \right\} \Rightarrow \boxed{\text{Chain rule}}$$

$W_1 = \text{mat}(3\times 3)$

$b_1 = (1 \times 1)$

$$\left\{ \frac{\partial L}{\partial b_2} = \frac{\partial L}{\partial A_2} \times \frac{\partial A_2}{\partial Z_2} \times \frac{\partial Z_2}{\partial b_2} \right\} \Rightarrow \boxed{\text{chain rule}}$$

$$\left\{ \frac{\partial L}{\partial W_1} = \frac{\partial L}{\partial A_2} \times \frac{\partial A_2}{\partial Z_2} \times \frac{\partial Z_2}{\partial f} \times \boxed{\frac{\partial f}{\partial P_1}} \times \boxed{\frac{\partial P_1}{\partial A_1}} \times \frac{\partial A_1}{\partial Z_1} \times \boxed{\frac{\partial Z_1}{\partial W_1}} \right\} \Rightarrow \text{chain rule}$$

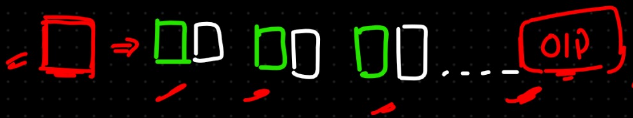
$$\left\{ \frac{\partial L}{\partial b_1} = \frac{\partial L}{\partial A_2} \times \frac{\partial A_2}{\partial Z_2} \times \frac{\partial Z_2}{\partial f} \times \frac{\partial f}{\partial P_1} \times \frac{\partial P_1}{\partial A_1} \times \frac{\partial A_1}{\partial Z_1} \times \frac{\partial Z_1}{\partial b_1} \right\} \Rightarrow \text{Chain rule}$$

(chain rule eq. wrt to the weight which will take during convolution)

- { Open google colab
- functional API from keras }

Sequential API

I/P → **Linear seq** → **O/P** ⇒ Linear topology

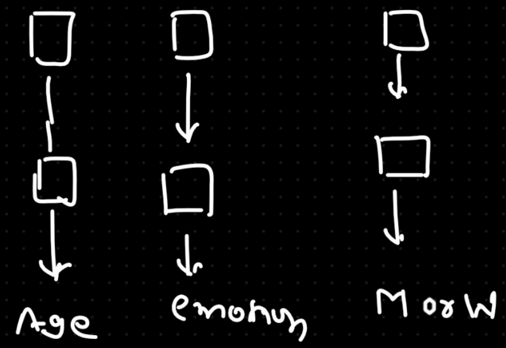


$\hat{y} - y \Rightarrow$ loss

⇒ Age, emotion, Men or women

First approach ⇒ Naive approach

3 network ⇒ 1st Network Age
2nd Network emotion
3rd Network Men or women



Sequential model ← CNN

2nd approach ⇒ functional Network