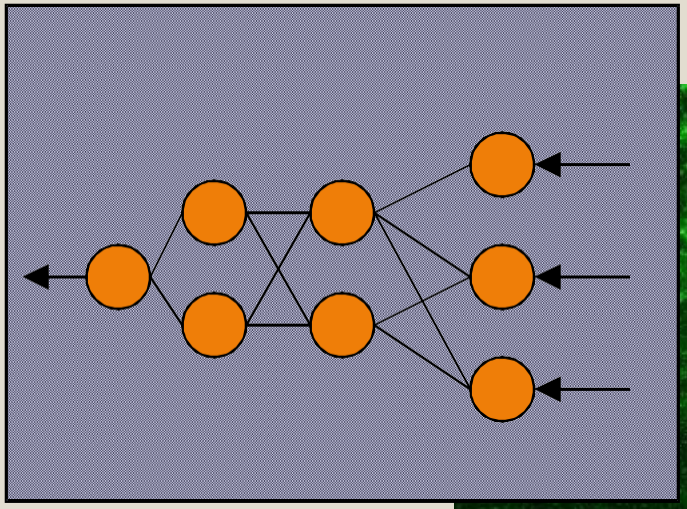
Introduction to
Artificial Neural Network Models



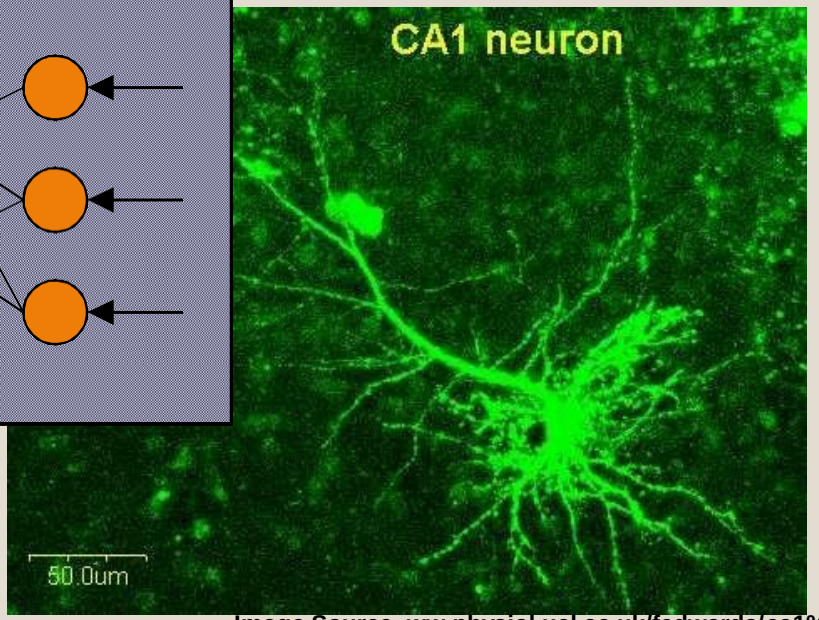


Image Source: ww.physiol.ucl.ac.uk/fedwards/ca1%20neuron.jpg

Definition

Neural Network

A broad class of models that mimic functioning inside the human brain

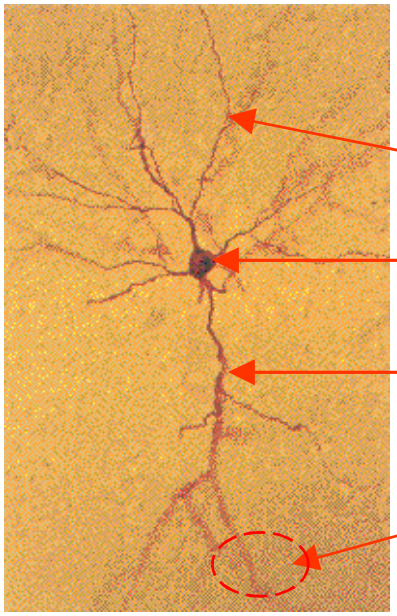
There are various classes of NN models.
They are different from each other depending on

- Problem types
  Prediction, Classification , Clustering
- Structure of the model
- Model building algorithm

For this discussion we are going to focus on
Feed-forward Back-propagation Neural Network
(used for Prediction and Classification problems)

A bit of biology . . .

Most important functional unit in human brain – a class of cells called – **NEURON**



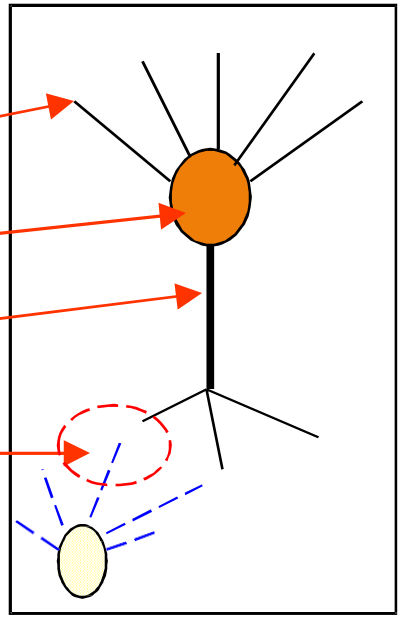
Hippocampal Neurons
Source: heart.cbl.utoronto.ca/ ~berj/projects.html

Dendrites

Cell Body

Axon

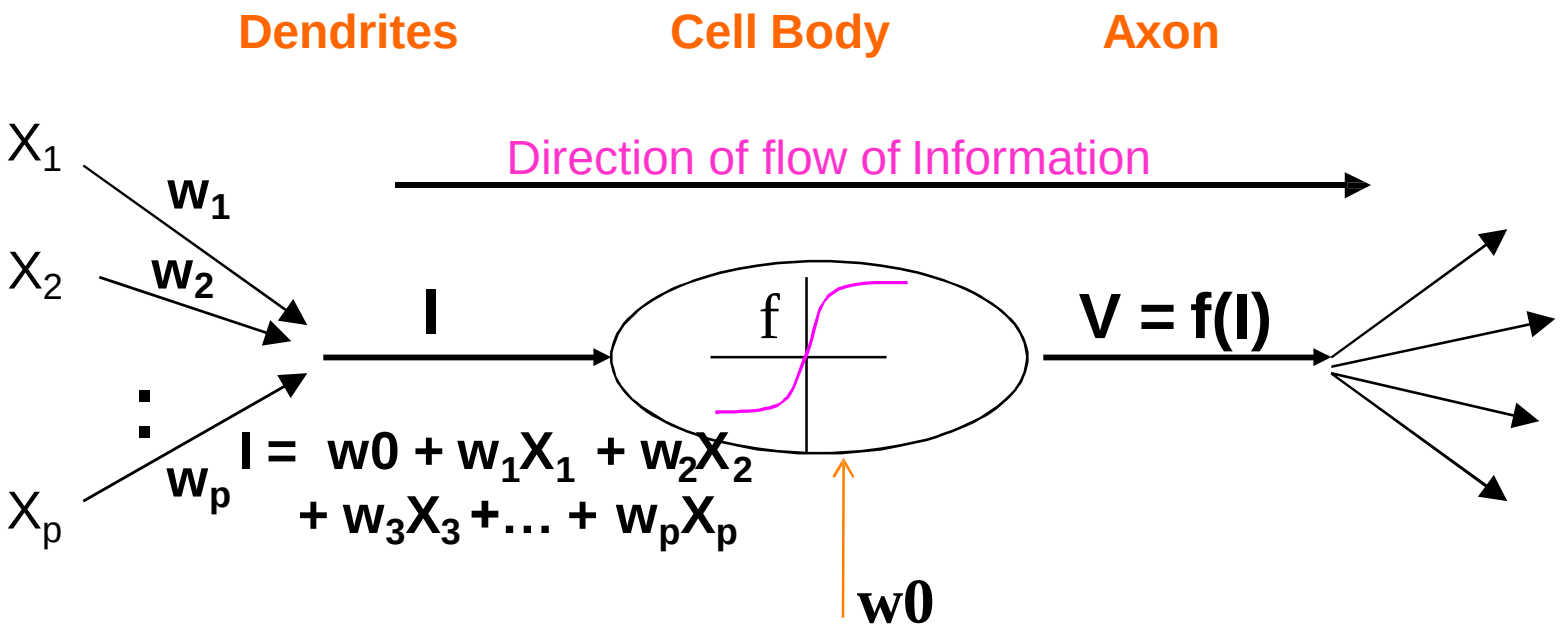
Synapse



Schematic

- **Dendrites** – Receive information
- **Cell Body** – Process information
- **Axon** – Carries processed information to other neurons
- **Synapse** – Junction between Axon end and Dendrites of other Neurons

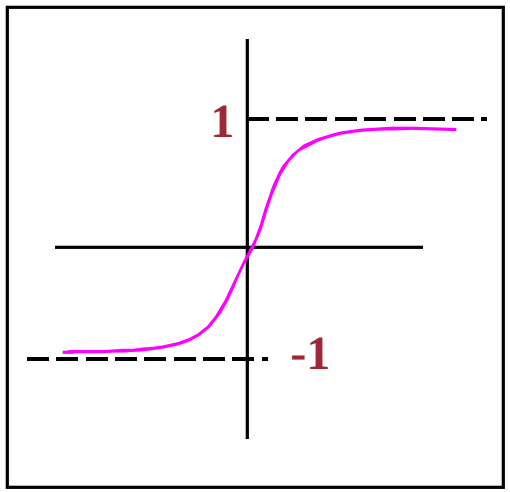
An Artificial Neuron



- Receives Inputs X_1 X_2 … X_p from other neurons or environment
- Inputs fed-in through connections with 'weights'
- Total Input = Weighted sum of inputs from all sources
- Transfer function (Activation function) converts the input to output
- Output goes to other neurons or environment

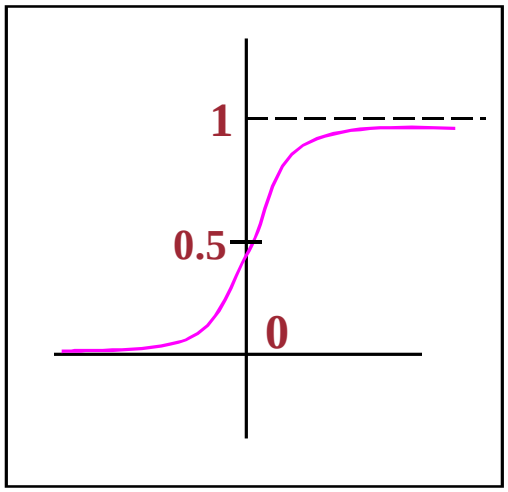
Transfer Functions

There are various choices for Transfer / Activation functions



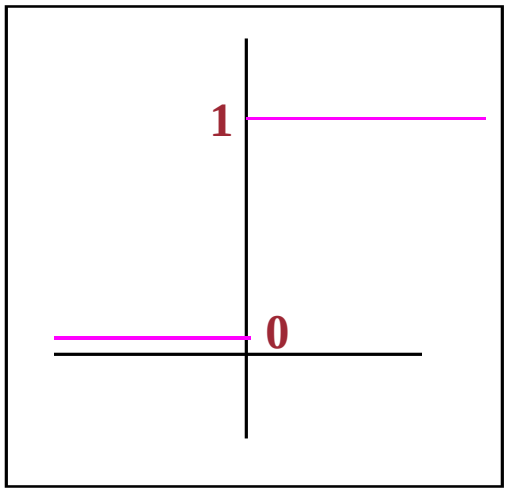
Tanh

$f(x) = (e^{x} - e^{-x}) / (e^{x} + e^{-x})$



Logistic

$f(x) = e^{x} / (1 + e^{x})$



Threshold

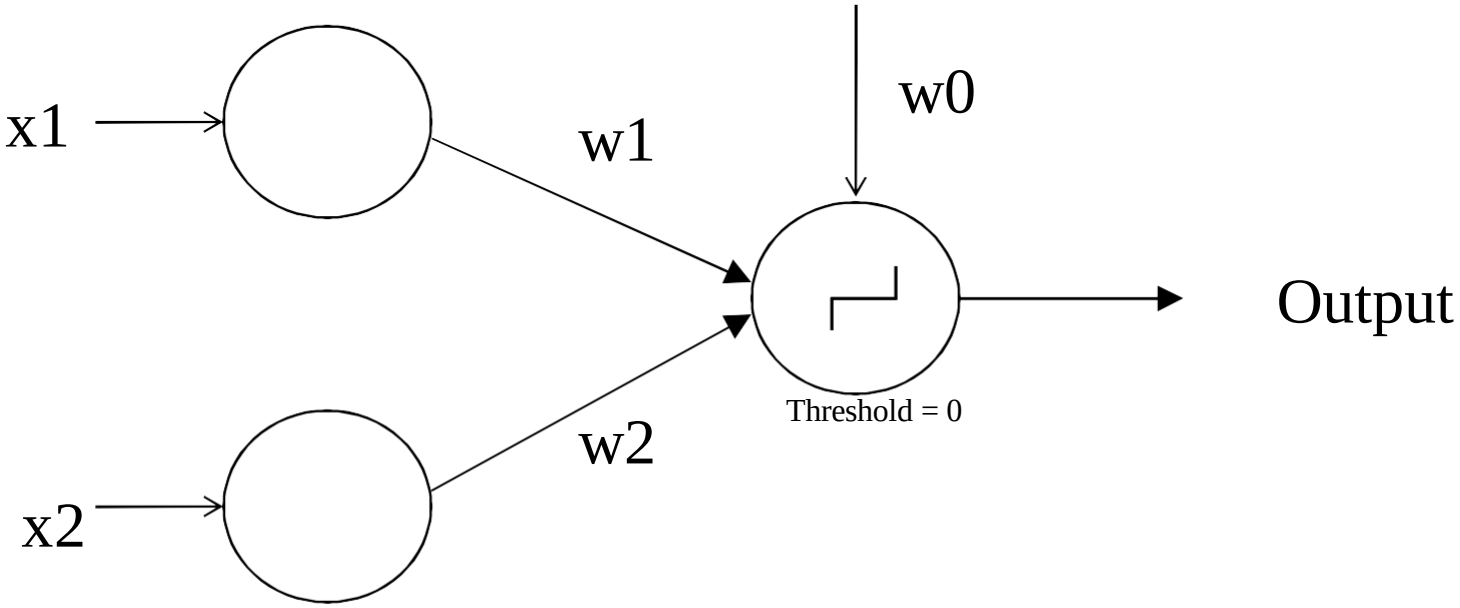
$$f(x) = \begin{cases} 1 \text{ if } x >= 1 \\ 0 \text{ if } x < 0 \end{cases}$$

(Perceptron)

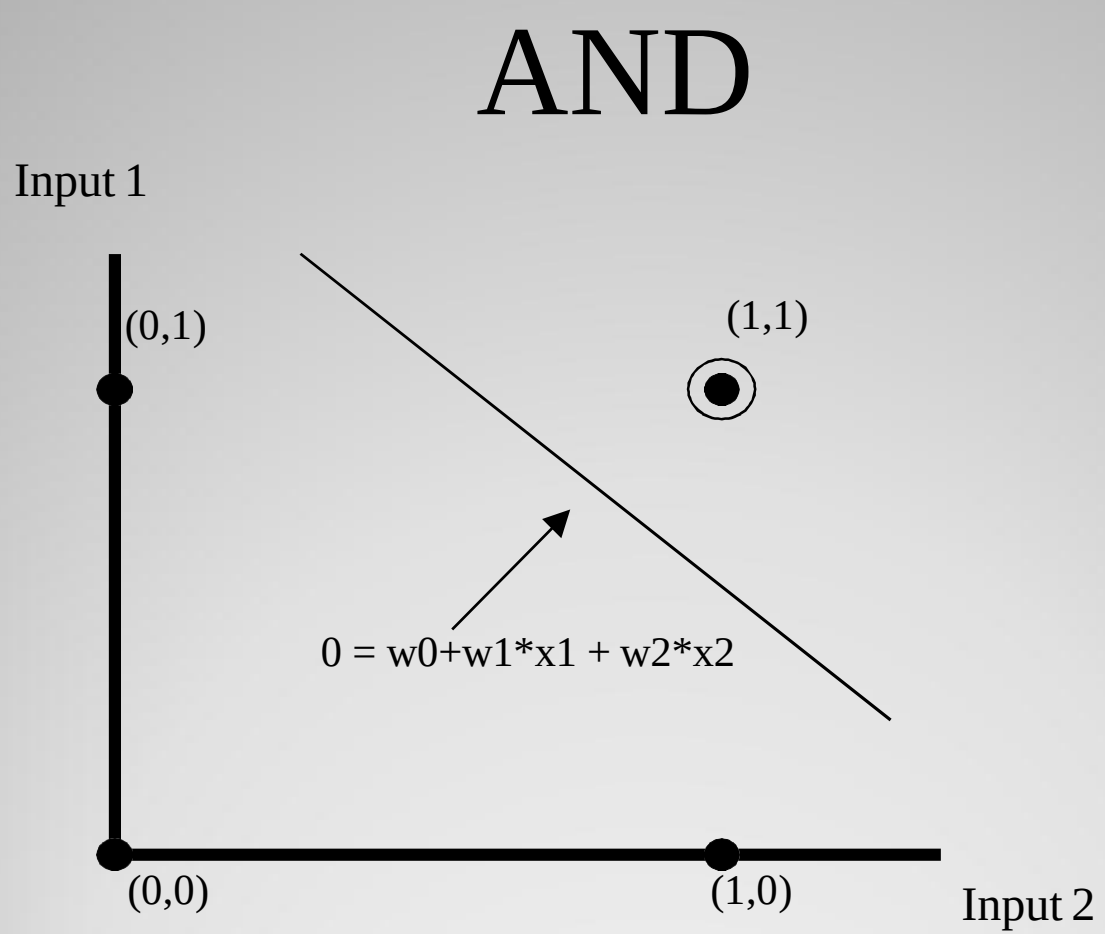
Linearly and non-linearly separable

Linearly separable – OR / AND
Linearly inseparable - XOR

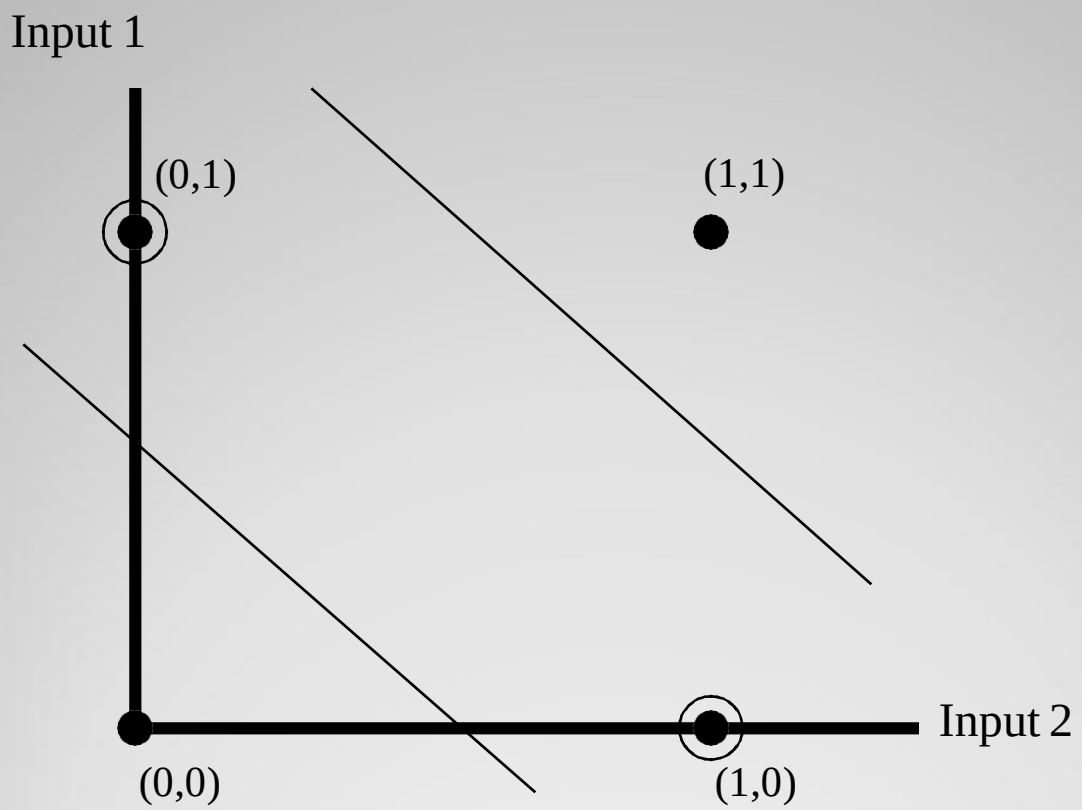
Single layer perceptron



Output space for AND gate



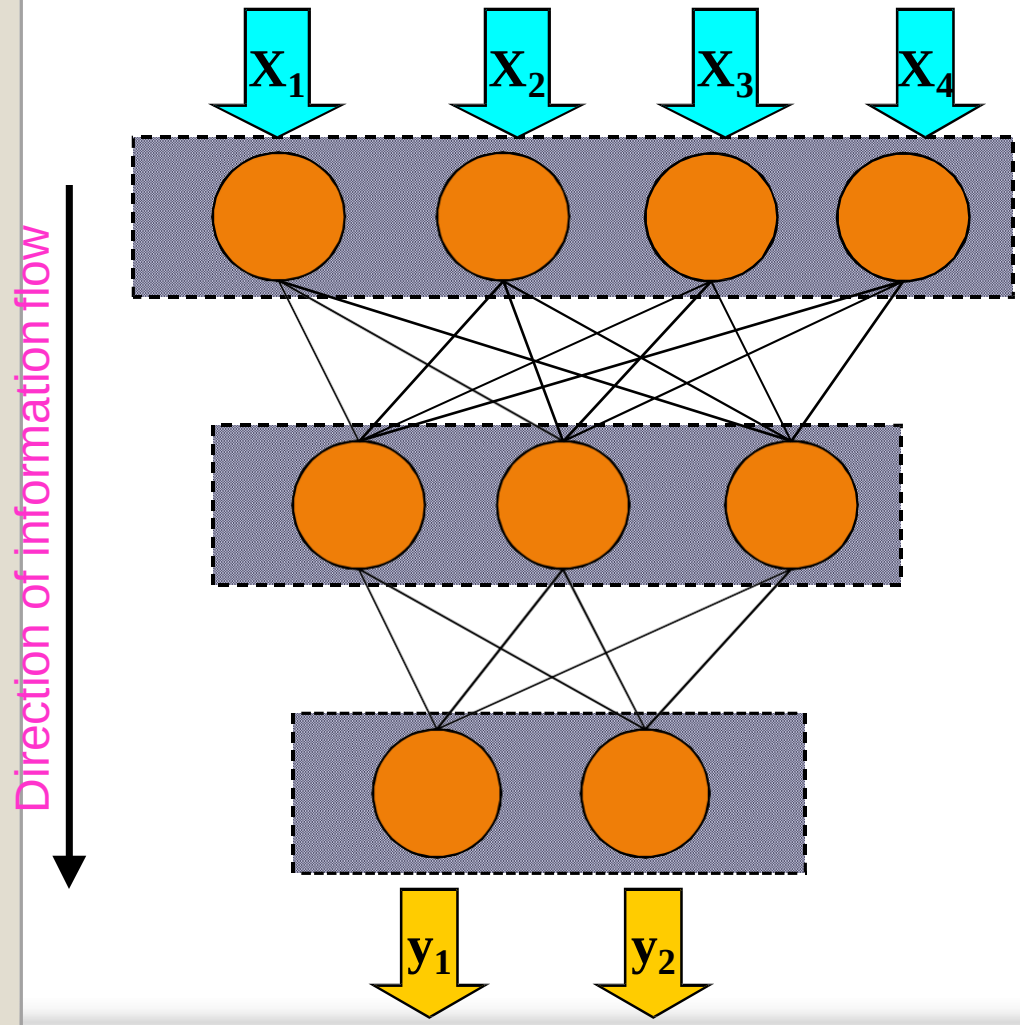
- Output space for XOR gate
- Demonstrates need for hidden layer



XOR

ANN – Feed-forward Network

A collection of neurons form a ‘Layer’



Input Layer

- Each neuron gets ONLY one input, directly from outside

Hidden Layer

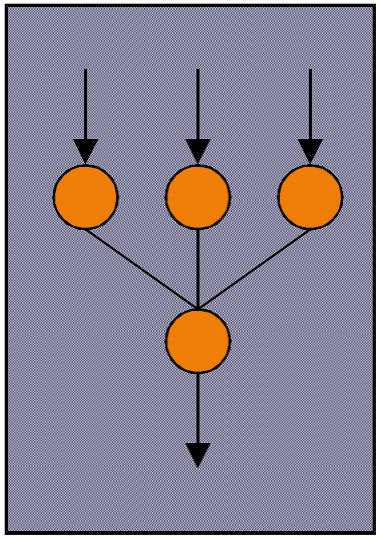
- Connects Input and Output layers

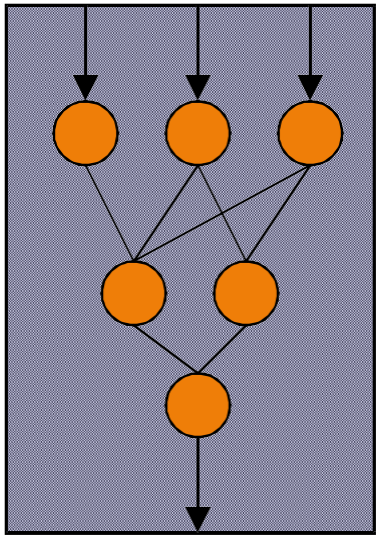
Output Layer

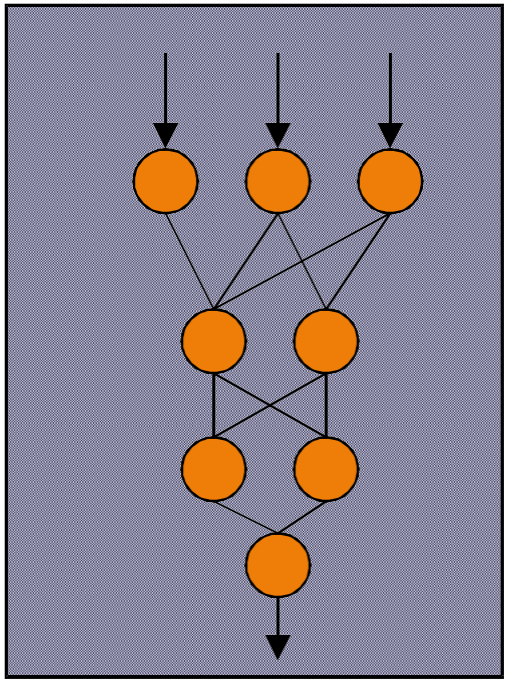
- Output of each neuron directly goes to outside

ANN – Feed-forward Network

Number of hidden layers can be **None** **One** **More**

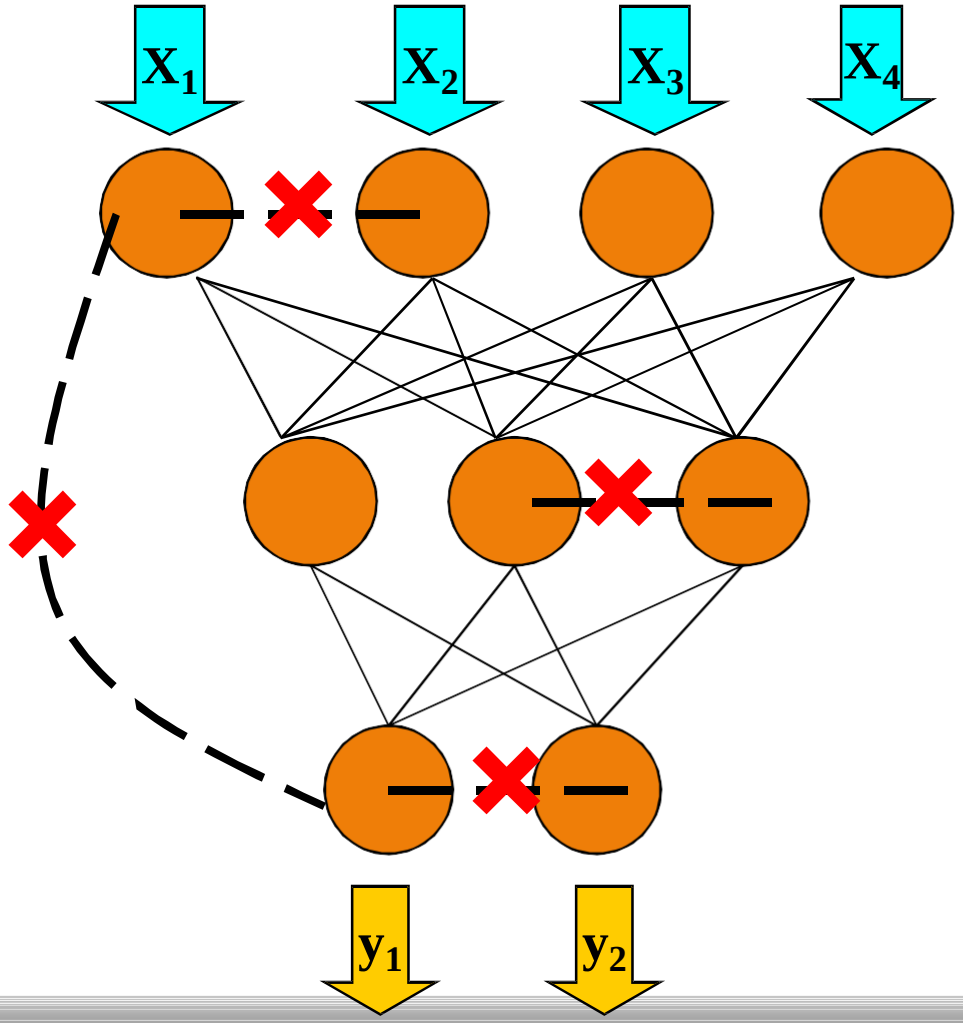






ANN – Feed-forward Network

Couple of things to note



- Within a layer neurons are NOT connected to each other.
- Neuron in one layer is connected to neurons ONLY in the NEXT layer. (Feed-forward)
- Jumping of layer is NOT allowed

One particular ANN model

What do we mean by ‘ A particular Model ‘ ?

Input: X_1 X_2 X_3 Output: Y Model: $Y = f(X_1 X_2 X_3)$

For an ANN : Algebraic form of f(.) is too complicated to write down.

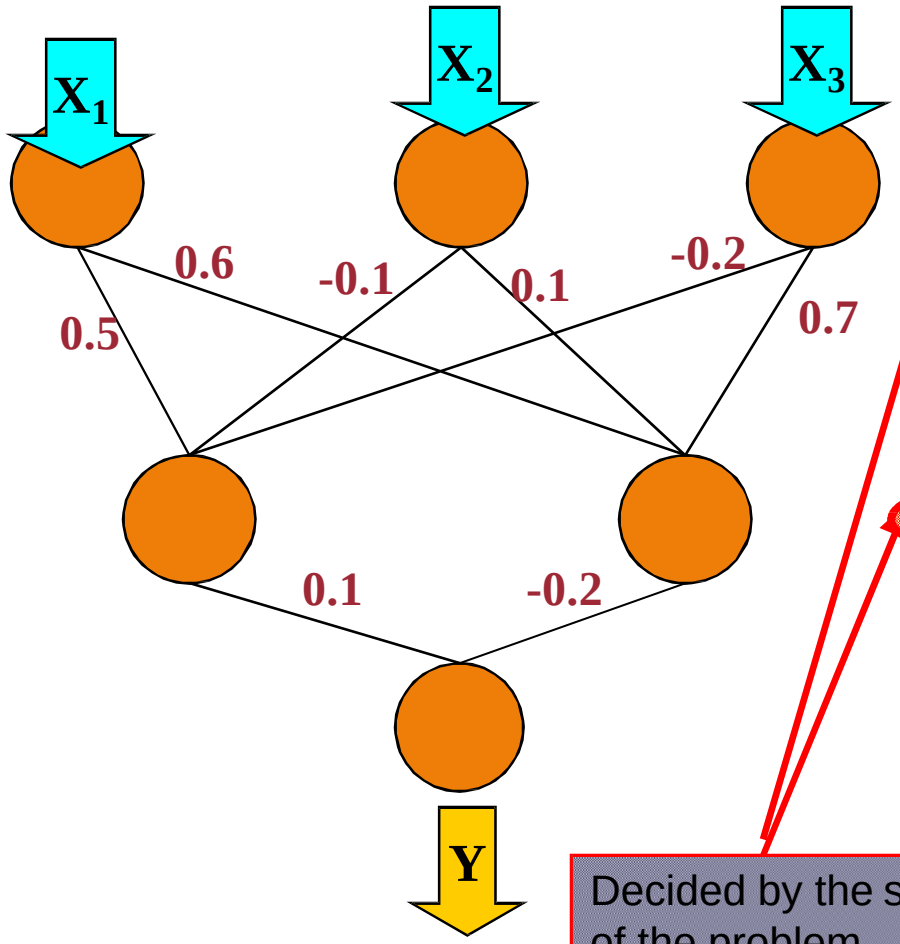
However it is characterized by

- # Input Neurons
- # Hidden Layers
- # Neurons in each Hidden Layer
- # Output Neurons
- WEIGHTS for all the connections

‘ Fitting ‘ an ANN model = Specifying values for all those parameters

One particular Model – an Example

Input: $X_1\ X_2\ X_3$ **Output:** Y **Model:** $Y = f(X_1\ X_2\ X_3)$



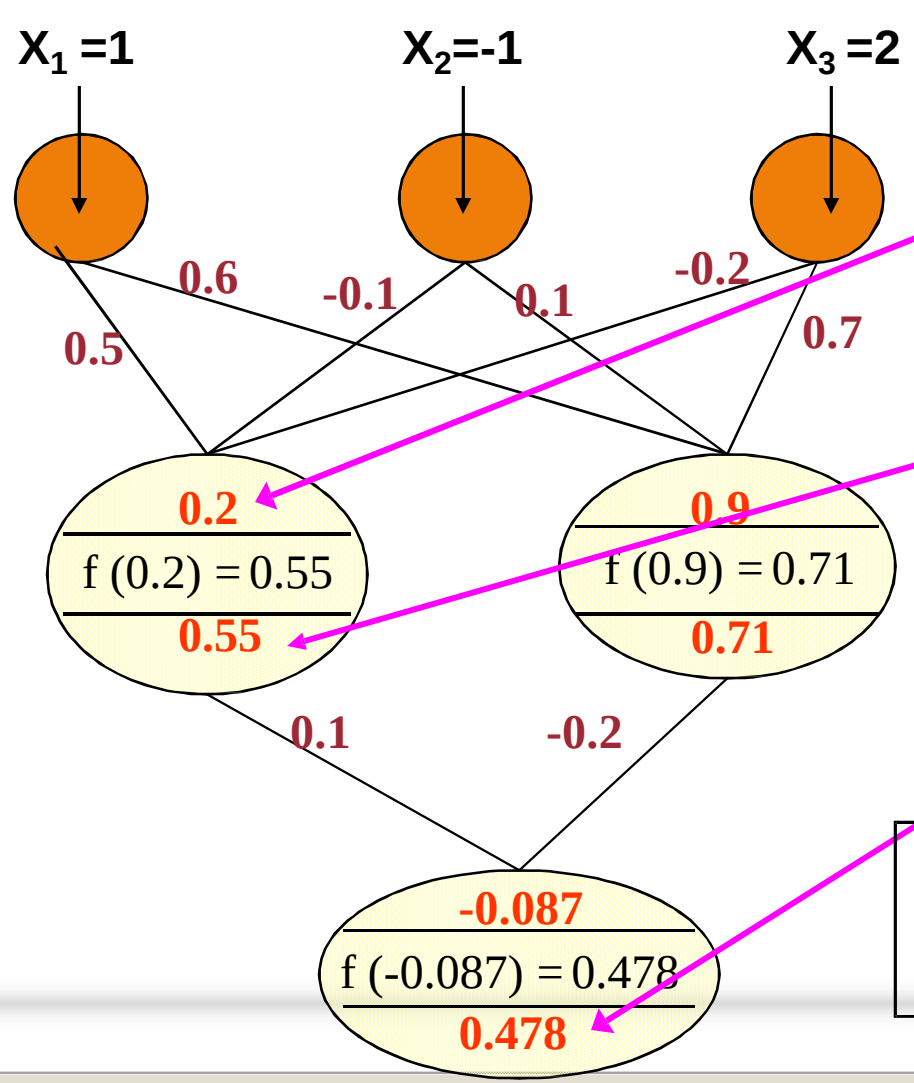
Parameters	Example
# Input Neurons	3
# Hidden Layers	1
# Hidden Layer Size	3
# Output Neurons	3
Weights	Specified

Decided by the structure of the problem
Input Nrns = # of X's
Output Nrns = # of Y's

Free parameters

Prediction using a particular ANN Model

Input: X_1 X_2 X_3 **Output:** Y **Model:** $Y = f(X_1 X_2 X_3)$



$0.2 = 0.5 * 1 - 0.1*(-1) - 0.2 * 2$

$f(x) = e^x / (1 + e^x)$

$f(0.2) = e^{0.2} / (1 + e^{0.2}) = 0.55$

Predicted Y = 0.478

Suppose Actual Y = 2

Then

Prediction Error = (2-0.478) =1.522