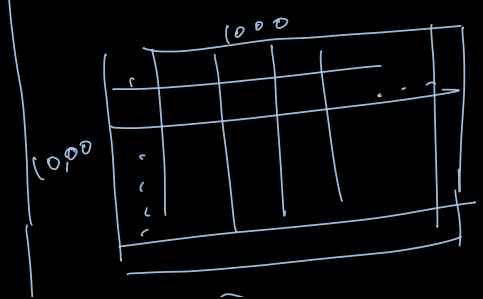
Deep Learning —

Biologist + Computer programmer

Neural Networks —

Machine Learning	Deep Learning
Columns ↑ increases ML struggle Curse of Dimensionality	⟷ Exponential amount of Dimensionality
10,00,000 95% ↓ 1,00,000 — 92 — Same result ↓ 50,000 — 80	⟷ humungous amount of Sample
Simple task can be handled	Complex task 1000 × 1000 

Maruti 800

1 litre → 20 km/lit

Speed - 80 km/hr

daily office

economical traffic

Ferrari

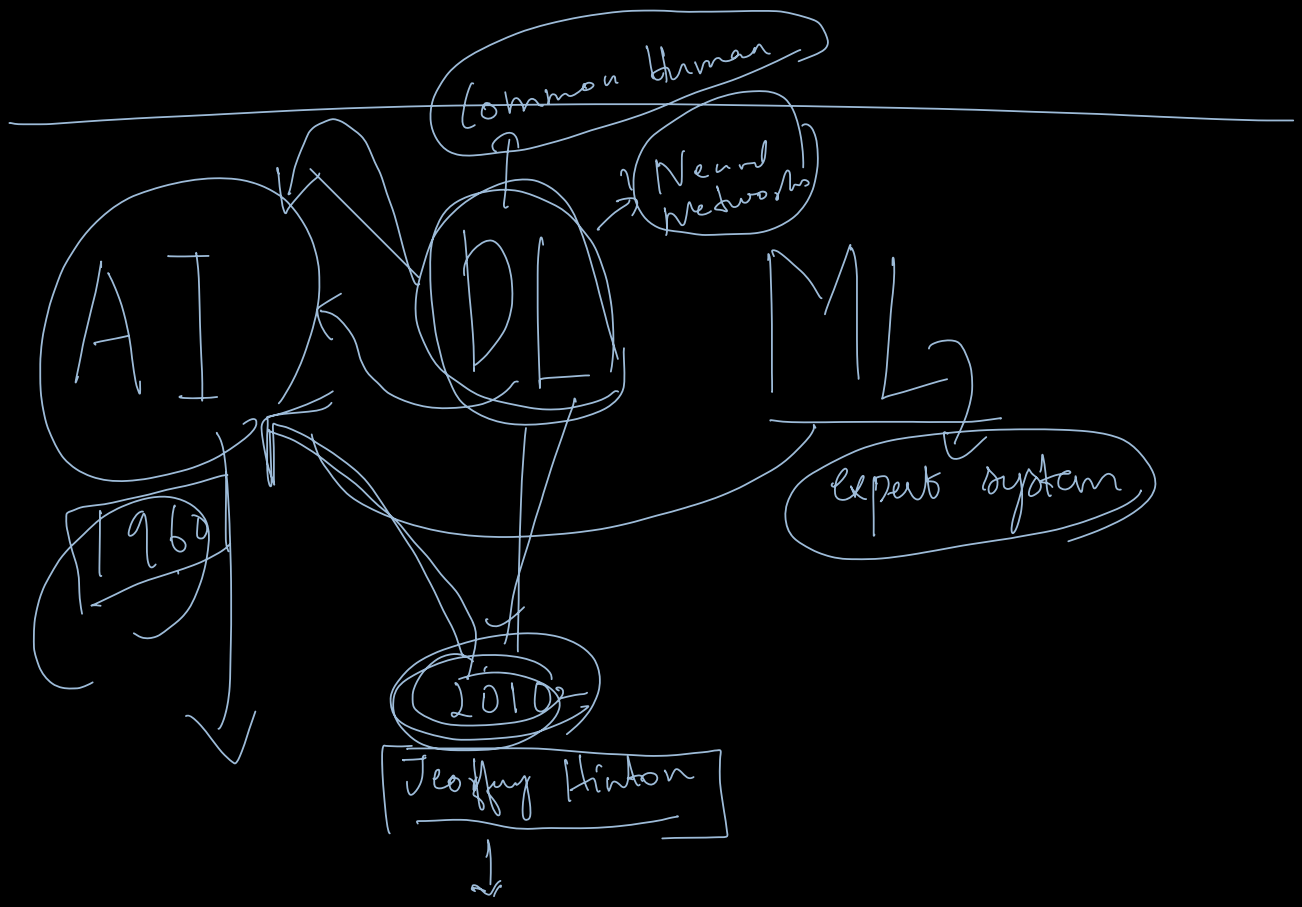
1 litre → ≤5 km

Speed → 250 km/hr

Outing

Timeless

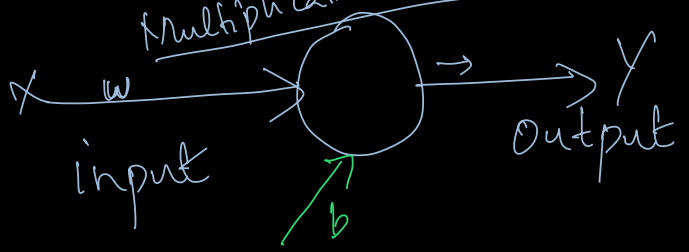
Budget



Neural Network

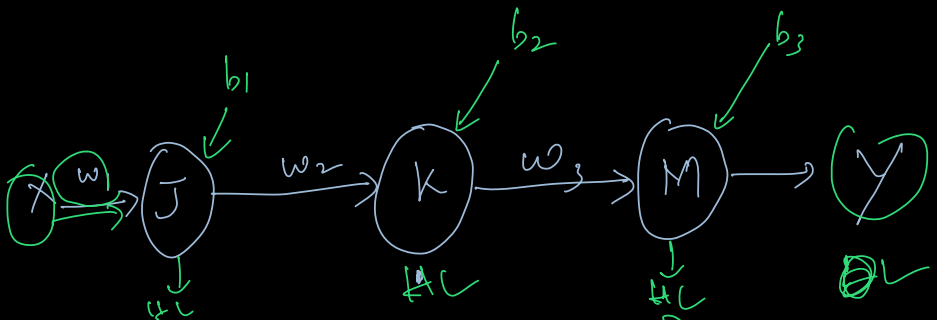
Neuron

Multiplication and addition

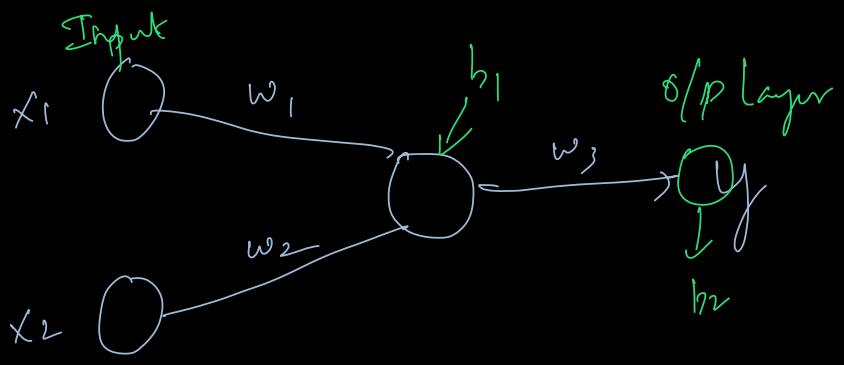


$$y = f(x)$$

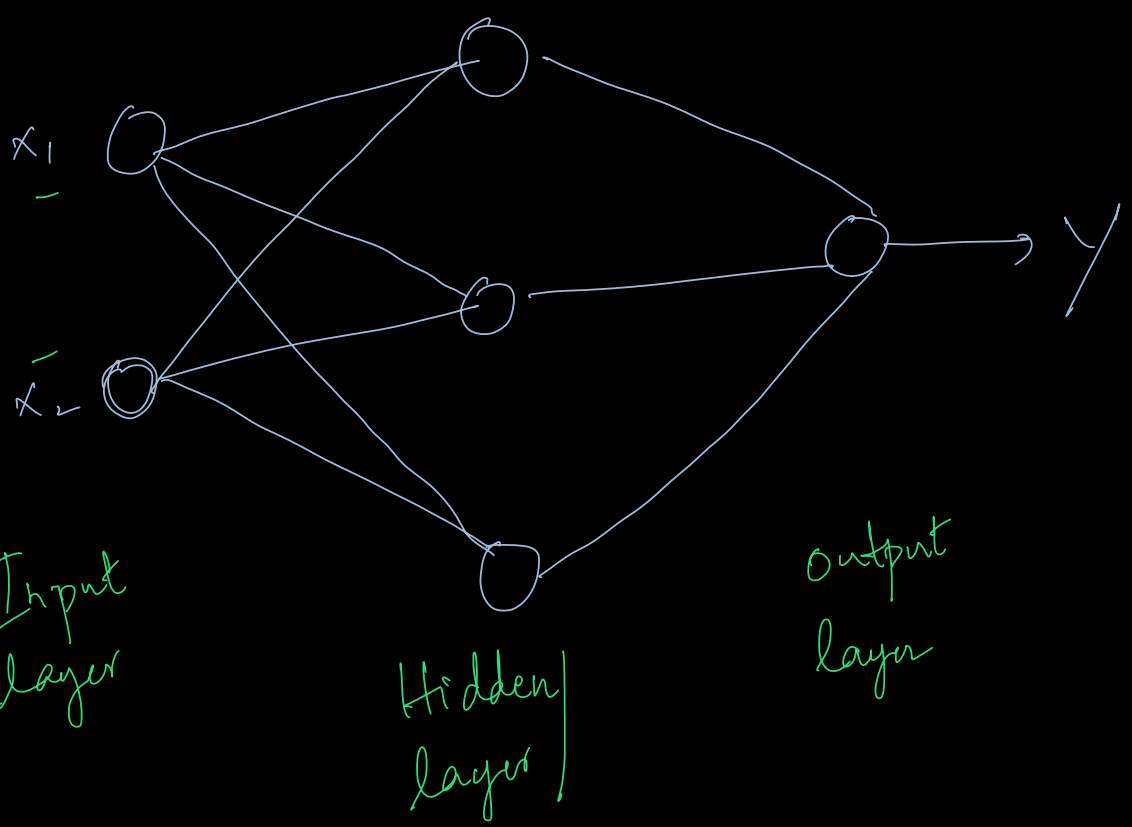
$$f(x) = x \cdot w + b$$

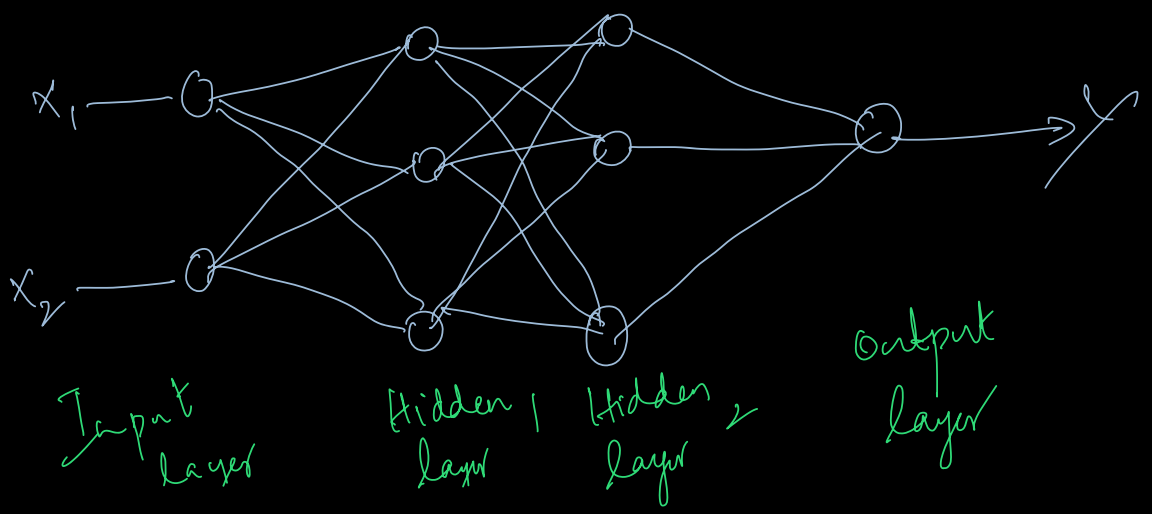


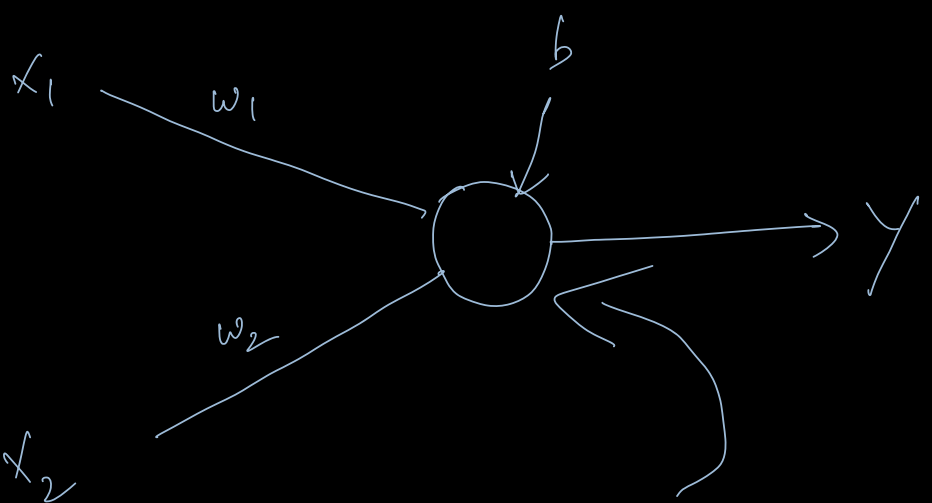
$$Y = \left(\left(x_1 \cdot w + b_1 \right) \cdot w_2 + b_2 \right) \cdot w_2 + b_3$$



$$y = \Big(\big(X_1 \cdot w_1 + b_1 \big) + \big(X_2 \, w_2 + b_1 \big) \Big) \cdot w_3 + b_2$$



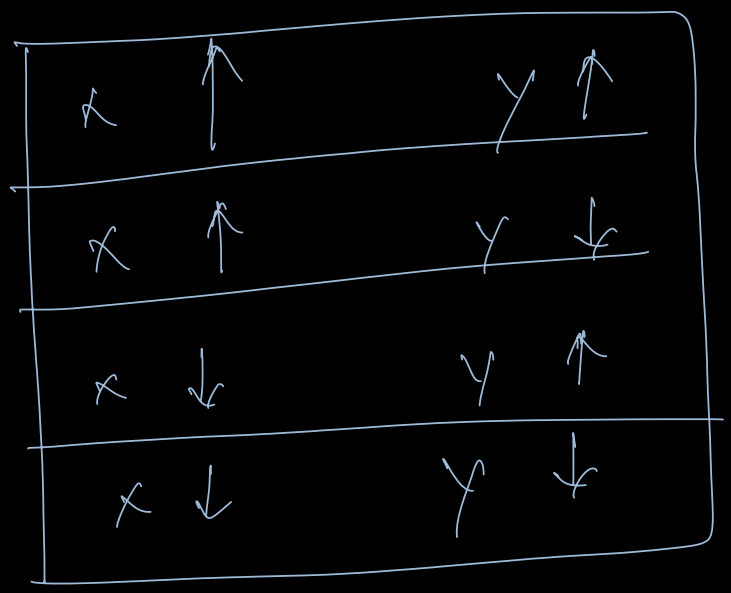


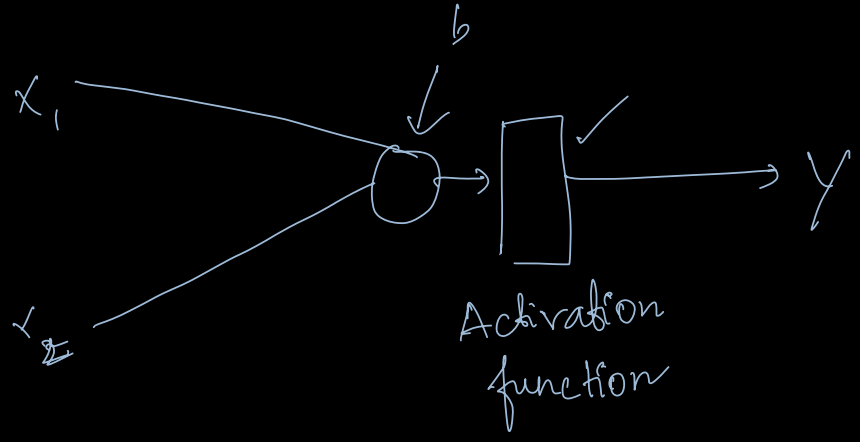


$$Y = x_1 w_1 + x_2 w_2 + b$$

Linear Regression

Linear Information





Activation function =

Identity function.

$5 \cdot I \Rightarrow 5$

$100 \cdot I \Rightarrow 100$

{ Identity function ⇒ Linear Regression
{ Sigmoid function ⇒ Logistic Regression

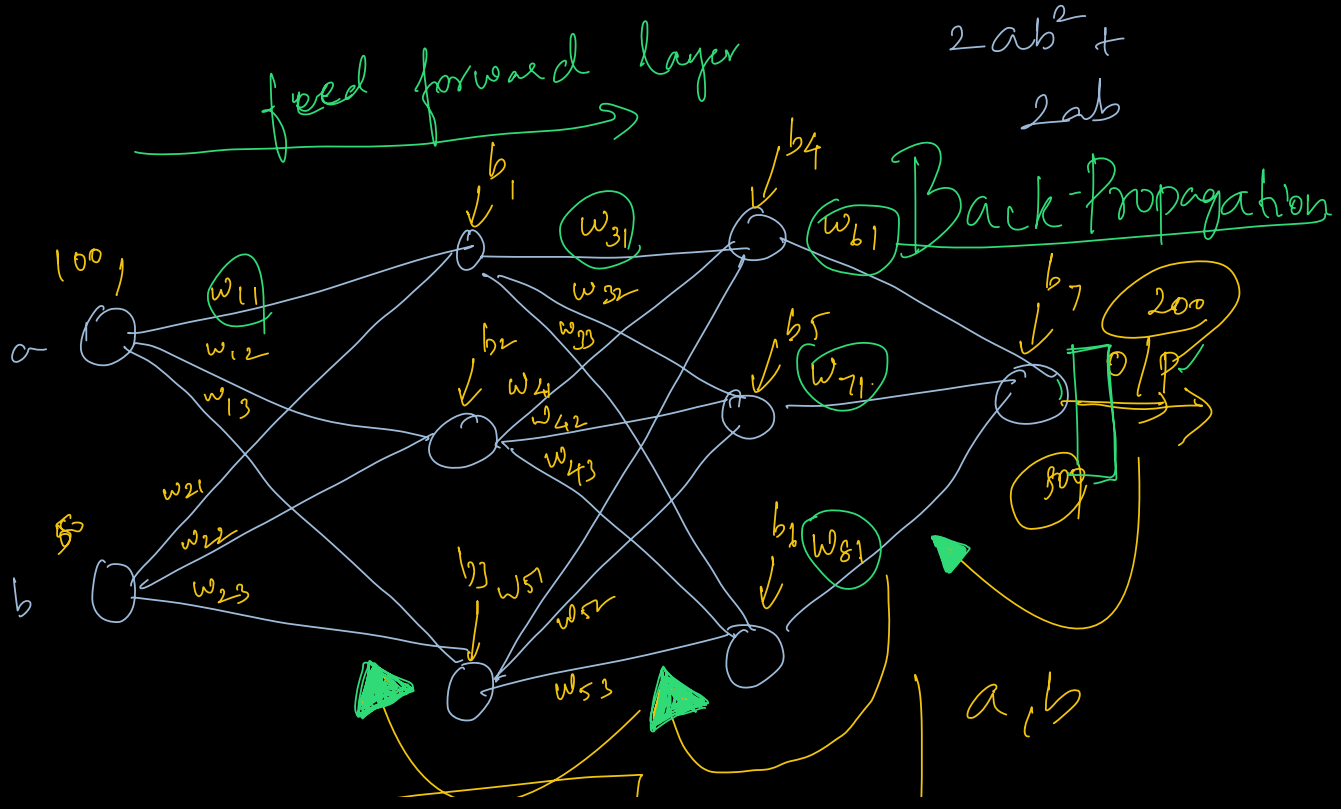
$$\frac{1}{1 + e^{-(mx+b)}}$$

Linear Regression

10 Columns ⟶ 11 parameters

10 slope + 1 Intercept

$a, b \Rightarrow 1, a, b, a^2, b^2, 2ab + 2ab^2 + 2ab$



25 parameters

$2 \times 3 \times 3$

$18 + (3 + 3 + 1)$

25

3 parameters

$a^2, b^2, 2ab$

6 →

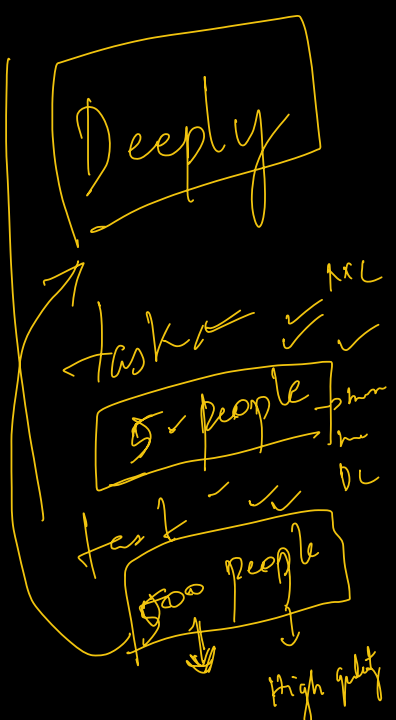
2 Hidden Layers

10 Neurons

$(2 \times (10 \times 10)$

$(200) + (10 + 10 + 1)$

221 → parameter



Sigmoid function
Tanh function
Softmax function
Step function

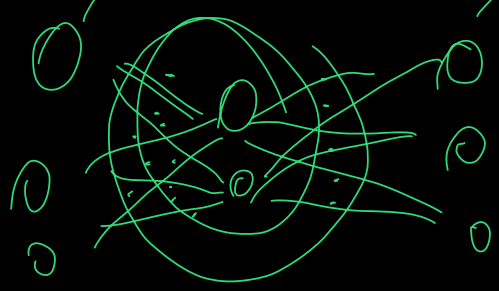
before 2000 →

1960's → 2010

M.L

RELU function
ELU function } Up 20000 ↓

Auto Encoders — Unsupervised



B1
→ 10 → (W_n random) → Back propgtn
W_n Optimised 1

B2
→ 10 → W_n optimised 1 → Back propgtn
W_n Optimised 2

B2
→ 10 → W_n optimised 2 → Back
W_n Optimised 3

Epoch
Iteration

Why NN was Abondoned

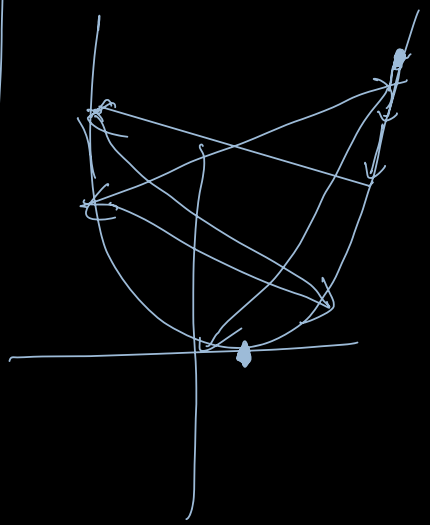
1. Vanishing Gradient descent
2. Exploding Gradient descent

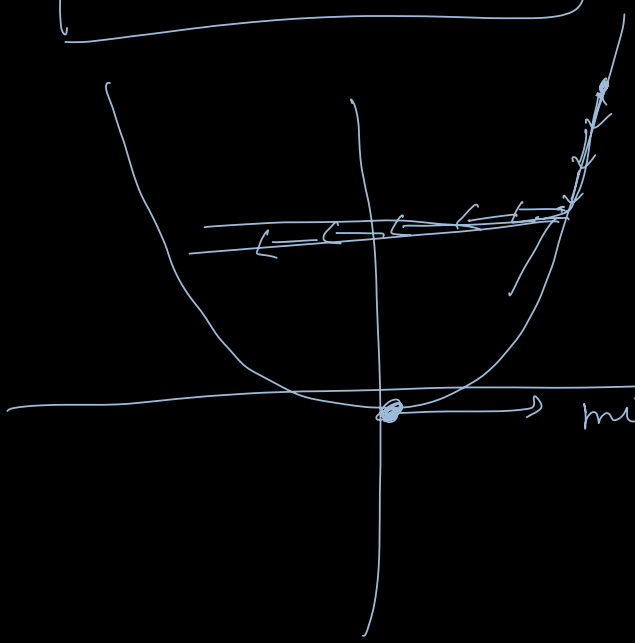
1. Vanishing Gradient descent / Exploding grad

$$m = \left[m - \left(\frac{6}{6m} \cdot l_r \right) \right] \rightarrow .$$

$\frac{6}{6m} \rightarrow 0$

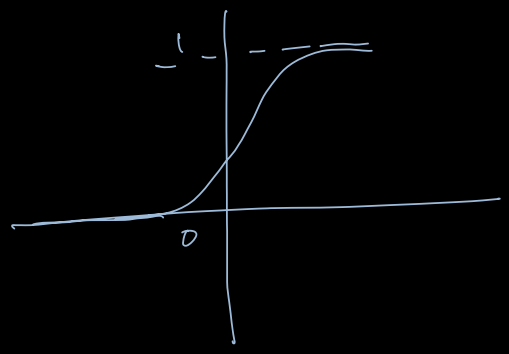
m = Constant





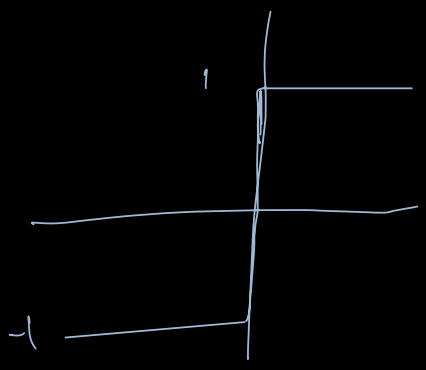
Activation function

Sigmoid



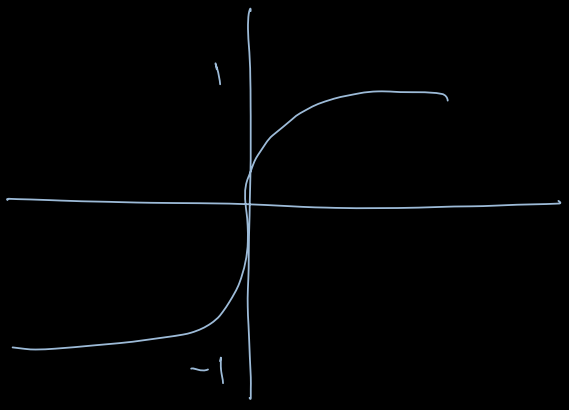
0 ⟷ 1

Step function



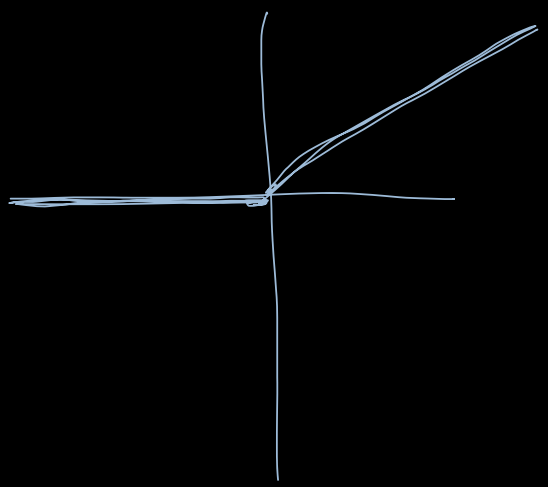
-1, 0, 1

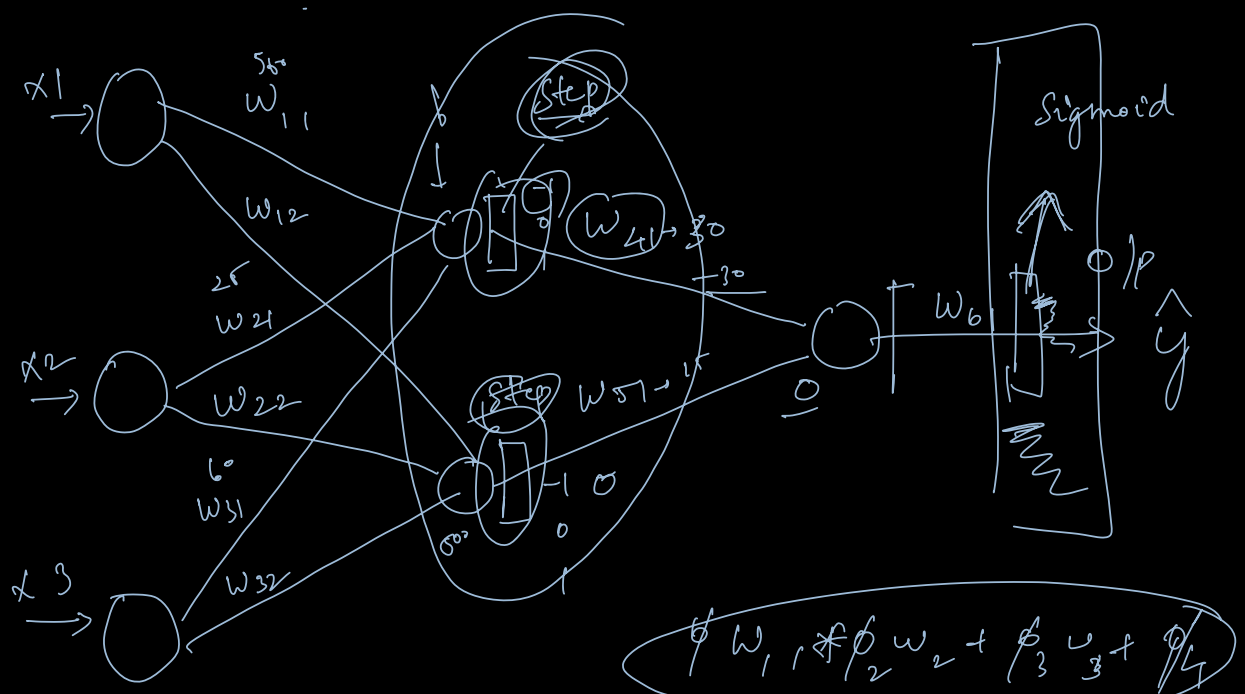
tanh



-1 ⟷ +1

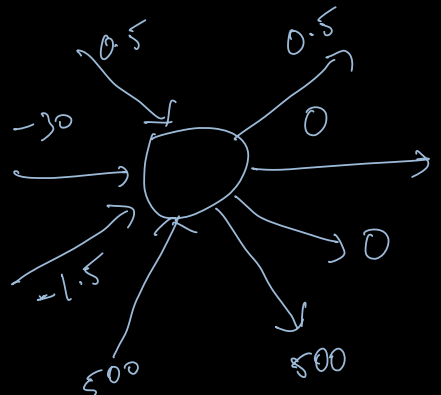
Relu





$$\frac{1}{(1+e^{-(\rightarrow)})}$$

Relu



Softmax function

Sigmoid

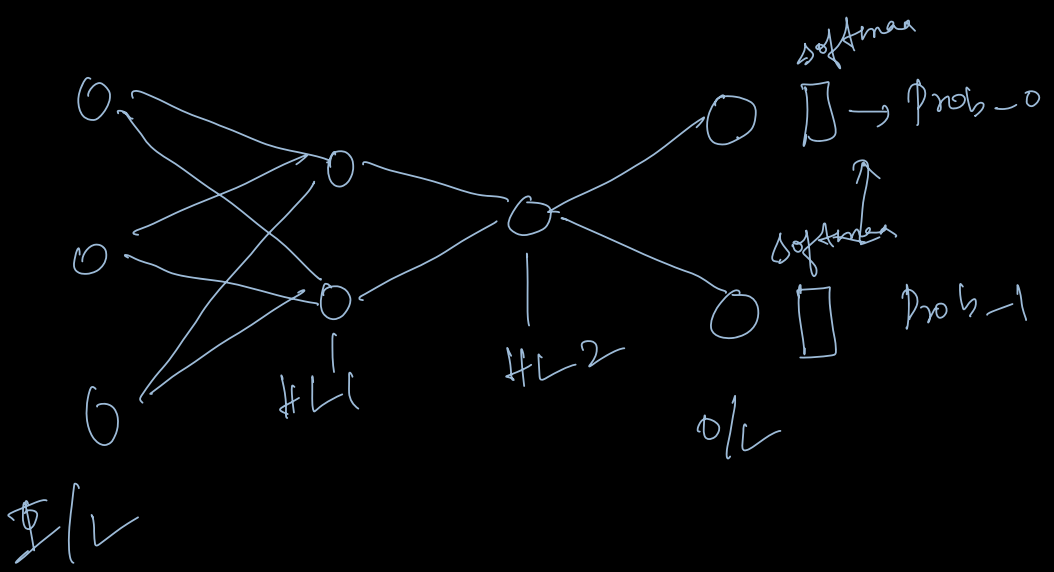
Binary classification

0 1

0 →	0.8
1 →	0.2

Multiclass [0, 1, 2]

0 →	0.3
1 →	0.1
2 →	0.6



Eradicating Vanishing and Exploding Gradient Descent

1. Proper Initialization

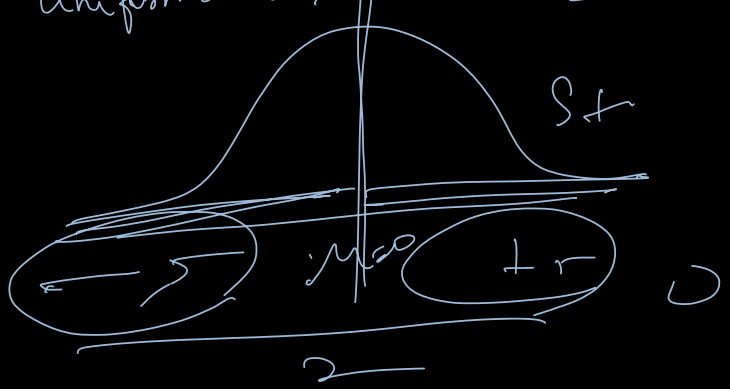
(10)

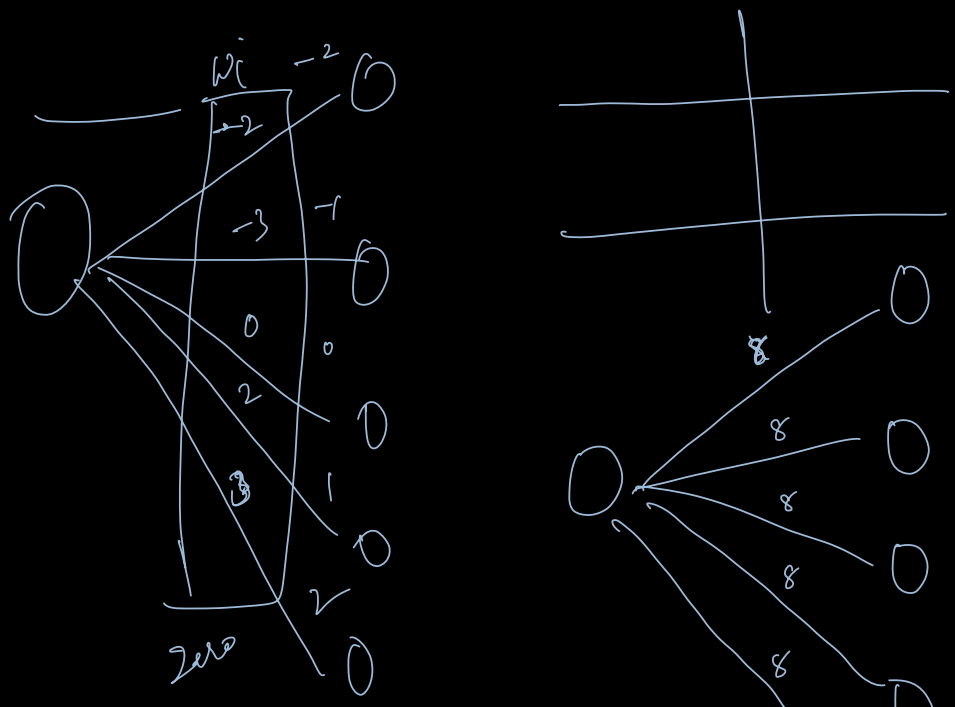
Normal distribution ✓

mean = 0

Uniform distribution → $[-r, +r]$

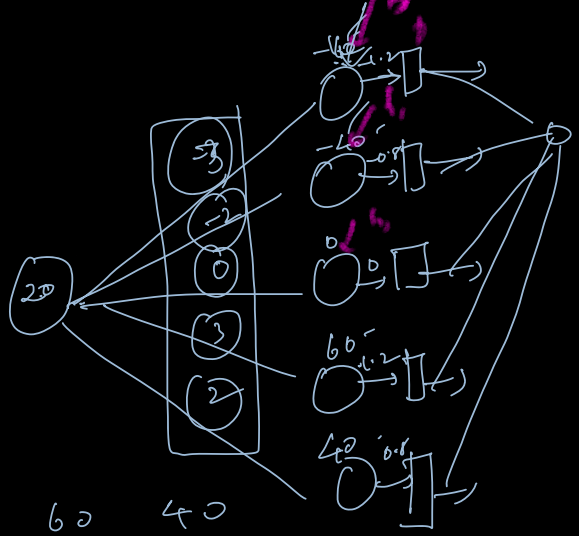
number I/P number inputs





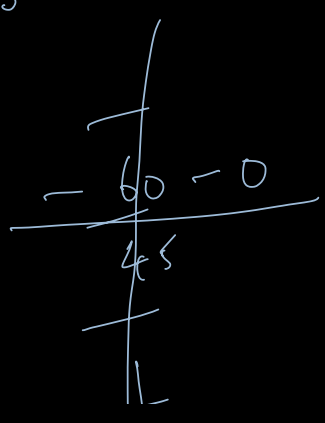
2. Batch Normalization

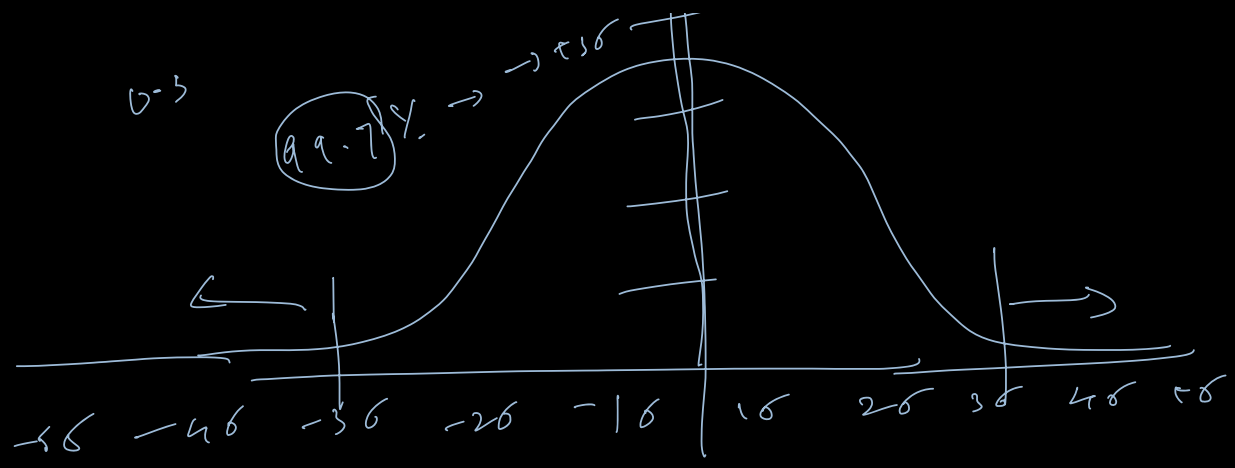
$$z \Rightarrow \frac{x - \mu}{\sigma}$$

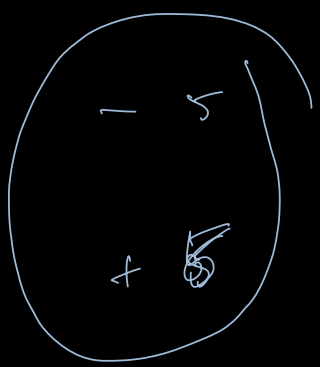


$\mu = 0$

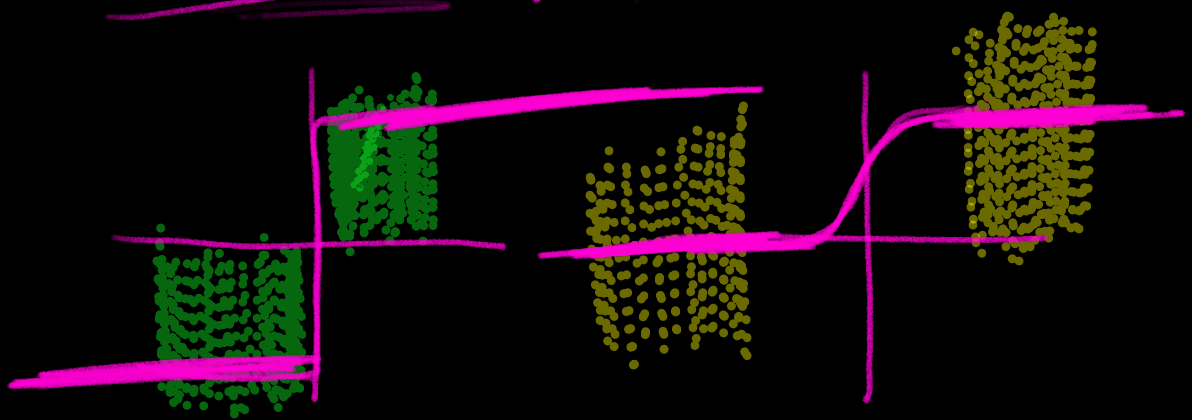
$\sigma = 45.6$







3. Non Saturating Activation Function



0 → 0

1 → 1

-1 → -1

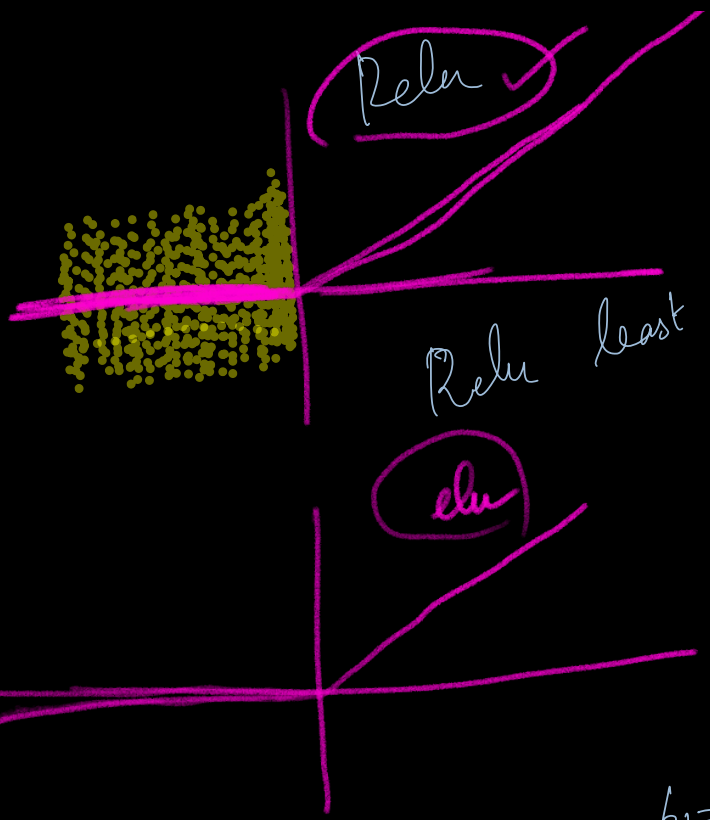
0.01 → 1

-0.001 → -1

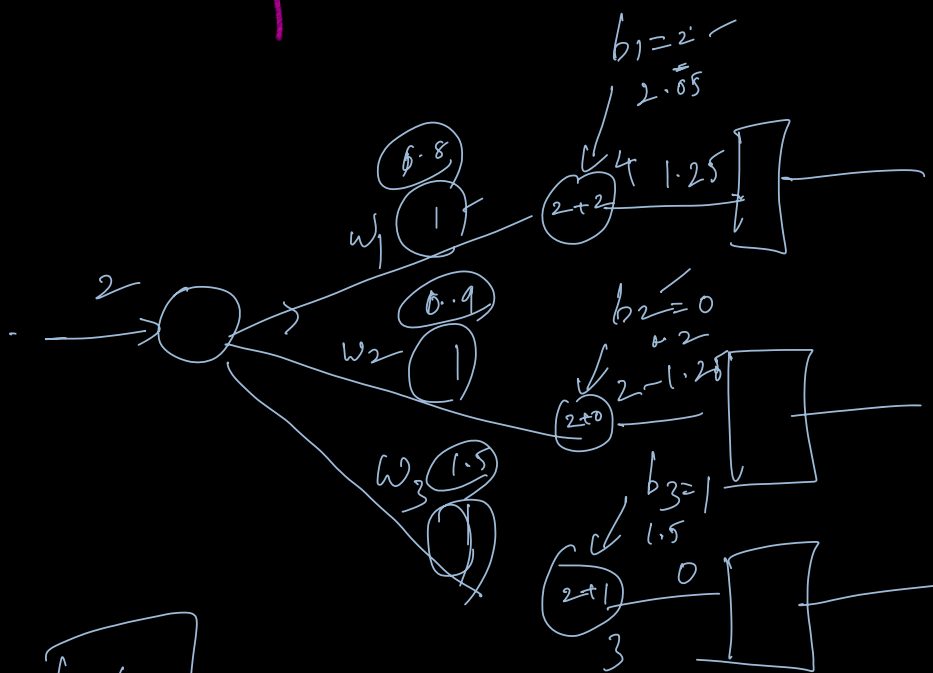
0 → 0

20 → 1

-10 → 0



Relu least Non Saturating Activation funct.



$$\begin{bmatrix} 4 \\ 2 \\ 3 \end{bmatrix}$$

$\mu = 3$

$\sigma = 0.8$

$$\frac{4-3}{0.8}, \frac{2-3}{0.8}, \frac{3-3}{0.8}$$

$$\frac{1}{0.8}, \frac{-1}{0.8}, 0$$

$$1.25 \quad -1.25 \quad 0$$

Decisions to be taken in NN

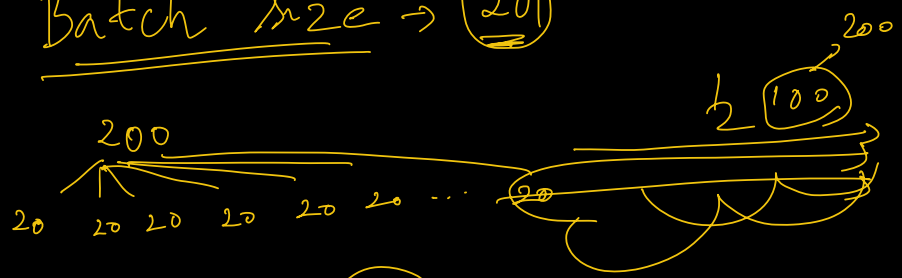
1. Epoch

100 epochs

200 → sample

20 → Batch size

2. Batch size → 20

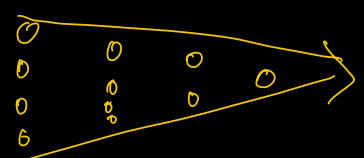


3. Iteration (10) → 1 epochs

1,000

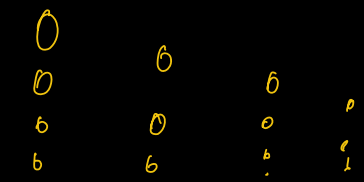
4. Hidden layers

Number hidden layers



5. Number of Neurons in Hidden Layers

2^n 8, 32, 64, 128, 256, 512, 1024



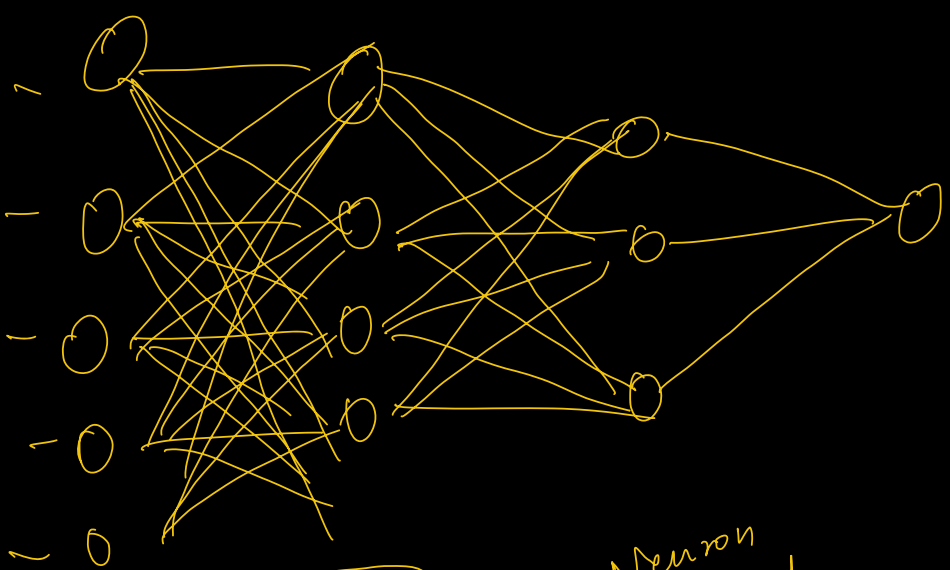
6. Activation function for each layer
   Relu

7. Optimizer
8. loss function

Model Hyperparameter	Model Parameter
Design of actual model to make the prediction better	O/p from the model
Input to the training process	Result from Training Process
KNN → K	used to make predictions
logistic → lasso, Ridge, alpha	
RF → number of trees	
Generate best models	

Overfitting in NN

1. Regularization
2. Cross Validation
3. Drop Out



Dropout 50%

80% Neuron will be dead connection