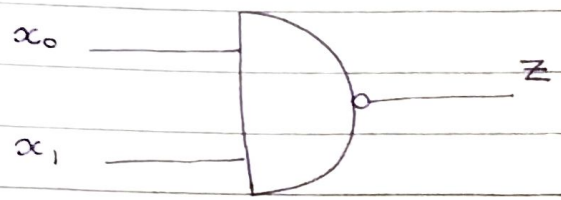
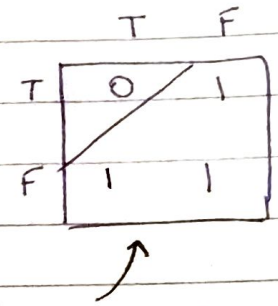
TRAINING A PERCEPTRON TO BE A NAND GATE

25/04/19

NAND GATE



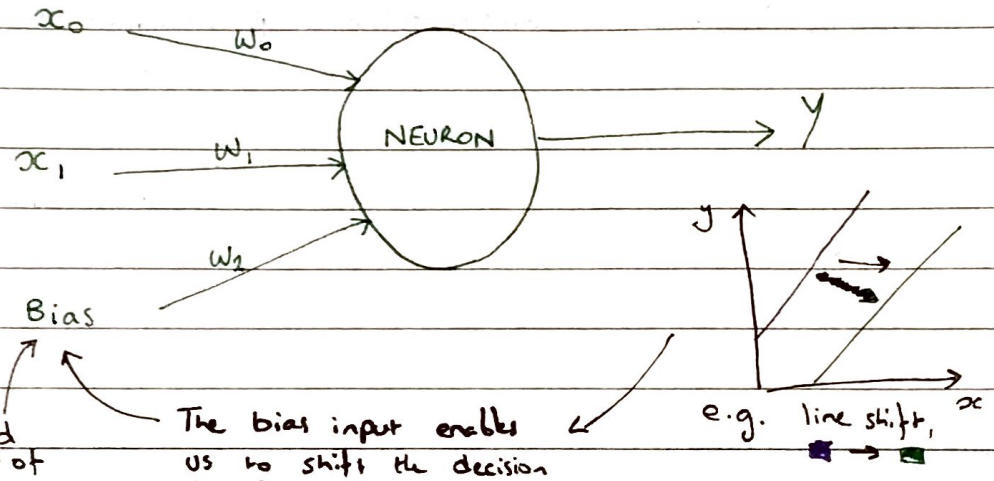
x_0	x_1	Z
0	0	1
0	1	1
1	0	1
1	1	0



NAND IS A LINEAR PROBLEM THAT IS SEPERABLE

PERCEPTRON ALGORITHM

INPUT **OUTPUT**



Fixed input of 1

The bias input enables us to shift the decision boundry. Without it, this would not be possible.

① Sum all inputs * weights, $x_0 W_0 + x_1 W_1 + \text{Bias}\, W_2$

decides whether to fire neuron or not, classifies output based on input

② Pass the result through the following activation function, f:

$f(x_0 W_0 + x_1 W_1 + \text{Bias}\, W_2)$ when f is as follows:

if $f(\text{Summed Inputs}) > 0 \rightarrow 1$

if $f(\text{Summed Inputs}) < 0 \rightarrow 0$

③ Compute error by evaluating guessed value against known answer

ERROR = DESIRED OUTPUT - GUESSED OUTPUT

DESIRED OUTPUT	GUESSED OUTPUT	ERROR
1	1	0
1	0	1
0	1	-1
0	0	0

(4) Adjust all weights according to the error

▲ NEW WEIGHT = WEIGHT + ΔWEIGHT

ΔWEIGHT = ERROR * INPUT * LEARNING RATE

Reduces rate at which weights change, improve networks overall accuracy

(5) Repeat (1) → (4)