UCS2136 PRINCIPLES OF PATTERN RECOGNITION

Unit 5

Fuzzy Pattern Recognition

Fuzzy set theory, Fuzzy and crisp classification, Fuzzy clustering. Fuzzy Pattern Recognition; Elementary Neural Network for Pattern Recognition: Hebbnet, Perceptron, ADALINE, Back propagation

* Elementary Neural Networks & Learning Rules

1. Hebbian Learning Rule

Concept: a form of unsupervised learning

→ based on the rule – 'cells that fire together, wire together'

when neuron (cell A) repeatedly excites another neuron (cell B), the synaptic connection between them is strengthened

→ This leads to increased efficiency in their communication.

Mathematical Expression: The weight update can be described as:

$$\Delta w = \eta\, o_i\, x_j$$

o_i → output of neuron i and x_j

x_j → input from neuron j

IF $o_i x_j$ is positive, the weight increases, otherwise, it decreased

2. Perceptron learning Rule :

→ a kind of supervised learning

→ The learning signal in this rule is the difference between the desired response and the actual response

$$\Delta w = \eta \cdot (d - o) \cdot x$$

d = desired output

o = actual output

x = input

3. Delta learning Rule

→ For supervised learning algorithms

→ only applicable for neurons with continuous activation functions.

It aims to minimize the overall error across training patterns.

→ Learning Signal - represented by Δ, and it represents the error. Given by

$$\Delta w_j = -\eta \frac{\partial E}{\partial w_j}$$

E = error function

→ Minimizing this error requires adjusting the weights in the direction opposite to the gradient (i.e the -ve gradient)

4. Widrow Hoff (least mean Squares) learning Rule

→ used for supervised learning algorithms

→ a special case of the Δ rule, where the activation function is an identity function f(net) = net.

→ The goal is to minimize the squared error between the desired output d_i, and the actual net input net_i.

Formula

$$\Delta w_j = \eta \cdot (d_i - net_i) \cdot x_j$$

→ This rule adjusts the weights to reduce the mean-squared error.

5. Winner-Take-All Learning Rule

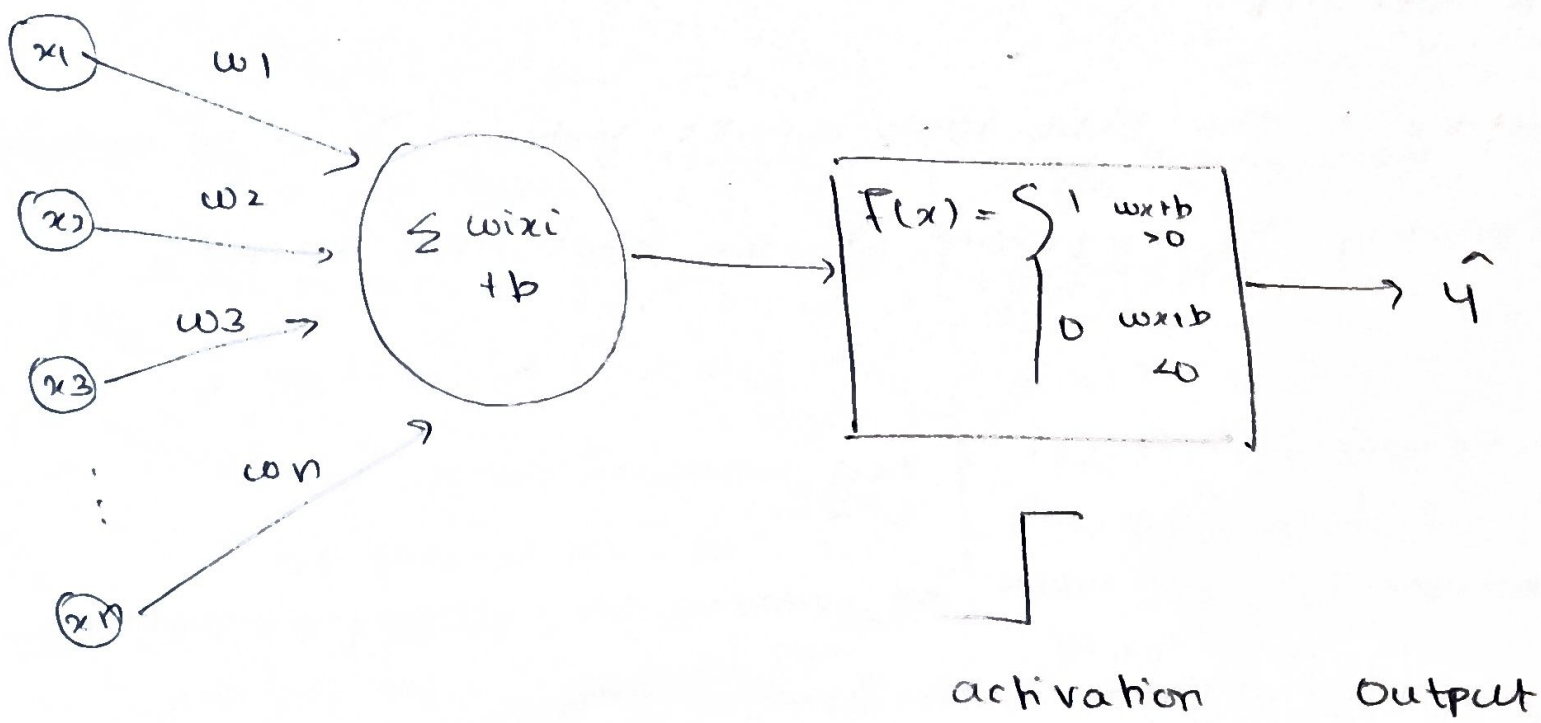
→ used in unsupervised learning

→ In this rule, used in competitive learning networks, only the neuron with the maximum response to the input is updated.

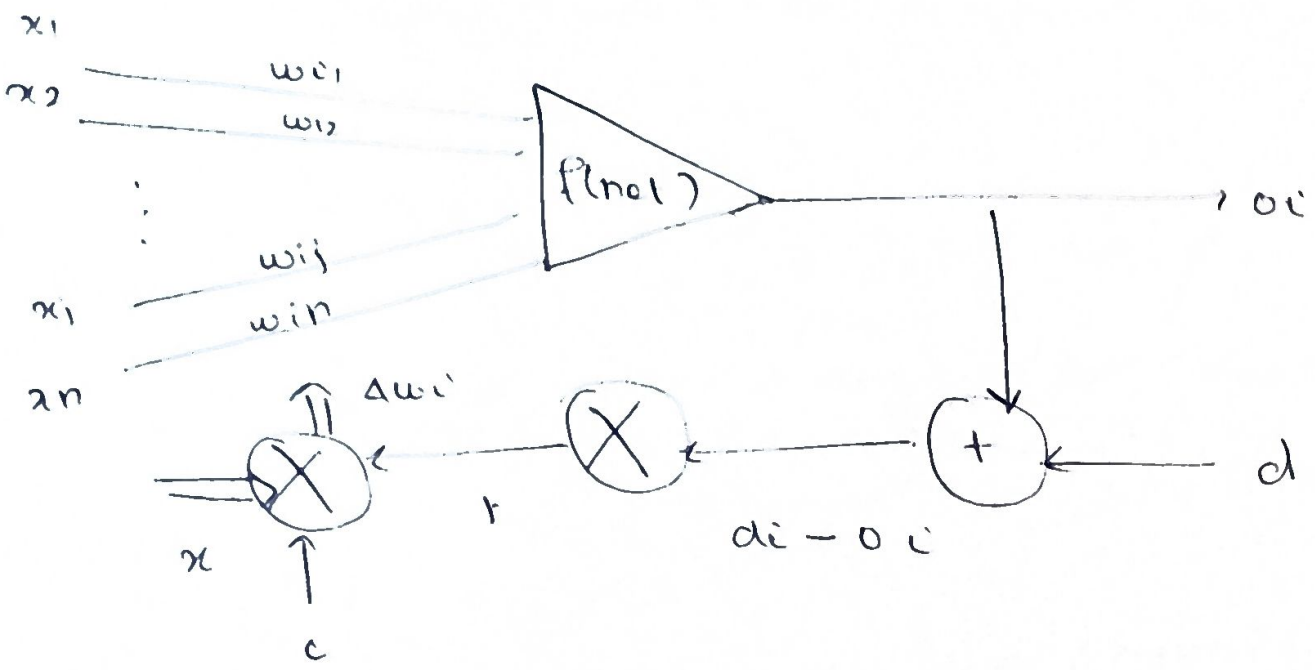
→ This neuron is declared the 'winner', and its weights are adjusted, while other neurons remain unchanged.

* Architecture Diagrams for Neural Networks

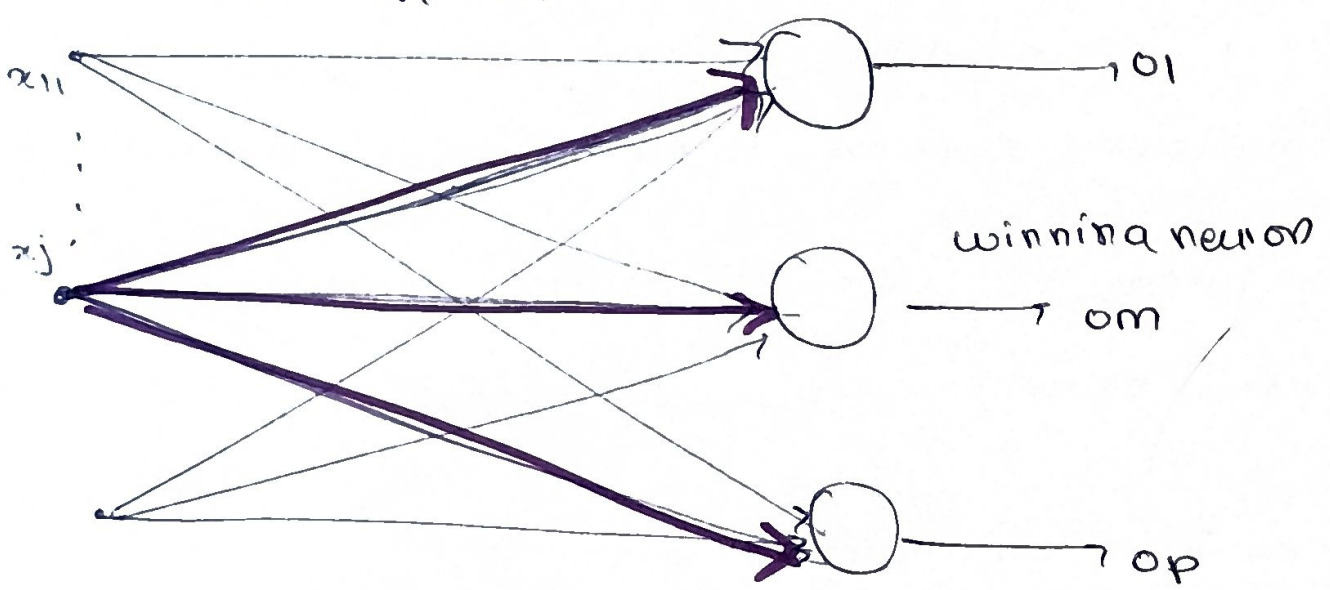
(i) Perceptron



(i) Delta Learning Rule



(iii) Winner Take All



highlighted = adjusted weights

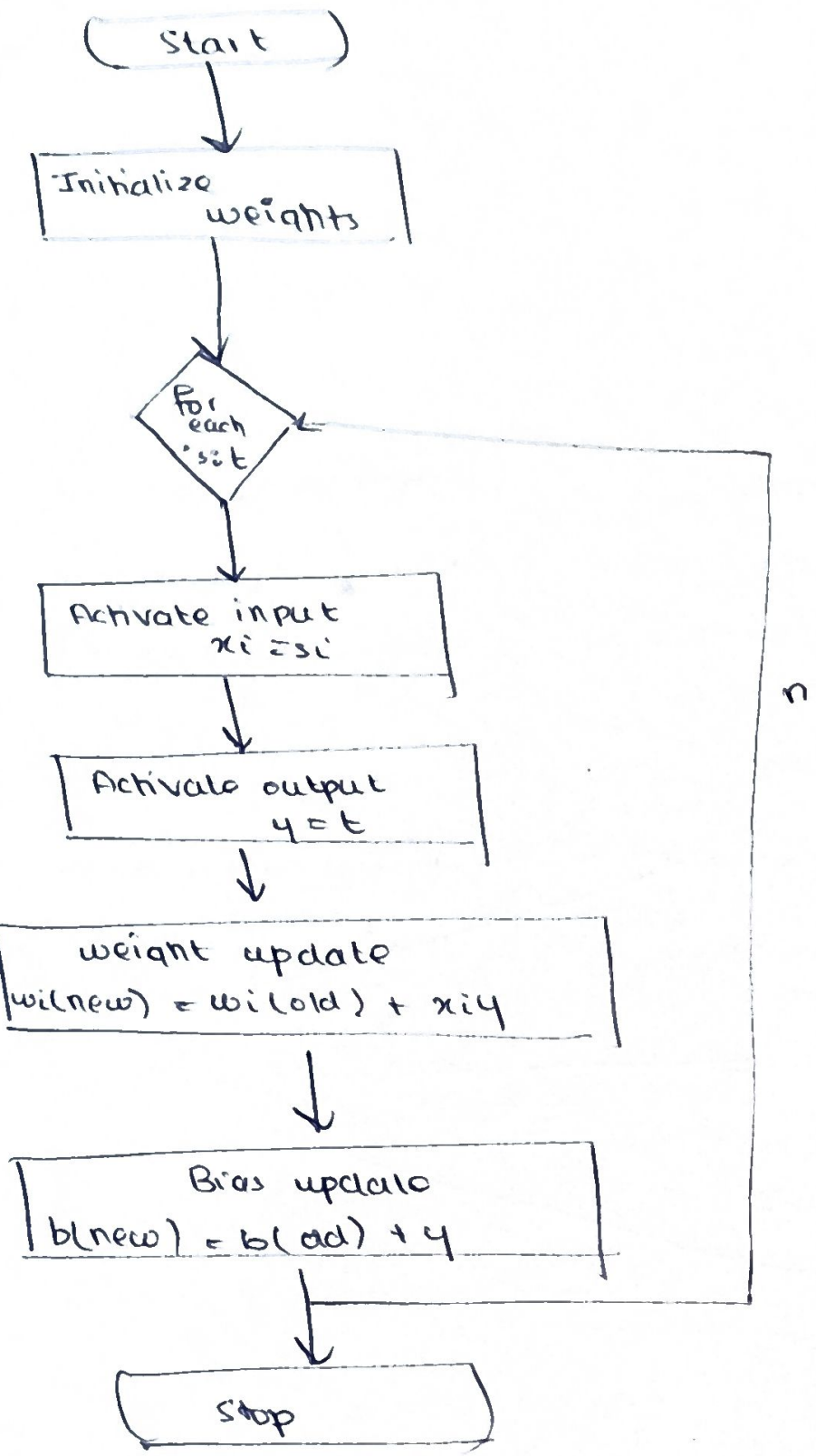
* Hebb Network

→ According to the Hebb rule, weight vector is found to increase proportionately to the product of the input and learning signal

$$w_i(new) = w_i(old) + x_{iy}$$

→ Hebb rule can be used for pattern association, pattern categorization, pattern classification etc.

Hebb Training Algorithm Flowchart



* Hebbnet vs. Perceptron

Criteria	Hebbnet	Perception
Type of Learning	unsupervised	supervised
activation function	no activation function - operates directly on weighted sum	step activation function

Criteria	Hebbnet	Perceptron
Error calculation	No explicit error calculation	error = d - o ↙ desired ↘ actual
Weight Update	$\Delta w = \eta \cdot o_i \cdot x_j$	$\Delta w = \eta \cdot (d-o)\, x_j$
Linearly Separable problems	cannot handle non linearly separable problems like XOR	Capable of solving linearly separable problems like AND, OR
Learning Objective	Strengthens connections between co-active neurons	Adjust weights to minimize classification error