Fundamentals of Neural Networks : AI Course lecture 37 Œ 38, notes, slides

www.myreaders.info/ , RC Chakraborty, e-mail rcchak@gmail.com , June 01, 2010

www.myreaders.info/html/artificial_intelligence.html

Fundamentals of Neural Networks

Artificial Intelligence

www.myreaders.info

Return to Website

Neural network, topics : Introduction, biological neuron model,

artificial neuron model, notations,

functions; Model of artificial

neuron - McCulloch-Pitts neuron equation; Artificial neuron Œ basic

elements, activation functions, threshold

function, piecewise linear

function, sigmoidal function; Neural
network architectures - single
layer feed-forward network, mult
i layer feed-forward network,
recurrent networks; Learning Methods in Neural
Networks -
classification of learning al
gorithms, supervised learning,
unsupervised learning, reinforced learning,
Hebbian learning,
gradient descent learning, competitive learning,
stochastic

learning. Single-Layer NN System - single layer
perceptron ,
learning algorithm for training, linearly separable
task, XOR
Problem, learning algorithm, AD
Aptive LINear Element (ADALINE)
architecture and training mechanism; Ap

plications of neural

networks - clustering, classificati

on, pattern recognition, function

approximation, prediction systems.

Fundamentals of Neural Networks

Artificial Intelligence

Topics

(Lectures 37, 38 2 hours)

Slides

1.

Introduction

Why neural network ?, Researchhistory, Biological

neuron model,

Artificial neuron model,

Notations, Functions.

03-12

2.

Model of Artificial Neuron

McCulloch-Pitts Neuron Equation, Ar

tificial neuron Œ basic elements,

Activation functions Œ threshold f

unction, piecewise linear function,

sigmoidal function.

13-19

3.

Neural Network Architectures

Single layer feed-forward network, Multi layer

feed-forward network,

Recurrent networks.

20-23

4 Learning Methods in Neural Networks

Classification of learning algorithms

, Supervised learning, Unsupervised

learning, Reinforced learning, Hebbian Learning,

Gradient descent

learning, Competitive learning, Stochastic

learning.

Neural Networks

What is Neural Net ?

Ł A neural net

is an artificial representation of the human brain that

tries to simulate its learning process. An artificial neural network

(ANN) is often called a "

Neural Network

" or simply Neural Net (NN).

Ł Traditionally, the word neural network is referred to a network of

biological neurons

in the nervous system that process and transmit

information.

Ł Artificial neural network is an interconnected group of

artificial neurons

that uses a mathematical model or computational model for information processing based on a connectionist approach to computation.

- The artificial neural networks are made of interconnecting artificial neurons which may share some properties of biological neural networks.
- Artificial Neural network is a network of simple processing elements (neurons) which can exhibit complex global behavior, determined by the connections between the processing elements and element parameters.
- Artificial neural network is an adaptive system that changes its structure based on external or internal information that flows

through the network.

03

AI-Neural Network Œ Introduction

1. Introduction

Neural Computers mimic certain processing capabilities of the human brain.

- Neural Computing is an information processing paradigm , inspired by biological system, composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems.
- Artificial Neural Networks (ANNs), like people, learn by example .
- An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process.
- Learning in biological systems involves

adjustments to the synaptic
connections that exist between the neurons. This is true of ANNs as well.

04

AI-Neural Network Œ Introduction

1.1 Why Neural Network

The conventional computers are good for

-

fast arithmetic and does
what programmer programs, ask them to do.

The conventional computers are
not so good for - interacting with
noisy data or data from the enviro
nment, massive parallelism, fault
tolerance, and adapting to circumstances.

The neural network systems help where we can not formulate an
algorithmic solution or where we can get lots of examples of the

behavior we require.

Neural Networks follow different paradigm for computing.

The von Neumann machinesare based on the processing/memory abstraction of human information processing.

The neural networks are based on the parallel architecture of

biological brains.

Neural networks are a form of multiprocessor computer system, with

- simple processing elements ,
- a high degree of interconnection,
- simple scalar messages, and
- adaptive interaction between elements.

Research History

The history is relevant because for nearly two decades the future of Neural network remained uncertain. McCulloch and Pitts (1943) are generally recognized as the designers of the

first neural network. They combined many simple processing units together that could lead to an overall increase in computational power. They suggested many ideas like : a neuron has a threshold level and once that level is reached the neuron fires. It is still the fundamental way in which ANNs operate. The McCulloch and Pitts's network had a fixed set of weights. Hebb (1949) developed the first learning rule

, that is if two neurons are
active at the same time then the strength
between them should be

increased.
In the 1950 and 60's, many researchers (Block, Minsky, Papert, and
Rosenblatt worked on
perceptron. The neural network model could be proved to converge to the correct weights, that will solve the problem. The
weight adjustment
(learning algorithm) used in the perceptron was found
more powerful than the learning rules used by Hebb. The perceptron caused
great excitement. It was thought to produce programs that could think.

Minsky & Papert (1969) showed that perceptron

could not learn those
functions which are not linearly separable.
The neural networks
research declined
throughout the 1970 and until mid
80's because the perceptron could not learn
certain important functions.
Neural network
regained importance
in 1985-86. The researchers, Parker
and LeCun discovered a learning algorithm for
multi-layer networks called
back propagation
that could solve problems that were not linearly
separable.

1.3

Biological Neuron Model

The human brain consists of a large number, more than a
billion of
neural cells
that process information. Each cell works like a simple
processor. The massive interaction between all cells and their parallel
processing only makes the brain's abilities possible.

Fig. Structure of Neuron

Dendrites
are branching fibers that
extend from the cell body or soma.
Soma or cell body
of a neuron contains
the nucleus and other structures, support

chemical processing and production of

neurotransmitters.

Axon is a singular fiber carries
information away from the soma to the

synaptic sites of other neurons (dendrites
and somas), muscles, or glands.
Axon hillock
is the site of summation
for incomin
g information. At any
moment, the collective influence of all

neurons that conduct impulses to a
given neuron will determine whether or not an

action potential will be initiated at the
axon hillock and propagated along the axon.
Myelin Sheath
consists of fat-containin

g cells that insulate the axon from electrical activity. This insulation acts to inc
rease the rate of transmission of si
gnals. A gap exists between each myelin sheath cell
along the axon. Since fat inhibits the propagation of electricity, the signal
s jump from one gap to the next.
Nodes of Ranvier
are the gaps (about 1
µm) between myelin sheath cells lon
g axons
are Since fat serves as a
good insulator, the myelin sheaths speed the rate of
transmission of an electrical impulse along the axon.

Synapse
is the point of connection between two neurons

or a neuron and a muscle or

a gland. Electrochemical communication between neurons takes place at these junctions. Terminal Buttons of a neuron are the small knobs at the end of an axon that release chemicals called neurotransmitters.

07

AI-Neural Network Œ Introduction

Ł Information flow in a Neural Cell

The input /output and the propagation of information are shown below.

Structure of a Neural Cell in the Human Brain

Dendrites receive activation from other neurons.

Soma processes the incoming activations and converts them into output activations.

Axons act as transmission lines to send activation to other neurons.

Synapses the junctions allow signal transmission between the axons and dendrites.

The process of transmission is by diffusion of chemicals called neuro-transmitters.

McCulloch-Pitts introduced a simplified model of this real neurons.

1.4 Artificial Neuron Model

An artificial neuron is a mathematical function conceived as a simple model of a real (biological) neuron.

Ł The McCulloch-Pitts Neuron

This is a simplified model of real ne
urons, known as a Threshold Logic Unit.

Input
1 Input 2 Input
n

A set of input connections brings
in activations from other neurons. A
processing unit sums the inputs, and then applies
a non-linear
activation function (i.e. squashin
g / transfer / threshold function).

An output line transmits the result to other
neurons.

In other words ,

- The input to a neuron arrives in the form of
signals.
- The signals build up in the cell.
- Finally the cell discharges (cell fires) through

the output .

- The cell can start building up signals again.

09

Output

AI-Neural Network Œ Introduction

1.5 Notations

Recaps : Scalar, Vectors, Matrices and Functions

Ł Scalar : The number

xi can be added up to give a scalar number.

s = x1 + x

2 + x3 + ... + x

n = x

i Ł Vectors : An ordered sets of related numbers. Row Vectors

(1 x n) X = (x

1 , x2 , x3 , . . ., x

n) , Y = (y

1 , y2 , y

3 , . . ., y

n) Add :

Two vectors of same length added to give

another vector.

Z = X + Y = (x

1 + y1 , x

2 + y2 , ... , x

n + y

n) Multiply:

Two vectors of same length multiplied to give a

scalar.

p = X . Y = x

1 y1 + x2 y2 + ... + x

nyn = x

i yi 10

i=1

ni=1n

AI-Neural Network Œ Introduction

Ł Matrices : m x n matrix , row no = m ,

column no = n

w11w11. . . . w

1n

w21w21. . . . w

21

W =

. . . .

. . . .

wm1w11. . . . w

mn

Add or Subtract

: Matrices of the same size are added or

subtracted

component by component.

A + B = C ,

cij = aij + b

ij a11 a12 b

11 b12 c11 = a11+b11 c12 = a12+b12a21 a22 b

21 b22 C21 = a21+b21 C

22 = a22 +b22

Multiply :

matrix

A

multiplied by matrix

B gives matrix

C.

(m x n) (n x p) (m

x p)

elements

cij =

aik bkj a11 a12 b

11b12 c11 c12 a21 a22 b

21b22 c21 c22

c11 = (a

11 xb

11)+ (a12xB21)c12 = (a

11 xb

12)+ (a12xB22)C21 = (a

21 xb

11)+ (a22xB21)C22 = (a

21 xb

12)+ (a22xB22)

11

+ =k=1nx =

AI-Neural Network Œ Introduction

1.6 Functions The

Function y= f(x)

describes a relationship, an input-output mapping,

from x to y. Threshold or Sign function

: sgn(x)

defined as

1

if x 0 sgn (x)

=

0 if x

< 0

Sign(x)

O/P

-4 -3 -2 -1 0 1 2 3

4

I/P

Threshold or Sign function

: sigmoid(x)

defined as a smoothed

(differentiable) form of the threshold function

1

sigmoid (x)

=

1 + e

-x Sign(x)

O/P

-4 -3 -2 -1 0 1 2 3

4

I/P

2. Model of Artificial Neuron

A very simplified model of real neurons is known as a Threshold Logic Unit (TLU). The model is said to have :

- A set of synapses (connections) brings in activations from other neurons.
- A processing unit sums the inputs, and then applies a non-linear activation function (i.e. squashing / tran sfer / threshold function).
- An output line transmits the result to other neurons.

2.1 McCulloch-Pitts (M-P) Neuron Equation

McCulloch-Pitts neuron is a simplif ied model of real biological neuron.

Input 1 Input 2 Input n

Simplified Model of Real Neuron

(Threshold Logic

Unit)

The equation for the output of a

McCulloch-Pitts neuron as a function

of 1 to n inputs is written as

Output

=

sgn

(

Input

i -

)

where is the neuron™s activation threshold.

If

Input i

then Output

= 1

If

Input i <

then Output

= 0 In this McCulloch-Pitts neuron model, the missing features are :

- Non-binary input and output,
- Non-linear summation,
- Smooth thresholding,
- Stochastic, and
- Temporal information processing.

13

Output

i=1n i=1 n i=1 n

AI-Neural Network Œ Model Neuron

2.2 Artificial Neuron - Basic Elements

Neuron consists of three basic components - weights, thresholds, and a single activation function

.

Fig Basic Elements of an Artificial Linear Neuron

Weighting Factors

w The values w1 , w

2 , . . . w

n are weightsto determine the strength of

input vector

X = [x1 , x

2 , . . . , x

n]T. Each input is multiplied by the

associated weight of the neuron connection

XT W. The +ve weight

excites and the -ve weight inhibits the node

output.

I = XT.W = x

1 w1 + x

2 w2 + ... + x

nwn = x

i wi Threshold The node™s internal threshold

is the magnitude offset. It affects the

activation of the node output

y as:

Y = f (I)

= f { x

i wi - k } To generate the final output

Y , the sum is passed on to a non-linear

filter f called Activation Function or Tran

sfer function or Squash function

which releases the output

Y. 14

W1 W2Wnx1 x2 xn Activation

Function

i=1

SynapticWeights

Threshold

y i=1 ni=1

n

AI-Neural Network Œ Model of Neuron

Threshold for a Neuron

In practice, neurons generally do not fire (produce an output) unless their total input goes above a threshold value.

The total input for each neuron is the sum of the weighted inputs to the neuron minus its threshold value. This is then passed through the sigmoid function. The equation for the transition in a neuron is :

$$a = 1/(1 + \exp(- x))$$

where x = ai wi - Q a is the activation for the neuron

ai is the activation for neuron

i wi is the weight

Q is the threshold subtracted

Activation Function

An activation function

f performs a mathematical operation on the signal output. The most common activation functions are:

- Linear Function,
- Piecewise Linear Function,
- Tangent hyperbolic function
- Threshold Function,
- Sigmoidal (S shaped) function,

The activation functions are chosendepending upon the type of

problem to be solved by the network.

i

2.2 Activation Functions

f -

Types

Over the years, researches tried several functions to convert the input into an outputs. The most commonly used functions are described below.

- I/P Horizontal axis shows sum of inputs .
- O/P Vertical axis shows the value the function produces ie output.
- All functions f are designed to produce values between
0 and

1. Ł Threshold Function

A threshold (hard-limiter) activation function is either a binary type or a bipolar type as shown below.

binary threshold

O/p

I/P

Output of a binary threshold function produces :

1 if the weighted sum of the inputs is +ve,

0 if the weighted sum of the inputs is -ve.

1 if I

0 Y = f (I) =

0 if I

< 0

bipolar threshold

O/p

I/P Output of a bipolar threshold function produces :

1 if the weighted sum of the inputs is +ve,

-1 if the weighted sum of the inputs is -ve.

1

if I 0

Y = f (I) =

-1 if I

< 0 Neuron with hard limiter activation func

tion is called McCulloch-Pitts model.

16

1

1

-1

AI-Neural Network Œ Model of Neuron

Ł Piecewise Linear Function

This activation function

is also called saturating

linear function and can

have either a binary or bipolar range for the

saturation limits of the output.

The mathematical model for a symmetri

c saturation function is described

below.

Piecewise Linear

O/p

I/P This is a sloping function that produces :

-1 for a -ve weighted sum of inputs,

1 for a +ve weighted sum of inputs.

I proportional to input for values between

+1 and

-1 weighted sum,

1

if I 0 Y = f (I) = I if -1

I 1

-1 if I

< 0

17

+1 -1

AI-Neural Network Œ Model of Neuron

Ł Sigmoidal Function

(S-shape function)

The nonlinear curved S-shape function is called the sigmoid function.

This is most common type of activation used to construct the neural networks. It is mathematically well behaved, differentiable and strictly increasing function.

Sigmoidal function

A sigmoidal transfer function can be written in the form:

1

Y = f (I) = , 0

f(I)

1 1 + e

- I

=

1/(1 + exp(-

I)) , 0 f(I)

1This is explained as

0 for large -ve input values,

1 for large +ve values, with a smooth transition between the two.

is slope parameter also called shape parameter;

symbol the

is also used to represented this parameter.

The sigmoidal function is achieved using exponential equation.

By varying

different shapes of the function can be obtained which adjusts the abruptness of the function as it changes between the two asymptotic values.

1

O/P

0.5

I/P

-4 -2 0 1 2

= 1.0

= 0.5

= 2.0

AI-Neural Network Œ Model of Neuron

Ł Example :

The neuron shown consists of four inputs with the weights.

Fig Neuron Structure of Example

The output

I of the network, prior to the activation function stage, is

+1

+1 I = X

T. W =

1 2 5 8

= 14

-1

+2

= (1 x 1) + (2 x 1) + (5 x -1) + (8 x 2) = 14

With a binary activation function

the outputs of the neuron is:

y (threshold

) = 1;

19

+1+1+2 -1x1=1 x2=2 xn=8 Activation

Function

Summing

Junction

Synaptic

Weights

= 0 Threshold

yX3=5 I

AI-Neural Network Œ Architecture

3. Neural Network Architectures

An Artificial Neural Network (ANN) is

a data processing system, consisting

large number of simple

highly interconnected processing elements

as

artificial neuron in a network structure that can

be represented using a

directed graph

G, an ordered 2-tuple

(V, E) , consisting a set

V of vertices

and a set

E of edges.

- The vertices may represent neurons

(input/output) and

- The edges may represent synaptic links

labeled by the weights attached.

Example :

Fig. Directed Graph

Vertices V = {

v1 , v

2 , v3 ,

v4,

v5 } Edges E = {

e1 , e

2 , e3 , e4, e5 } 20

V1 V3V2 V4V5e3e2e5e4e5

AI-Neural Network Œ Architecture

3.1 Single Layer Feed-forward Network

The Single Layer Feed-forward Network consists of a single layer of weights , where the inputs are directly connected to the outputs, via a series of weights.

The synaptic links carrying weights connect every input to every output , but not other way.

This way it is considered a network of feed-forward type. The sum of the products of the weights and the inputs is calculated in each neuron node, and if the value is above some threshold (typically 0) the neuron fires and takes the activated value (typically 1); otherwise it takes the deactivated value (typically -1).

Fig. Single Layer Feed-forward Network

w21w11w12wn2wn1w1mw2mwnmw22y1 y2 ym x1

x2 xn out

put yj input xi weights wijSingle layer

Neurons

AI-Neural Network Œ Architecture

3.2 Multi Layer Feed-forward Network

The name suggests, itconsists of multiple layers.

The architecture of

this class of network, besides having the input

and the output layers,

also have one or more intermediary layers called

hidden layers

. The

computational units of the hidden layer are

known as

hidden neurons

.

Fig. Multilayer feed-forward network in

(Œ m Œ n)

configuration

. - The hidden layer does intermediate

computation before directing the

input to output layer.

- The input layer neurons are linked to the hidden

layer neurons; the

weights on these links are referred to as

input-hidden layer weights

. - The hidden layer neurons and the co

rresponding weights are referred to

as output-hidden layer weights

. - A multi-layer feed-forward network with

input neurons,

m1 neurons in

the first hidden layers, m2 neurons in the second

hidden layers, and n

output neurons in the output layers is written as

(- m1 - m2 Œ n).

- Fig. above illustrates a multilayer feed-forward network with a configuration
(- m Œ n).

22
w11w12v21v11w1mvn1v1m v2m Vmw11x1 x2 x
y3y1y2yny1ymHidden Layer
neurons
yj Output Layer
neurons
zk Input Layer
neurons
xi Input
hidden layer
weights
vijOutput
hidden layer
wei

ghts

wjk

3.3

Recurrent Networks

The Recurrent Networks differ from feed-forward architecture.

A Recurrent network has at least one feed back loop.

Example :

Fig. Recurrent neural network

There could be neurons with self-feedback links; that is the output of a neuron is feedback into itself as input.

x1 x2 X y2y1Yny1ymHidden Layer

neurons

yj Output Layer

neurons

zk Input Layer

neurons

xi Feedback

links

4. Learning methods in Neural Networks

The learning methods in neural networks

are classified into three basic types :

- Supervised Learning,
- Unsupervised Learning
- Reinforced Learning

These three types are classified based on :

- presence or absence of

teacher

and

- the information provided for the system to learn.

These are further categorized, based on the rules used, as

- Hebbian, - Gradient descent,
- Competitive
- Stochastic learning.

AI-Neural Network ŒLearning methods

Classification of Learning Algorithms

Fig. below indicate the hierarchical re presentation of the algorithms mentioned in the previous slide. These algorithms are explained in subsequent slides.

Fig. Classification of learning algorithms

Neural Network

Learning algorithms

Unsupervised Learning

Supervised Learning

(Error based)

Reinforced Learning

(Output based)

Error Correction

Gradient descent

Stochastic

BackPropagationLeast Mean

Square

Hebbian

Competitive

AI-Neural Network ŒLearning methods

Ł Supervised Learning

- A teacher is present during learning process and presents expected output.
- Every input pattern is used to train the network.
- Learning process is based on comparison, between network's computed output and the correct expected output, generating "error".
- The "error" generated is used to change network parameters that result improved performance.

Ł Unsupervised Learning

- No teacher is present.
- The expected or desired output is not presented to the network.

- The system learns of it own by discovering and adapting to the structural features in the input patterns.

Ł Reinforced Learning

- A teacher is present but does not present the expected or desired output but only indicated if the computed output is correct or incorrect.
- The information provided helps the network in its learning process.
- A reward is given for correct answer computed and a penalty fora wrong answer.

Note :

The Supervised and Unsupervised learning methods are most popular forms of learning compared to Reinforced learning.

AI-Neural Network ŒLearning methods

Ł Hebbian Learning

Hebb proposed a rule based on correlative weight adjustment.

In this rule, the input-output pattern pairs (X_i , Y_i) are associated by the weight matrix W, known as correlation matrix computed as

W = Xi Yi

T

where

YiT is the transpose of the associated output vector

Yi There are many variations of this rule proposed by the other researchers (Kosko, Anderson, Lippman) i=1

n

AI-Neural Network ŒLearning methods

Ł Gradient Descent Learning

This is based on the minimization of errors

E defined in terms of weights

and the activation function of the network.

- Here, the activation function of the network is

required to be

differentiable, because the updates of weight

is dependent on

the gradient of the error

E. - If Wij is the weight update of the link

connecting the

i th and the

j thneuron of the two neighboring layers, then

Wij is defined as

Wij

= (E / Wij)

where

is the learning rate parameters and

(E / Wij) is error

gradient with reference to the weight

Wij

. Note :

The Hoffs Delta rule and Back-propagation learning rule are

the examples of Gradient descent learning.

28

AI-Neural Network ŒLearning methods

Ł Competitive Learning

- In this method, those neurons which respond strongly to the input

stimuli have their weights updated.

- When an input pattern is presented, all neurons in the layer compete, and the winning neuron undergoes weight adjustment .

- This strategy is called "winner-takes-all"

.

Ł Stochastic learning

- In this method the weights are adjusted in a probabilistic fashion.

- Example : Simulated annealing which is a learning mechanism employed by Boltzmann and Cauchy machines.

5. Single-Layer NN Systems

Here, a simple Perceptron Model and an ADALINE Network Model is presented.

5.1 Single Layer Perceptron

Definition : An arrangement of one input layer of neurons feed forward to one output layer of neurons is known as Single Layer Perceptron.

Fig. Simple Perceptron Model

1

if net j 0 y j = f (net j) =

where net j =

xi wij

0

if

net j

< 0 30

i=1 nw21w11w12wn2wn1w1mw2mwnmw22y1 y2

ym x1 x2 xn output yjinput xi weights

wijSingle layer

Perceptron

AI-Neural Network ŒSingle Layer learning

Ł Learning Algorithm for Training Perceptron

The training of Perceptron is a supervised learning algorithm where weights are adjusted to minimize error when ever the output does not match the desired output.

– If the output is correct then no adjustment of weights is done.

i.e.

=

– If the output is

1 but should have been 0then the weights are decreased on the active input link

i.e.

=

– . x

i – If the output is

0 but should have been

1 then the weights are

increased on the active input link

i.e.

=

\+ . xi Where is the new

adjusted weight, is the old weight

x

i is the input and

is the learning rate parameter.

small leads to slow and

large leads to fast learning.

31

W i j

K+1

W i j

K+1 W i jKW i j

K+1 W i jKW i j K+1 W i jKWi jK

AI-Neural Network ŒSingle Layer learning

Ł Perceptron and Linearly Separable Task

Perceptron can not handle tasks which are not separable.

- Definition : Sets of points in 2-D space are linearly separableif the sets can be separated by a straight line.

- Generalizing, a set of points in n-dimensional space are linearly separable if there is a hyper plane of (n-1) dimensions separates the sets.

Example

S1

S2

S1

S2

(a) Linearly separable patterns

(b) Not Linearly separable patterns

Note :

Perceptron cannot find weights for classification problems that are not linearly separable.

Ł XOR Problem :

Exclusive OR operation

Input x1 Input x2 Output

0 0 0

1 1 0

0 1 1

1 0 1

XOR

truth table

Even parity means even number of 1 bits in the input

Odd parity means odd number of 1 bits in the input

X2

(0, 1) (1, 1)

(0, 0) X1

(0, 1)

Output of XOR

in X1 , x2 plane

- There is no way to draw a single straight line so that the circles
are on one side of the line and the dots on the other side.

- Perceptron is unable to find a
line separating even parity input
patterns from odd parity input patterns.

33 • °• °Even
parit
y •Odd parit
y °

AI-Neural Network ŒSingle Layer learning

Ł Perceptron Learning Algorithm

The algorithm is illustrated step-by-step.

Step 1 :

Create a peceptron with
(n+1)
input neurons

x0 , x1 , ... , . x

n , where

x

0 = 1 is the bias input

. Let

O be the output neuron

. Step 2 :

Initialize weight

W = (w

0 , w

1 , ... , . w

n) to random weights

. Step 3 :

Iterate through the input patterns

Xj of the training set using the

weight set;

ie compute the weighted sum of inputs

net j = x

i wifor each input pattern

j . Step 4 :

Compute the output

y j using the step function

1

if net j

0 y

j = f (net

j) =

where

net

j =

xi wij

0

if net

j

< 0

Step 5 :

Compare the computed output

yj with the target output

yj

for each input pattern

j .

If all the input patterns have been

classified correctly, then output

(read) the weights and exit.

Step 6 :

Otherwise, update the weights as given below :

If the computed outputs

yj is 1 but should have been

0,

Then wi = wi -

xi , i= 0, 1, 2, ... , n

If the computed outputs

yj is 0 but should have been

1,

Then wi = wi +

x_i , i= 0, 1, 2, ... , n

where

is the learning parameter and is constant.

Step 7 :

goto step 3

END

34

i=1

n i=1

n

AI-Neural Network ŒADALINE

5.2 ADAptive LINear Element (ADALINE)

An ADALINE consists of a single ne

uron of the McCulloch-Pitts type,

where its weights are determined by the

normalized least mean

square (LMS) training law. The LMS lear

ning rule is also referred to as
delta rule
. It is a well-established
supervised training
method that
has been used over a wide range of diverse applications.

Ł Architecture of a simple
ADALINE

The basic structure of an ADALINE is similar to a neuron with a
linear activation function and a feedback loop. During the training
phase of ADALINE, the input vector as well as the desired output
are presented to the network.

[The complete training mechanism has been explained in the next slide.
]35

W1W2Wnx1 x2 xn Neuron

ErrorDesiredOutput

Output

Œ +

AI-Neural Network ŒADALINE

Ł ADALINE

Training Mechanism

(Ref. Fig. in the previous slide - Architecture of a simple ADALINE)

The basic structure of an ADALINE is similar to a linear neuron
with an extra feedback loop
. During the training phase of ADALINE, the input vector

X = [x

1 , x

2 , . . . , x

n]T

as well as desired output are presented

to the network.

The weights are adaptively adjusted

based on delta rule.

After the ADALINE is trained, an input vector

presented to the

network with fixed weights will

result in a scalar output.

Thus, the network performs an

ndimensional mapping to a

scalar value.

The activation function is not used during the

training phase.

Once the weights are properly adjusted, the

response of the

trained unit can be tested by applying various inputs, which are

not in the training set. If the network produces consistent

responses to a high degree with the test inputs, it is said

that the network could generalize. The process of training and

generalization are two important attributes of this network.

Usage of ADLINE : In practice, an ADALINE is used to

- Make binary decisions; the output is sent through a binary threshold.

- Realizations of logic gates such as AND, NOT and OR .

- Realize only those logic functions

that are linearly separable.

6. Applications of Neural Network

Neural Network Applications can be grouped in following categories:

Clustering:

A clustering algorithm explores the similarity between patterns and places similar patterns in a cluster. Best known applications include data compression and data mining.

Classification/Pattern recognition:

The task of pattern recognition is to assignan input pattern (like handwritten symbol) to one of many classes. This category

includes algorithmic implementations such as associative memory.

Function approximation :

The tasks of function approximation is to find an estimate of the unknown function subject to noise.

Various engineering and scientific disciplines require function approximation.

Prediction Systems:

The task is to forecast some fu ture values of a time-sequenced data. Prediction has a significant impact on decision support systems.

Prediction differs from function approximation by considering time factor.

System may be dynamic and may pr oduce different results for the same input data based on system state (time).

AI-AI-Neural Network ŒReferences

7. References : Textbooks

1.
"Neural Networks: A Comprehensive Founda
tion", by Simon S. Haykin, (1999),
Prentice Hall, Chapter 1-15, page 1-889.

2.
"Elements of Artificial Neural Networks",
by Kishan Mehrotra, Chilukuri K. Mohan
and Sanjay Ranka, (1996), MIT Press, Chapter 1-7,
page 1-339.

3.
"Fundamentals of Neural Networks: Architec
ture, Algorithms and Applications", by
Laurene V. Fausett, (1993), Prentice Hall,
Chapter1-7, page 1-449.

4.
"Neural Network Design", by Martin T. Hagan, Howard B. Demuth and Mark Hudson Beale, (1996) , PWS Publ. Company, Chapter 1-19, page 1-1 to 19-14.

5.
"An Introduction to Neural Networks", by James A. Anderson, (1997), MIT Press,

Chapter 1- 17, page 1-585.

6.
"AI: A New Synthesis", by Nils J. Nilsson, (1998), Morgan Kaufmann Inc., Chapter 3, Page 37-48.

7. Related documents from open source, mainly internet. An exhaustive list is being prepared for inclusion at a later date.