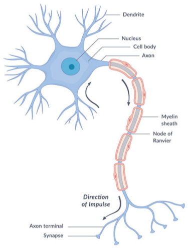
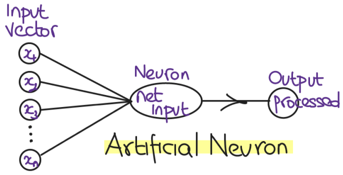
Artificial Neural Network

Artificial Neuron:

To understand artificial neural network or just neural network or in short NN. We need to start with understanding the most fundamental unit of it, which is an artificial neuron. It is designed by taking inspiration from a real human neuron. But it doesn't mean that the working principles of the ANN and the real biological neuron are similar. In fact we don't have full understanding of how the brain works as of today.





The above is merely a single neuron, it is not a 'neural network'. It is just a rough structure of single artificial neuron. Here we did not explain how it works.

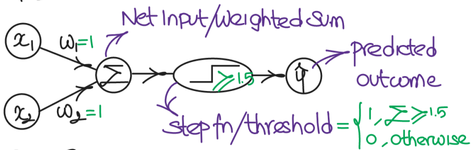
Perceptron is widely considered to be the first neuron model. To be technically more precise perceptron is the first learnable model which is capable of learning some parameter after being trained on a dataset. But truly the first neuron model is McCulloch & Pitts neuron model some people refer to this also as perceptron. Honestly there's no single precise definition of what a perceptron is, lets stick to calling the first neuron model as McCulloch & Pitts model and discuss more about it.

McCulloch & Pitts Neuron Model:

Historically if this is the mankind's first neuron model, then why is it not popular? and why perceptron is considered to be the first neuron model? The answer is 'learnability'. McCulloch Pitts model can do basic logical operations such as OR, AND, NOT and with additional layer it can perform XOR as well. But we have to manually set the weights and biases. In case of Frank Rosenblatt's Perceptron, we don't have to hand code those weights and biases. The model will learn them on its own. Now lets check the architecture and the working of McCulloch & Pitts neuron model and solve basic logic gates problems using them.

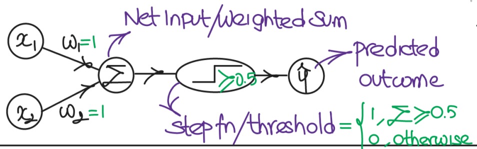
The inputs of McCulloch & Pitts neuron model are binaries and the output is also binary. That's what it is going to be for logic gates. Only the weights and biases can possibly non-binary

AND Gate:



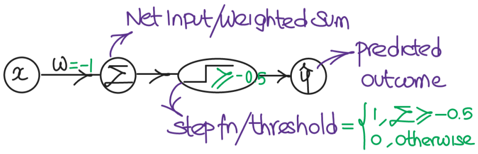
x_1	x_2	$\hat{y}$
0	0	0
0	1	0
1	0	0
1	1	1

OR Gate: Note that all these w & b are handpicked



x_1	x_2	$\hat{y}$
0	0	0
0	1	1
1	0	1
1	1	1

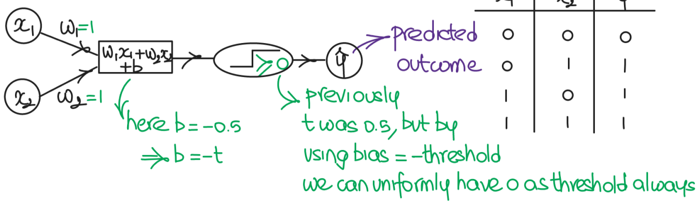
NOT Gate:



x	$\hat{y}$
0	1
1	0

Solving AND, OR, NOT using McCulloch Pitts neuron model was quite straight forward though we had to handpick the weights and biases. You wouldn't have come across the terminology of 'bias' in any of the previous diagram because we use bias and threshold alternatively but not interchangeably. Because bias is basically negative threshold. Let me explain that in the following diagram.

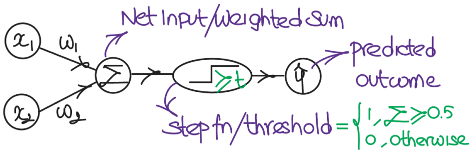
OR Gate:



x_1	x_2	$\hat{y}$
0	0	0
0	1	1
1	0	1
1	1	1

Now let us try to solve XOR problem using McCulloch Pitts neuron model. We'll learn that no matter what we do, it is impossible to solve without involving another layer of neuron.

XOR Gate:



x_1	x_2	$\hat{y}$
0	0	0
0	1	1
1	0	1
1	1	0

No matter what value we assign for w_1, w_2, t, we never obtain the above table. Why?

Consider the net input $w_1x_1+w_2x_2$ & we compare it with t

$\Rightarrow$ $\hat{y} = \begin{cases} 1, & w_1x_1+w_2x_2 \geq t \\ 0, & \text{otherwise} \end{cases}$ lets substitute the values from the table

$\Rightarrow$ $(x_1,x_2)=(0,0) \Rightarrow \hat{y} = \begin{cases} 1, & 0 \geq t \\ 0, & 0 < t \end{cases}$ $\Rightarrow t > 0$

$(x_1,x_2)=(0,1) \Rightarrow \hat{y} = \begin{cases} 1, & w_2 \geq t \\ 0, & w_2 < t \end{cases}$ $\rightarrow w_2 \geq t > 0$

$(x_1,x_2)=(1,0) \Rightarrow \hat{y} = \begin{cases} 1, & w_1 \geq t \\ 0, & w_1 < t \end{cases}$ $\rightarrow w_1 \geq t > 0$ $\oplus$

$\Rightarrow$ $w_1+w_2 \geq 2t$

$(x_1,x_2)=(1,1) \Rightarrow \hat{y} = \begin{cases} 1, & w_1+w_2 \geq t \\ 0, & w_1+w_2 < t \end{cases}$

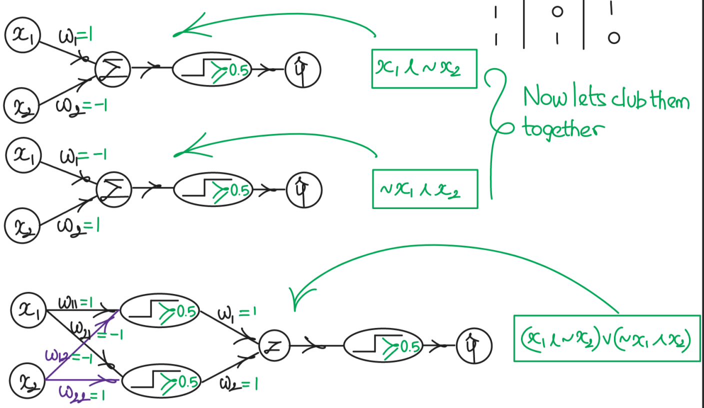
Contradiction

XOR Gate:

$a \oplus b = (a \land \sim b) \lor (\sim a \land b)$ $\rightarrow$ a standard result

Using the above result we see that we have to create two AND models & connect them using OR

x_1	x_2	$\hat{y}$
0	0	0
0	1	1
1	0	1
1	1	0



Hence we solved XOR using the McCulloch Pitts neuron model but by adding another layer to the network.

Hope you found it useful!
Lets connect
https://www.linkedin.com/in/fazil-mohammed-4062711b2/