Q.1. Deep Learning Assgn 2   Yash Patel, Gaurav Maheshwari

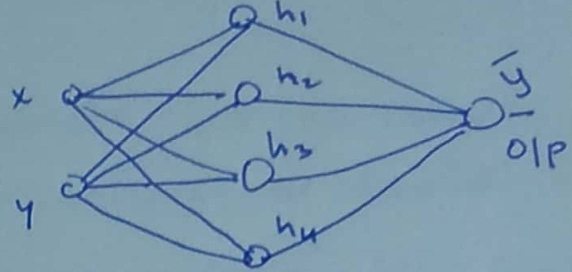
Now, we have to first get the eqn's of lines (segments)

$\overline{PQ} = x - 3y - 2 = 0.$ (h_1)

$\overline{PK} = 2x - y + 6 = 0$ (h_2)

$\overline{RK} = x - 2y + 6 = 0.$ (h_3)

$\overline{QR} = x + 3y - 14 = 0.$ (h_4).



Now, we are given as inputs (x, y), in order to design a NN, hidden layer has 4 perceptrons and o/p layer has only one perceptron.

Also, each perceptron in the hidden layer models one line segment of the trapezoid. And we can do AND/OR accordingly to classify the pt's in class A & class B.

Hence, the o/p layer perceptron is AND/OR gate

Now, as we have designed the NN, we need to calculate the wt's & bias.

$$h_1 = \left([x \;\; y] \underbrace{\begin{bmatrix} 1 \\ -3 \end{bmatrix}}_{\text{wt's}} \right) - \underbrace{2}_{\text{bias}}$$

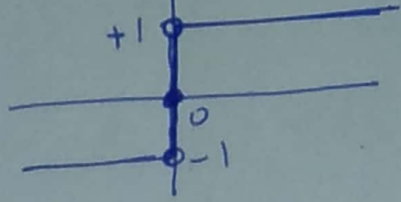
$$h_2 = \left([x \;\; y] \underbrace{\begin{bmatrix} 2 \\ -1 \end{bmatrix}}_{\text{wt's}} \right) + \underbrace{6.}_{\text{bias}}$$

$$h_3 = \left([x \;\; y] \underbrace{\begin{bmatrix} 1 \\ -2 \end{bmatrix}}_{\text{wt's}} \right) + \underbrace{6}_{\text{bias}}$$

$$h_4 = \left([x \;\; y] \underbrace{\begin{bmatrix} 1 \\ +3 \end{bmatrix}}_{\text{wt's}} \right) - \underbrace{14.}_{\text{bias}}$$

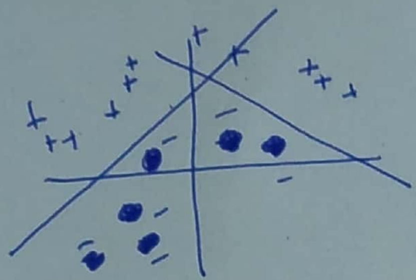
Activatn function: Sgn(x). i.e.

$$f(x) = \begin{cases} -1 & x < 0 \\ +1 & x > 0 \\ 0 & x = 0. \end{cases}$$



→ Using sgn(x) as activatn functn, we can classify easily whether a pt. (x, y) is in class (A) or class (B).

→ Hence, we can implement this logic with an AND gate



Output layer wts $[-1, -1, -1, -1]$ and bias 2.

Output Classes: if $\bar{y} = \text{output} = 1$, then class B

$\bar{y} = -1$, then class A.

Q.2

(a) Following equations are used:

$s(x) = \text{sigmoid}(x) = \frac{1}{1+e^{-x}}$.

$h_1 = s(w_1x_1 + w_2x_2 + b_1) = 0.596$

$h_2 = s(w_3x_1 + w_4x_2 + b_1) = 0.608$

$s'(x) = s(x)(1 - s(x))$

$o_1 = s(w_5h_1 + w_6h_2 + b_2) = 0.758$

$o_2 = s(w_7h_1 + w_8h_2 + b_2) = 0.779$

Now, we know that total error is used as mean squared error. Hence,

Error $E = \frac{1}{2}((o_1 - y_1)^2 + (o_2 - y_2)^2)$.

$= \frac{1}{2} \times 0.446 = \boxed{0.2238}$

(b) Now, for back propogation we need to update the wt's & bias by doing gradient descent and then update the wt's & bias by this formula's.

new_weight = old_weight $- \underbrace{\alpha}_{0.5} \cdot \underbrace{\text{grad\_descent\_w}}_{\text{w.r.t. w}}$

new_bias = old_bias $- \underbrace{\alpha}_{0.5} \cdot \underbrace{\text{grad\_descent\_b}}_{\text{w.r.t. b}}$.

In order to ~~upat~~ update all the wt's, we need to differentiate Error "E" w.r.t. all the wt's. Also as most of the terms in the differentiation are common, I ~~wont~~ will write in the following manner.

$\frac{\delta E}{\delta w_1} = \frac{\delta E}{\delta h_1} \times \frac{\delta h_1}{\delta w_1}$ & $\frac{\delta E}{\delta w_2} = \frac{\delta E}{\delta h_1} \times \frac{\delta h_1}{\delta w_2}$.

$$\frac{\delta E}{\delta h_1} = \underbrace{(O_1-y_1)\,O_1(1-O_1)}_{0.120} w_5 + \underbrace{(O_2-y_2)\,O_2(1-O_2)}_{-0.020.} w_7 \;—\; ⓐ = 0.038$$

$$\frac{\delta h_1}{\delta w_1} = \underbrace{h_1(1-h_1)}_{0.240} x_1 = 0.024 \;\&\; \frac{\delta h_1}{\delta w_2} = \underbrace{h_1(1-h_1)}_{0.240} x_2 = 0.096.$$

Here only x_1 & x_2 differ in $\frac{\delta h_1}{\delta w_1}$ & $\frac{\delta h_2}{\delta w_2}$.

~~So I will write~~

$$\cancel{\frac{\delta h_1}{\delta w_{1,2}} = h_1(1-h_1)\,w_{1,2} \;—\; ⓑ}$$

$$\therefore \frac{\delta E}{\delta w_1} = \underbrace{ⓐ}_{0.038} \times 0.024.$$

$$\frac{\delta E}{\delta w_2} = ⓐ \times 0.096.$$

$$= 3.648 \times 10^{-3}.$$

$$\therefore \frac{\delta E}{\delta w_1} = 9.12 \times 10^{-4}.$$

$$\frac{\delta E}{\delta h_2} = \underbrace{(O_1-y_1)\,O_1(1-O_1)}_{0.120} w_6 + \underbrace{(O_2-y_2)\,O_2(1-O_2)}_{-0.020} w_8. = 0.048 \;ⓑ$$

$$\frac{\partial h_2}{\partial w_3} = \underbrace{h_2(1-h_2)}_{0.238} x_1 = 0.023$$

$$\frac{\partial h_2}{\partial w_4} = \underbrace{h_2(1-h_2)}_{0.238} x_2 = 0.095$$

$$\frac{\delta E}{\delta w_3} = \underbrace{ⓑ}_{0.048} \times 0.023.$$

$$= 1.104 \times 10^{-3}$$

$$\frac{\delta E}{\delta w_4} = ⓑ \times 0.095$$

$$= 4.56 \times 10^{-3}$$

$$\frac{\partial E}{\partial w_5} = \underbrace{(O_1-y_1)\,O_1(1-O_1)}_{0.120} \underbrace{h_1}_{0.596} = 0.071$$

$$\frac{\partial E}{\partial w_6} = \underbrace{(O_1-y_1)\,O_1(1-O_1)}_{0.120} \underbrace{h_2}_{0.608} = 0.072.$$

$$\frac{\partial E}{\partial w_7} = \underbrace{(O_2 - Y_2)\, O_2 (1 - O_2)}_{-0.020}\, h_1 = -0.011$$

$$\frac{\partial E}{\partial w_8} = \underbrace{(O_2 - Y_2)\, O_2 (1 - O_2)}_{-0.020}\, h_2 = -0.012$$

Now, we will update the wt's accordingly

$$\text{new\_}w_1 = w_1 - \alpha \cdot \frac{\partial E}{\partial w_1}$$

$$= 0.1 - 0.5 \times 9.12 \times 10^{-4}.$$

$$= \cancel{4.56 \times 10^{-4}}\ 0.0999544$$

$$\text{new\_}w_2 = w_2 - \alpha \cdot \frac{\partial E}{\partial w_2}$$

$$= 0.2 - 0.5 \times 3.648 \times 10^{-3} = 0.198176$$

$$\text{new\_}w_3 = w_3 - \alpha \cdot \frac{\partial E}{\partial w_3} = 0.19948$$

$$\text{new\_}w_4 = w_4 - \alpha \cdot \frac{\partial E}{\partial w_4} = 0.29772.$$

$$\text{new\_}w_5 = w_5 - \alpha \frac{\partial E}{\partial w_5} = 0.3645$$

$$\text{new\_}w_6 = w_6 - \alpha \frac{\partial E}{\partial w_6} = 0.464$$

$$\text{new\_}w_7 = w_7 - \alpha \cdot \frac{\partial E}{\partial w_7} = 0.5055$$

$$\text{new\_}w_8 = w_8 - \alpha \cdot \frac{\partial E}{\partial w_8} = 0.606.$$

Now feed forward analysis with new wt's.

new_h_1 = 0.596105952

new_h_2 = 0.608029304.

new_O_1 = 0.750148831

new_O_2 = 0.780708049.

Now, new_Error = 0.2184.

Yess, the error after 1st iteratn of back propogatn has ~~increased~~ decreased !!.