Hebbian Rule

- This rule was proposed by Donald Hebb in 1949.
- If neuron i is near enough to excite neuron j and repeatedly participate in its activation, the synaptic connection between these neurons is strenghtned and neuron j becomes more sensitive to stimuli from neuron i.
- Hebb's law can be represented in the form of 2 rules:
  i) If 2 neurons on either side of connection are activated synchronously, then the weight of that connection is increased.
  ii) If 2 neurons on either ~~of~~ side of connection are activated asynchronously, then the weight of that connection is decreased.
- If 2 neurons don't have any relationship, then the weight will not increase.

- The weight update in Hebb rule is given by,

$$w_i(new) = w_i(old) + x_i y$$

- The hebb rule is more suitable for bipolar data, than binary data.

~~Flowchart~~ :

Training algorithm:

step 0: initialize weights = 0

step 1: step 2 have to be performed for each training input vector, and target o/p pair i.e. s:t

step 2: weight adjustments and bias adjustments are as follows :

$$w_i(new) = w_i(old) + x_i y$$

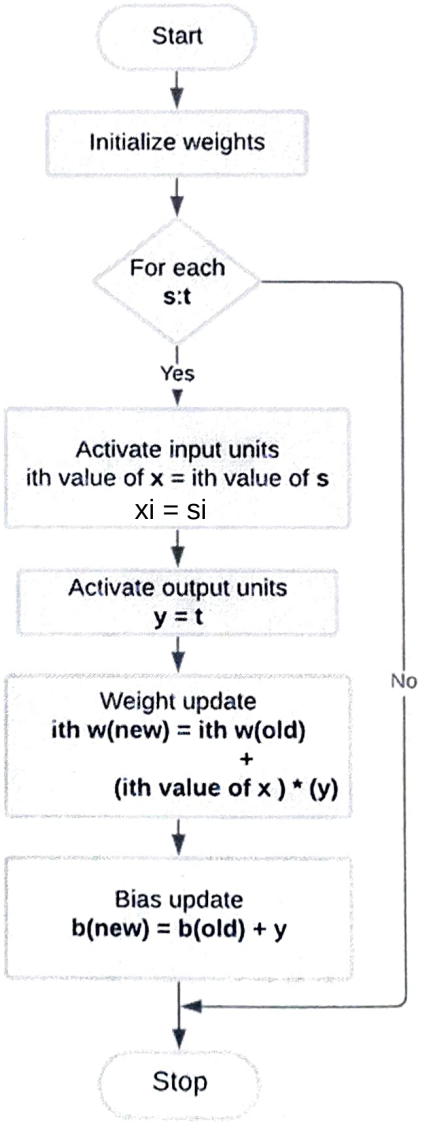
$$b(new) = b(old) + y$$

Here, the change in weight can be expressed as

$$\Delta w = xy$$

$$\therefore \quad w(new) = w(old) + \Delta w$$

Flowchart :


Start
Initialize weights
For each
s:t
Yes
No
Activate input units
ith value of x = ith value of s
xi = si
Activate output units
y = t
Weight update
ith w(new) = ith w(old)
+
(ith value of x) * (y)
Bias update
b(new) = b(old) + y
Stop