EE 583
PATTERN RECOGNITION

NeuroPR : Fundamentals

Neurons and Neural Nets

Neural Network Structures

Artificial Neural Networks

Matrix Approaches to NeuroPR

Introduction

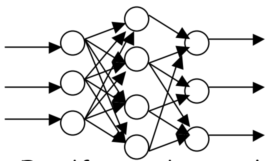
- NeuroPR is a relatively new approach to pattern recognition problem involving large interconnected networks of nonlinear units, simple called *neural nets*
- The main idea is imitating-partially known- human neural system which has a similar interconnective structure
- Since the basic element of human information processing system is slow, the basic of the biological computation should be based on parallel processing
- An (artificial) neural net has 3 entities :
  - Network topology
  - Characteristic of each neuron
  - Learning/training strategy

Key NeuralNet Concepts

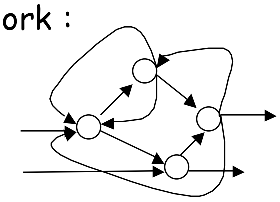
- Overall computation model consists of variable interconnection of simple elements
- Experience/training is stored in the form of network connections
- Objective is to develop an internal structure during experience/training so that this structure correctly classifies similar patterns
- Neural networks are dynamic since their outputs and interconnection weights may change over time

Neural Network Structures

- There are several generic neural network structures
  - Pattern Associator (PA) : Feedwork network
  - Content-Addressable or Associative Memory Model (CAM or AM) : Hopfield model
  - Self-Organizing Network :



Feedforwardnetwork



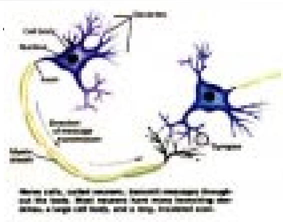
Hopfield-like model

Note that for networks with feedbacks, stability might be a problem

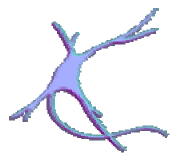
Learning in Neural Networks

- Similar to other PR approaches, there are two types of learning in Neuro PR :
  - Supervised learning :
    - without feedback (nonrecurrent)
      - Feedforward network with generalized delta rule
    - with feedback (recurrent)
      - Hopfield (CAM) network
  - Unsupervised learning :
    - without feedback (nonrecurrent)
      - Kohonen self-organizing network
    - with feedback (recurrent)
      - Networks based on Adaptive Resonance Theory (ART)

Physical Neural Networks



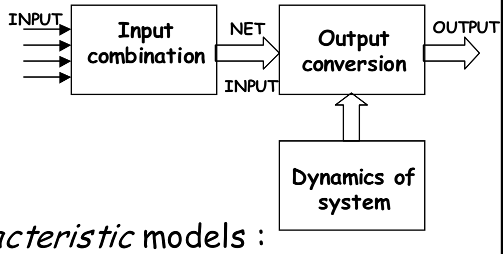
- 20 billion neural cells (neurons) exist in our brain
- Three major parts of a neuron
  - Soma : Cell nucleus
  - Axon : Output receptors
  - Dendrite : Input receptors
- Usually 1000 to 10,000 connections (*synapse*) to other neurons



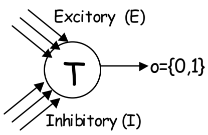
- There are sensor and motor neurons
- Light/sound→SensorNeuron→Motor Neuron

Artificial Neural Networks (1/2)

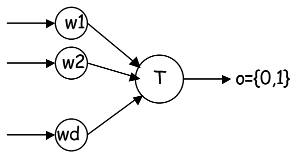
- An abstract representation of a single neuron



- Main *activation characteristic* models :



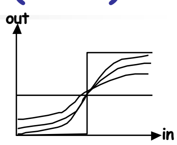
Inputs			*Output*
$E \geq T$	$I = 0$	$\Rightarrow$	1
$E \geq T$	$I > 0$	$\Rightarrow$	0
$E < T$	$I = 0$	$\Rightarrow$	0
$E < T$	$I > 0$	$\Rightarrow$	0



Input	*Output*
$\sum_{k=1}^{d} x_k w_k < T$	$\Rightarrow 0$
$\sum_{k=1}^{d} x_k w_k \geq T$	$\Rightarrow 1$

Artificial Neural Networks (2/2)

- *Activity functions* map neuron input activation into an output signal



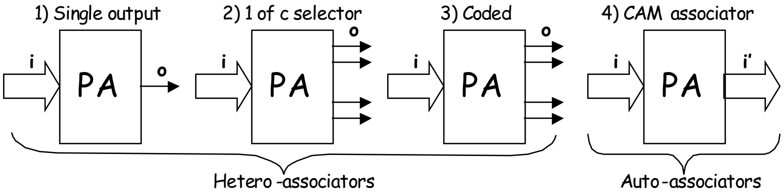
$$out = f(in) = \frac{1}{1+e^{-\lambda \cdot in}} \qquad out = f(in) = \begin{cases} 1 & in \geq 0 \\ 0 & in < 0 \end{cases} \qquad out = f(in) = \begin{cases} +1 & in \geq 0 \\ -1 & in < 0 \end{cases}$$

- Neural unit interconnection strenghts (weights) : w_{ab} represents connection to unit a from unit b and for a linear relation :

$$in_a = \sum_b w_{ab} out_b \quad \Rightarrow \quad \begin{bmatrix} in_1 \\ \vdots \\ in_e \end{bmatrix} = [w_{ab}]_{exd} \begin{bmatrix} out_1 \\ \vdots \\ out_d \end{bmatrix}$$

Neural Pattern Associators

- Linear input/output relations allows to use the tools of linear algebra to develop some insights into pattern storage and recall
- Let *i* be the input (stimulus) to network and *o* be the corresponding desired output (response); some typical pattern associators can be shown as below :



- Desirable PA should classify (or correct) and be supervised/unsupervised trainable

Matrix Approaches (1/4)

- For the linear case, since input/output relations can be written in terms of matrices, interconnection weights can be determined, easily
- However, training data is assumed to be general
- For a single layer network :

$$i_a = \sum_b w_{ab} o_b \quad \Rightarrow \quad \begin{bmatrix} i_1 \\ \vdots \\ i_e \end{bmatrix} = [w_{ab}]_{exd} \begin{bmatrix} o_1 \\ \vdots \\ o_d \end{bmatrix}, \quad i^1 = W_1 o^1$$

- For a multi-layer network :

$$i^2 = W_2 o^2 = W_2 W_1 o^1$$

Matrix Approaches (2/4)

- Assume a training set of *n* stimulus-response pairs
- For developing a network that recognizes each s^i (dx1 vector) to form the correct response, r^i (dx1), use a bank of "matched filters", w_i (1xd) to obtain

$$w_i.s^i = p^i \quad i=1,\ldots,n$$

- For the entire bank of n filters,

$$W_{nxd}\, s_{dx1} = p_{nx1}$$

- By determining the largest element of p_{nx1}, the corresponding closest stimulus r^i is found
- Such a system can store only *n* stimulus-response pairs in its internal structure

Matrix Approaches (3/4)

- As another approach, given a single stimulus-response pair (s,r) with s normalized to unit length, one can obtain $W=rs^T$
- In order to find response to stimulus s, we obtain

$$Ws = rs^Ts = r|s|^2 = r$$

- For a distorted stimulus s', network response becomes

$$Ws' = rs^Ts' = (s^Ts')\, r = \alpha\, r$$

a scaled version of the trained response (note that $s^Ts'<1$)

- For several stimulus-response pairs (s^i,r^i), use superposition of $W=rs^T$ for every pair as

$$W=\sum_{i=1}^{n} r^i (s^i)^T \quad \Rightarrow \quad \text{response for } s_u, \quad Ws_u=\sum_{i=1}^{n} r^i (s^i)^T s_u = \sum_{i=1}^{n} [(s^i)^T s_u] r^i$$

- If desired response to input s_u is r^p, then one should have

$$< s^i s^p > = \delta_{ip} = \begin{cases} 1 & \text{for } i=p \\ 0 & \text{elsewhere} \end{cases}$$

However, this requires orthonormality of stimulus which is not controllable

Matrix Approaches (4/4)

- For a general case, in order to determine unknown W matrix for n stimulus-response pairs, given the relation

  $W s_k = r_k \quad k=1,\ldots,n \quad [r_k : c\times1, s_k : d\times1]$

  [Note that the dimensions of r_k and s_k may not be equal]
- Using pseudo-inverse matrices, for the overall relation WS=R, the unknown matrix W is obtained as

  $W = R\,(S^TS)^{-1}S^T$
- If S is orthogonal → (S^TS)=Identity matrix → $W = RS^T$

$$W = \begin{bmatrix} r_1 & \cdots & r_n \end{bmatrix} \begin{bmatrix} s_1^T \\ \vdots \\ s_n^T \end{bmatrix} = \sum_{i=1}^{n} r_i s_i^T$$

- If S is orthogonal, the result reduces to the previous outer product formulation

Example

- For the given 3 training stimulus-response pairs, find weight matrix W

$$\left\{ s_1 = \begin{pmatrix}4\\1\\1\end{pmatrix}, r_1 = \begin{pmatrix}1\\0\\0\end{pmatrix} \quad s_2 = \begin{pmatrix}4\\1\\0\end{pmatrix}, r_2 = \begin{pmatrix}0\\1\\0\end{pmatrix} \quad s_3 = \begin{pmatrix}2\\0\\1\end{pmatrix}, r_3 = \begin{pmatrix}0\\0\\1\end{pmatrix} \right\}$$

- For this example, R is simply equal to identity matrix and $S=(s_1,s_2,s_3)$

$$S = \left\{ \begin{pmatrix}4\\1\\1\end{pmatrix} \begin{pmatrix}4\\1\\0\end{pmatrix} \begin{pmatrix}2\\0\\1\end{pmatrix} \right\}$$

- Since S has full column rank and has an inverse, W is obtained as

$$W = S^{-1} = \begin{pmatrix} -\frac{1}{2} & 2 & 1 \\ \frac{1}{2} & -1 & -1 \\ \frac{1}{2} & -2 & 0 \end{pmatrix}$$

- For perturbed version of s, we have s'=(4 1 2) which gives the interpolative response (2 -1 0)