ARTIFICIAL INTELLIGENCE

Unit – 3 Artificial Neural Network

HISTORY OF NEURAL NETWORK

- The roots of all work of NN are in neurobiological studies.
- Some questions: William James (1890)

  - how nervous system works

- The amount of activity at any given point in the brain cortex(coating) is the sum of the tendencies of all other points to discharge into it, such tendencies being proportionate.

1. to the number of times the excitement of other points may have accompanied that of the point in question;
2. to the intensities of such excitements;
3. to the absence of any rival point functionally disconnected with the first point, into which the discharges may be diverted.

HISTORY OF NEURAL NETWORK

- How do nerves behave when stimulated by different magnitudes of electric current?
- Is there a minimal threshold (quantity of current) needed for nerves to be activated?
- how do different nerve cells communicate electrical currents among one another?
- How do various nerve cells differ in behavior?

- //up to mid-twentieth century not answered……

HISTORY OF NEURAL NETWORK

- From psychologists striving to understand exactly how learning, forgetting, recognition, and other such tasks are accomplished by animals.
- McCulloch and Pitts (1943)

  - credited first mathematical model of a single neuron
- Dreyfus (1962), Bryson and Ho (1969), and Werbos (1974)

  - combinations of many neurons
- Hybrid systems Gallant (1986)

  - combining neural networks and non-connectionist components

and many more……..

BIOLOGICAL NEURAL NETWORKS

- A biological neural network is a series of interconnected neurons whose activation defines a recognizable linear pathway.

- The interface through which neurons interact with their neighbours usually consists of several axon terminals connected via synapses to dendrites on other neurons.

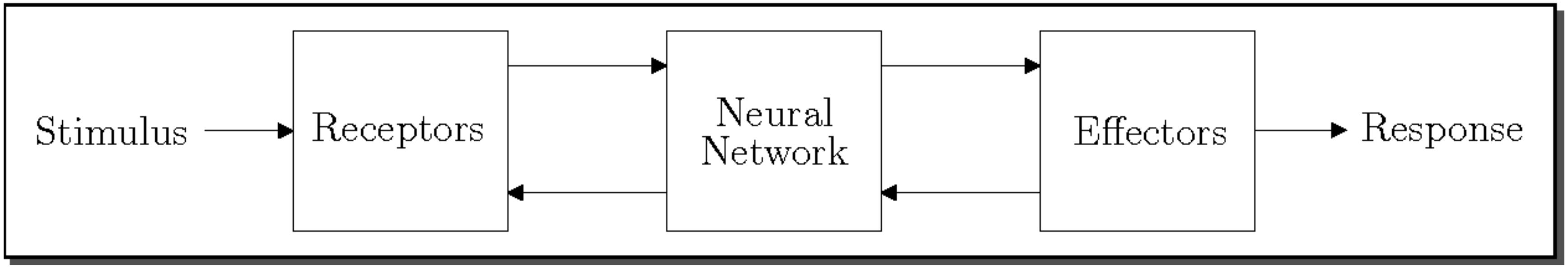
- If the sum of the input signals into one neuron surpasses a certain threshold, the neuron sends an action potential at the axon hillock and transmits this electrical signal along the axon.

BIOLOGICAL NEURAL NETWORKS

- Nervous System
  - Neurons
    - What?→ A neuron is a nerve cell that is fundamental building block of the biological nervous system. Neurons are similar to other cells in the human body in a number of ways, but there is one key difference between neurons and other cells. Neurons are specialized to transmit information throughout the body

    - 10 – 100 billions Neurons
    - connection to 100 – 10000 other neurons
    - 100 different types
    - signal

NERVOUS SYSTEM



NERVOUS SYSTEM

- The receptors collect information from the environment

  – e.g. photons on the retina

- The effectors generate interactions with the environment

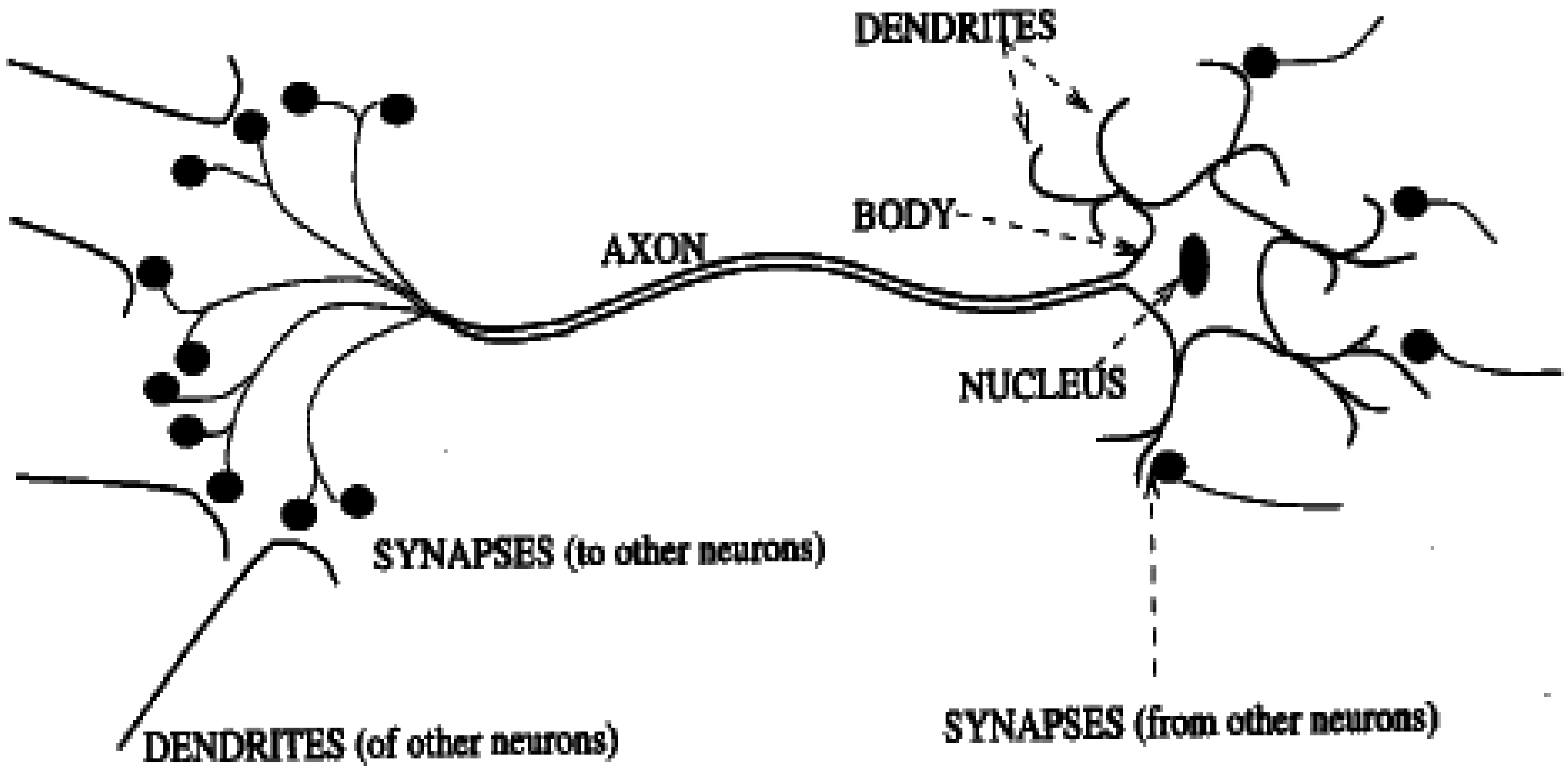
  – e.g. activate muscles

- The flow of information/activation is represented by arrows feedforward and feedback.

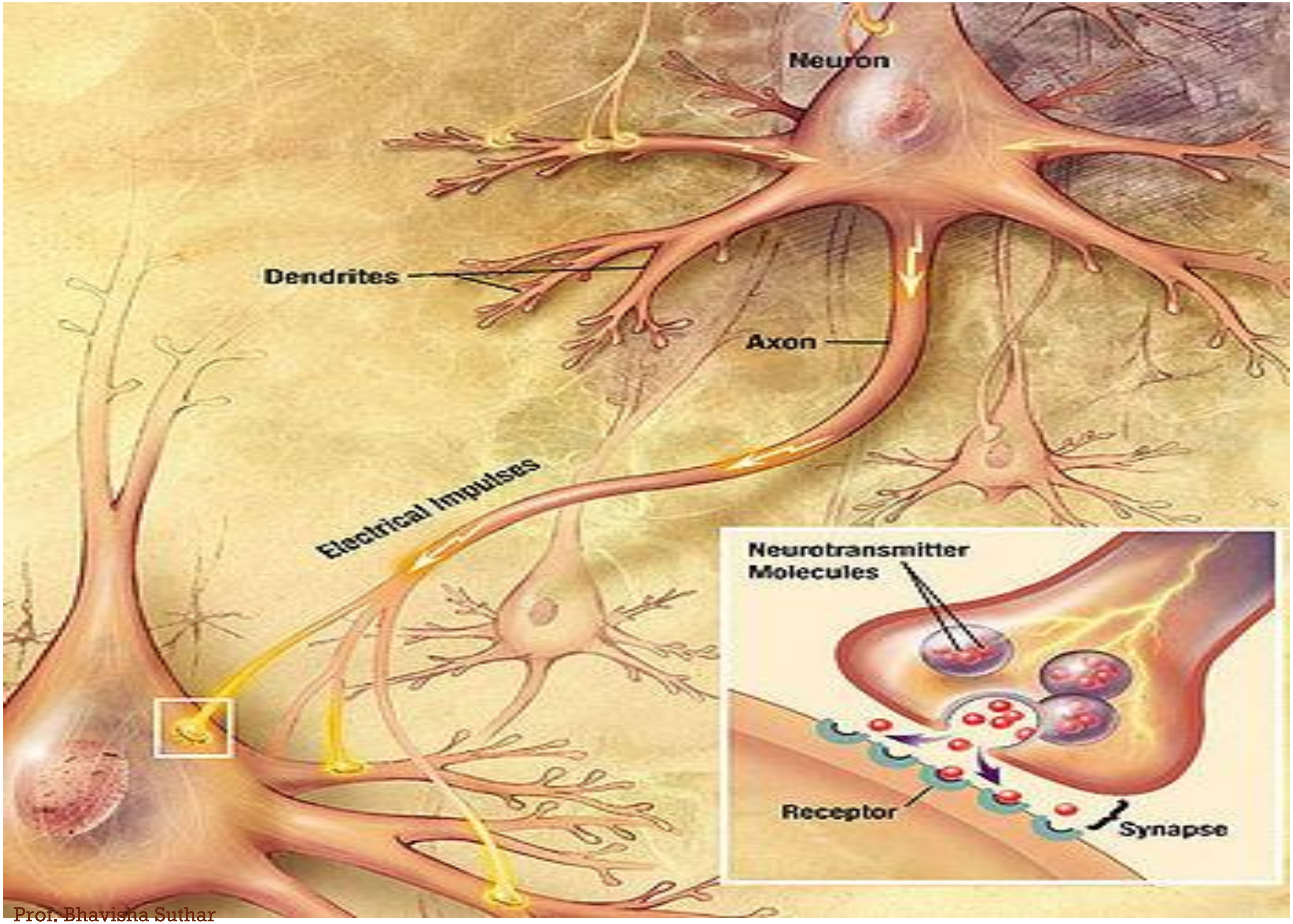
- Naturally, this module will be primarily concerned with how the neural network in the middle works.

BIOLOGICAL NEURON

- How it works?



BIOLOGICAL NEURONS



BASIC COMPONENTS OF BIOLOGICAL NEURONS

- The majority of ***neurons*** encode their activations or outputs as a series of brief electrical pulses (i.e. spikes or action potentials).

- The neuron's ***cell body (soma)*** processes the incoming activations and converts them into output activations.

- The neuron's ***nucleus*** contains the genetic material in the form of DNA. This exists in most types of cells, not just neurons.

BASIC COMPONENTS OF BIOLOGICAL NEURONS

- ***Dendrites*** are fibres which start from the cell body and provide the receptive zones that receive activation from other neurons.

- ***Axons*** are fibres acting as transmission lines that send activation to other neurons.

- The junctions that allow signal transmission between the axons and dendrites are called ***synapses.*** The process of transmission is by diffusion of chemicals called ***neurotransmitters*** across the synaptic cleft.
- At the other end of axon, there exits an inhibiting unit called ***synapse.*** This unit controls flow of neuronal current from originating neuron to receiving dendrites of neighborhood neurons.
- ***Synapses have processing value or weight.***

BASIC COMPONENTS OF BIOLOGICAL NEURONS

- **Communication Between Synapses**

- Once an electrical impulse has reached the end of an axon, the information must be transmitted across the synaptic gap to the dendrites of the adjoining neuron. In some cases, the electrical signal can almost instantaneously bridge the gap between the neurons and continue along its path.

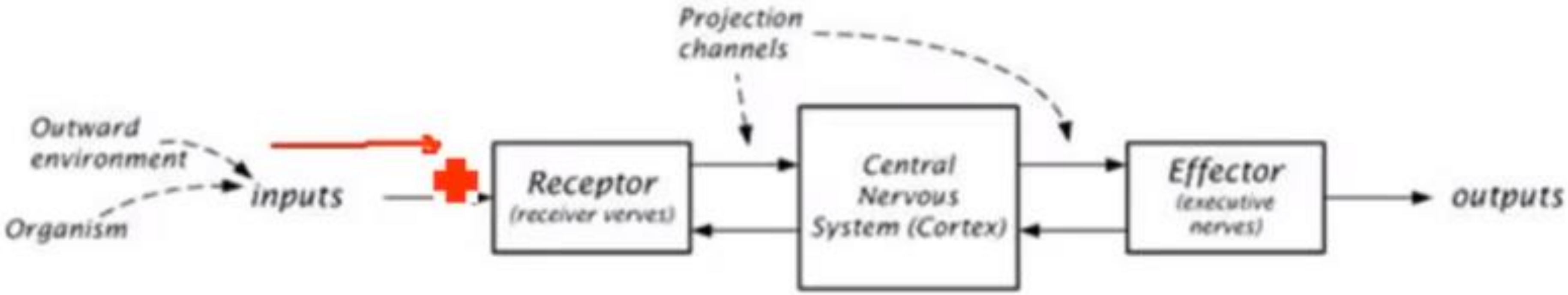
- In other cases, neurotransmitters are needed to send the information from one neuron to the next. Neurotransmitters are chemical messengers that are released from the axon terminals to cross the synaptic gap and reach the receptor sites of other neurons. In a process known as reuptake, these neurotransmitters attach to the receptor site and are reabsorbed by the neuron to be reused.

BASIC COMPONENTS OF BIOLOGICAL NEURONS

- **Neurotransmitters:**

  Neurotransmitters are an essential part of our everyday functioning. While it is not known exactly how many neurotransmitters exist, scientists have identified more than 100 of these chemical messengers.

BNN



Projection
channels
Outward
environment
Organism
inputs
Receptor
(receiver verves)
Central
Nervous
System (Cortex)
Effector
(executive
nerves)
outputs

SOME CHARACTERISTICS OF BIOLOGICAL NEURON NETWORK

- Massive connectivity and parallel

  - each neuron can be connected with about more than 5,000 other neurons

- Robust and fault tolerant

  - tolerance against damage to individual neurons

  -tolerance to loss of neuron is a high priority since a graceful degradation of performance is very important to the survival of the organism

- Capability to adapt to surroundings

  - with change in our surroundings/ work conditions, our brain helps us gradually adapt to the new environment

SOME CHARACTERISTICS OF BIOLOGICAL NEURON NETWORK

- Ability to learn and generalize from known examples

  - e.g: meet a friend after a long with new look, human can recognize

- Collective behavior is more important than individual behavior

  - BNN is more complex, meaningless to understand single neuron and its interaction with other neuron.

PROPERTIES OF BIOLOGICAL NEURON

- Signals are received by the processing elements, This element sums the weighted inputs.
- The weight at the receiving end has the capability to modify the incoming signal.
- The neuron fires(transmit output), when sufficient input is obtained.
- The output produced from one neuron may be transmitted to other neuron.
- Processing of information is found to be local.

PROPERTIES OF BIOLOGICAL NEURON

- The weights can be modified by experience.
- Neuron transmitters for the synapses may be excitatory or inhibitory.
- Both Artificial and Biological neurons have an inbuilt fault tolerance.

→After multiplication all values are summed with each other. This value is known as **net value.**

→After passing the net value in a threshold function we can get the output. These operations is formed inside cell body.

NEURAL NETWORK

- Task involving intelligence/ Pattern recognition

  - difficult to automate
  - easily done by animals (by sense)
  - computation/ simulation of process (to extend physical limitations)
  - necessity the study and simulation of neural network

- NN

  - part of Nervous sytem (interconnected neurons)

ARTIFICIAL NEURAL NETWORK

- Borrowed from the analogy of biological neural networks.

- Also known as neural nets, Artificial Neural System, Parallel distributed processing systems, connectionist systems.

- For computing system

  - directed graph (nodes/vertices & edges/links/arcs)

ARTIFICIAL NEURAL NETWORK

- In ANN

- each node performs some simple computations.

- each arc conveys a signal from one node to another.

- labelled by **weight** (indicating the extent to which a signal is amplified or diminished by a connection)

- Every graph not be NN

ARTIFICIAL NEURAL NETWORK-WEIGHT

- Example:

"AND" of two binary inputs, implemented in hardware using AND gate.

Inputs:

X1 {0,1} and X2 {0,1}

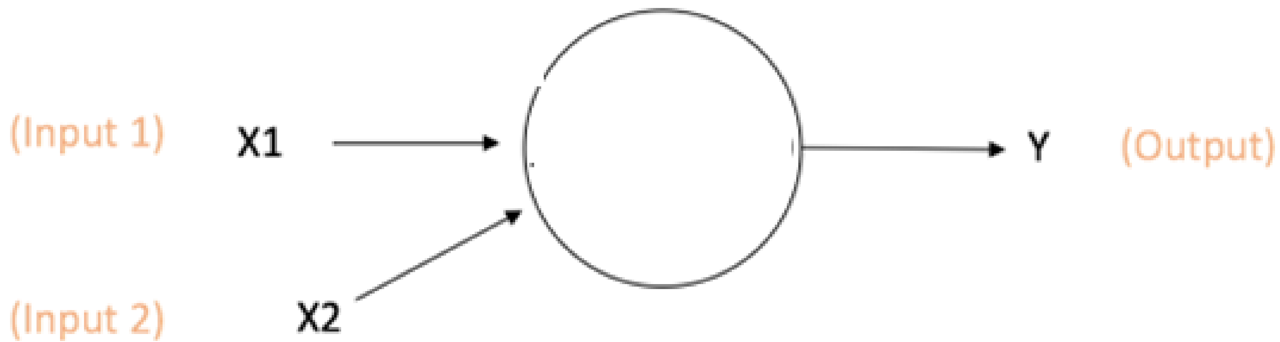
Output:

= 1 ; if X1=X2=1

= 0 ; otherwise

CONT…

- Not a neural network

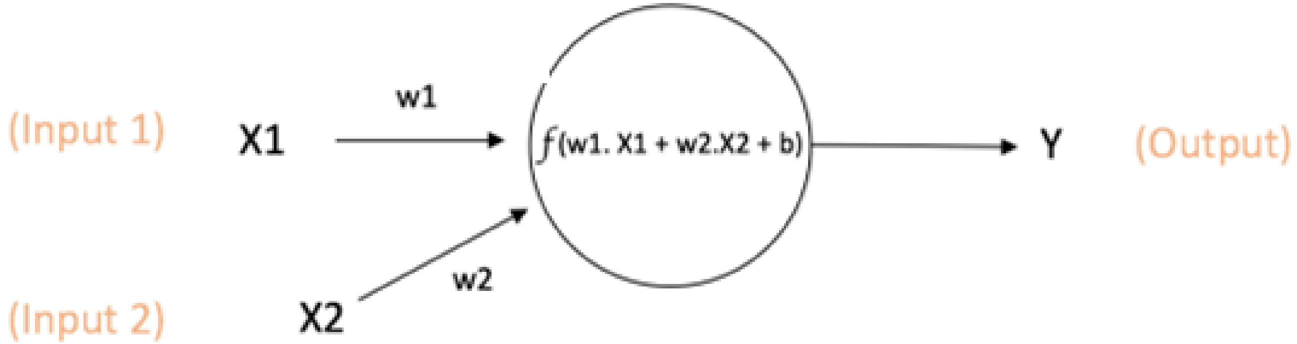


CONT…

- Neural network

  - weights are initially random

  - based on learning algorithm the value of weight will be change.



WEIGHT IN NN

- In computer science, synaptic weight refers to the strength or amplitude of a connection between two nodes, corresponding in biology to the amount of influence the firing of one neuron on another.

WHAT DOES WEIGHT MEAN IN TERMS OF NEURAL NETWORKS?

- What is a Perceptron ?

  - A Perceptron is a type of artificial neuron which takes in several binary inputs x1, x2, … , xn and produces a single binary output.

- To compute output (O):

a) Sum up the inputs. O = x1 + x2 + x3

b) Apply a linear mathematical operation to the inputs. O = (x1 * x2) + x3

WHAT DOES WEIGHT MEAN IN TERMS OF NEURAL NETWORKS?

- **In Case b)**

  - If x1 and x2 both are **0,** then output O=x3.

  - If x1 or x2(any one) is **0,** then output O=x3. (loss info.)

- **In case a)**

  - Inputs are not relatively connected.

  - so correct

WHAT DOES WEIGHT MEAN IN TERMS OF NEURAL NETWORKS?

- Example:

Que: Make a perceptron for predicting whether it will rain today or not.

- Binary o/p as **O**, take values **0** or **1** and inputs **x1** and **x2.**

Let, **x1** be **1** if the weather is humid today,

0 if the opposite.

Let, **x2** be **1** if you are wearing a white cloth today,

0 if not.

here that **wearing a white cloth** has almost **no correlation** with the possibility of rainfall.

So, a possible output function can be: **O = x1 + <u>0.1</u> * x2**

WHAT DOES WEIGHT MEAN IN TERMS OF NEURAL NETWORKS?

- **if x2= 0 ,** 0.1*x2 is still 0,
- **if x2=1**, 0.1*x2 will be 0.1, not 1.
- **It basically brings down the importance of the input x2 from 1 to 0.1, and hence, it is called the 'weight' of the input x2.**

- Consider x2 to be 1 for now.
- O will still be (0 + 0.1) = 0.1 or (1 + 0.1) = 1.1,

  i.e. not a binary value. In order to make it one, what we can do is use a '**threshold**' for the total value of O.
- So, O is 1 if (x1 + 0.1 * x2) > 1

  O is 0 if (x1 + 0.1 * x2) < 1. We have thus solved our problem.

WHAT DOES WEIGHT MEAN IN TERMS OF NEURAL NETWORKS?

Simplification:

w = [w1 w2 w3 …. wn],

x = [x1 x2 x3 …. xn]

B(bias) = -(threshold)

Where,

O = 1 ; {x * (weight of x) + y * (weight of y) + b} > 0,

O = 0 ; otherwise

COMPARISON BETWEEN ANN AND BNN

BNN	ANN
Slower in speed.	Faster in speed.
Adaptable is possible because new information is added without destroying old information.	Adaptable is not possible because new information destroys the old information.
Fault-tolerant: if irrespective of faults in network connections, then also information is still preserved.	No fault-tolerant: If information corrupted in the memory, it can not be restored back.
Memory and processing elements are collocated.	Memory and processing are separate.
There is self organization during the learning.	This is software dependent.
Parallel and asynchronous.	Sequential synchronous.

WHAT DOES BIAS MEAN IN TERMS OF NEURAL NETWORKS?

- A bias unit is an extra neuron added to each pre-output layer that stores the value of 1.

  - are not connected to any previous layer

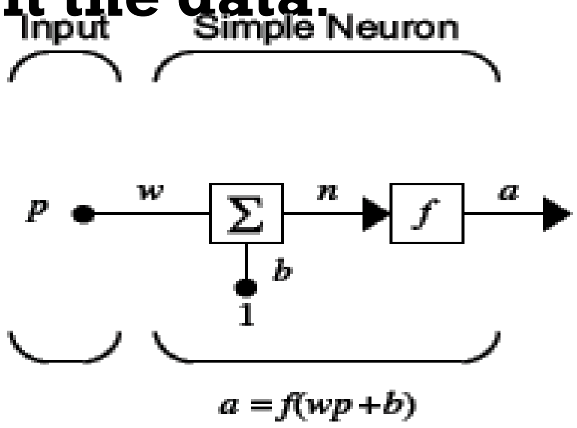
  - It is just appended to the start/end of the input and each hidden layer,

  - is not influenced by the values in the previous layer

  - neurons don't have any incoming connections.

WHAT DOES BIAS MEAN IN TERMS OF NEURAL NETWORKS?

- bias units still have outgoing connections and they can contribute to the output of the ANN.
- It improves the performance of the neural network.
- **Bias nodes are added to increase the flexibility of the model to fit the data.**



ACTIVATION FUNCTIONS

- In ANN defines the output of neuron given a set of inputs.
- Biologically inspired by activity in our brains, where different neurons fire or activated by different stimuli.
- A standard computer chip circuit can be seen as a digital network of activation function that can be on or off(1 or 0), depending on input. This is same behavior of the linear perception in NN.

ACTIVATION FUNCTIONS

- The activation function is used to calculate output of a neuron, net is the linear combiner output. To generate final output **Y.**
- It'**s** basically decide whether a neuron should be activated or not.
- Whether the information that the neuron is receiving is relevant for the given information or should it be ignored.

$$Y = \text{Activation}(\Sigma(weight * input) + bias)$$

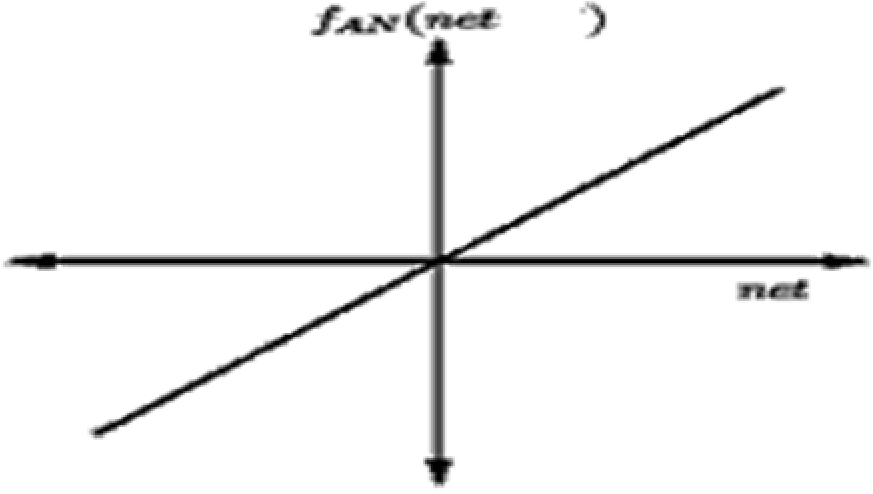
ACTIVATION FUNCTION

- Types:

  - linear (straight shape)

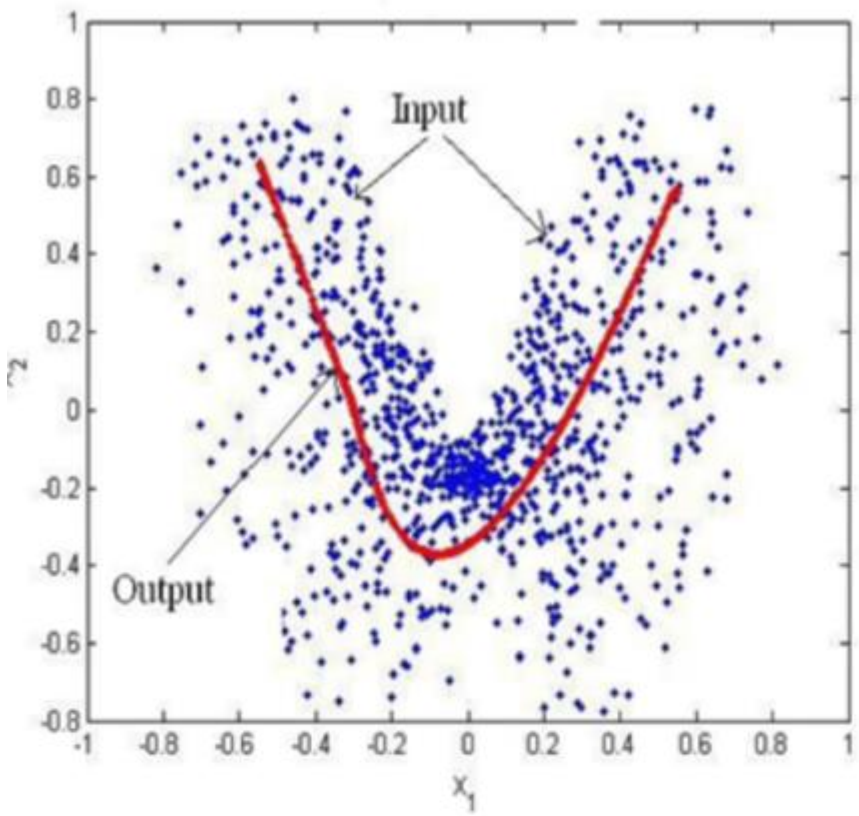
  - non-linear (in curve)

LINEAR ACTIVATION FUNCTION



Linear function

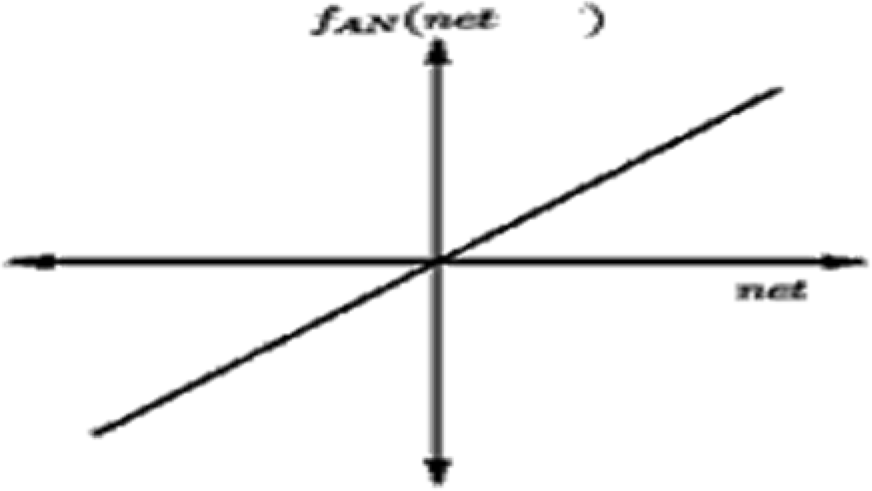
NON-LINEAR ACTIVATION FUNCTION



ACTIVATION FUNCTIONS

- Linear function:

Y= net



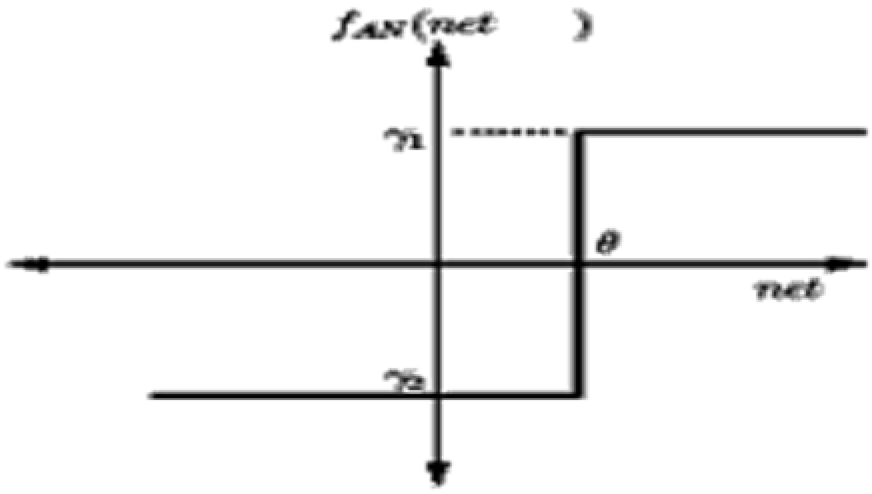
Linear function

ACTIVATION FUNCTIONS

- Binary step function:

Y= f(net) = +1, if net>0

= -0, if net<=0

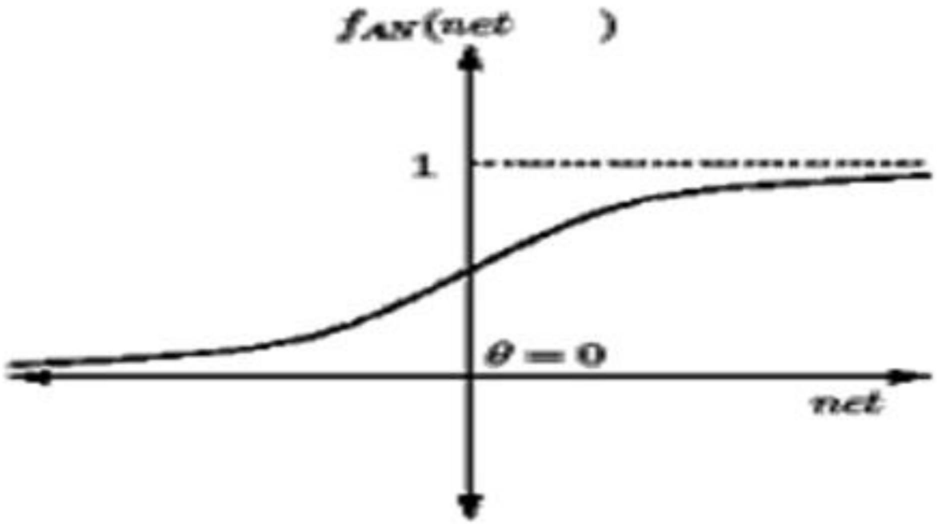


Step function

ACTIVATION FUNCTIONS

- Sigmoid function:

Y= 1/(1+e ^-net)

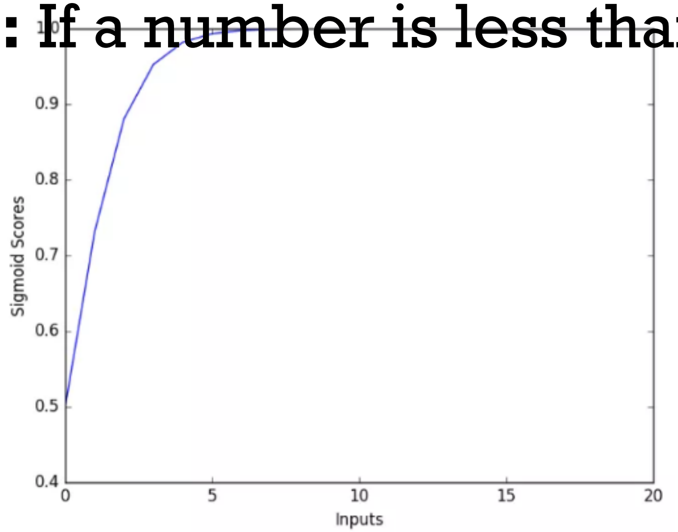


Sigmoid function

ACTIVATION FUNCTION-CONT...

Non-Negative: If a number is greater than or equal to zero.

Negative: If a number is less than or equal to Zero.



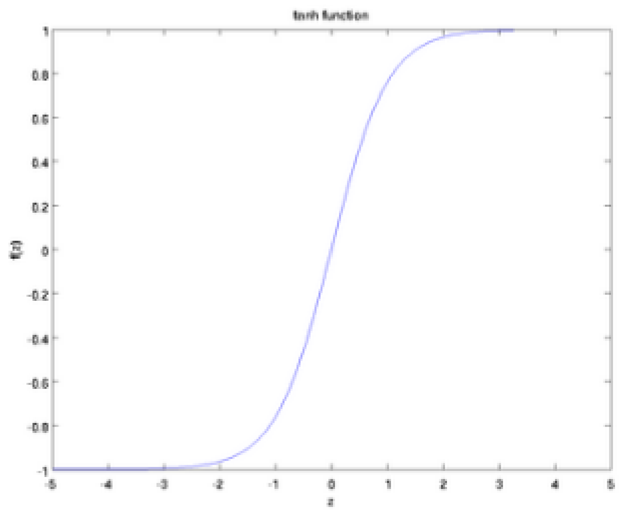
Sigmoid graph

The Sigmoid function used for **binary classification** in logistic regression model.

ACTIVATION FUNCTIONS

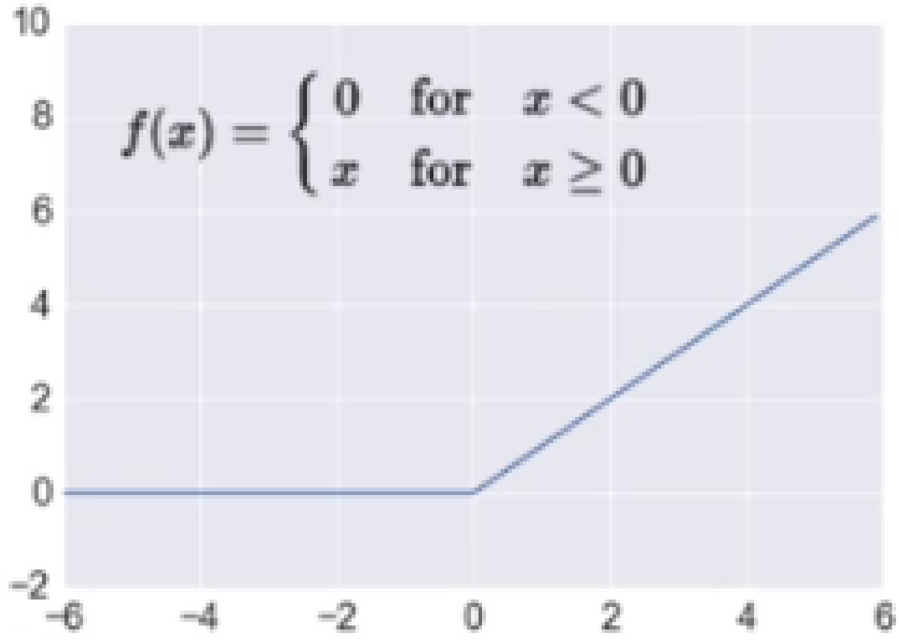
- **Tanh function:**

$$f(x) = \tanh(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})}$$



ACTIVATION FUNCTIONS

- **ReLU function:**



ACTIVATION FUNCTIONS

- ReLU is best for hidden layers
- for O/P layer use softmax for classification or linear for regression.

THRESHOLD VALUE

$$net = \sum x_i w_i + b; \quad i=1 \text{ to } n$$

$$y = f\,(net)$$

- **Threshold Value:** The net value is known as activation value. The threshold value is sometimes used to qualify the output of a neuron in the output layer.

→ The net value or activation value is compared with threshold and neuron fires if the threshold value is exceeded.

WHY USE NEURAL NETWORK

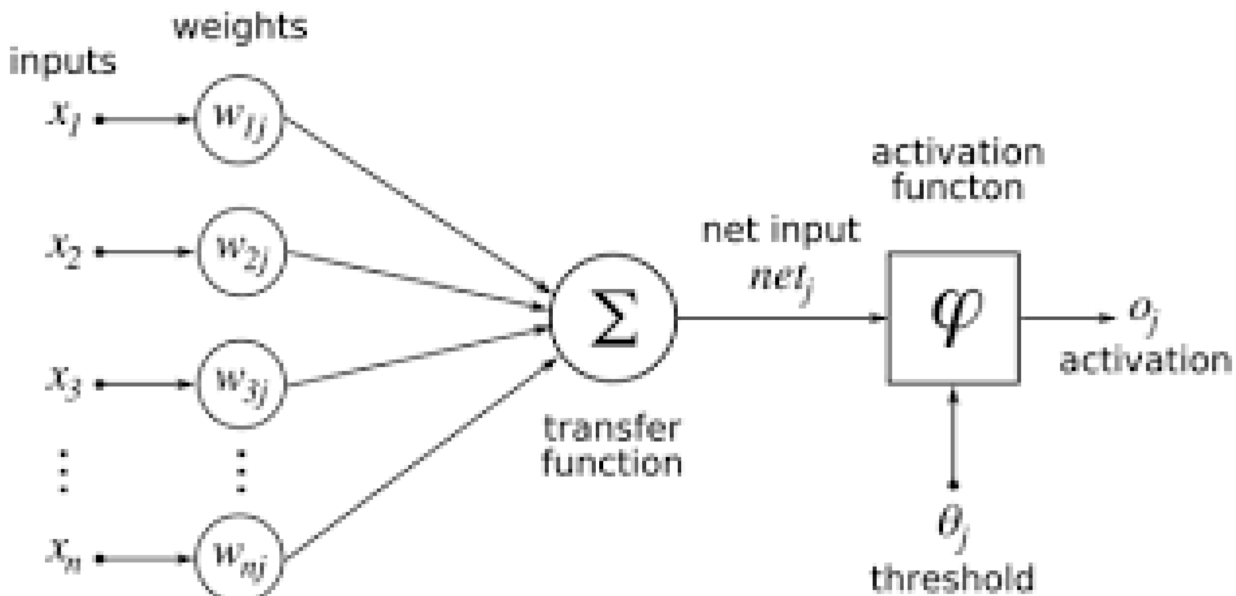
- It can be used to extract patterns and detect trends that are too complex to be notices by wither humans or other computer techniques.

- A trained neural network can be thought of as an "expert" in the category of information it has been given to analyze. This expert can then be used to provide projections given new situations of interest and answer "what if" questions.

APPLICATION OF NEURAL NETWORK

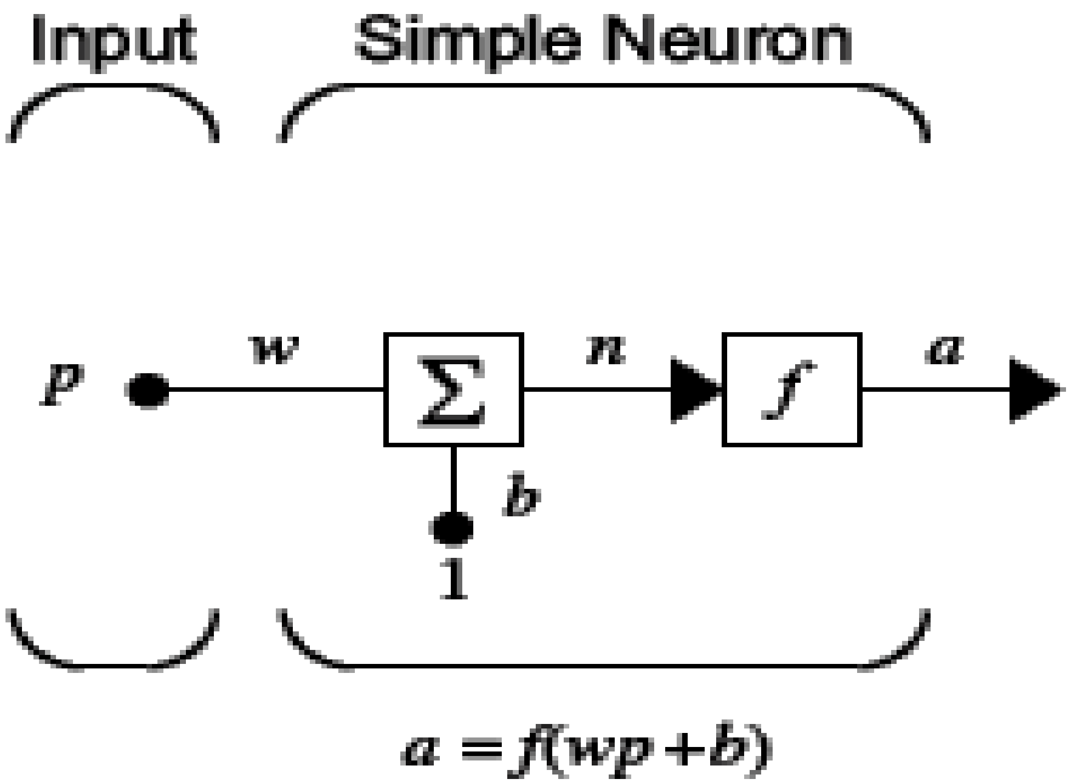
- Process modeling and control
- Machine Diagnostics
- Portfolio Management
- Target Recognition
- Medical Diagnosis
- Credit Rating
- Targeted Marketing
- Voice recognition
- Financial Forecasting
- Intelligent searching
- Fraud detection
- Essay scoring

ARTIFICIAL NEURAL NET (ANN)

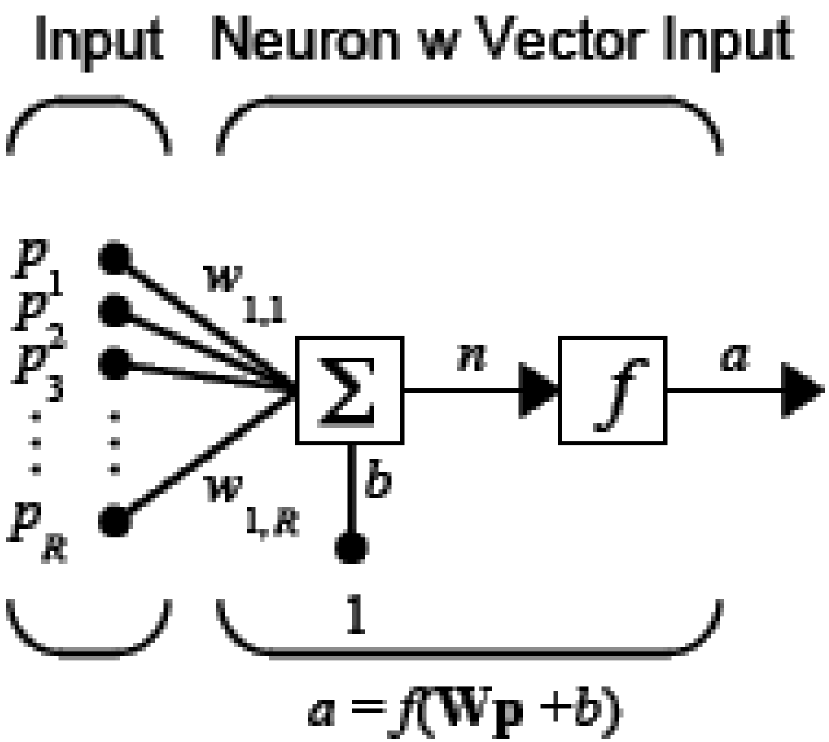


Dendrons Cell body Axon Synapse

ARTIFICIAL NEURAL NET (ANN)



NEURON WITH VECTOR INPUT

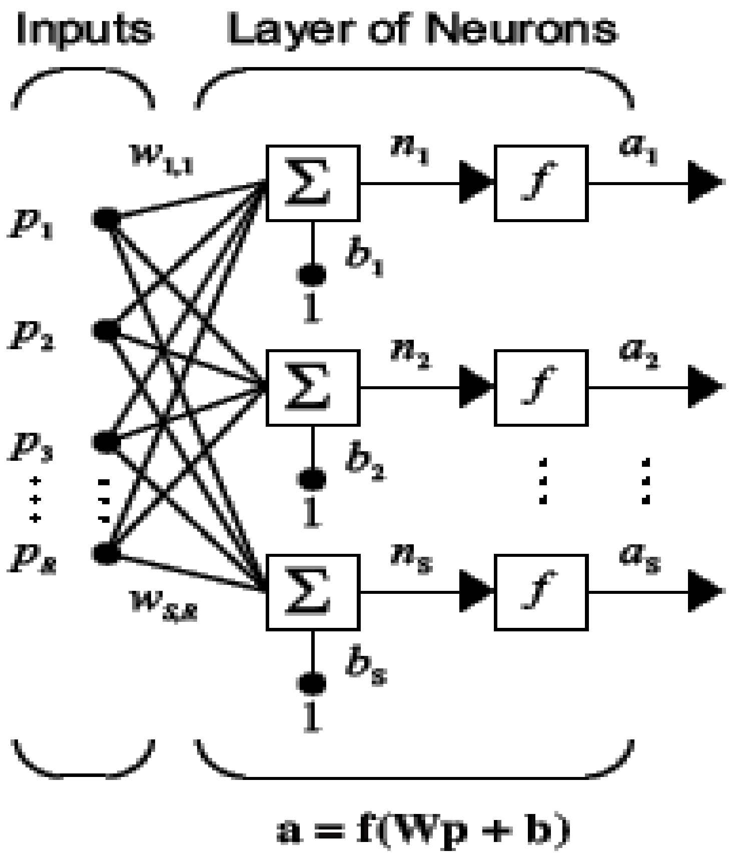


Where

R = number of elements in input vector

$$n = w_{1,1}p_1 + w_{1,2}p_2 + \ldots + w_{1,R}p_R + b$$

A LAYER OF NEURONS

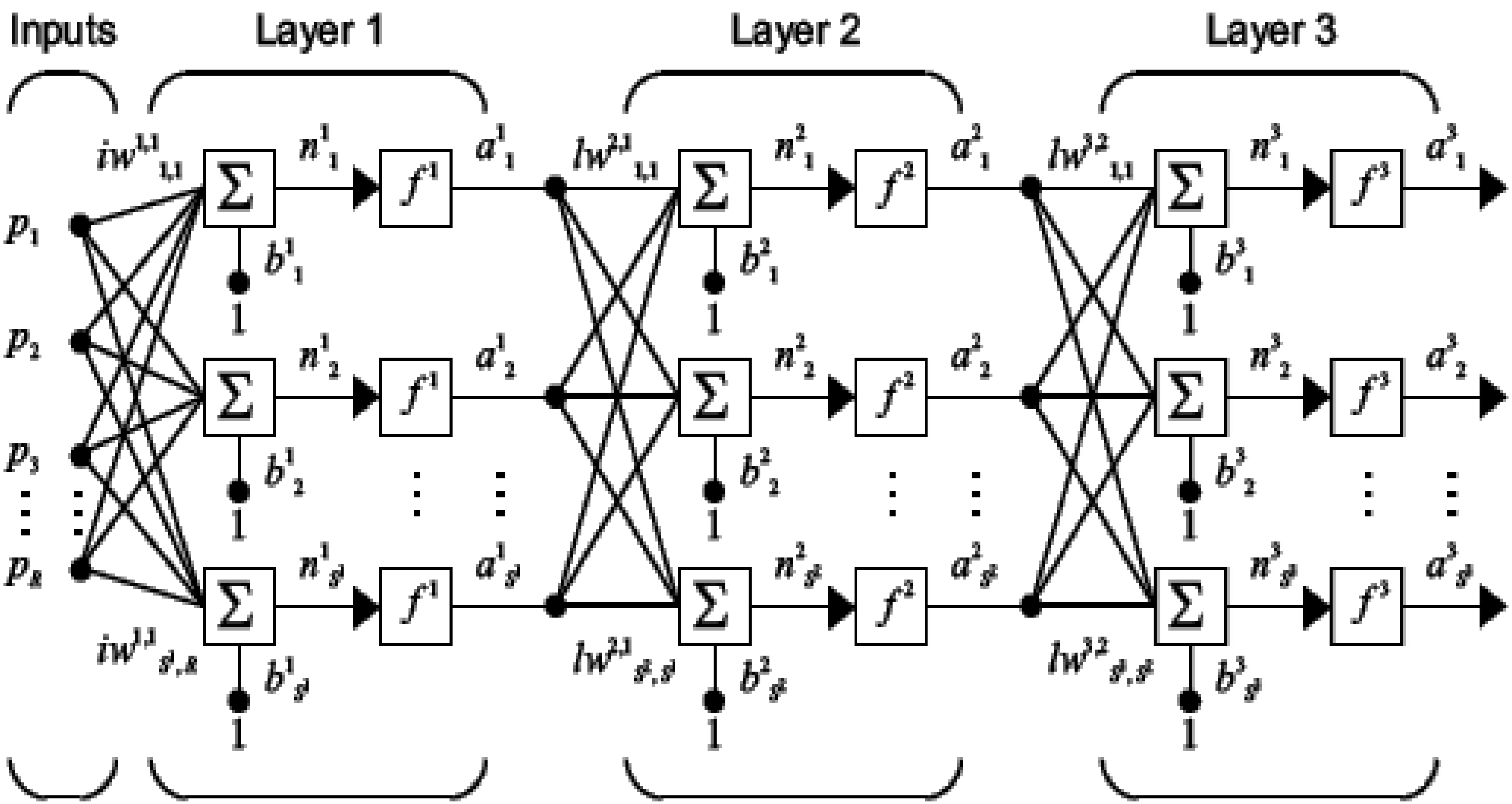


Where

R = number of elements in input vector

S = number of neurons in layer

MULTIPLE LAYERS OF NEURONS



$$a^1 = f^1(IW^{1,1}p + b^1) \qquad a^2 = f^2(LW^{2,1}a^1 + b^2) \qquad a^3 = f^3(LW^{3,2}a^2 + b^3)$$

$$a^3 = f^3(LW^{3,2}f^2(LW^{2,1}f^1(IW^{1,1}p + b^1) + b^2) + b^3)$$

TYPES OF NEURAL NETWORK

Parameter	Types	Description
Based on connection pattern	FeedForward, Recurrent	• **Feedforward** - In which graphs have no loops. • **Recurrent** - Loops occur because of feedback.
Based on the number of hidden layer	Single layer, Multi-Layer	• **Single Layer** - Having one hidden layer. E.g. , Single Perceptron • **Multilayer** - Having multiple hidden layers. Multilayer Perceptron
Based on nature of weights	Fixed, Adaptive	• **Fixed** - Weights are fixed a priori and not changed at all. • **Adaptive** - Weights are updated and changed during training.
Based on Memory unit	Static, Dynamic	• **Static** - Memoryless unit. The current output depends on the current input. E.g. , Feedforward network • **Dynamic** - Memory unit - The output depends upon the current input as well as the current output. E.g. , Recurrent Neural Network

CONT…

- **Feedforward Network**

  - Single layer feed forward network

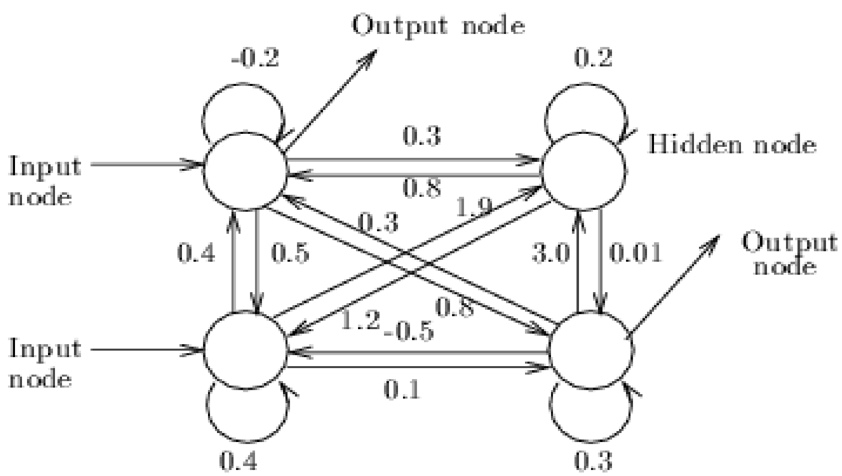
  - Multilayer feed forward network

ANN ARCHITECTURES

- Fully Connected Network
- Layered Network
- Acyclic Network
- Feedforward Network

ANN ARCHITECTURE

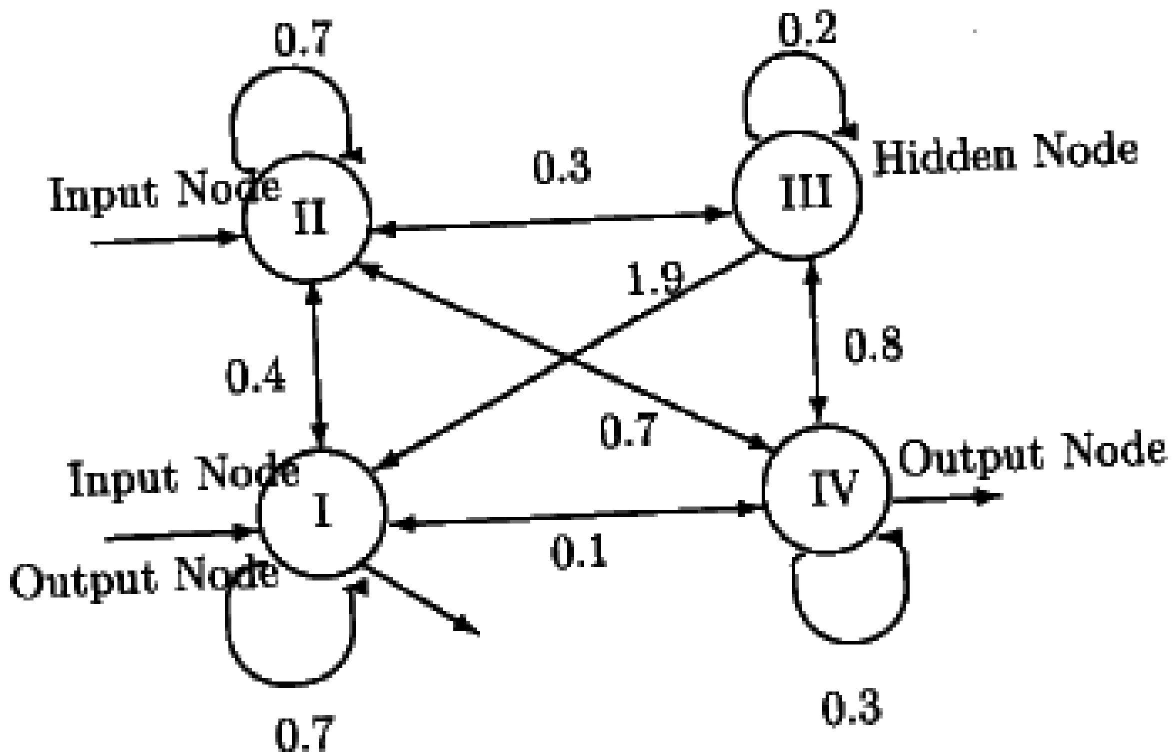
- **(Asymmetric) Fully Connected Networks**
  - Every node is connected to every other node
  - Connection may be excitatory (positive), inhibitory (negative), or irrelevant (≈ 0).
  - Most general
  - Symmetric fully connected nets: weights are symmetric ($w_{ij} = w_{ji}$)



Input nodes: receive input from the environment
Output nodes: send signals to the environment
Hidden nodes: no direct interaction to the environment

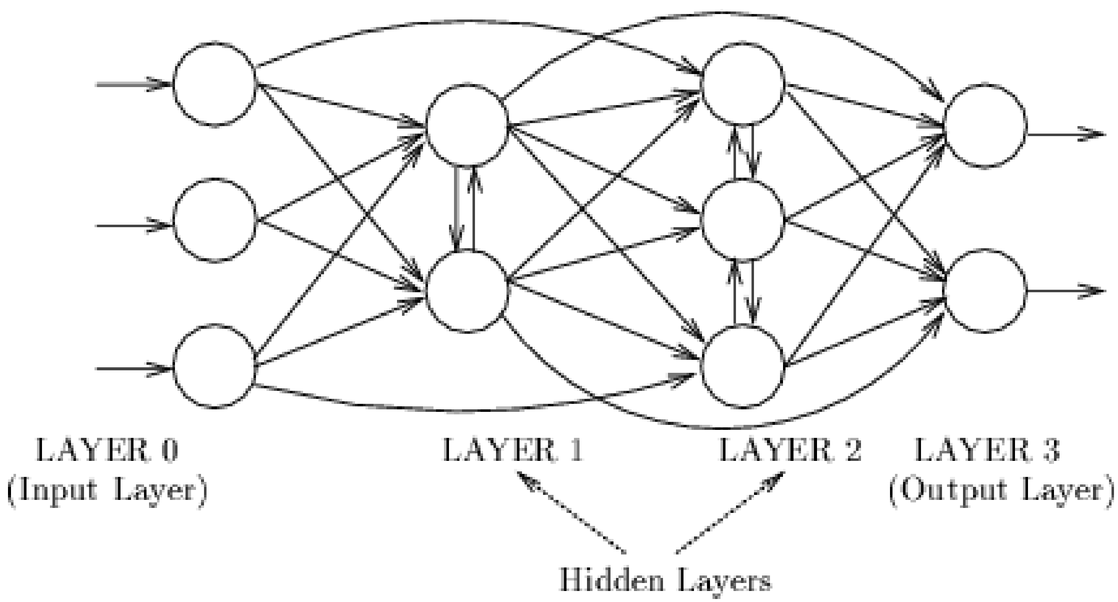
ANN ARCHITECTURE

- Fully Connected Network



ANN ARCHITECTURE

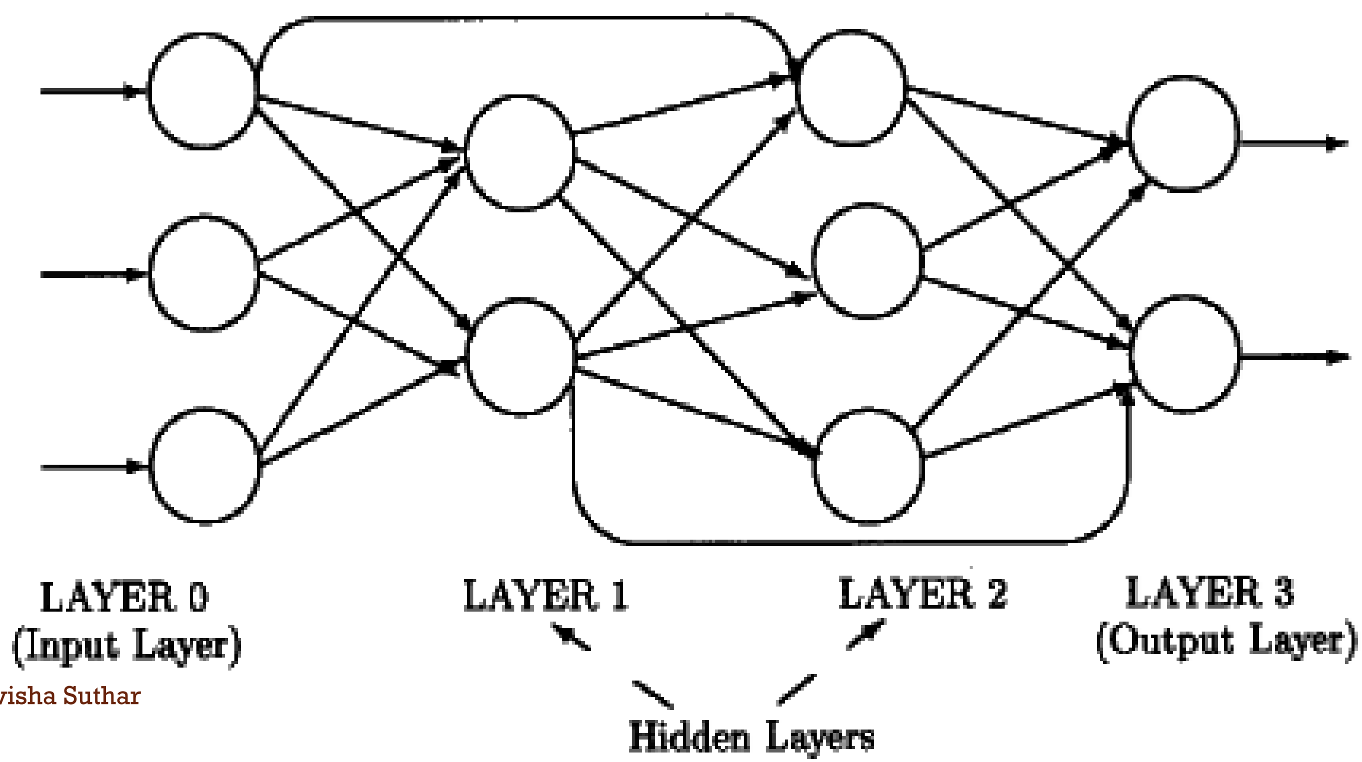
- **Layered Networks**
  - Nodes are partitioned into subsets, called layers.
  - No connections that lead from nodes in layer *j* to those in layer *k* if *j* > *k*.



- Inputs from the environment are applied to nodes in layer 0 (**input layer**).
- Nodes in input layer are place holders with no computation occurring (i.e., their node functions are identity function)

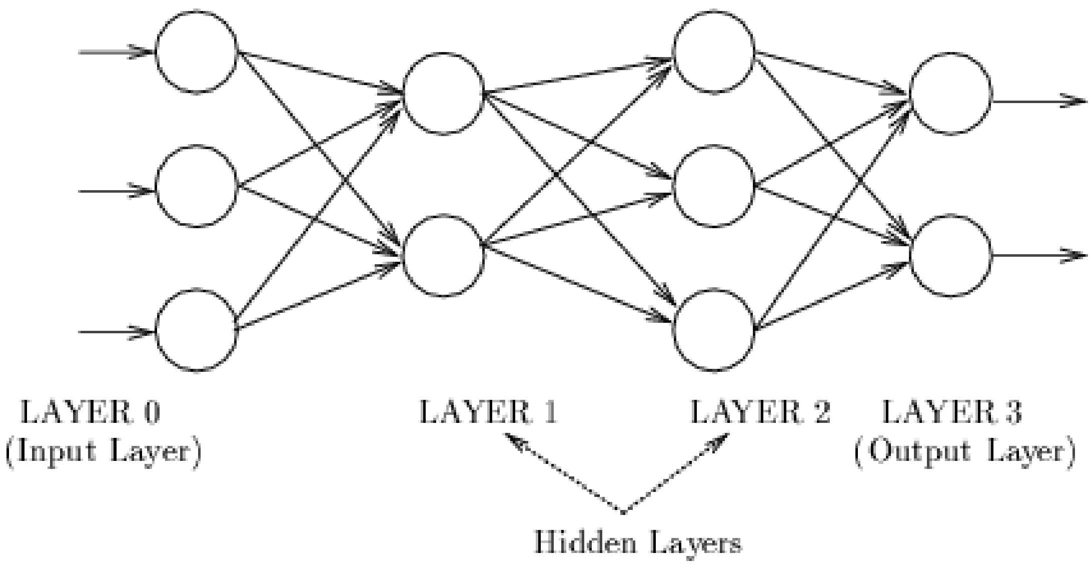
ANN ARCHITECTURE

- ***Acyclic Network***
- The connection may exists between any node in layer but it not allows if two layers are i & j then i=j is not possible.



ANN ARCHITECTURE

- **Feedforward Networks**
  - A connection is allowed from a node in layer *i* only to nodes in layer *i* + 1.
  - Most widely used architecture.

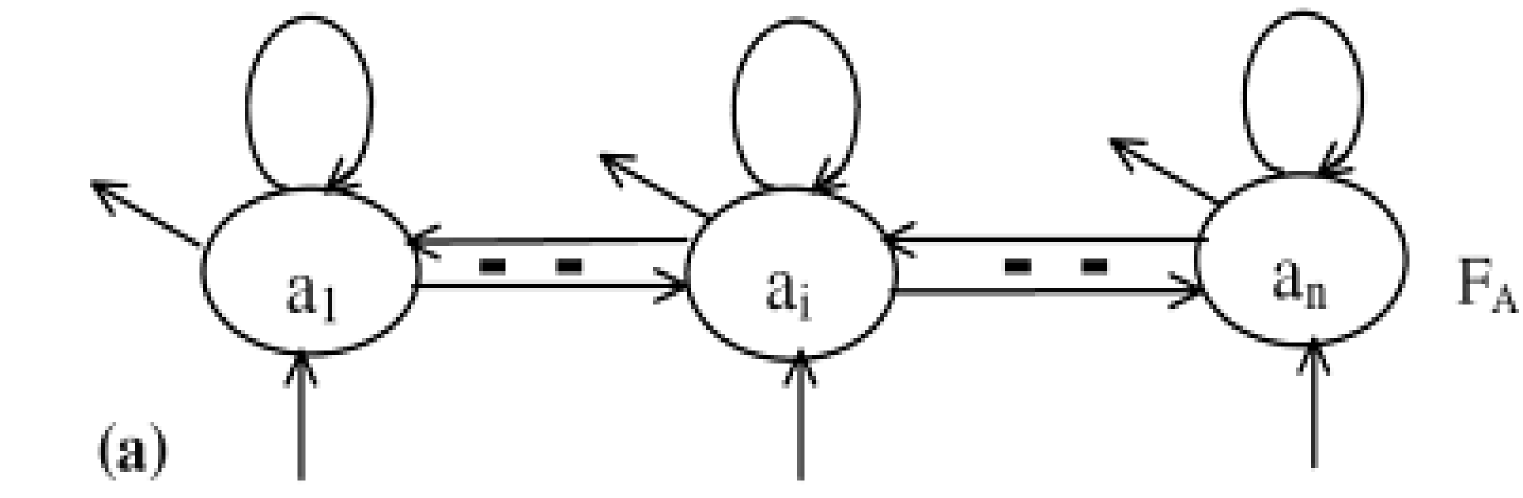


Conceptually, nodes at higher levels successively abstract features from preceding layers

TOPOLOGIES OF NEURAL NETWORK

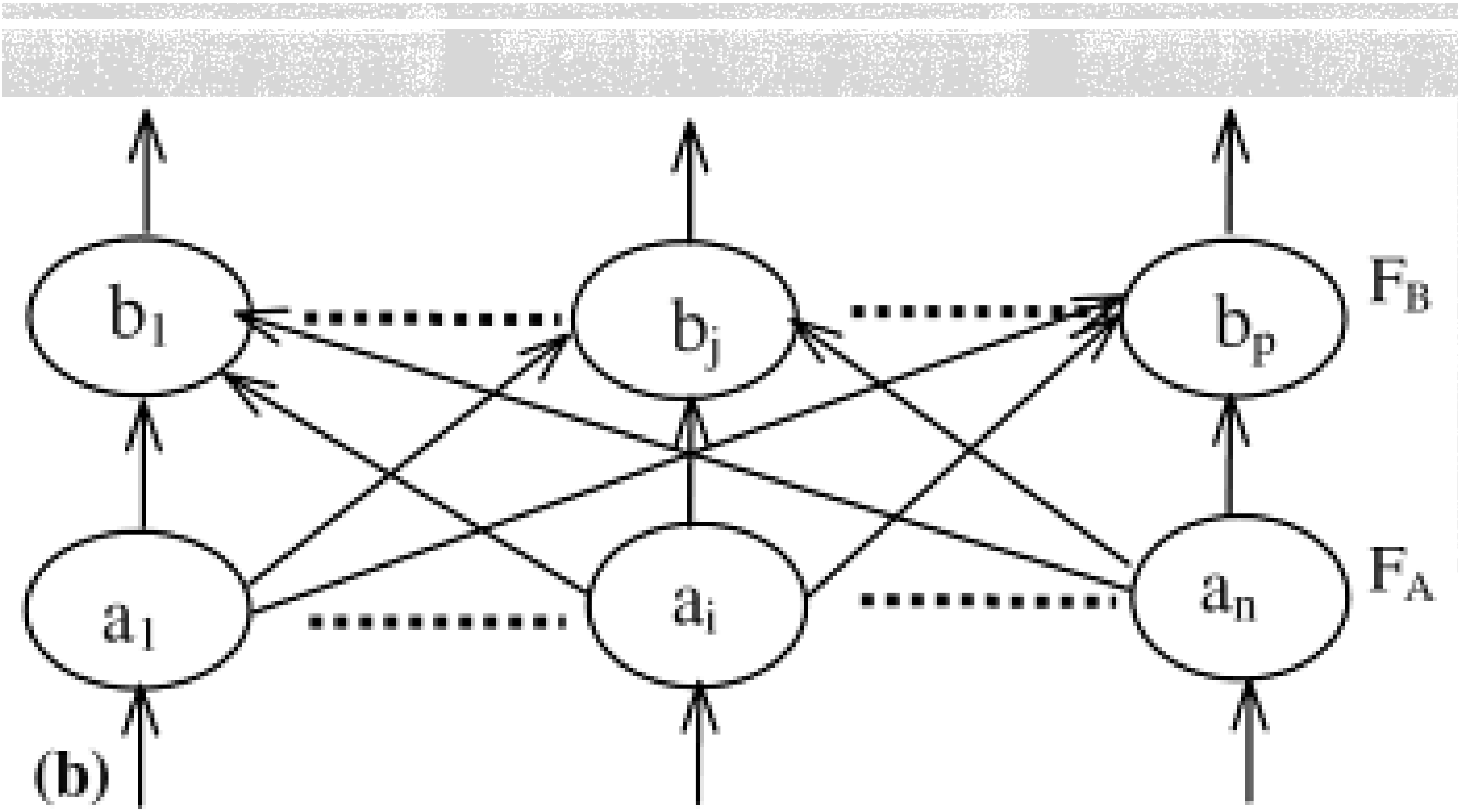
- Depending on nature of problems, the artificial neural net is organized in different structural arrangements.

a) Single layered recurrent net with lateral feedback



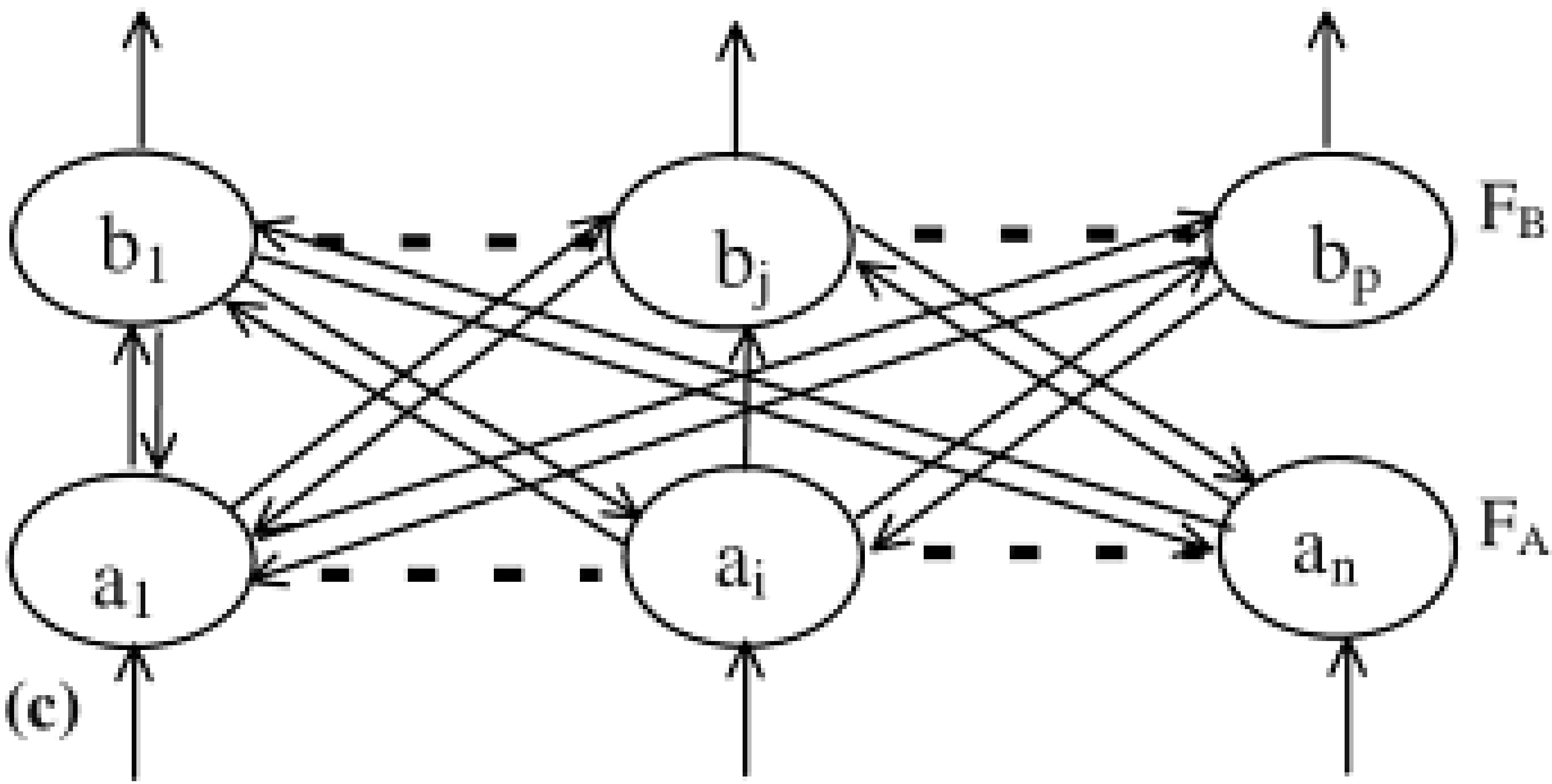
TOPOLOGIES OF NEURAL NETWORK

b) Two layered feed-forward structure



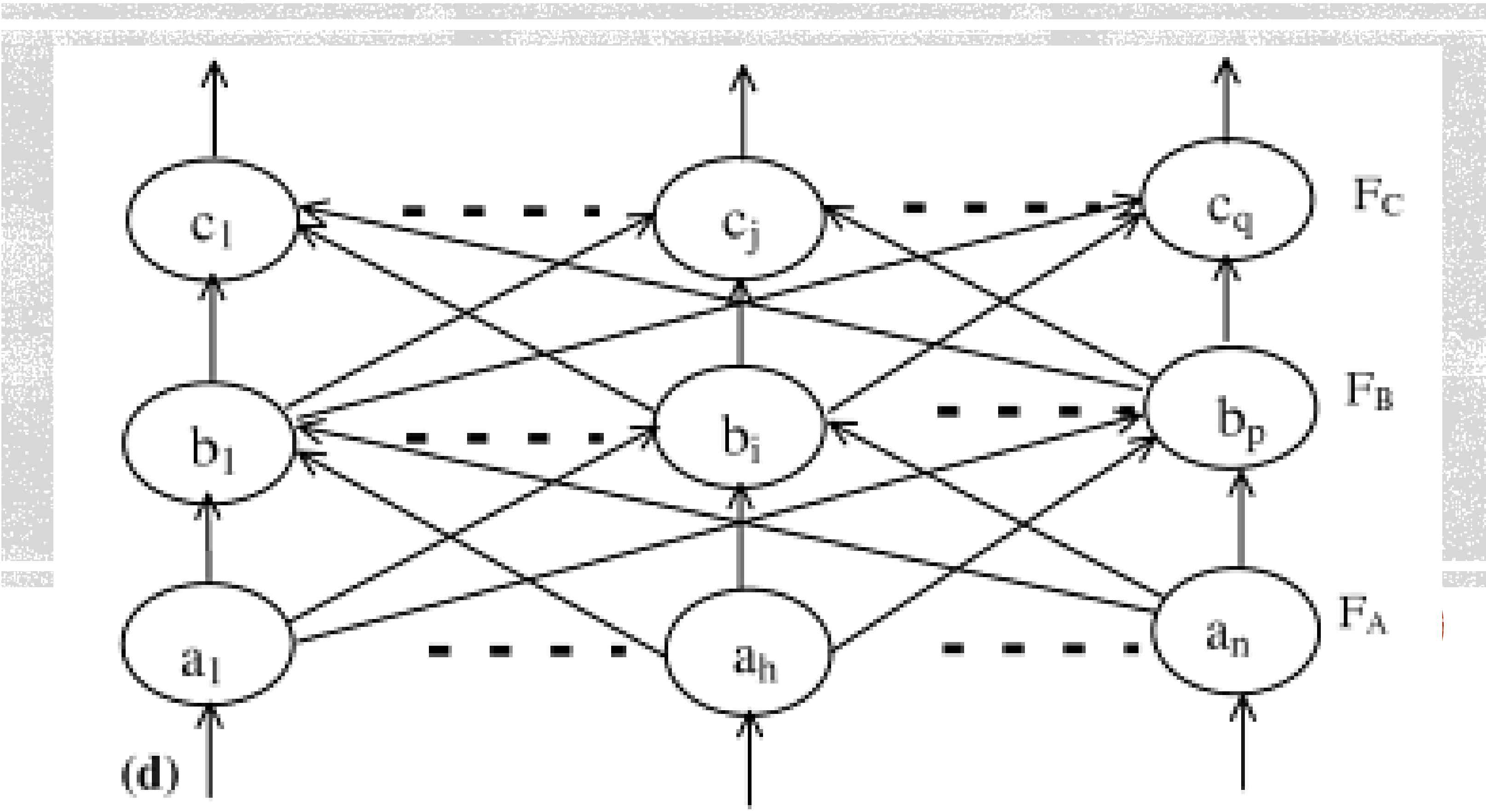
TOPOLOGIES OF NEURAL NETWORK

c) Two layered structure with feedback



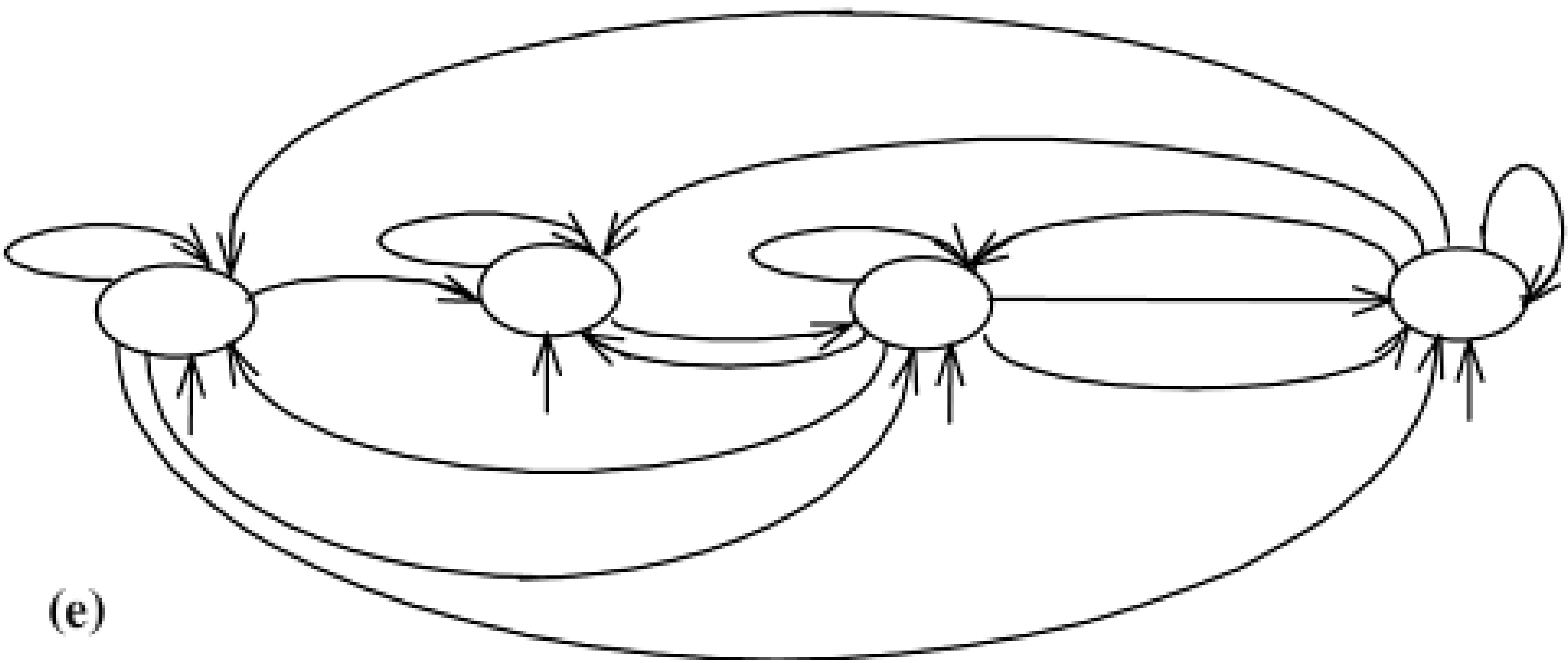
TOPOLOGIES OF NEURAL NETWORK

d) Three layered feed-forward structure



TOPOLOGIES OF NEURAL NETWORK

e) A recurrent structure with self-loops



(e)

APPLICATIONS OF ANN

- Aerospace
- Automotive
- Banking
- Defense
- Electronics
- Financial
- Manufacturing
- Medical
- Robotics
- Securities
- Telecommunications

LEARNING PROCESS

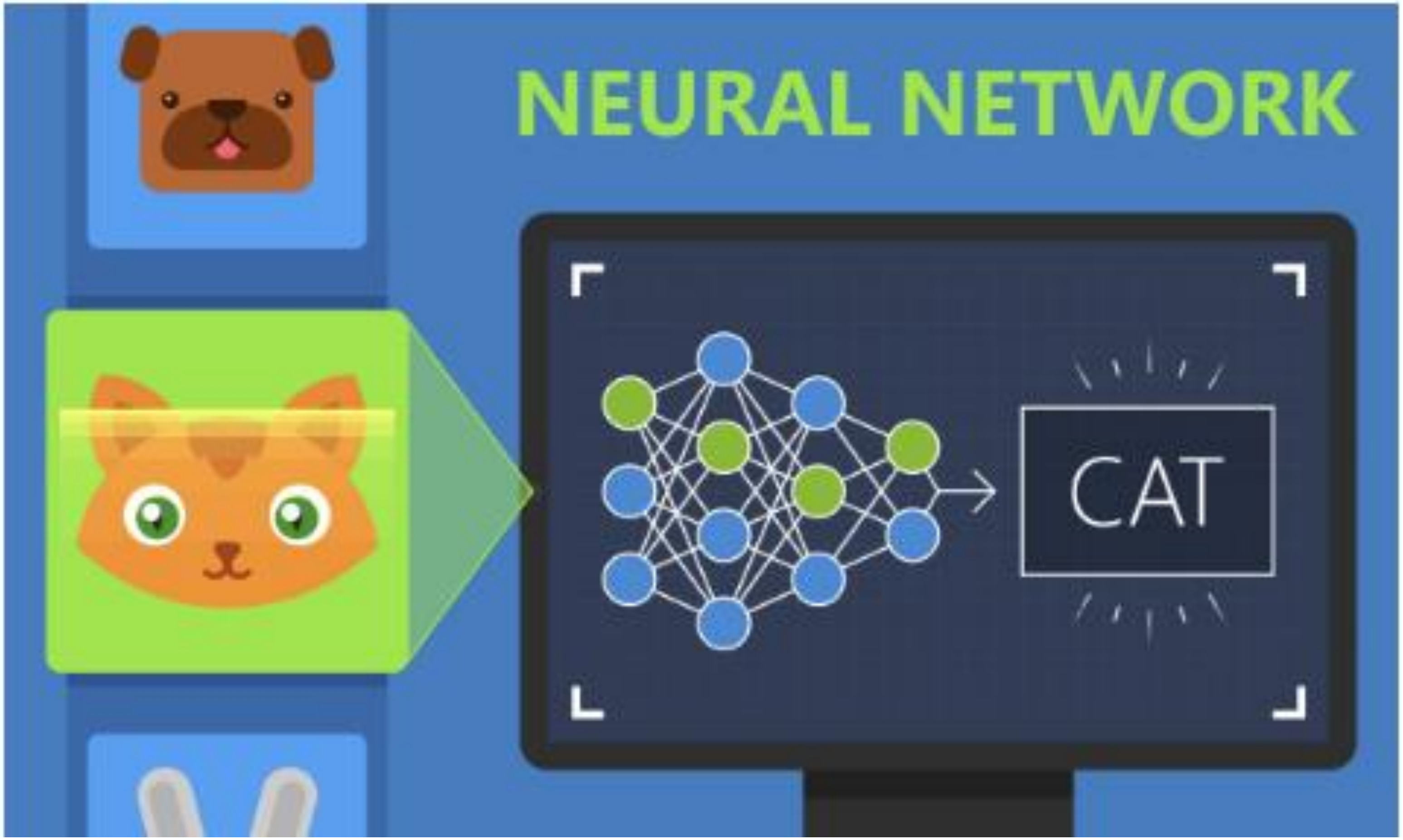
Three types of learning have been used.

1. Supervised learning
2. Unsupervised learning:
3. Reinforcement learning

SUPERVISED LEARNING

- Every input pattern associated with output pattern.
- Output pattern is known as target or desired pattern.
- A teacher is present during learning process.
- A comparison is required between the network's computed output and correct output to determine the error.
- The error can then be used to change network parameters which result improve the performance.
- X(i) is an input vector and d(i) is corresponding target value, the learning rule calculates the updated value of the neuron weights and bias. **Eg. Perceptron, back propagation etc.**

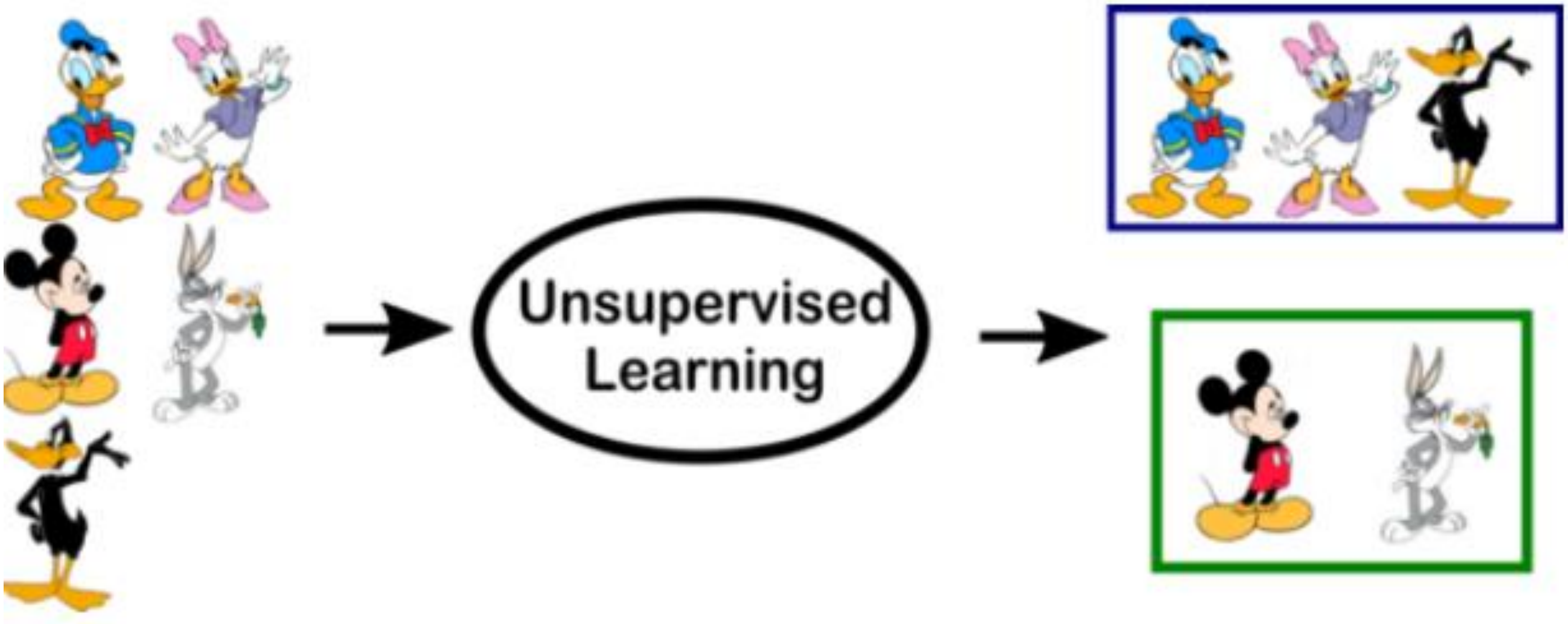
SUPERVISED LEARNING



UNSUPERVISED LEARNING

- In this method target output is not present to the network.
- No teacher to present the desired patterns hence the system learns of its own by discovering and adapting to structural features in the input pattern.
- The weight and bias are adjusted on inputs only.
- To learn cluster input patterns into a finite number of classes.

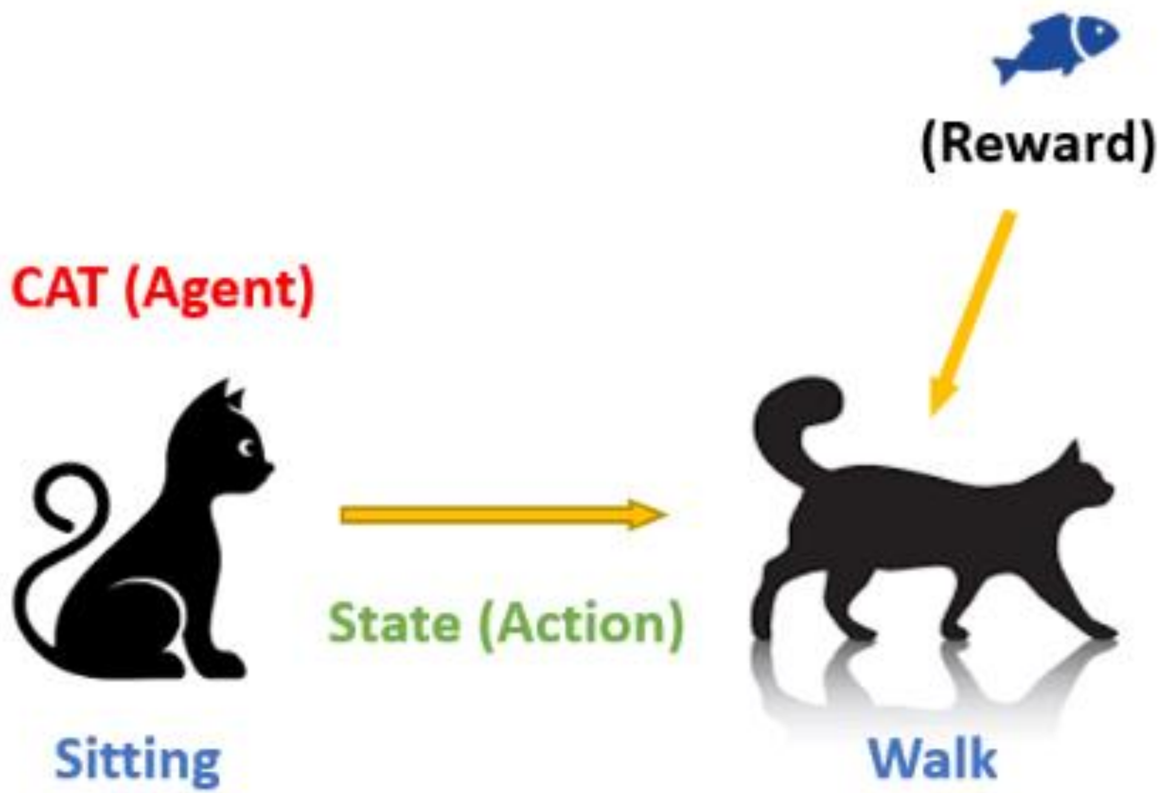
UNSUPERVISED LEARNING



REINFORCEMENT LEARNING

- The value of the output is unknown, but the network provides the feedback whether the output is right or wrong.

REINFORCEMENT LEARNING



LEARNING PROCESS: EXAMPLES

- Supervised learning:
  ex. Image is of cat or dog? Classify it based on features
- Unsupervised learning:

  ex. Different age group people falls in different group
- Reinforcement learning

  ex. Give rewards to cat if it follows some rules

ADVANTAGES OF NN

- A neural network can perform tasks that a linear program can not.
- When an element of the neural network fails, it can continue without any problem by their parallel nature.
- A neural network learns and does not need to be reprogrammed.
- It can be implemented in any application.
- It can be performed without any problem.

LIMITATIONS OF NN

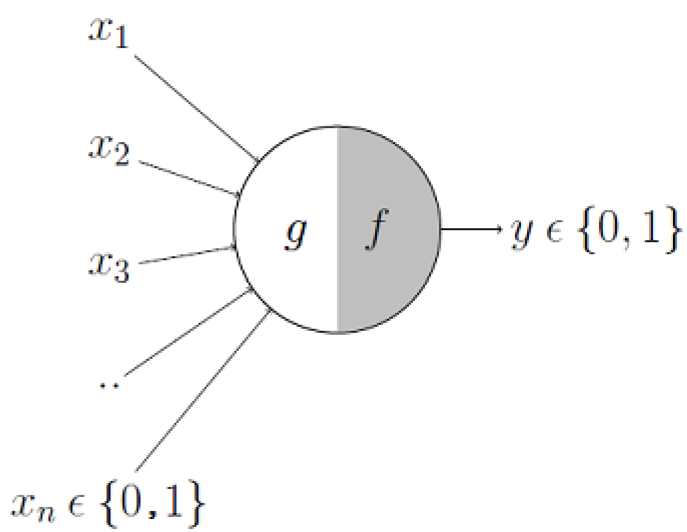
- The neural network needs amount of Data for training to operate.
- Computationally expensive
- Requires high processing time for large neural networks.

PERCEPTRON

- It is very well known that the most fundamental unit of deep neural networks is called an ***artificial neuron/perceptron***.
- McCulloch-Pitts Neuron
- But the very first step towards the *perceptron* we use today was taken in 1943 by McCulloch and Pitts, by mimicking the functionality of a biological neuron.

MCCULLOCH-PITTS NEURON

- The first computational model of a neuron was proposed by Warren MuCulloch (neuroscientist) and Walter Pitts (logician) in 1943.



PERCEPTRON

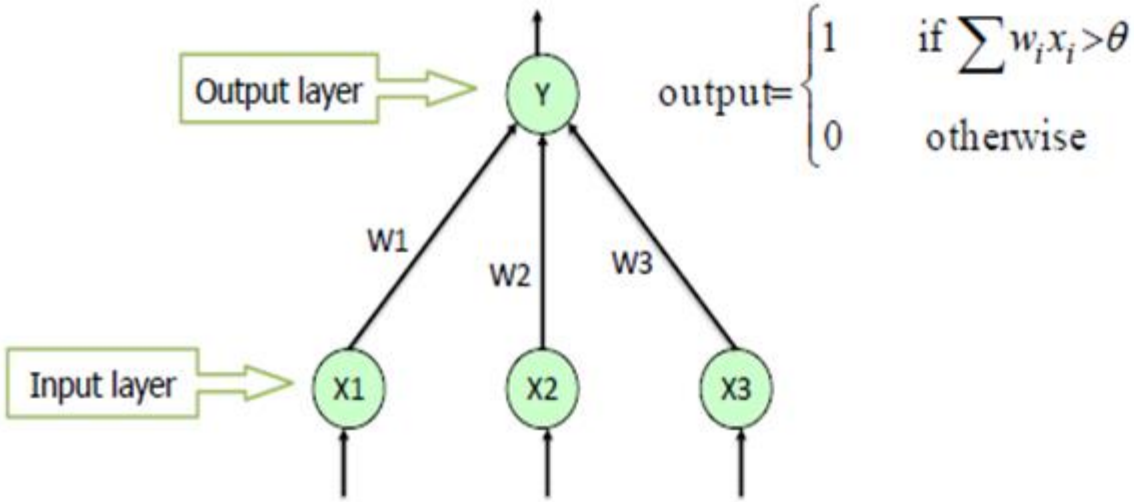
- In machine learning, the **perceptron** is an algorithm for supervised learning of binary classifiers.

$$f(x) = \begin{cases} 1 & \text{if } \mathbf{w} \cdot \mathbf{x} + b > 0 \\ 0 & \text{otherwise} \end{cases}$$

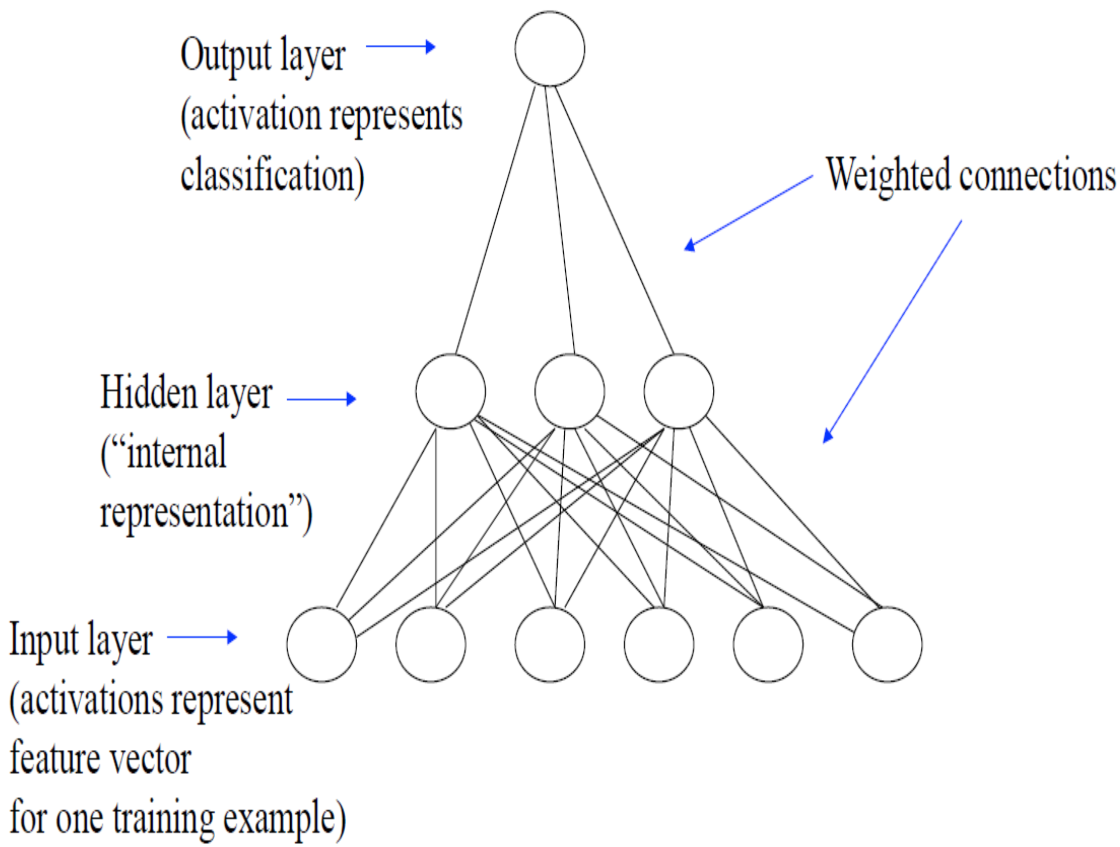
PERCEPTRON NETWORK

- Types:
- Single layer perceptron network
- Multilayer perceptron network

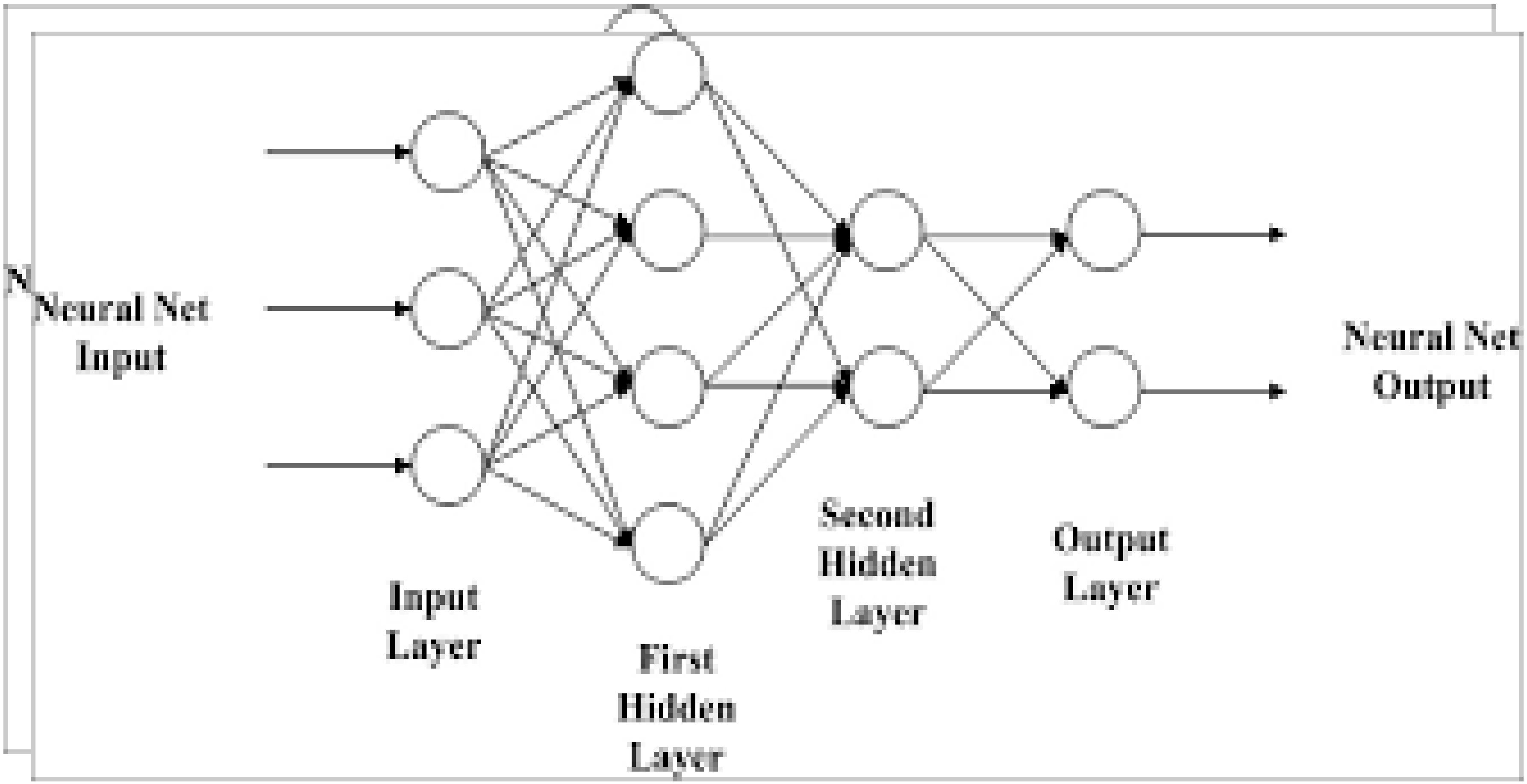
SINGLE LAYER PERCEPTRON NETWORK



A TWO-LAYER NEURAL NETWORK



MULTI LAYER PERCEPTRON NETWORK



SUPERVISED LEARNING (PERCEPTRON)

- To start training process ,initially the weights and the bias are set to zero.
- Also set learning parameter, ranges between 0 and 1.
- Then find output by multiplying input with weights and adding bias entity. The after apply activation function on it.
- This output is compared with the target, where if any difference occurs, we go in for weight updation based on **perceptron learning rule**, else the network training is stopped.
- The algorithm can be used for both binary and bipolar input vectors. It uses a bipolar target with fixed threshold and adjustable bias.

THE TRAINING ALGORITHM (SINGLE LAYER PERCEPTRON)

1. Initialize weights and bias θ (initially it can be zero). Set learning rate α **(o to 1).**
2. While stopping condition is false do steps 3-7.
3. For each training pair s:t do steps 4-6.
4. Set Activations of input units.

   xi = sj for i= 1 to n

5. Compute the output unit response.

   $$\mathbf{Yin = \sum xiwi + b;}$$

   The activation function used is,

   $$Y = f(Yin) = \begin{cases} 1, & \text{if } Yin > \theta \\ 0, & \text{if } Yin = \theta \\ -1, & \text{if } Yin < \theta \end{cases}$$

THE TRAINING ALGORITHM (SINGLE LAYER PERCEPTRON)

6. The weights and bias are updated if the target is not equal to the output response.

   If $t != y$ and the value of X_i != 0.

   $$\mathbf{W_{i(new)} = w_{i(old)} + \alpha * t * x_i}$$

   $$\mathbf{b_{(new)} = b_{(old)} + \alpha * t}$$

   else

   $$\mathbf{W_{i(new)} = w_{i(old)}}$$

   $$\mathbf{b_{(new)} = b_{(old)}}$$

7. Test for stopping condition.

   The stopping conditions may be the weight changes.

NOTE: Only weights connecting active input units (xi!=0) are updated.

Weights are updated only for patterns that do not produce the correct value of y.

LINEAR SEPARABILITY

- If two classes of patterns can be separated by a decision boundary, represented by the linear equation

$$b + \sum_{i=1}^{n} x_i w_i = 0$$

- then they are said to be linearly separable. The simple network can correctly classify any patterns.
- If such a decision boundary does not exist, then the two classes are said to be linearly inseparable.
- Linearly inseparable problems cannot be solved by the simple network , more sophisticated architecture is needed.

EXAMPLE-I

Example 4.1 Develop a perceptron for the AND function with bipolar inputs and targets.

Solution The training pattern for AND function can be,

Input			Target
X_1	X_2	b	t
1	1	1	1
−1	1	1	−1
1	−1	1	−1
−1	−1	1	−1

SOLUTION

Step 1: Initial weights $w_1 = w_2 = 0$ and $b = 0$, $\alpha = 1$, $\theta = 0$.

Step 2: Begin computation.

Step 3: For input pair (1, 1): 1, do Steps 4–6

Step 4: Set activations of input units

$$x_i = (1, 1).$$

Step 5: Calculate the net input.

$$y_{-in} = b + \sum x_i w_i = 0 + 1 \times 0 + 1 \times 0 = 0$$

Applying the activation,

$$y = f(y_{-in}) = \begin{cases} 1, & \text{if} \quad y_{-in} > 0 \\ 0, & \text{if} \quad -0 \le y_{-in} \le 0 \\ -1, & \text{if} \quad y_{-in} < -0 \end{cases}$$

Therefore $y = 0$.

SOLUTION

Step 6: $t = 1$ and $y = 0$

Since $t \neq y$, the new weights are,

$$w_{i(new)} = w_{i(old)} + \alpha t x_i$$

$$w_{1(new)} = w_{i(old)} + \alpha t x_1 = 0 + 1 \times 1 \times 1 = 1$$

$$w_{2(n)} = w_{2(o)} + \alpha t x_2 = 0 + 1 \times 1 \times 1 = 1$$

$$b_{(new)} = b_{(old)} + \alpha t$$

$$b_{(n)} = b_{(0)} + \alpha t = 0 + 1 \times 1 = 1$$

The new weights and bias are [1 1 1].

The algorithmic steps are repeated for all the input vectors with their initial weights as the previously calculated weights.

By presenting all the input vectors, the updated weights are shown in table below:

Input			Net	Output	Target	Weight Changes			Weights		
x_1	x_2	B	y_{in}	y	t	Δw_1	Δw_2	Δb	w_1	w_2	B
								(0	0	0)	
1	1	1	0	0	1	1	1	1	1	1	1
−1	1	1	1	1	−1	1	−1	−1	2	0	0
1	−1	1	2	1	−1	−1	1	−1	1	1	−1
−1	−1	1	−3	−1	−1	0	0	0	1	1	−1

SOLUTION

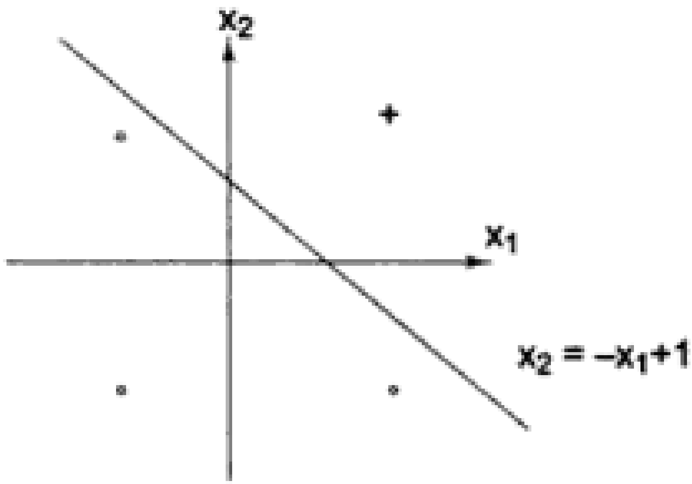
This completes one epoch of the training.

The final weights after the first epoch is completed are, $w_1 = 1, w_2 = 1, b = -1$

We know that $b + x_1 w_1 + x_2 w_2 = 0$

$$x_2 = -x_1 \frac{w_1}{w_2} - \frac{b}{w_2}$$

$$x_2 = -x_1 \frac{1}{1} - \frac{(-1)}{1}$$



$x_2 = -x_1 + 1$ is the separating line equation.

The decision boundary for AND function trained by perceptron network is given as,

In a similar way, the perceptron network can be developed for logic functions OR, NOT, AND NOT etc.