XOR function:

x_1	x_2	XOR function (Actual output - 0)
0	0	0
0	1	1
1	0	1
1	1	0

2 inputs - (x_1, x_2)
2 weights = (w_1, w_2)

For inputs (0,0)

$$y = \sum_{i=1}^{m} (x_i * w_i) + b$$

Assume $b = w_0$

$$y = 0(w_1) + 0(w_2) + w_0 < 0$$

Consider the class <0 since, the actual output is 0

$$y = w_0 < 0 \Rightarrow w_0 < 0 \quad \text{—(1)}$$

For inputs (0,1)

$$\sum x_i w_i + w_0 \geq 0, \text{ since actual o/p is 1}$$

$$0(w_1) + 1(w_2) + w_0 \geq 0$$

$$w_2 \geq -w_0 \quad \text{— (2)}$$

For inputs (1,0)

$$\sum x_i w_i + w_0 \geq 0$$

$$1(w_1) + 0(w_2) + w_0 \geq 0$$

$$w_1 \geq -w_0 \quad \text{— (3)}$$

For inputs (1,1)

$$\sum w_i . x_i + w_0 < 0$$

$$(1)w_1 + (1)(w_2) + w_0 < 0$$

$$w_1 + w_2 < -w_0 \quad \text{—(4)}$$

By adding ② and ⑤, we get

$$w_1 + w_2 \geq -w_0$$

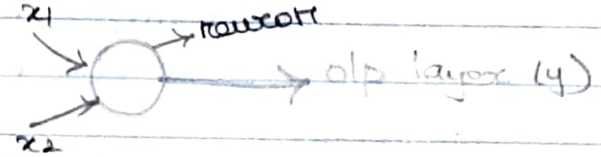
But the ④ Equation defines, $w_1 + w_2 < -w_0$

So, it is a non-linear data. Hence, we cannot perform perceptron.

Note: In order to solve non-linear data, we go for Multi-layer Perceptron (MLP)

Linear network/machine	Non-linear network/machine
* Separate with a line	separate with curves
* degree of function is ≤ 1	degree of function is > 1
* Slope is constant	slope will varies

Single layer neural network:

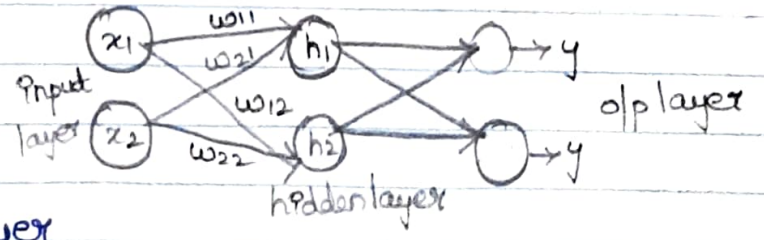


Single output layer (only one output) input layer

Multi-layer neural network:

→ fully connected layer

→ layer between input and output layer called hidden layer



→ All the outputs are interconnected to hidden layer

(All the neurons are interconnected)

we can have n - number of hidden layers

Based on the features of input datasets, input layer comes

Based on the class, the output layer comes

MLP (Multilayer Perceptron) has weights for each and every edges.