1) Single layer perceptrons - Only linearly Separable

2) Multi-layer perceptrons - two or more layers
- higher processing power.

Single layer perceptron :-

Loss function = $(y_i - \bar{y})^2$

while building deep learning models our cost functions are (dst, time, height)

→ Mean squared Error (Regression loss)

(or)

L2 Loss / MSE :

$$\boxed{MSE = \frac{1}{N}\sum_{i}^{N}(y_i - \bar{y}_i)^2}$$

→ Mean Absolute error / L1 Loss.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \bar{y}_i|$$

→ Huber loss :-

$$Huber = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{2}(y_i - \bar{y}_i)^2 \quad |y_i - \bar{y}_i| \leq \delta$$

$$Huber = \frac{1}{n}\sum_{i=1}^{n}\Delta(|y_i - y_i| - \tfrac{1}{2}) \quad |y_i - y_i| > \Delta$$

Classification Loss :-

→ Binary cross entropy / Log Loss

y: → Actual values

$$Log\ loss = \frac{-1}{N}\sum_{i=1}^{N} y_i \log \bar{y}_i + (1-y)\log(1-\bar{y}_i)$$

cal Cross entropy

$$loss = \sum_{j=1}^{k} y_i \log(\bar{y}_j)$$

loss function = -sum up to k ($y_i \log(\bar{y}_i)$)

where k is classes.

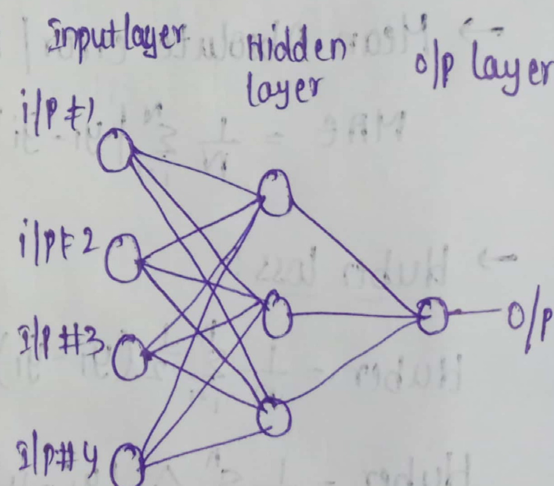
$$Cost = \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{k} [y_{ij} \log(\bar{y}_{ij})]$$

Architectural components of neural networks:-

Neural network:-

* Neural network is a computer model which resembles human brain designed to recognise patterns & make decisions based on data.

Input layer:- The i/p layer accepts the i/p data & passes to the next layer. It consists of interconnected nodes.



Hidden layer:- The activation functions all are present the main function is to extract features & abstract representation of i/p data.

Output layer:- It generates the final output depending on problem the no of neurons in the o/p layer may vary.