Assignment-2

AI 5100: Deep Learning
SM20MTECH12003.
Raj Kumar Surana.

Question -1 -

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \underbrace{\begin{bmatrix} a & b \\ c & d \end{bmatrix}}_{A} * \begin{bmatrix} x \\ y \end{bmatrix} + \underbrace{\begin{bmatrix} t_x \\ t_y \end{bmatrix}}_{t}$$

Assuming affine transformation with 6 parameters of A & t.

To estimate the transformation minimum 3 points are needed.

s = no. of sample points required = 3

e = probability that a point is outlier = 1/2

N = Number of iterations required for 95% chance of computing correct homography

P = Probability that atleast one sample out of N iterations is only composed of inliers
= 95/100

Teacher's Sign. .....

$$P = 1 - \left(1 - (1-e)^5\right)^N = \frac{95}{100}$$

$$\left(1 - (1-e)^5\right)^N = \frac{5}{100}$$

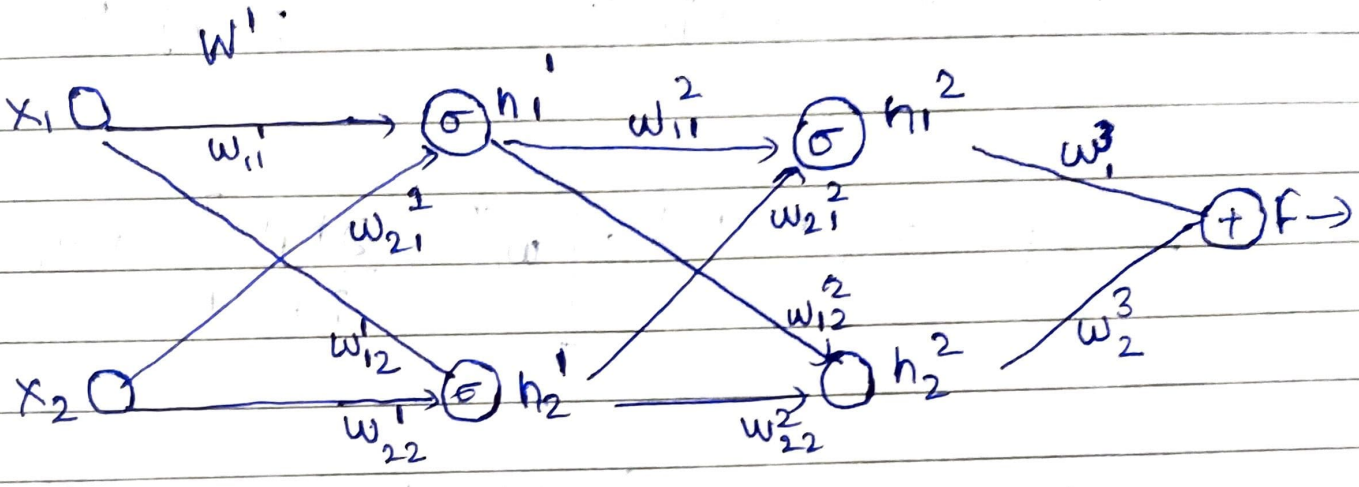
$$\left(1 - \left(1 - \frac{1}{2}\right)^3\right)^N = \frac{5}{100}$$

$$\left(\frac{7}{8}\right)^N = \frac{5}{100}$$

$$N = 22.43$$

$$\therefore 23 \text{ Ans.}$$

Question-2-



$x \xrightarrow{\cdot w^{[1]}} w^{[1]} x (\delta^{[1]})$

$h^{[2]}\ w^{[3]} = f$

$\sigma(\delta^{[1]}) = h^{[1]}$

Last Layer

$h^{[1]} \cdot w^{[2]} = \delta^{[2]}$

$\sigma(\delta^{2}) = h^{[2]}$

Ans→

$$\frac{\partial f}{\partial w_{ij}^{[1]}} = x_i \times \sigma'(\delta_j^{[1]}) \times \left[w_{j1}^{[2]} \times \sigma'[\delta_1^{[2]}] \times w_1^{[3]} + w_{j2}^{[2]} \times \sigma'[\delta_2^{[2]}] \times w_2^{[3]} \right]$$

Teacher's Sign.

Proof →

$$\frac{\partial F}{\partial w_{ij}^{[1]}} = \frac{\partial h_j^{[1]}}{\partial w_{ij}^{[1]}} \times \frac{\partial f}{\partial h_j^{[1]}}$$

$$\rightarrow \frac{\partial h_j^{[1]}}{\partial z_j^{[1]}} \times \frac{\partial z_j^{[1]}}{\partial w_{ij}^{[1]}} \times \left[\frac{\partial \left(h_1^{[2]} w_1^{[3]} + h_2^{[2]} w_2^{[3]} \right)}{\partial h_j^{[1]}} \right]$$

$$\rightarrow x_i \times \sigma'(z_j^{[1]}) \times \left[w_1^{[3]} \times \frac{\partial h_1^{[2]}}{\partial h_j^{[1]}} + w_2^{[3]} \times \frac{\partial h_2^{[2]}}{\partial h_j^{[1]}} \right]$$

$$\rightarrow x_i \times \sigma'[z_j^{[1]}] \times \left[w_1^{[3]} \times \sigma'[z_1^{[2]}] \times w_{j1}^{[2]} + w_2^{[3]} \times \sigma'[z_2^{[2]}] \times w_{j2}^{[2]} \right]$$

Ans ✓

Question -3 -

$$\Delta w_{ij}^{[2]} = \Delta w_{ij}^{[2]} + \delta_i^{[3]} \cdot a_j^{[2]}$$

Δ_{ij} can be considered an element in the ith row & jth column of matrix Δ

$\delta_i^{[3]} \cdot a_j^{[2]}$

we have to write 2 vectors $\delta^{[3]}$ & $a^{[2]}$ in such a way that their multiplication result in a matrix whose ith raw & jth column element can be represented by $\delta_i^{[3]} \cdot a_j^{[2]}$

Let $\delta^{[3]}$ is

$$\begin{bmatrix} \delta_0^{[3]} \\ \delta_1^{[3]} \\ \delta_2^{[3]} \\ \vdots \\ \delta_i^{[3]} \end{bmatrix}$$

& $a^{[2]}$ is

$$\begin{bmatrix} a_0^{[2]} \\ a_1^{[2]} \\ \vdots \\ a_j^{[2]} \\ \vdots \end{bmatrix}$$

$\delta^{[3]} \cdot (a^{[2]})^T$ will give a matrix whose ith raw, jth column element will be represented by $\delta_i^{[3]} \cdot a_j^{[2]}$

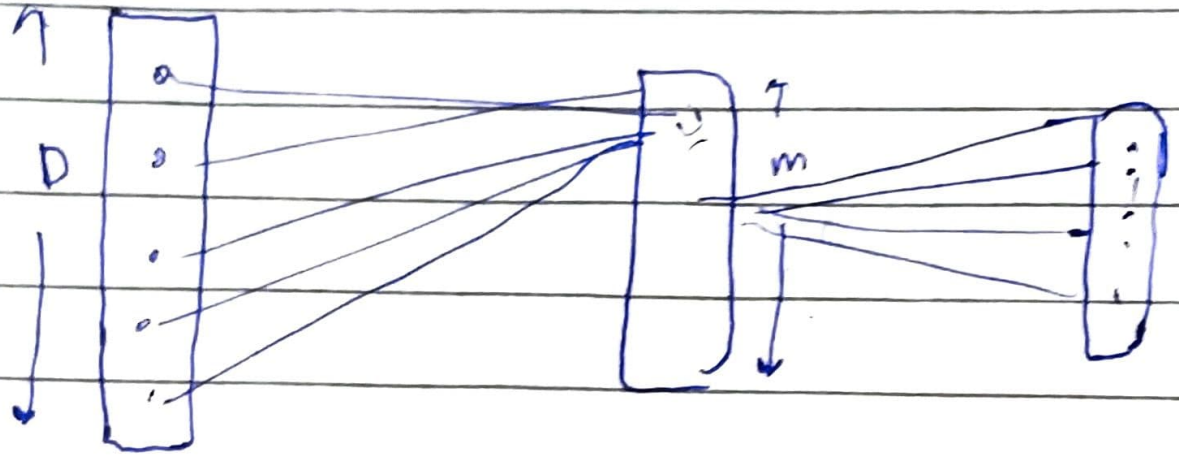
So finally

$$\Delta^{[2]} = \Delta^{[2]} + \delta^{[3]} \cdot a^{[2]}$$

Teacher's Sign.

Question-4 →

Independent Derivatives = ~~mxDxc~~ $M^2 \times C$



Size D input Layer

M size Hidden Layer

Size C output Layer

Total Number of weights = $(M \times D) + (M \times C)$

Bias = $M + C$

Question 5 :

Given $y_n = f(x_n; w) + \epsilon_n$

where ϵ_n is zero mean gaussian distributed error having fixed covariance matrix Σ.

so $E[y_i] = f(x_i; w) + E[\epsilon_n]$

$= f(x_i, w)$

& $var(y_i) = var(f(x_i; w) + \epsilon_n)$

$= var(\epsilon_n) = \Sigma$

$y \sim N(x | f(x_i; w), \Sigma)$

$$p[y | x, F, w, \Sigma] = \prod_{i=1}^{n} N(y_i | x_i, f(x_i; w), \Sigma)$$

$$L = \prod_{i=1}^{n} (2\pi\Sigma)^{-n/2} \cdot e^{\sum_{i=1}^{n} \frac{[y_i - f(x_i; w)]^2}{2\Sigma}}$$

Taking log of likelihood function we get

$$\log(L) = \frac{-n}{2}\log(2\pi\Sigma) \frac{-1}{2}\left[(y_i - F(x_i;w))^T \Sigma^{-1} \right.$$
$$\left. [y_i - F(x_i;w)]\right]$$

As maximising Log Likelihood is equivalent to the minimising sum of squared error

Therefore we have

$$\underset{\min}{\frac{\partial E(w)}{\partial w}} = \underset{\max}{\frac{\partial L(w)}{\partial w}} \Rightarrow \quad [\text{as per definition}]$$

$$\frac{\partial E(w)}{\partial w} = -0 - \frac{1}{2}\frac{\partial}{\partial w}\left[\sum_{i=1}^{n}\left[(y_i - f(x_i;w))^T \Sigma^{-1}(y_i - f(x_i;w))\right]\right.$$

Error function can be obtained by integrating the above equation.

$$\boxed{E(w)} = \frac{1}{2}\sum_{i=1}^{n}\left[(y_i - f(x_i;w))^T \Sigma^{-1} [y_i - f(x_i;w)]\right]$$

Error function

Question - 6→

(a)

As given in the question scale symmetry is assuming a simple scaling i.e. $\{\gamma, \theta_t\}$, $\gamma \in \mathbb{R}$ for weights in layer I and multiplying the next layer's weight by $1/\gamma$.

Doing this Loss function value remains the same, that's why it is called symmetry by scale.

→ But scale symmetry present in the weight space of a deep neural network causes failure of Euclidean gradient based SSD (Stochastic Gradient Descent) based optimization

Scale Symmetry result in unbalanced Networks & SSD performs very poorly on unbalanced Network

In such networks our model can converge on different Local minima

Question - 6-b

Permutation Symmetry:-

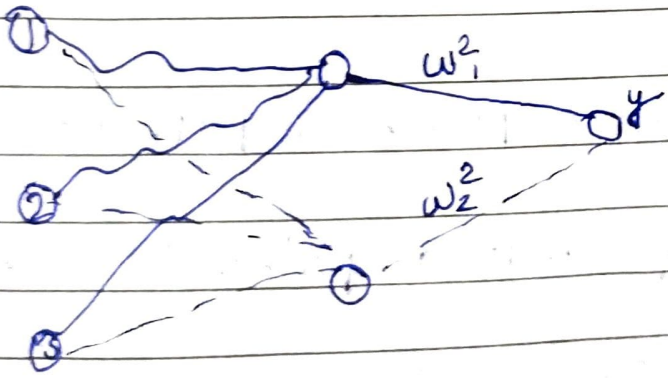
Teacher's Sign.

As per permutation symmetry we can get the equivalent models by swapping the incoming weight vectors & outgoing weight vectors of the neuron i & neuron j in the same layer of a trained network.

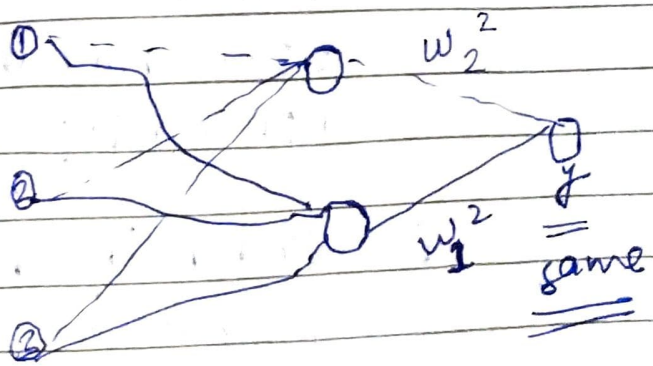
This means, for any neural network if there are million ~~weights~~ neurons, there can be millions of swap and these different weight combinations will give the same model error.

So there will be a lot of other minima points in weight space.

Example -



Swap



Teacher's Sign.....