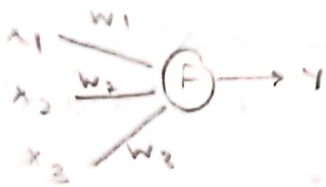
(*) McCulloch Pitts Neuron

↓

fundamental block of DL → Artificial neuron

↓



takes a weighted aggregate of the inputs, and applies a function to the aggregate, giving an apt o/p.

↓

motivation: biological neuron.

↓

dendrite: i/ps from other neurons / sensory organs

synapse: strength of interaction b/w 2 neurons.

↓

weights in a NN.

soma: processing unit, post processing, the o/p is sent to other neurons.

↓

Model: early model of an artificial neuron introduced by warren McCulloch and walter Pitts.

↓

MP model is also called linear threshold gate.

↓

walter Pitts & warren McCulloch

(•) a model is our approximation of the true relationship between y and x.

$$y = \hat{f}(x)$$

$$y = ax + b$$

$$y = ax^2 + bx + c$$

i) function
ii) parameters
iii) kind of i/ps
iv) kind of o/ps
↓
things to be mentioned when specifying a model.

For MCP :

(i) i/ps → boolean
(ii) o/ps → boolean
(iii) function is split into two : g : aggregates the i/ps,
f : takes a decision based on the aggregation.

$$g(x_1, x_2, \ldots, x_n) = g(x) = \sum_{i=1}^{n} x_i$$

$$y = f(g(x)) = 1 \text{ if } g(x) \geq b \rightarrow \text{threshold}$$
$$0 \text{ if } g(x) < b.$$

inhibitory i/p : if this input is on, y = 0 (at all points / situations)
↓
overrides all i/ps.

(iv) one parameter : b
↓
The threshold needs to be adjusted in such a way that the model gives the highest possible accuracy. (correct prediction)

(•) data : big constraint for the i/p to be boolean,

but an example which could be applied using this model, is the LBW decision problem,

where the i/p features are :

a) pitch in line
b) impact
c) missing stumps

→ data is naturally boolean

↓

$$y = \left(\sum_{i=1}^{3} x_i \geq b \right)$$

↓

task / objective of the model : finding a b, such that the o/ps can be matched to i/p.

↓ data not naturally boolean, can also be converted to bool in exchange for information loss

↓

mapping to a boolean space → more bifercations can be added to add specificity

phone example

↓

launch (< 6 months)
weight (< 160g).
screen size (< 5.9 in)

↓

similarly for other features.

↓

and for the o/p labels as well.

(•) loss function : loss/error per sample = $Y - \hat{Y}$ (simply, the difference b/w the true & predicted value)

total loss : $\sum_{i=1}^{n} Y_i - \hat{Y}_i$

But here, what can happen is that the loss for one (-1) prediction might cut out the loss from another prediction (+1), leading to a wrong idea about the loss. To negate this effect, we modify the total loss function to.

errors are getting cancelled out

$$\text{loss} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

L2 loss

true value

predicted value

why not take the modulus fn instead of the squared error loss?

L1 loss : The modulus fn. is non differentiable at $x = 0$ (absolute value fn.)

$$f(x) = |x|$$

$$f'(0) = \lim_{h \to 0} \frac{|0+h| - |0|}{h}$$

$$= \lim_{h \to 0} \frac{|h|}{h} \quad \text{DNE (LHS} \neq \text{RHS)}$$

since $\lim_{h \to 0^+} \frac{|h|}{h} = 1$ (LHS)

$\lim_{h \to 0^-} \frac{|h|}{h} = -1$ (RHS)

(·) learning algorithm: $\hat{y} = \sum_{i=1}^{n} x_i \geq b$

task : bi classification

$$\text{loss} = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

learning algorithm to find b, such that loss is minimized

Things to note : the aggregation will be between 0 to n (no. of features) and would be discrete and not continous.

↓

Brute force search for 1 parameter

↓

start for b = 0 and increase the value to find b, such that loss gets minimized.

(•) evaluation :

$$\text{accuracy} = \frac{\text{correct predictions}}{\text{total predictions}}$$

↓

for test dataset (data the model has / might not have encountered before)

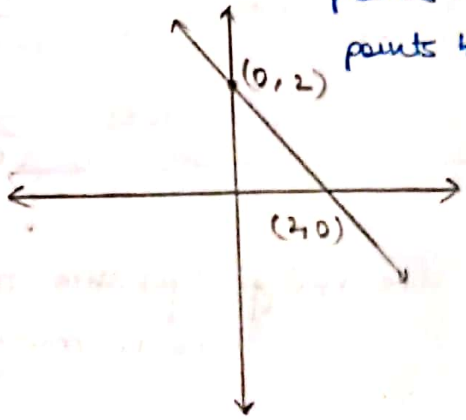
↓

more on the lines of a test, of concepts in school.

*) geometric interpretation of the MP neuron (for 2 dims)

points above or on → 1

points below → 0



$\Rightarrow x_2 = mx_1 + c$

$\Rightarrow mx_1 - x_2 + c = 0$

↓

for example

$x_1 + x_2 - 2 = 0$

$ax_1 + bx_2 + c = 0$ (general form)

↓

points above or on the line means the aggregation is +ve, and will give 1 o/p.

↓ can be extended to 3 dims.

eqn of model : plane

(again, all points above & on plane are +ve. ∴ o/p = 1)

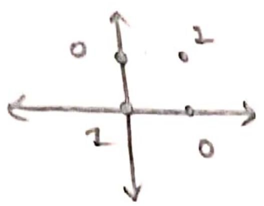
↓

the line & plane separates the points of different class (0 & 1) → boundary which separates the points

↓

can be conceptually extended to n dimensions.

↓



↓

cannot be separated. for any value for b, using the MCP neuron.

for MCP,

$$x_1 + x_2 - b = 0$$

$y - 5 = 4x + 6$

$y = 4x + 11$

$2x + 3y = 5 \Rightarrow$ 11 $\quad 3y = 5 - 2x$

$y = \frac{5}{3} - \frac{2}{3}x$

(divides the set of points, above which we get o/p 1, and below, we get o/p 0)

linear (x)

fixed slope (x)

few possible intercepts (x)

↓

2^n total possible points → n : no. of features

↓

only 4 values when n = 2

↓

(0,0), (0,1), (1,0), (1,1)

↓

the slope = 1, and even y can only take discrete values

↓

hence constrained (very).

↓

finding a line such that, line separates ~~the~~ 1, and 0 o/p points.