

Neural Networks

- Artificial neural networks → Tabular
- Convolution neural network → images
- NLP (RNN, LSTM, Transformer) → text

ANN -

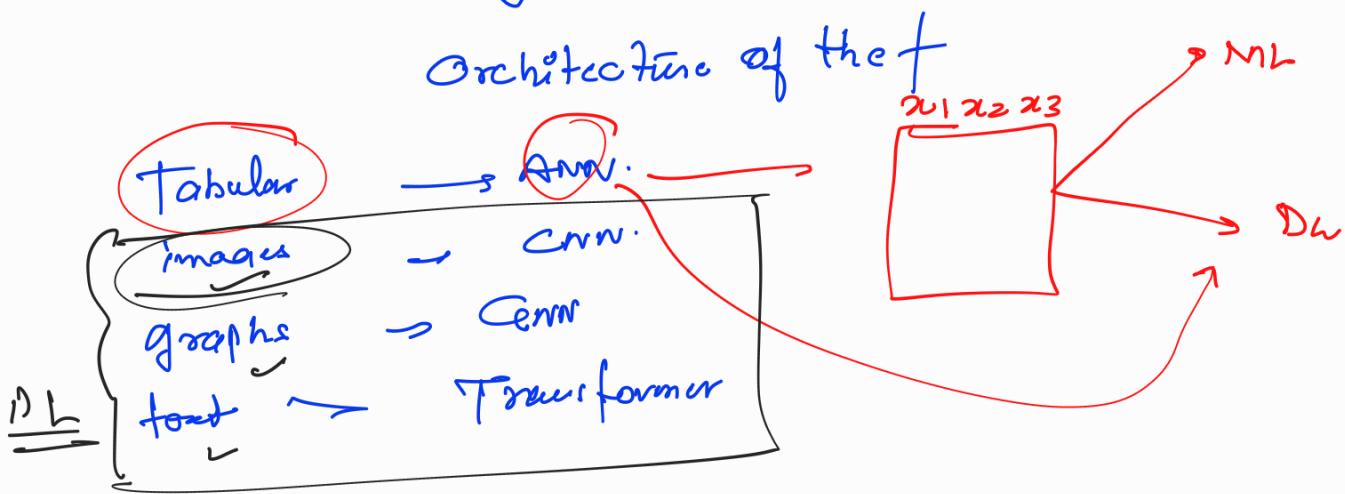
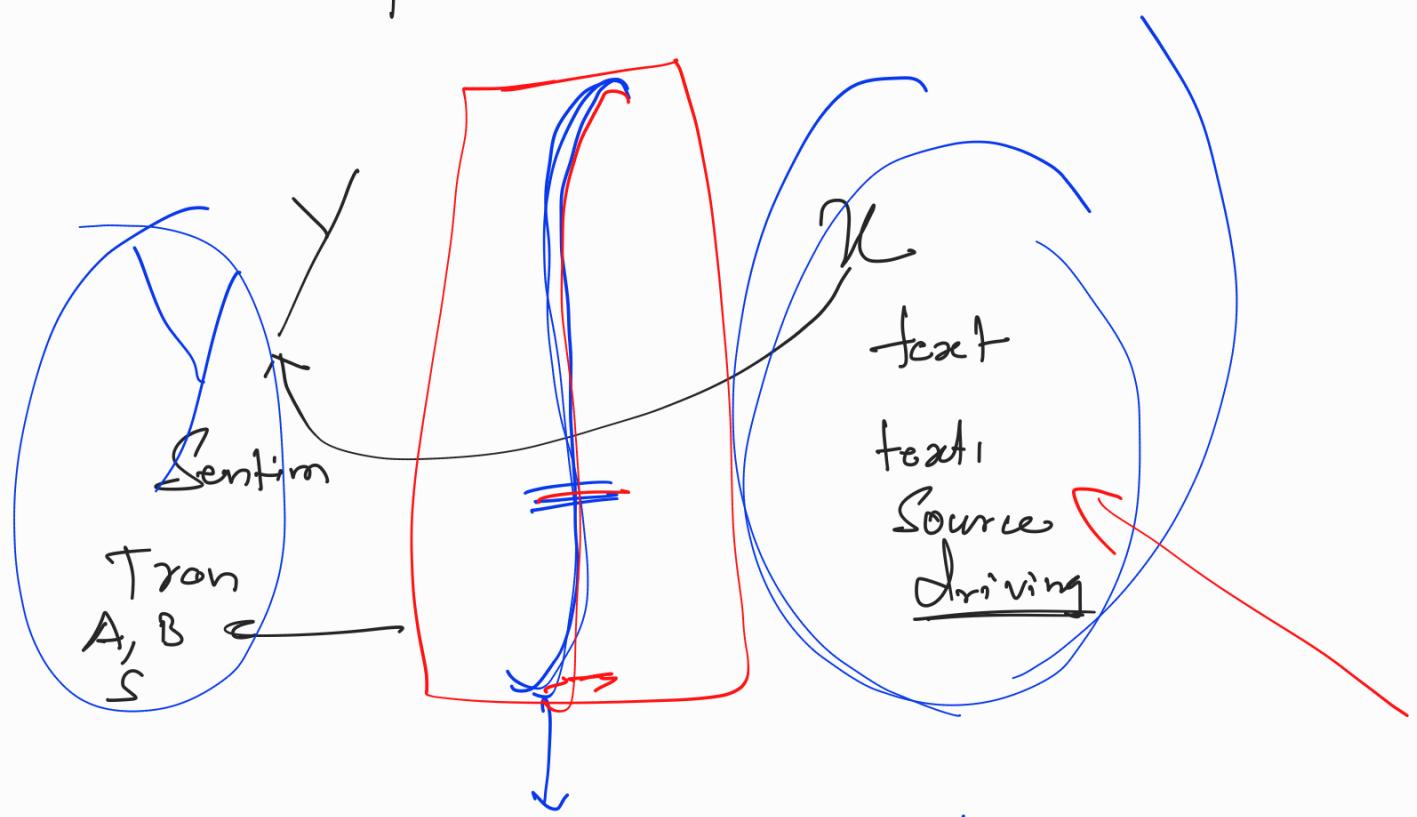
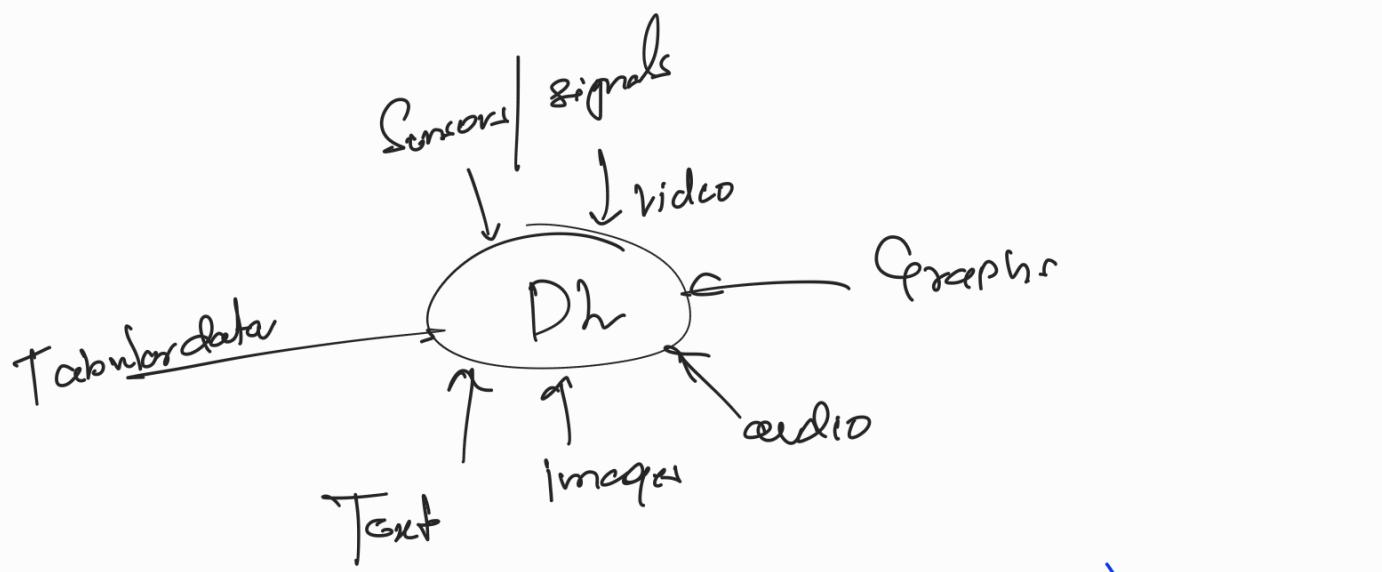
- Why DL / What is DL → What is ANN
- What is a (Neuron)
- What is MLP
- Forward Prop / Backward Prop.
- Training steps of ANN

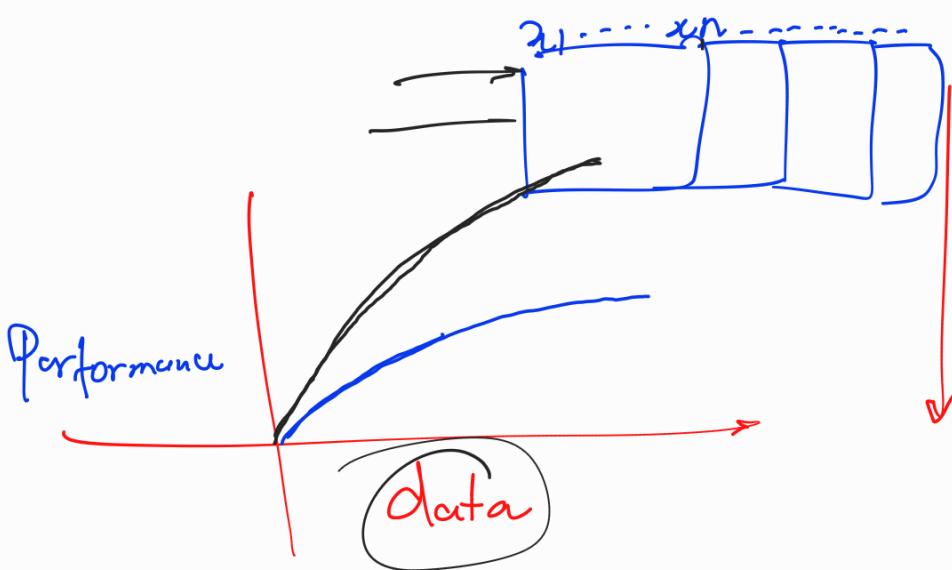
{ implement nn in a code (Tensorflow / Keras)
Dropout, Regularization, Batch Normalization, losses
Callbacks, Optimizers.

{ Explainability / Denoising
→ Autoencoders

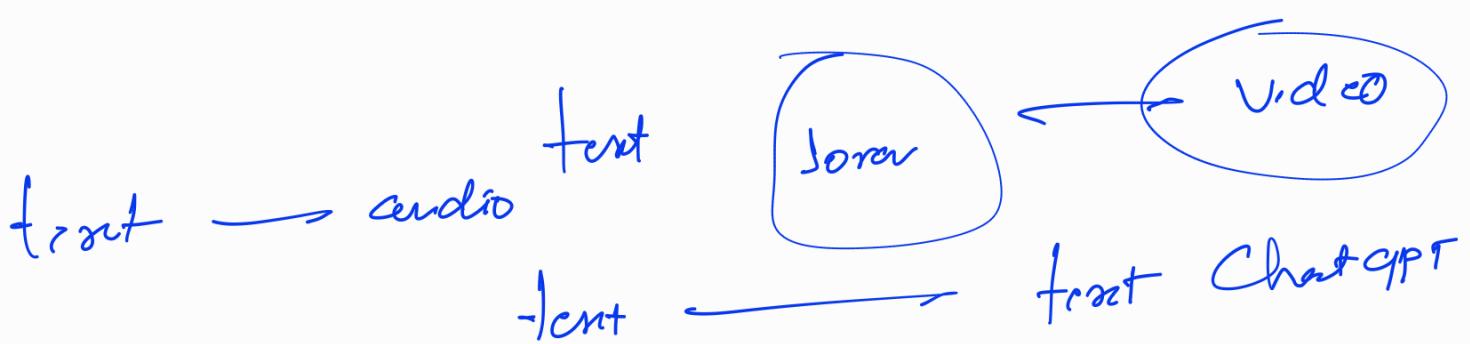
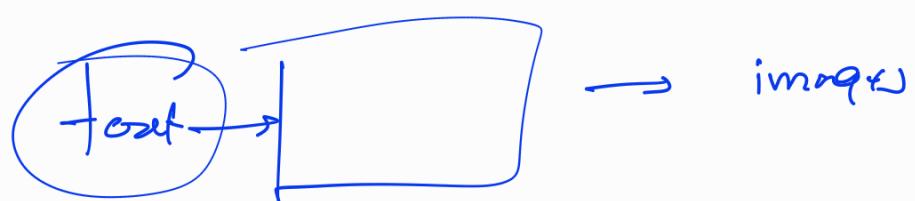
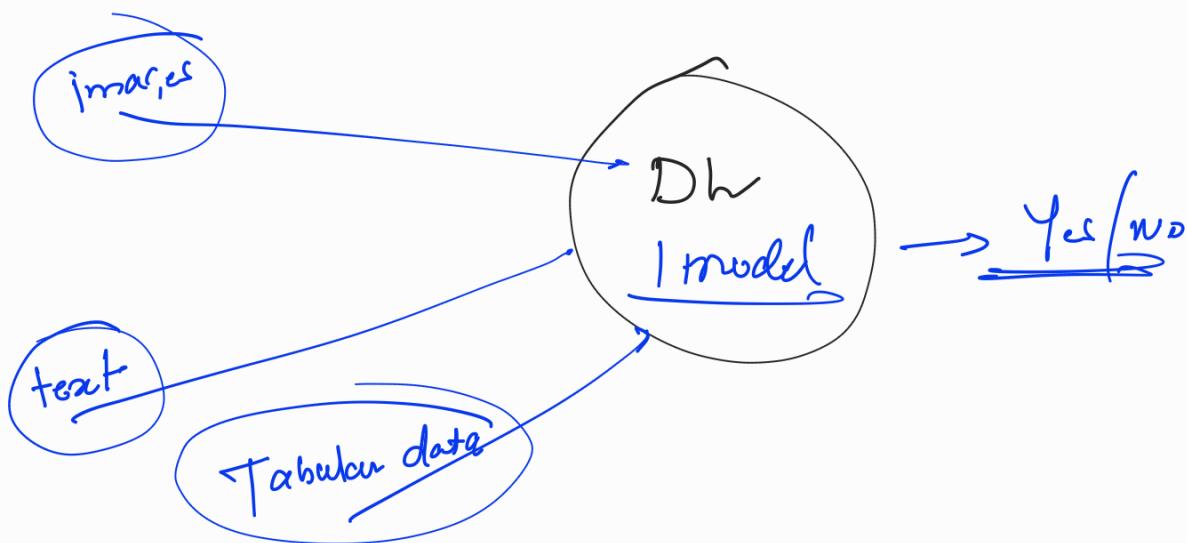
Optimization using ANN.

Deep learning is a field of study wherein we use parametric models which enables automated feature engineering.



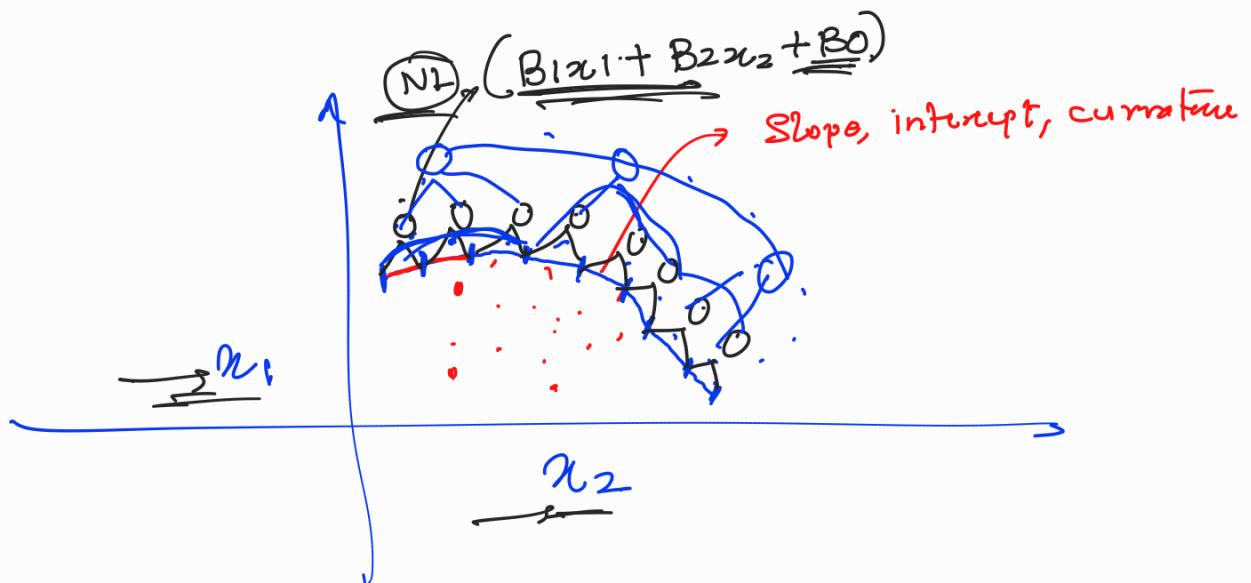


Multi-modality

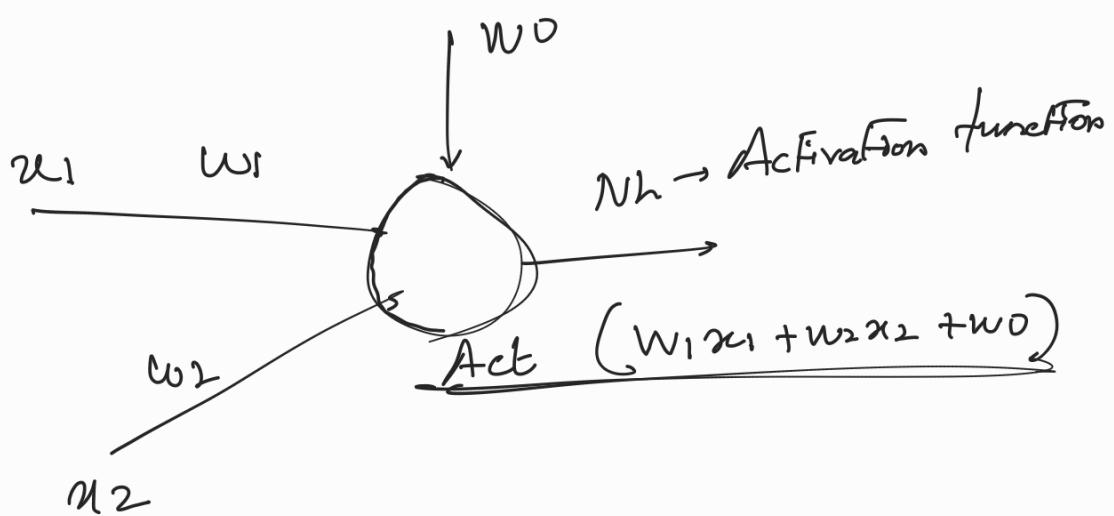
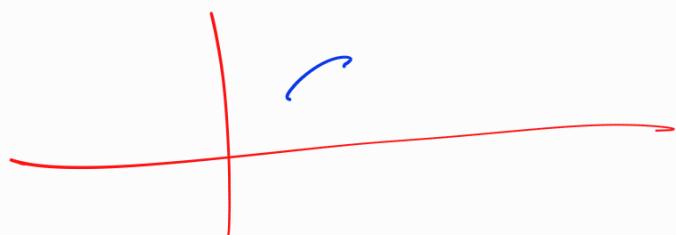


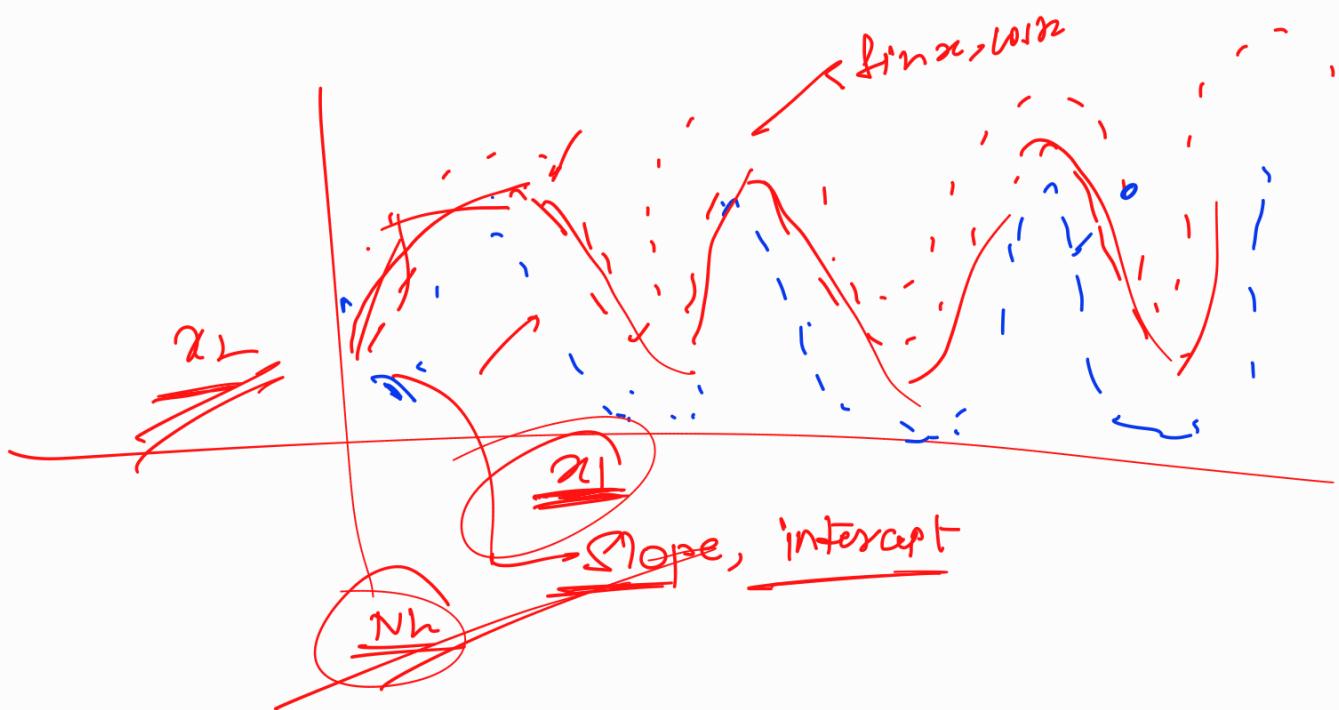
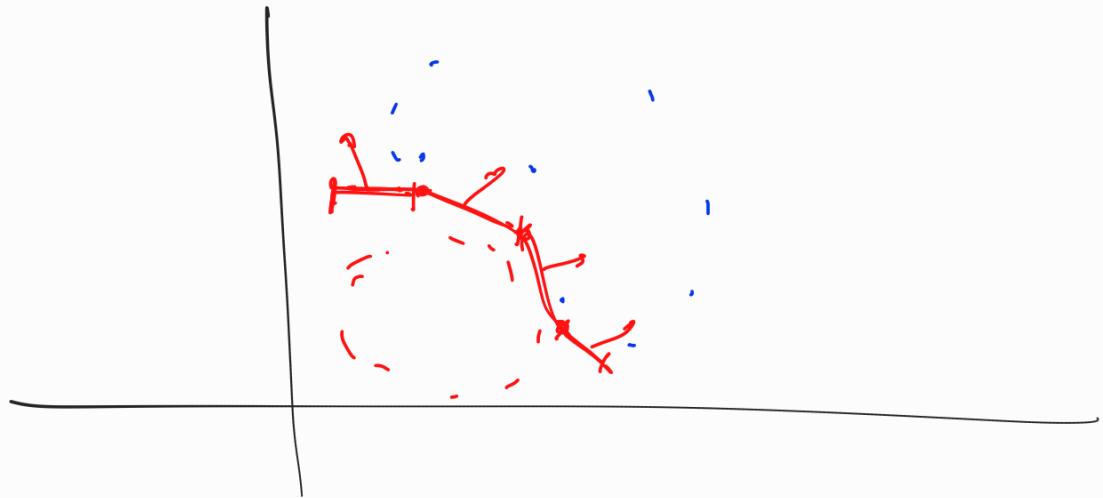
Dh

↳ Neurons

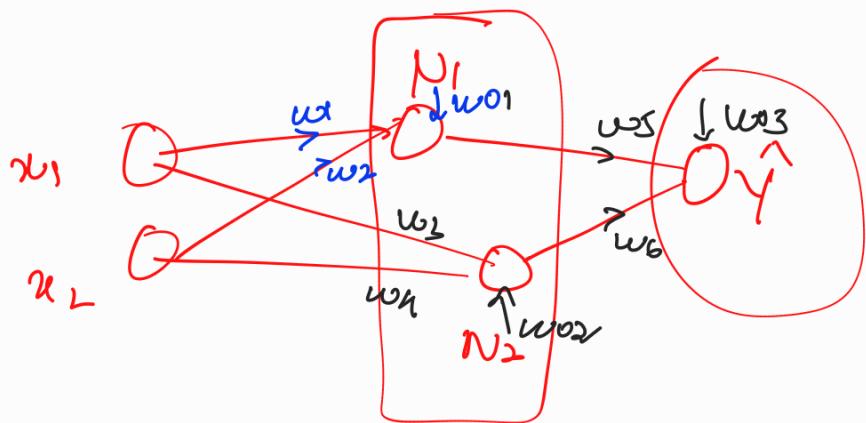


Can \neq use multiple "small"
Create this blue boundary. Curvature
implies migrations to





22:19 9m



$$N_1 = w_1 x_1 + w_2 x_2 + w_{01}$$

$$N_2 = w_3 x_1 + w_4 x_2 + w_{02}$$

$$\hat{y} = w_5 N_1 + w_6 N_2 + w_{03}$$

$$\hat{y} = w_5 (w_1 x_1 + w_2 x_2 + w_{01}) + w_6 (w_3 x_1 + w_4 x_2 + w_{02}) + w_{03}$$

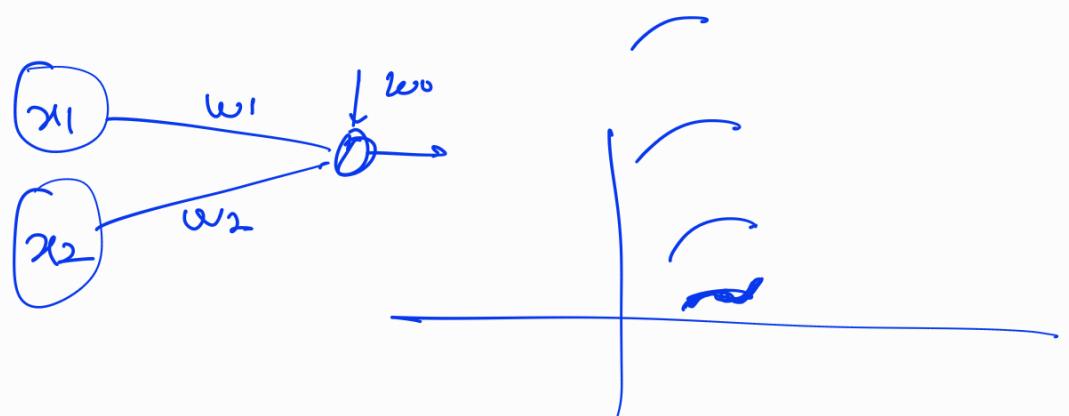
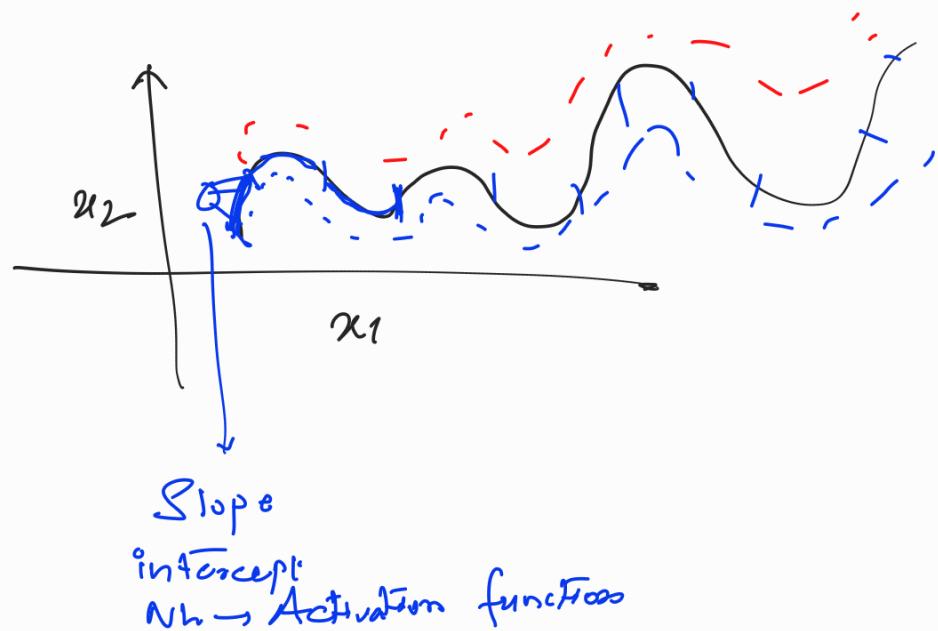
$$\begin{aligned} \hat{y} = & x_1 (w_5 w_1 + w_6 w_3) + x_2 (w_5 w_2 + w_6 w_4) \\ & + w_{01} w_5 + w_{02} w_6 + w_{03} \end{aligned}$$

$$\hat{y} = n_1 A + n_2 B + C.$$

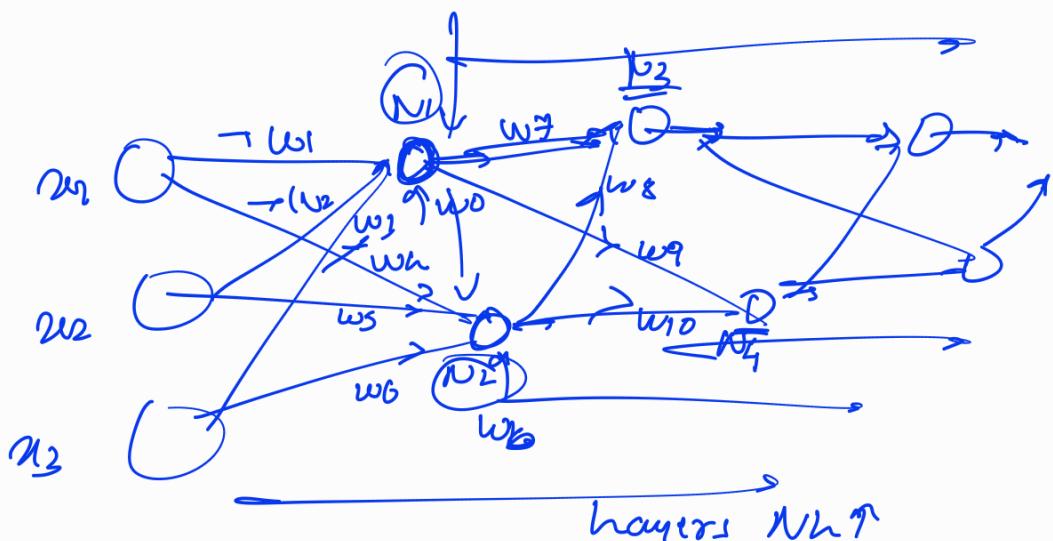
$$\boxed{\hat{y} = x_1 A + x_2 B + C}$$

22/04/2022

$B \rightarrow$ coefficient
 $Nh \rightarrow$ Activation



$\tanh \left(\underline{w_1 x_1 + w_2 x_2 + w_0} \right)$

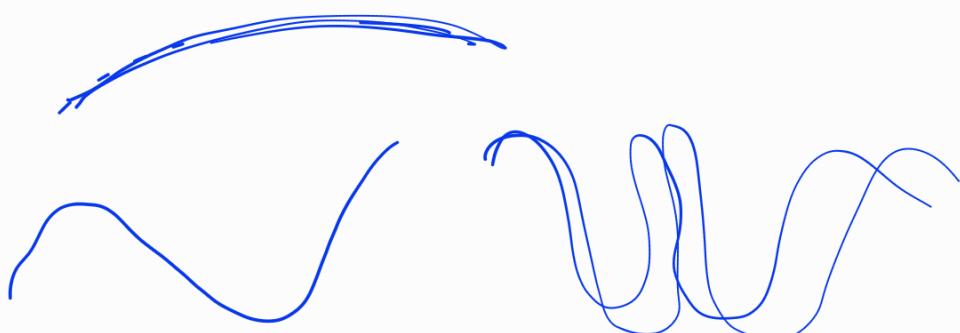


$$N_1 = \tanh(w_1x_1 + w_2x_2 + w_3x_3 + w_0)$$

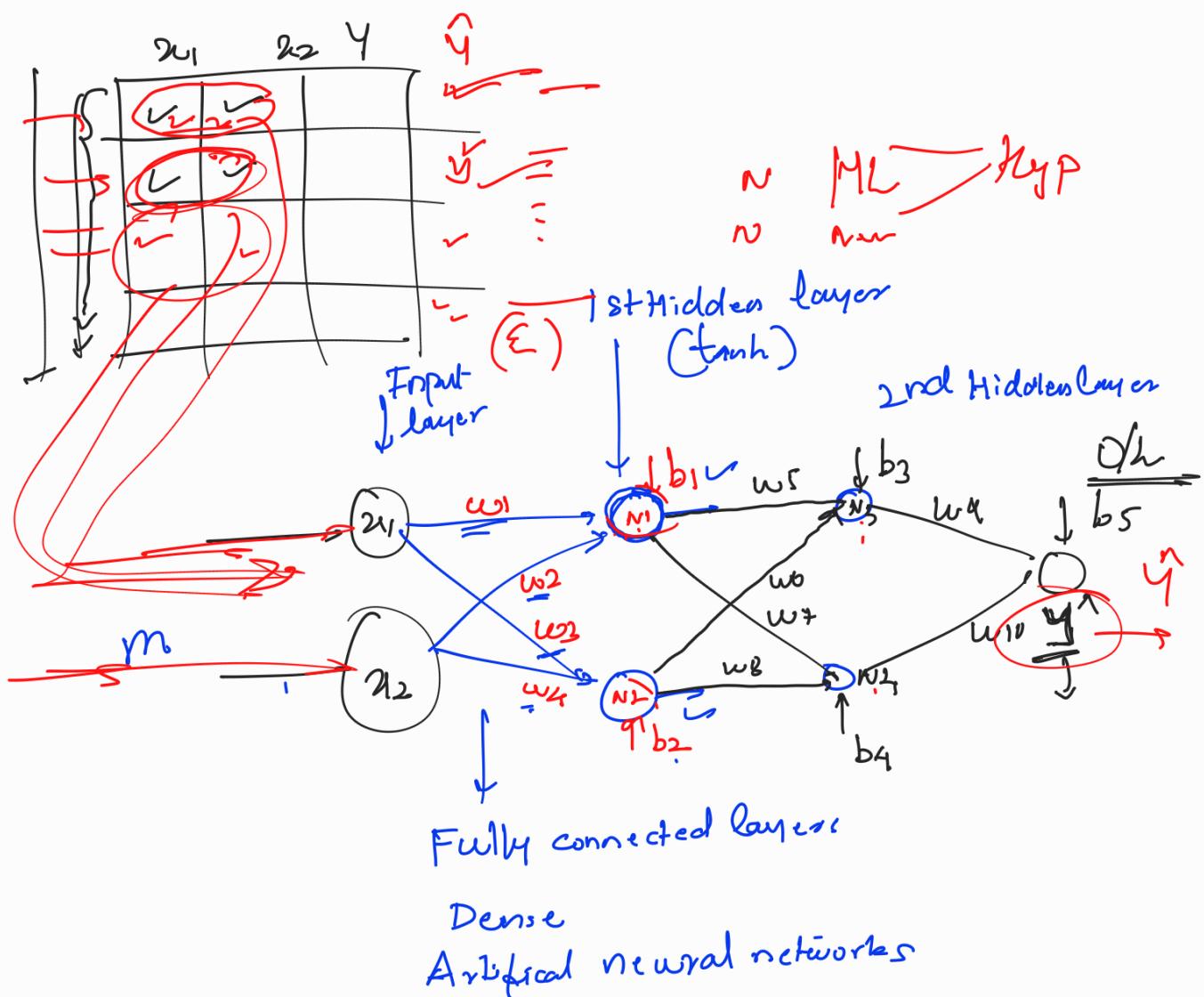
$$N_2 = \tanh(w_4x_1 + w_5x_2 + w_6x_3 + w_b)$$

$$N_3 = \tanh(w_7N_1 + w_8N_2 + b)$$

$$N_4 = \tanh(w_9N_1 + w_{10}N_2 + b)$$



Forward Propagation & Backward Propagation



As soon as the DC introduces the first hidden layers by mentioning the number of neurons in the HC, immediately all connections are established and weights are assigned randomly.

$$\text{Activation function: } N_1 = \tanh(w_1 x_1 + w_2 x_2 + b)$$

$$N_2 = \tanh(w_3 x_1 + w_4 x_2 + b)$$

$$N_3 = \tanh(w_5 N_1 + w_6 N_2 + b_3)$$

$$N_4 = \tanh(w_7 N_1 + w_8 N_2 + b_4)$$

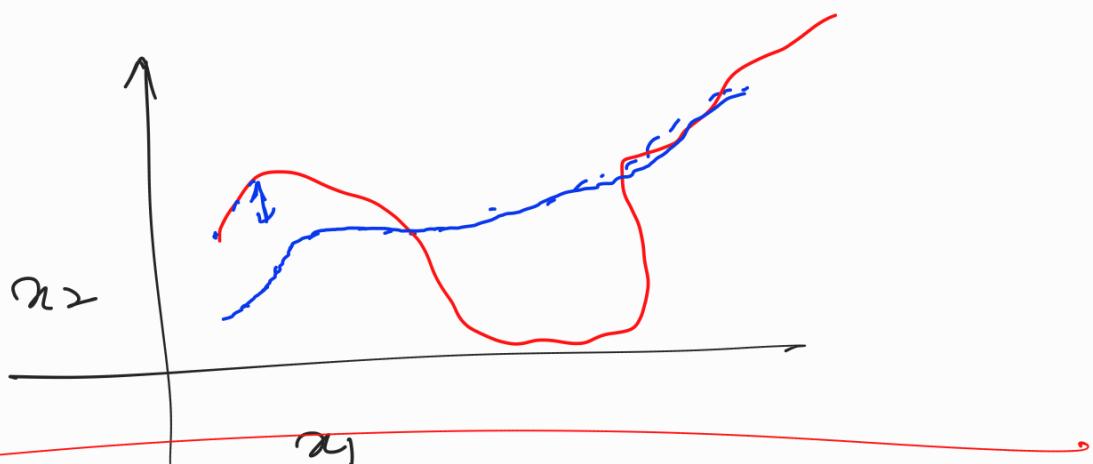
$$\hat{y} = (w_9 N_3 + w_{10} N_4 + b_5)$$

$$L = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

Forward Propagation

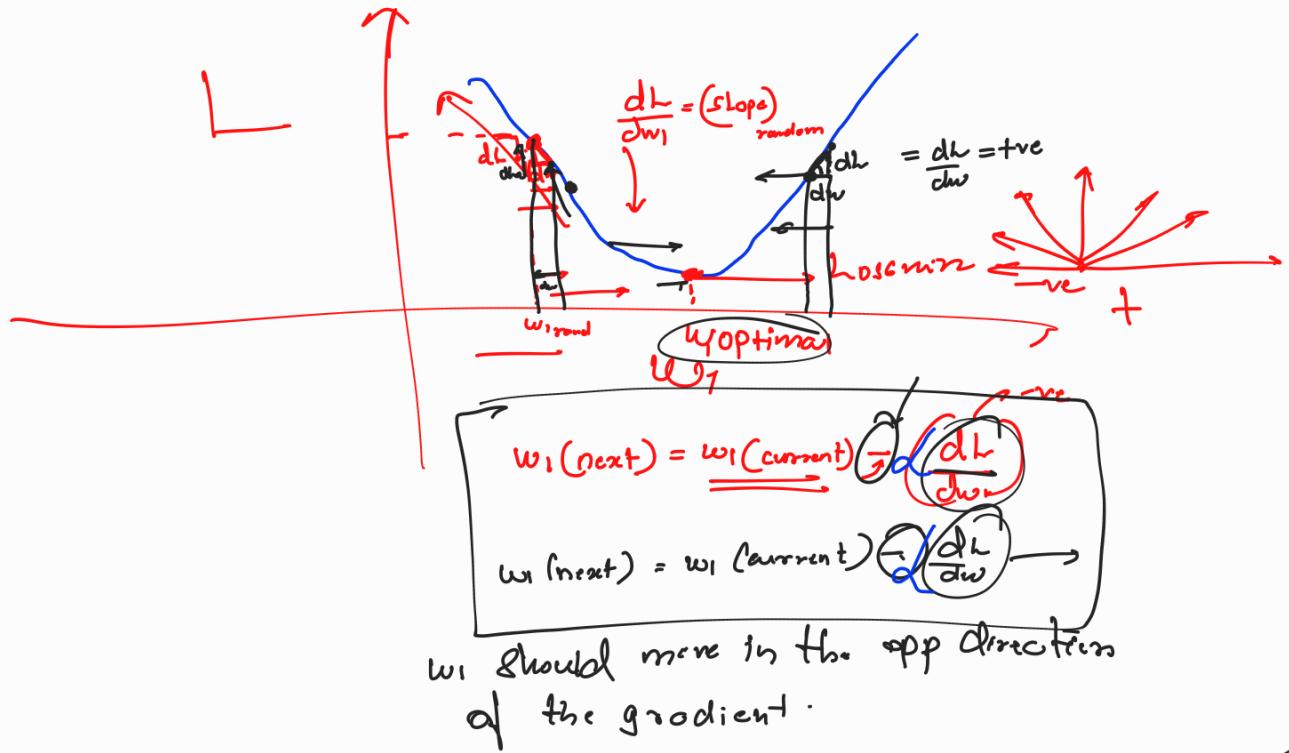
1st Forward Propagation

- ① Define the architecture of the NN (2HL, ...)
- ② The weights will be randomly assigned (d_{ij}).
- ③ Define the loss function
- ④ Take all the sample data and calculate the loss.



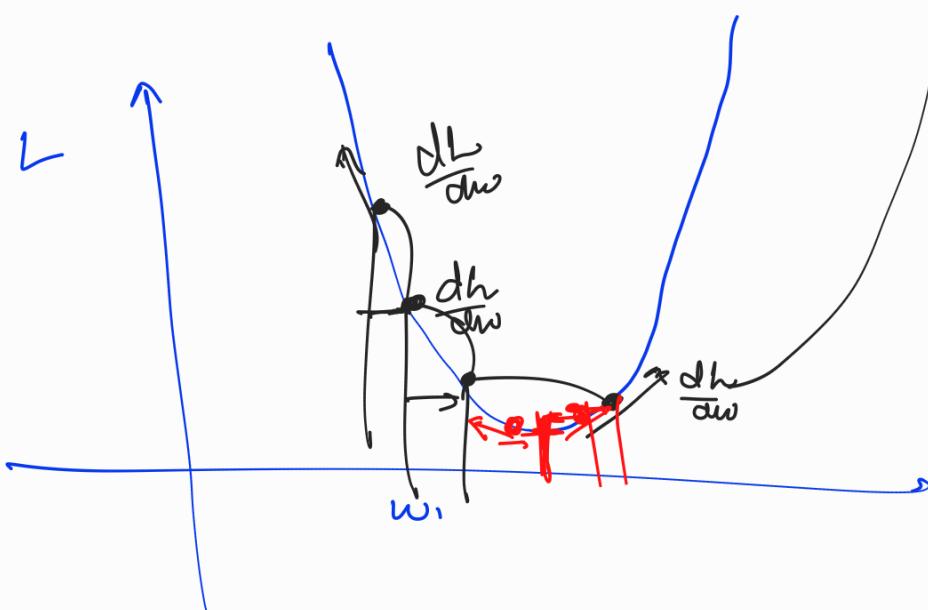
- ⑤ Minimise the loss by changing the weights
- Backward propagation \rightarrow Gradient Descent

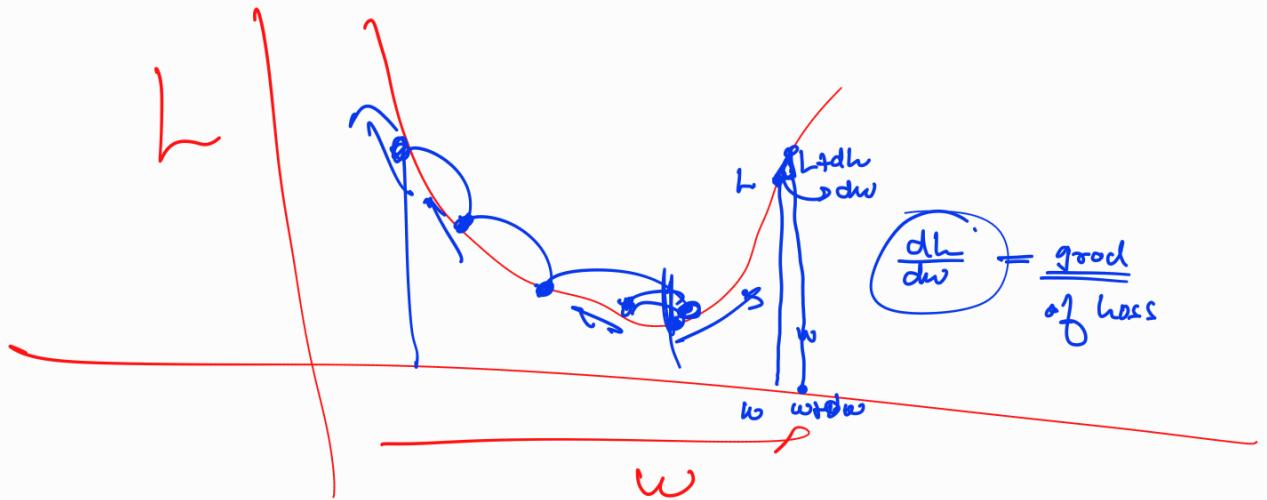
$$L = f(\underline{w})$$



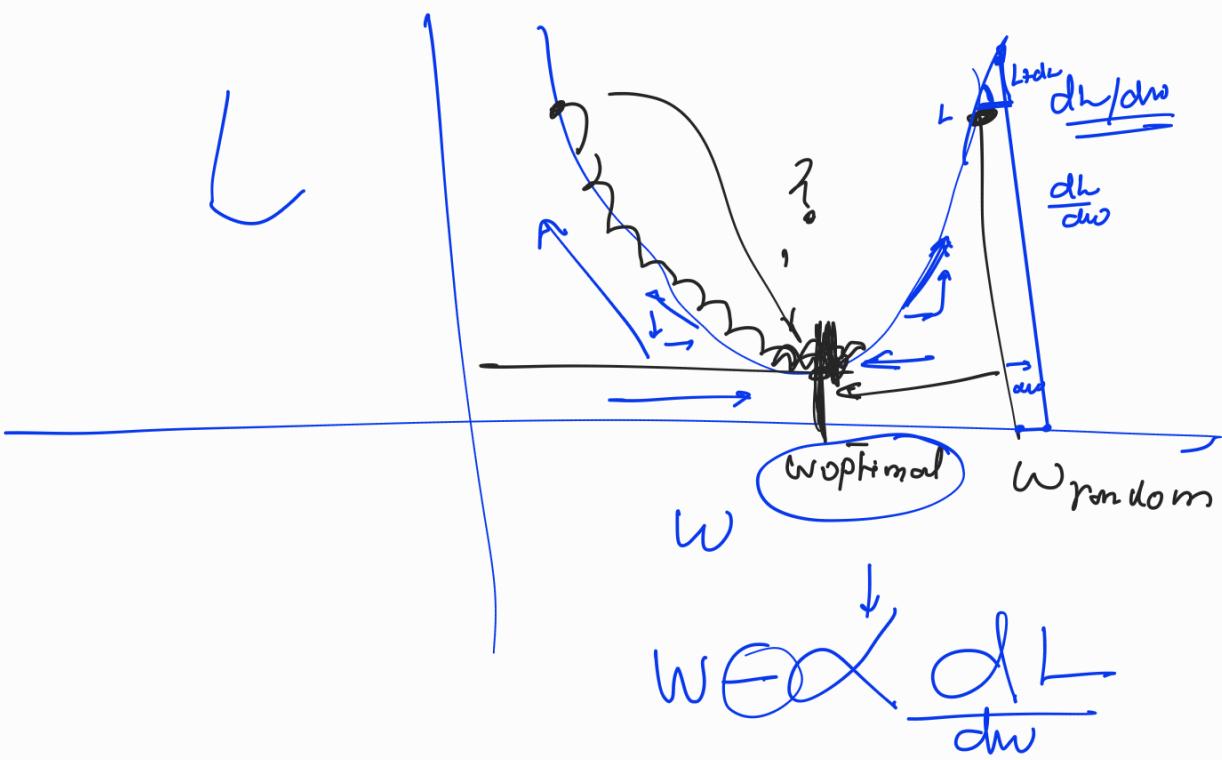
→

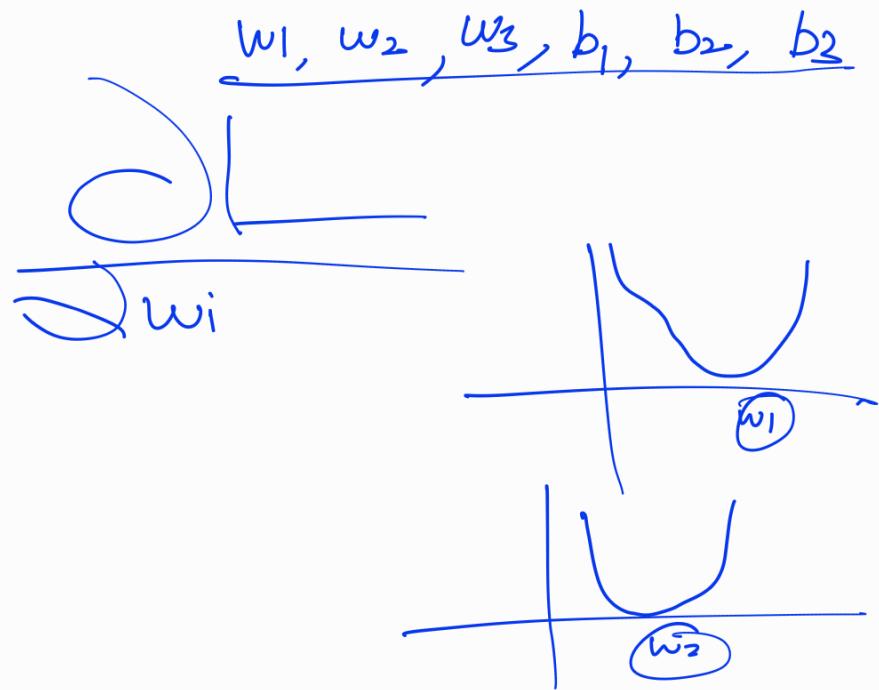
$$w_{\text{S}}^{(t+1)} = w_{\text{S}}^{(t)} - \alpha \frac{\partial L}{\partial w(t)}$$





$$\left. \frac{dh}{dw} \right|_w = \text{Slope of the curve at } w$$





Gradient Descent

Break. (22:20pm)

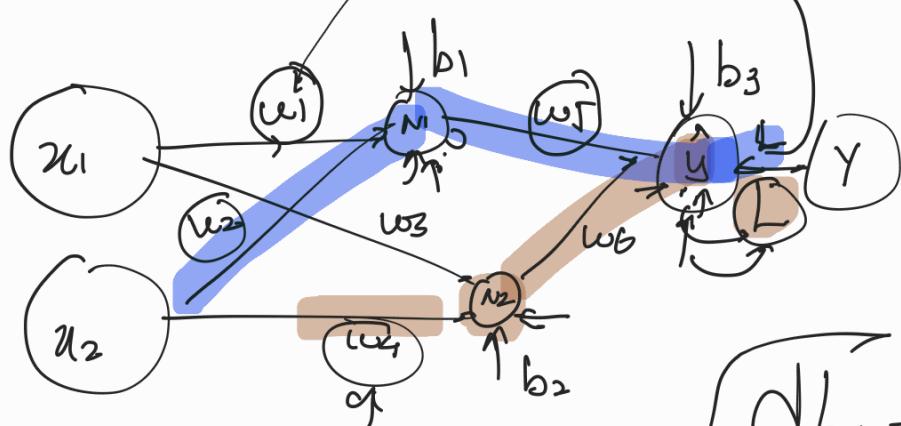
- ① Create the NN architecture and define the loss function.
- ① all weights are assigned randomly.
- ① Send all the (training data) through the architecture and calculate the loss for the entire data. } F.P.
- ② given the loss, find out the derivative of the loss w.r.t all weights & biases.
- ③ Perform gradient descent $(w_{t+1}) = w_t - \alpha \frac{dh}{dw_t})$ } → BP

⑥ update the weights)

⑦ Using the update weights, goto step 3.

⑧ The model's loss doesn't reduce further (saturation).

⑨ Each iteration of (I-FP + I-BP) \rightarrow 1 epoch.



$$N_1 = w_1 x_1 + w_2 x_2 + b_1$$

$$N_2 = w_3 x_1 + w_4 \underline{x_2} + b_2$$

$$\hat{y} = w_5 N_1 + w_6 N_2 + b_3$$

$$L = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\frac{\partial L}{\partial w_5} = \frac{\partial L}{\partial \hat{y}} \times \frac{\partial \hat{y}}{\partial w_5}$$

$$\frac{\partial L}{\partial w_6} = \frac{\partial L}{\partial \hat{y}} \times \frac{\partial \hat{y}}{\partial w_6}$$

$$\frac{\partial \hat{y}}{\partial x_2} = 2x$$

$$\rightarrow \frac{\partial L}{\partial w_1}, \frac{\partial L}{\partial w_2}, \frac{\partial L}{\partial w_3}, \dots, \frac{\partial L}{\partial w_6}, \frac{\partial L}{\partial b_1}, \frac{\partial L}{\partial b_2}, \frac{\partial L}{\partial b_3}$$

$$\left[\sum_{i=1}^n 2(y_i - \hat{y}_i)x_2 \right] [w_6] \times (x_2)$$

$$\frac{\partial L}{\partial w_1}$$

$$\frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial \hat{y}} \times \frac{\partial \hat{y}}{\partial w_2}$$

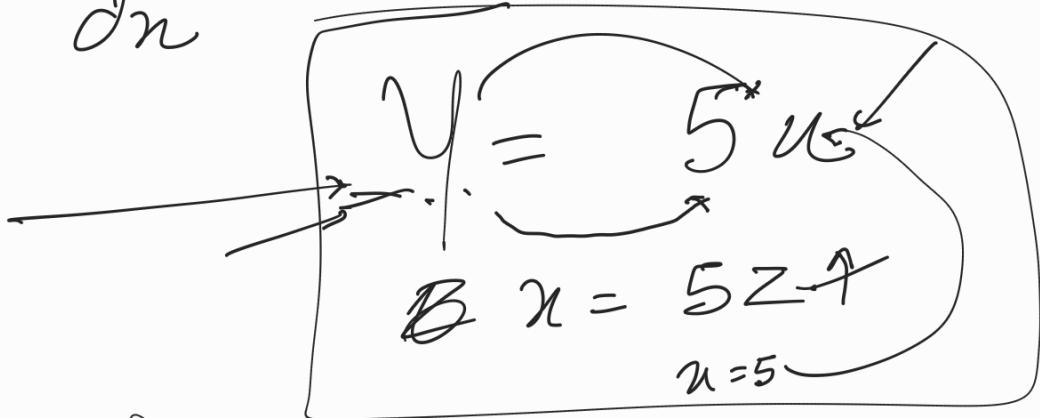
$$\frac{\partial L}{\partial \hat{y}} = \frac{\partial L}{\partial \hat{y}}$$

$$\times \frac{\partial \hat{y}}{\partial w_2} \times \frac{\partial w_2}{\partial w_2}$$

$$\frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial \hat{y}} \times \frac{\partial \hat{y}}{\partial N_1} \times \frac{\partial N_1}{\partial w_2}$$

$$\psi = 5\pi$$

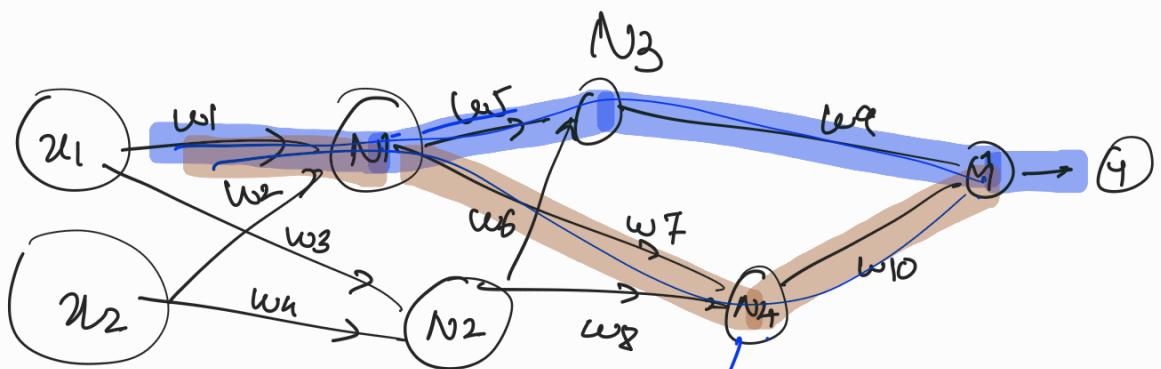
$$\frac{\partial Y}{\partial z} = 5$$



$$\frac{\partial Y}{\partial z} = \frac{\partial Y}{\partial x} \times \frac{\partial x}{\partial z} \rightarrow \text{Chain Rule}$$

$\frac{\partial h}{\partial w}$

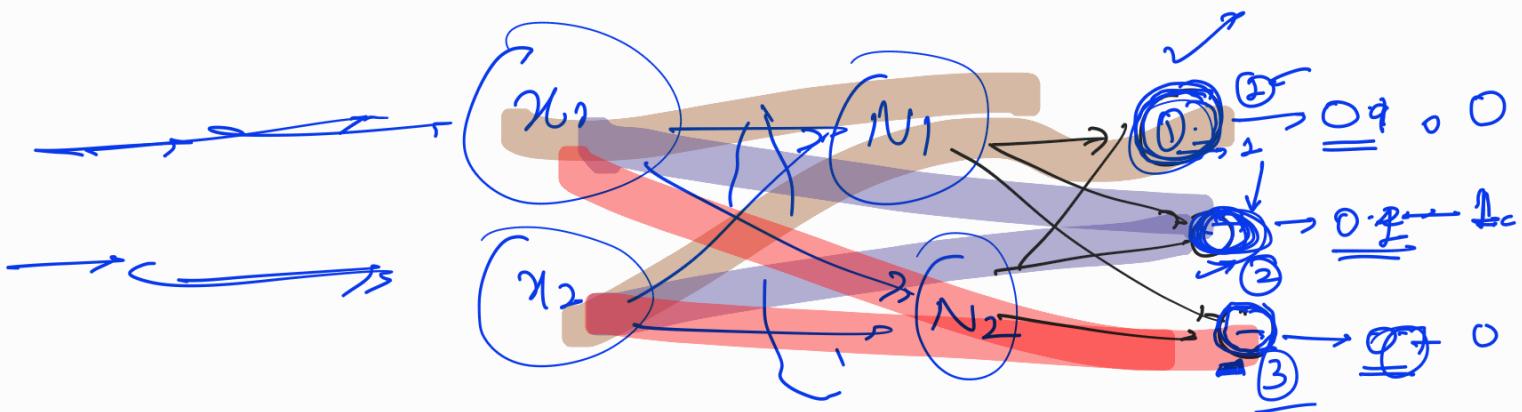
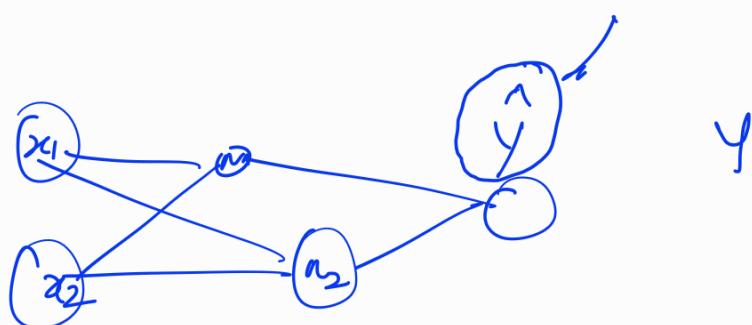
$$\frac{\partial Y}{\partial z}$$

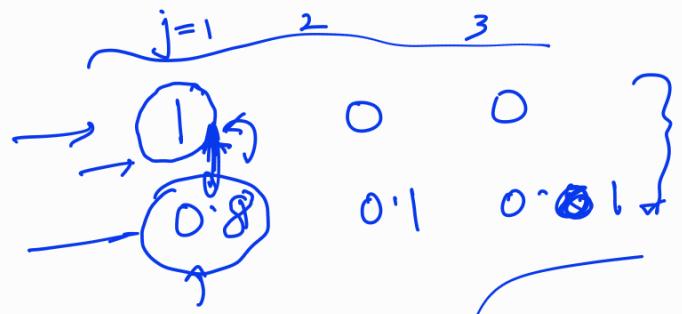


$$\frac{dh}{d\omega_1} = \frac{dh}{dy} \times \frac{dy}{dN_3} + \frac{dh}{dN_3} \times \frac{dN_3}{dN_1} + \frac{dh}{dN_1} \times \frac{dN_1}{d\omega_1}$$

$$= \frac{dh}{dy} \times \frac{dy}{dN_4} \times \frac{dN_4}{dN_2} \times \frac{dN_2}{dN_1} \times \frac{dN_1}{d\omega_1}$$

x_1	x_2	y	1	2	3
1	-	1 ←	1	0	0
1	-	2 ←	0	1	0
1	-	3 ←	0	0	1





$$\sum_{i=1}^n \sum_{j=1}^3 y_i \log \hat{y}_i = 1 \times \underline{\log 0.8} + 0 \times \log 0.1 + 0 \times \log 0.8$$