# Assignment 5

## Question 1

→ Logistic Regression uses probabilities above & below thresholds to classify, but perceptron after refining from weight optimization, directly gives output 0(1) binary.

→ Weights get updated in perceptron if there is misclassification. In logistic regression, there is loss function & it is optimized by gradient descent.

Tweak :- → Change the activation function to sigmoid
→ Use cross entropy loss function & reduce the loss to update weights

## Question 2

- Email → 2 chances of outputs → 1 neuron [0/1]
  At the end, we need to update weights.
  Sigmoid [0 to 1] will be better and easy to train.
  ∵ differentiable, can use gradient descent to update

- Digit classification → 10 possible → 10 neurons
  We can use SoftMax, which gives probability per every DIGIT

## Question 3

(a) Shape of input X = Total no. of inputs in the batch × 10
$= n \times 10$

(b) $\left.\begin{array}{l} W_h \Rightarrow 10 \times 50 \\ b_h \Rightarrow 1 \times 50 \end{array}\right\}$ Result $\Rightarrow Z = XW_h + b_h$

(c) $W_o \Rightarrow 50 \times 3$

$b_o \Rightarrow 3$

(d) shape of output $Y = (n \times 10) \times (10 \times 50) \times (50 \times 3)$

$= n \times 3$

(e) $Y = \text{ReLU}(\text{ReLU}(XW_h + b_h) W_o + b_o)$

Question 4

$$E(W) = -\sum_{n=1}^{N} \{t_n \ln y_n + (1-t_n) \ln(1-y_n)\}$$

$$\frac{\partial E(W)}{\partial a_k} = \frac{\partial E_k}{\partial y_k} \cdot \frac{\partial y_k}{\partial a_k}, \quad \text{where} \quad y_k = \frac{1}{1+e^{-a_k}}$$

$$\Rightarrow e^{-a_k} y_k + y_k = 1$$

$$\Rightarrow e^{-a_k} = \frac{1-y_k}{y_k}$$

$$\Rightarrow -a_k = \ln\left(\frac{1-y_k}{y_k}\right)$$

$$\Rightarrow a_k = \ln\left(\frac{y_k}{1-y_k}\right)$$

$$\rightarrow \frac{\partial E_k}{\partial y_k} = \frac{-t_k}{y_k} - \frac{1-t_k}{1-y_k}$$

$$= \frac{y_k - t_k}{y_k(1-y_k)}$$

$$\rightarrow \frac{\partial y_k}{\partial a_k} = \frac{1}{\frac{\partial a_k}{\partial y_k}} = \frac{1}{\frac{1}{y_k} + \frac{1}{1-y_k}}$$

$$\frac{\partial y_k}{\partial a_k} = y_k(1-y_k)$$

$$\text{LHS} \Rightarrow \frac{\partial E_k}{\partial a_k} = \frac{y_k - t_k}{y_k(1-y_k)} \cdot y_k(1-y_k) = y_k - t_k \Rightarrow \text{RHS}$$

## Question 5

Input $\Rightarrow 4 \times 4$

Filter $\Rightarrow 2 \times 2$

Output $\Rightarrow 3 \times 3$

Stride $= 1$

| | | |
|---|---|---|
| $-4+0+0+36 = 32$ | $-10+0+24+0 = 14$ | $6+0+0-24 = -18$ |
| $0+0-4-18 = -22$ | $-12+0-12+0 = -24$ | $0+0+0+12 = 12$ |
| $2+0+20+0 = 22$ | $6+0+0+0 = 6$ | $0+0+0+18 = 18$ |

$=$

| | | |
|---|---|---|
| 32 | 14 | -18 |
| -22 | -24 | 12 |
| 22 | 6 | 18 |

## Question 6

Stride $= 2$

Input - padded $= (0, x_1, x_2, x_3, x_4, 0)$

Filters $= (W_1, W_2, W_3)$

$\rightarrow$ 2 output activations :- $y_1 = 0 \cdot W_1 + x_1 \cdot W_2 + x_2 W_3$

$y_2 = x_2 W_1 + x_3 W_2 + x_4 W_3$

$\rightarrow$ We know, $y = Ax^T$ $\quad x^T = \begin{bmatrix} 0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ 0 \end{bmatrix}$ $\quad y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$ $\Rightarrow A = \begin{bmatrix} W_1 & W_2 & W_3 & 0 & 0 & 0 \\ 0 & 0 & W_1 & W_2 & W_3 & 0 \end{bmatrix}$

Transpose $\rightarrow$ Input $y = (y_1, y_2)$

Filter $= (W_1, W_2, W_3)$

$$z = A^T y \Rightarrow \begin{bmatrix} w_1 & 0 \\ w_2 & 0 \\ w_3 & w_1 \\ 0 & w_2 \\ 0 & w_3 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} w_1 y_1 \\ w_2 y_1 \\ w_3 y_1 + w_1 y_2 \\ w_2 y_2 \\ w_3 y_2 \\ 0 \end{bmatrix}$$

## Question 7

Input $X = \begin{bmatrix} 1 & 1 & 2 & 1 \\ 2 & 1 & 2 & 0 \\ 3 & 2 & 4 & 1 \\ 1 & 1 & 2 & 1 \end{bmatrix}$

### Encoder :-

$$W_1 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ -1 & 0 & 1 & 0 \end{bmatrix} \qquad W_2 = \begin{bmatrix} 1 & 0 & 1 \\ -1 & 1 & 0 \end{bmatrix}$$

$$b_1 = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} \qquad b_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$W_1 X + b_1 = \begin{bmatrix} 2 & 2 & 4 & 2 \\ 5 & 3 & 6 & 1 \\ 1 & 0 & 1 & -1 \end{bmatrix}$$

$$\underbrace{\text{ReLU}(W_1 X + b_1)}_{p} = \begin{bmatrix} 2 & 2 & 4 & 2 \\ 5 & 3 & 6 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

$$W_2 p + b_2 = \begin{bmatrix} 3 & 2 & 5 & 2 \\ 3 & 1 & 2 & -1 \end{bmatrix}$$

$$\underbrace{\text{ReLU}(W_2 p + b_2)}_{z} = \begin{bmatrix} 3 & 2 & 5 & 2 \\ 3 & 1 & 2 & 0 \end{bmatrix}$$

Decoder :-

$$W_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & -1 \end{bmatrix} \qquad W_2 = \begin{bmatrix} 1 & 0 & -1 \\ 1 & -1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

$$b_1 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \qquad b_2 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ -3 \end{bmatrix}$$

$$W_1 Z + b_1 = \begin{bmatrix} 3 & 2 & 5 & 2 \\ 4 & 2 & 3 & 1 \\ 0 & 1 & 3 & 2 \end{bmatrix} \qquad W_2 A + b_2 = \begin{bmatrix} 3 & +1 & 2 & 0 \\ -1 & 0 & 2 & 1 \\ 1 & 2 & 4 & 3 \\ 1 & 0 & 3 & 0 \end{bmatrix}$$

$$\underset{\Downarrow \; A}{\mathrm{ReLU}(W_1 Z + b_1)} = \begin{bmatrix} 3 & 2 & 5 & 2 \\ 4 & 2 & 3 & 1 \\ 0 & 1 & 3 & 2 \end{bmatrix} \qquad \Downarrow \; Y$$

$$Y - Y' = \begin{bmatrix} 3 & 1 & 2 & 0 \\ -1 & 0 & 2 & 1 \\ 1 & 2 & 4 & 3 \\ 1 & 0 & 3 & 0 \end{bmatrix} - \begin{bmatrix} 1 & 1 & 2 & 1 \\ 2 & 1 & 2 & 0 \\ 3 & 2 & 4 & 1 \\ 1 & 1 & 2 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 & -1 \\ -3 & -1 & 0 & 1 \\ -2 & 0 & 0 & 2 \\ 0 & -1 & 1 & -1 \end{bmatrix}$$

$\Downarrow$

Reconstruction loss

MSE Loss

Gradients $\frac{\partial L}{\partial y} = (Y - Y') \times 2 = \begin{bmatrix} 4 & 0 & 0 & -2 \\ -6 & -2 & 0 & 2 \\ -4 & 0 & 0 & 4 \\ 0 & -2 & 2 & -2 \end{bmatrix}$