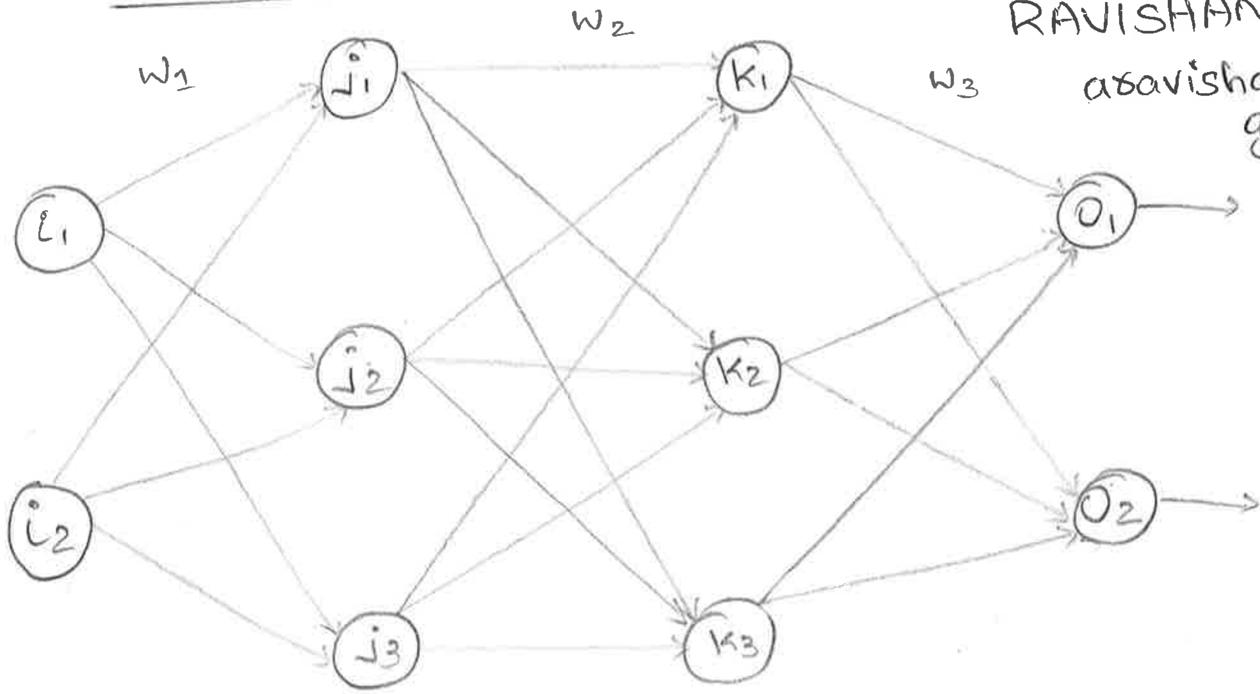
# CS 795 - HW 1

ASHWIN RAVISHANKAR
aravisha@gmu.edu






$$W_1 = W_{ij} = \begin{bmatrix} W_{i_1 j_1} & W_{i_1 j_2} & W_{i_1 j_3} \\ W_{i_2 j_1} & W_{i_2 j_2} & W_{i_2 j_3} \end{bmatrix}$$

$$W_2 = W_{jk} = \begin{bmatrix} W_{j_1 k_1} & W_{j_1 k_2} & W_{j_1 k_3} \\ W_{j_2 k_1} & W_{j_2 k_2} & W_{j_2 k_3} \\ W_{j_3 k_1} & W_{j_3 k_2} & W_{j_3 k_3} \end{bmatrix}$$

$$W_3 = W_{kO} = \begin{bmatrix} W_{k_1 O_1} & W_{k_1 O_2} \\ W_{k_2 O_1} & W_{k_2 O_2} \\ W_{k_3 O_1} & W_{k_3 O_2} \end{bmatrix}$$

Input = $[i_1, i_2]$
Hidden 1 = $[j_1, j_2, j_3]$
Hidden 2 = $[k_1, k_2, k_3]$
output = $[O_1, O_2]$

Activation:
-> Hidden: sigmoid
-> output: softmax

Sigmoid $\sigma(x) = \frac{1}{1+e^{-x}}$

Softmax $\sigma(x) = \frac{e^x}{e^x + e^y}$

loss function → $L = \|y - \hat{y}\|^2 = (y_1 - \hat{y}_1)^2 + (y_2 - \hat{y}_2)^2$

Neuron → $h_{in} \rightarrow (h) \rightarrow h_{out}$

$h_{in} = \Sigma W^T x$

$h_{out} = \sigma(h_{in})$.

Derivative → $\sigma'(x) = \sigma(x) \cdot (1 - \sigma(x))$
sigmoid

$\sigma'(x) = \sigma(x)[\delta_{ij} - \sigma(x)]$
softmax $\delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$

Hidden 1 - Forward Pass

$$\begin{bmatrix} j1_{in} & j2_{in} & j3_{in} \end{bmatrix}^T = \begin{bmatrix} W_{i1j1} & W_{i2j1} \\ W_{i1j2} & W_{i2j2} \\ W_{i1j3} & W_{i2j3} \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \end{bmatrix} = \begin{bmatrix} i_1 \cdot W_{i1j1} + i_2 W_{i2j1} \\ i_1 W_{i1j2} + i_2 W_{i2j2} \\ i_1 W_{i1j3} + i_2 W_{i2j3} \end{bmatrix}$$

$$\begin{bmatrix} j1_{out} & j2_{out} & j3_{out} \end{bmatrix}^T = \begin{bmatrix} \sigma(j1_{in}) & \sigma(j2_{in}) & \sigma(j3_{in}) \end{bmatrix}^T$$

Sigmoid

Hidden 2 - Forward Pass

$$\begin{bmatrix} K1_{in} & K2_{in} & K3_{in} \end{bmatrix}^T = \begin{bmatrix} W_{j1K1} & W_{j2K1} & W_{j3K1} \\ W_{j1K2} & W_{j2K2} & W_{j3K2} \\ W_{j1K3} & W_{j2K3} & W_{j3K3} \end{bmatrix} \begin{bmatrix} j1_{out} \\ j2_{out} \\ j3_{out} \end{bmatrix}$$

$$= \begin{bmatrix} j1_{out} W_{j1K1} + j2_{out} W_{j2K1} + j3_{out} W_{j3K1} \\ j1_{out} W_{j1K2} + j2_{out} W_{j2K2} + j3_{out} W_{j3K2} \\ j1_{out} W_{j1K3} + j2_{out} W_{j2K3} + j3_{out} W_{j3K3} \end{bmatrix}$$

$$\begin{bmatrix} K1_{out} & K2_{out} & K3_{out} \end{bmatrix}^T = \begin{bmatrix} \sigma(K1_{in}) & \sigma(K2_{in}) & \sigma(K3_{in}) \end{bmatrix}^T$$

Sigmoid.

Output Forward Pass

$$\begin{bmatrix} O1_{in} & O2_{in} \end{bmatrix}^T = \begin{bmatrix} W_{K1O1} & W_{K2O1} & W_{K3O1} \\ W_{K1O2} & W_{K2O2} & W_{K3O2} \end{bmatrix} \begin{bmatrix} K1_{out} \\ K2_{out} \\ K3_{out} \end{bmatrix}$$

$$= \begin{bmatrix} K1_{out} \cdot W_{K1O1} + K2_{out} W_{K2O1} + K3_{out} \cdot W_{K3O1} \\ K1_{out} \cdot W_{K1O2} + K2_{out} W_{K2O2} + K3_{out} \cdot W_{K3O2} \end{bmatrix}$$

$$\begin{bmatrix} O1_{out} & O2_{out} \end{bmatrix}^T = \begin{bmatrix} \sigma(O1_{in}) & \sigma(O2_{in}) \end{bmatrix}^T$$

Softmax

Total Loss:

$y_i$ - observed output

$\hat{y_i}$ - target output

$$L = \|y_i - \hat{y}_i\|^2 = (y_1 - \hat{y}_1)^2 + (y_2 - \hat{y}_2)^2$$
$$= (O1out - \hat{y}_1)^2 + (O2out - \hat{y}_2)^2$$

Activations:

$$\begin{bmatrix} j1out \\ j2out \\ j3out \end{bmatrix} = \begin{bmatrix} 1/1+e^{j1in} \\ 1/1+e^{j2in} \\ 1/1+e^{j3in} \end{bmatrix} \qquad \begin{bmatrix} K1out \\ K2out \\ K3out \end{bmatrix} = \begin{bmatrix} 1/1+e^{K1in} \\ 1/1+e^{K2in} \\ 1/1+e^{K3in} \end{bmatrix}$$

$$\begin{bmatrix} O1out \\ O2out \end{bmatrix} = \begin{bmatrix} \dfrac{e^{O1in}}{e^{O1in} + e^{O2in}} \\ \dfrac{e^{O2in}}{e^{O1in} + e^{O2in}} \end{bmatrix}$$

Also:

$y_1 = O1out$

$y_2 = O2out.$

$O = [O1 \quad O2]$

Back Propogation: output layer

$$\frac{\partial L_1}{\partial O1out} = 2(O1out - \hat{y}_1) \qquad \frac{\partial L_2}{\partial O2out} = 2(O2out - \hat{y}_2)$$

$$\frac{\partial O1out}{\partial O1in} = O1out(1 - O1out) \qquad \frac{\partial O2out}{\partial O2in} = O2out(1 - O2out)$$

$$\frac{\partial L}{\partial w_3} = \frac{\partial L}{\partial O_{out}} \cdot \frac{\partial O_{out}}{\partial O_{in}} \frac{\partial O_{in}}{\partial w_3} = 2(O_{out} - \hat{y}) \cdot O_{out}(1 - O_{out}) \cdot K_{out}$$

$$\boxed{\frac{\partial L}{\partial w_3} = \frac{\partial L_1}{\partial w_3} + \frac{\partial L_2}{\partial w_3} = \sum_{m=1,2} 2(y_m - \hat{y}_m)\, y_m(1 - y_m)\, K_{out} \quad m \in 1,2}$$

$L = L_1 + L_2.$

$K_{out} = [K1out \quad K2out \quad K3out]$

$$\frac{\partial L}{\partial W_3} \begin{cases} 2\,(O1out - \hat{y}_1)\; O1out\,(1 - O1out)\; Kout \\ 2\,(O2out - \hat{y}_2)\; O2out\,(1 - O2out)\; Kout \end{cases}$$

Back Propogation - Hidden Layer 2

$$Kout = [K1out \quad K2out \quad K3out]$$

$$\frac{\partial Kout}{\partial W_2} = \frac{\partial Kout}{\partial Kin} \frac{\partial Kin}{\partial W_2} = Kout \cdot (1 - Kout) \cdot Jout$$

$$\frac{\partial L}{\partial Kout} = \frac{\partial L}{\partial Oout} \frac{\partial Oout}{\partial Oin} \frac{\partial Oin}{\partial Kout} = 2(Oout - \hat{y}_m) \cdot Oout\,(1 - Oout)\, W_3$$

$$\boxed{\frac{\partial L}{\partial W_2} = \frac{\partial L}{\partial Kout} \cdot \frac{\partial Kout}{\partial W_2} = Kout\,(1 - Kout) \cdot Jout \cdot 2\,(Oout - \hat{y}_i)\; Oout\,(1 - Oout) \cdot W_3}$$

Back Propogation - Hidden Layer 1

$$Jout = [J1out \quad J2out \quad J3out]$$

$$\frac{\partial Jout}{\partial W_1} = \frac{\partial Jout}{\partial Jin} \frac{\partial Jin}{\partial W_1} = Jout\,(1 - Jout) \cdot Iout$$

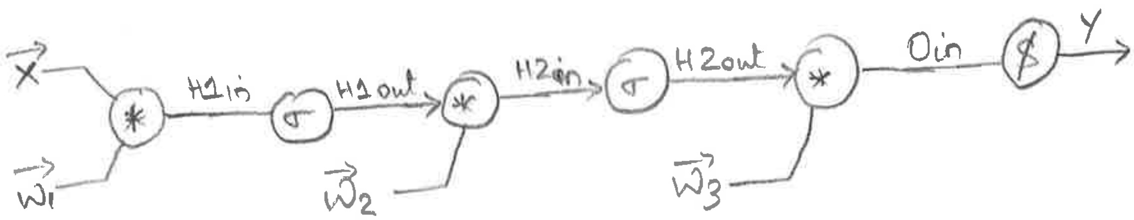
$$\frac{\partial L}{\partial Jout} = \frac{\partial L}{\partial Kout} \cdot \frac{\partial Kout}{\partial Kin} \frac{\partial Kin}{\partial Jout} =$$

$$= 2\,(Oout - \hat{y}_m) \cdot Oout\,(1 - Oout)\, W_3 \cdot Kout\,(1 - Kout)\, W_2$$

$$\frac{\partial L}{\partial W_1} = \frac{\partial L}{\partial J_{out}} \cdot \frac{\partial J_{out}}{\partial W_1}$$

$$= 2(O_{out} - \hat{y}_m)\, O_{out}(1 - O_{out})\, W_3\, K_{out}(1 - K_{out})\, W_2 \cdot J_{out}(1 - J_{out}) \cdot I_{out}$$

## Computational Graph:




$\vec{X}$ - Input vector
$\vec{W_1}, \vec{W_2}, \vec{W_3}$ - weight matrix
$*$ - dot product
$\sigma$ - Sigmoid activation function
$ - Softmax activation function

# CS 795 – Deep Learning – Home Work 1

Ashwin Ravishankar
aravisha@gmu.edu

Runtime Metrics:

| Sl. No | Metric | Model | |
|---|---|---|---|
| | | Keras Framework | My Implementation |
| 1 | Accuracy | 87.36% | 87.76% |
| 2 | Precision | 0.853968254 | 0.876190476 |
| 3 | Recall | 0.890728477 | 0.880382775 |
| 4 | F1-Score | 0.871961102 | 0.878281623 |