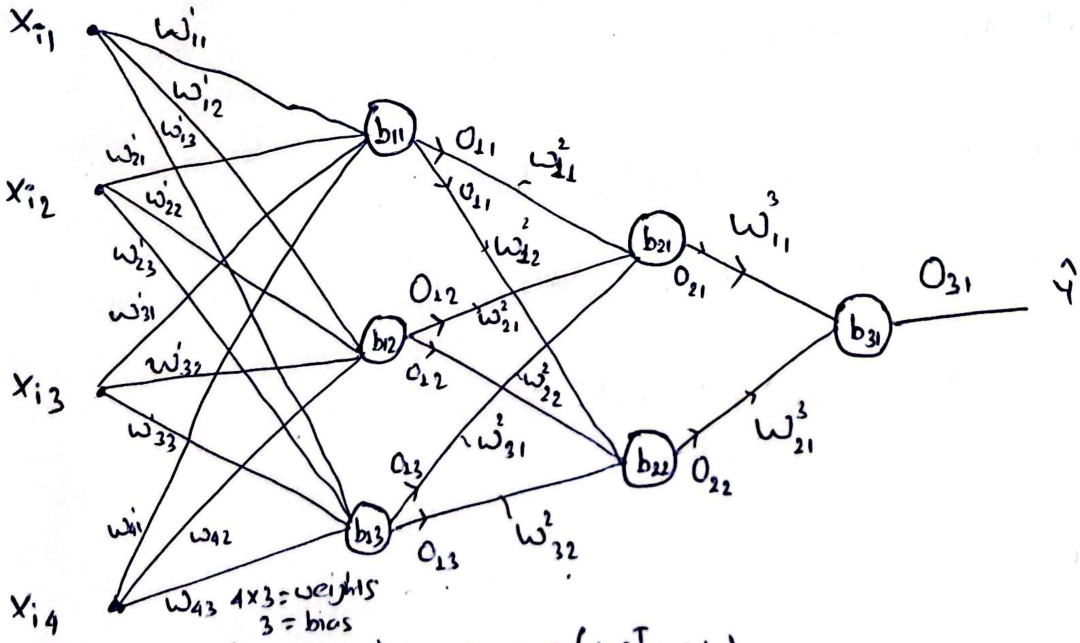
How Neural Network predicts

Suppose we have a data

gpa	iq	10th marks	12th marks	Placed
7.2	74	66	80	1
8.1	90	70	76	0



Prediction $\rightarrow \sigma(W^T x + b)$

Layer #1

$$\begin{bmatrix} w^1_{11} & w^1_{12} & w_{13} \\ w^1_{21} & w^1_{22} & w^1_{23} \\ w^1_{31} & w^1_{32} & w^1_{33} \\ w^1_{41} & w^1_{42} & w^1_{43} \end{bmatrix}^T \begin{bmatrix} x_{i1} \\ x_{i2} \\ x_{i3} \\ x_{i4} \end{bmatrix} + \begin{bmatrix} b_{11} \\ b_{12} \\ b_{13} \end{bmatrix}$$

$4\times3 \xrightarrow{T} 3\times4 \qquad 4\times1 \qquad 3\times1$

$$\begin{bmatrix} w^1_{11} & w^1_{21} & w^1_{31} & w^1_{41} \\ w^1_{12} & w^1_{22} & w^1_{32} & w^1_{42} \\ w^1_{13} & w^1_{23} & w^1_{33} & w^1_{43} \end{bmatrix} \begin{bmatrix} x_{i1} \\ x_{i2} \\ x_{i3} \\ x_{i4} \end{bmatrix} + \begin{bmatrix} b_{11} \\ b_{12} \\ b_{13} \end{bmatrix}$$

$$= \begin{bmatrix} w'_{11}x_{i1} + w'_{21}x_{i2} + w'_{31}x_{i3} + w'_{41}x_{i4} \\ w'_{12}x_{i1} + w'_{22}x_{i2} + w'_{32}x_{i3} + w'_{42}x_{i4} \\ w'_{13}x_{i1} + w'_{23}x_{i2} + w'_{33}x_{i3} + w'_{43}x_{i4} \end{bmatrix} + \begin{bmatrix} b_{11} \\ b_{12} \\ b_{13} \end{bmatrix}$$

since we also have to apply activation function

$$= \sigma\left(\begin{bmatrix} w'_{11}x_{i1} + w'_{21}x_{i2} + w'_{31}x_{i3} + w'_{41}x_{i4} + b_{11} \\ w'_{12}x_{i1} + w'_{22}x_{i2} + w'_{32}x_{i3} + w'_{42}x_{i4} + b_{12} \\ w'_{13}x_{i1} + w'_{23}x_{i2} + w'_{33}x_{i3} + w'_{43}x_{i4} + b_{13} \end{bmatrix}\right)$$

$$= \begin{bmatrix} O_{11} \\ O_{12} \\ O_{13} \end{bmatrix}$$

Layer #2

$$= \begin{bmatrix} w^2_{11} & w^2_{12} \\ w^2_{21} & w^2_{22} \\ w^2_{31} & w^2_{32} \end{bmatrix}^T \begin{bmatrix} O_{11} \\ O_{12} \\ O_{13} \end{bmatrix} + \begin{bmatrix} b_{21} \\ b_{22} \end{bmatrix}$$

$3\times 2 \xrightarrow{T} 2\times 3$ $\quad 3\times 1 \quad 2\times 1$

Transpose & multiply (dot products).

$$= \sigma\left(\begin{bmatrix} w^2_{11}O_{11} + w^2_{21}O_{12} + w^2_{31}O_{13} + b_{12} \\ w^2_{12}O_{11} + w^2_{22}O_{12} + w^2_{32}O_{13} + b_{22} \end{bmatrix}\right) = \begin{bmatrix} O_{21} \\ O_{22} \end{bmatrix}$$

Layer#3

$$\begin{bmatrix} w^3_{11} \\ w^3_{21} \end{bmatrix}^T \begin{bmatrix} O_{21} \\ O_{22} \end{bmatrix} + \begin{bmatrix} b_{31} \end{bmatrix}$$

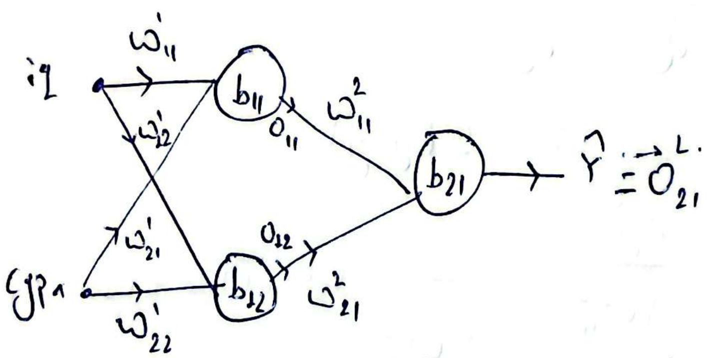
$$= \sigma\left(\left[w^3_{11} O_{21} + w^3_{21} O_{22} + b_{31}\right]\right)$$

$$= \hat{Y}_i$$

$$= O_{31}$$

Backpropagation in NN

iq	cgpa	lpa
80	8	3
60	9	5
70	5	8
120	7	11



Steps

0) Initialize weights & bias (random value or different initialization techniques).

$w \to 1, b \to 0.$

1) You select a point (row)
↳ Student.

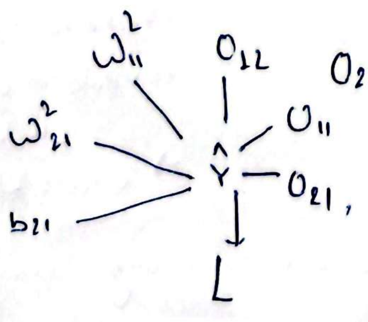
2) Predict (lpa) → forward prop [Dot product].
Suppose we got $\boxed{18 \text{ lpa}}$ but actual was 3.

3) Choose a loss function.
We chose MSE loss

$$L = (Y - \hat{Y})^2$$

$$(3 - 18)^2 = 225$$

error.



$$O_{21} = O_{11} w^2_{11} + O_{12} w^2_{21} + b_{21}$$

4) Weights & bias update
↳ Gradient Descent

$$W_{new} = W_{old} - \eta \, \frac{\partial L}{\partial W_{old}}$$

$$b_{new} = b_{old} - \eta \frac{\partial L}{\partial b_{old}}$$

Partial derivative → G.D

$$\frac{\partial L}{\partial w_{11}^2}, \frac{\partial L}{\partial w_{21}^2}, \frac{\partial L}{\partial b_{21}} \;\Big|\; \frac{\partial L}{\partial w_{11}^1}, \frac{\partial L}{\partial w_{21}^1}, \frac{\partial L}{\partial b_{11}} \;\Big|\; \frac{\partial L}{\partial w_{12}^1}, \frac{\partial L}{\partial w_{22}^1}, \frac{\partial L}{\partial b_{12}}$$

$$\frac{\partial L}{\partial w_{11}^2} = \frac{\partial L}{\partial \hat{y}} \times \frac{\partial \hat{y}}{\partial w_{11}^2}$$ → Chain rule

$$\frac{\partial L}{\partial \hat{y}} = \frac{\partial}{\partial \hat{y}} (Y - \hat{Y})^2$$

$$= -2(y - \hat{y}).$$

$$\frac{\partial \hat{y}}{\partial w_{11}^2} = \frac{\partial}{\partial w_{11}^2} [O_{11} w_{11}^2 + O_{12} w_{21}^2 + b_{21}].$$

$$= O_{11}$$

$$\boxed{\therefore \frac{\partial L}{\partial w_{11}^2} = -2(y - \hat{y}) O_{11}}$$

→ Loss function is a function of all trainable parameter.

Back propagation algorithm

epochs = 5.

for i in range (epochs)

for j in range (x.shape[0])

→ select 1 row (random)

→ Predict (using forward propagation)

→ Calculate loss

→ update weights & biase

→ $w_n = w_o - \eta \frac{\partial L}{\partial w}$

→ Calculate avg loss for the epoch