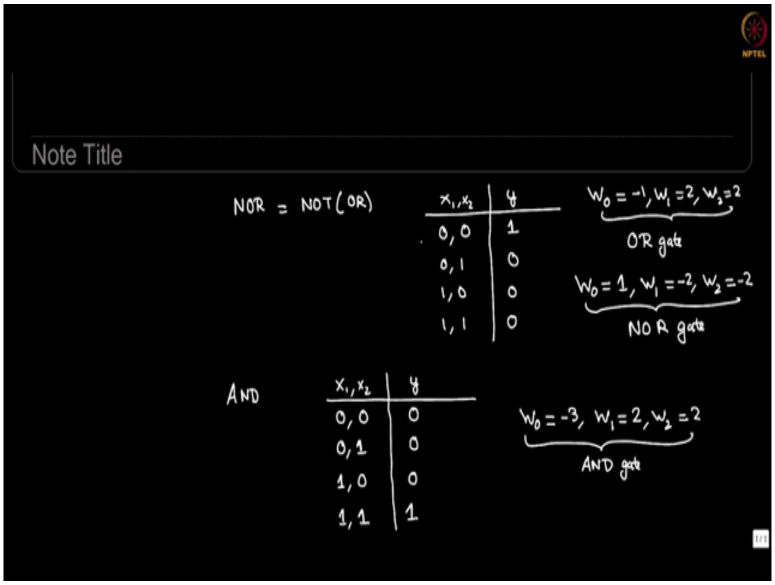
Machine Learning for Engineering and Science Applications

Professor Dr. Balaji Srinivasan
Department of Mechanical Engineering
Indian Institute of Technology, Madras

NOR, AND, NAND Gates

(Refer Slide Time: 0:25)



Note Title

NOR = NOT(OR)

x_1, x_2	y
0, 0	1
0, 1	0
1, 0	0
1, 1	0

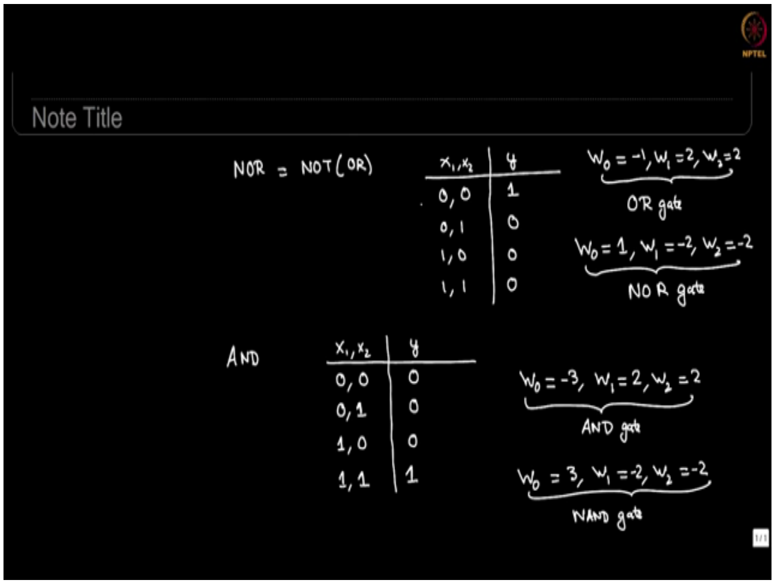
$$\underbrace{W_0 = -1, W_1 = 2, W_2 = 2}_{\text{OR gate}}$$

$$\underbrace{W_0 = 1, W_1 = -2, W_2 = -2}_{\text{NOR gate}}$$

AND

x_1, x_2	y
0, 0	0
0, 1	0
1, 0	0
1, 1	1

$$\underbrace{W_0 = -3, W_1 = 2, W_2 = 2}_{\text{AND gate}}$$



Note Title

NOR = NOT(OR)

x_1, x_2	y
0, 0	1
0, 1	0
1, 0	0
1, 1	0

$$\underbrace{W_0 = -1, W_1 = 2, W_2 = 2}_{\text{OR gate}}$$

$$\underbrace{W_0 = 1, W_1 = -2, W_2 = -2}_{\text{NOR gate}}$$

AND

x_1, x_2	y
0, 0	0
0, 1	0
1, 0	0
1, 1	1

$$\underbrace{W_0 = -3, W_1 = 2, W_2 = 2}_{\text{AND gate}}$$

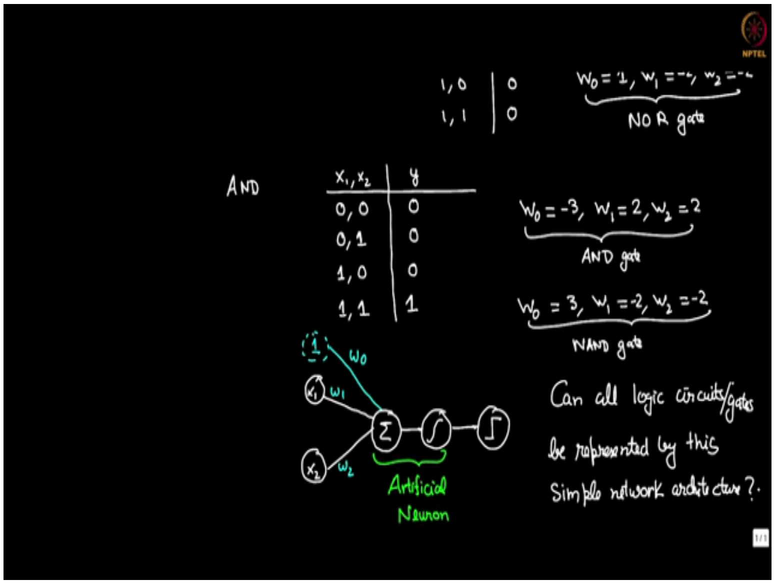
$$\underbrace{W_0 = 3, W_1 = -2, W_2 = -2}_{\text{NAND gate}}$$

In this video, we look at a few more gates very very quickly the basic algebra for this case had already been done in the previous video. So suppose you take the NOR gate. NOR gate is essentially the opposite of OR gate. So I will quickly write the truth table. So if you have 00, 01, 10, 11, so it is exactly the opposite of OR. How would we make weights for this? I will let you work through the algebra but essentially you can probably see that if you, remember that w0 was minus- 1 for 1 possible set of values.

This was the OR gate. You can guess that flipping the weights and making w0, 1 and w1, and w2 negative of this, will give us NOR gate. You can check this quickly. If you do w0 which is 1 plus+ w1 x1 plus- w2 x2, the value will come out to be positive. Therefore y after classification will come out to 1. Similarly in this case w0, plus+ w1 x1 plus+ w2 x2 will be minus- 1. So this will be 0, minus- 1, 0. In this case you will get minus- 4 or minus- 3 and you will get 0 again. So this works out.

You can similarly check how the AND gate works. I will write truth table. So this works only if both of these are non-zero. And a set of weights that will give you this result, now you have to weigh the biased unit a little bit heavier so that it does not trigger a positive result even if one of these is active. So w0 plus+ w1, x2, so w2 x2 in this case will be minus- 1 and it will still stay at 0. So this case is the AND gate. And you can easily guess that the NAND gate would simply be the negative of these weights.

(Refer Slide Time: 3:33)



1, 0	0
1, 1	0

$\underbrace{W_0 = 1,\ W_1 = -4,\ W_2 = -4}_{\text{NOR gate}}$

AND

x_1, x_2	y
0, 0	0
0, 1	0
1, 0	0
1, 1	1

$\underbrace{W_0 = -3,\ W_1 = 2,\ W_2 = 2}_{\text{AND gate}}$

$\underbrace{W_0 = 3,\ W_1 = -2,\ W_2 = -2}_{\text{NAND gate}}$

Can all logic circuits/gates be represented by this Simple network architecture?

So we saw in this video that very simple elementary weights, elementary gates OR, NOR, AND, NAND, all of them can be represented by the simple architecture, x1, x2, a biased unit which is simply 1 followed by our weights followed by the sigmoid followed by a classification. This portion of course put together as we saw earlier can simply be called an artificial neuron. We will write w0, w1, w2. So the question, I will stop this video with this, can all logic circuits or in fact can all logic gates be represented by this simple network?

We will see in the next video that this is not possible and since it is not possible, that is what leads us to extra layers within this neural network when we have to consider things like XOR gate.