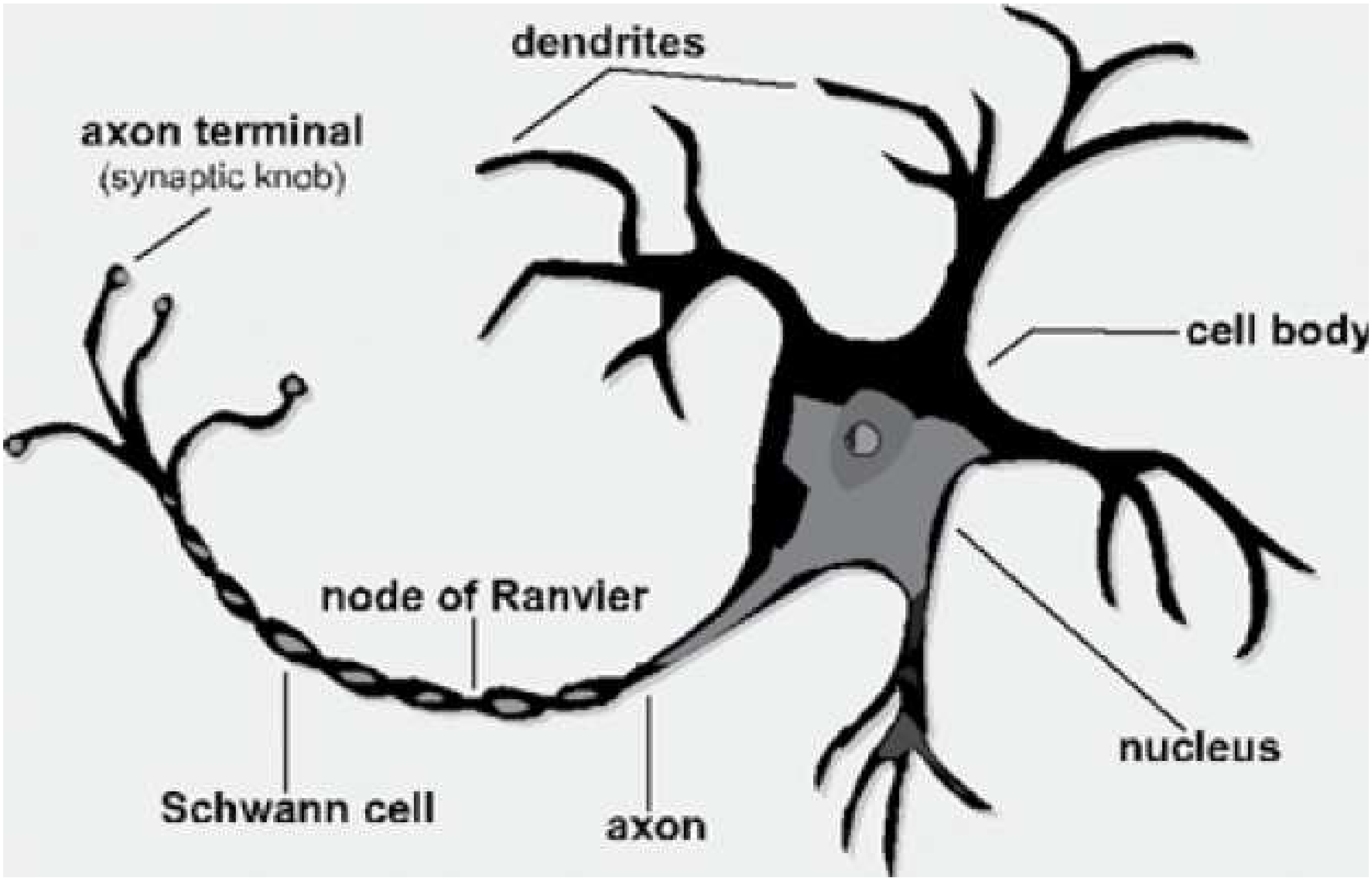
Pattern Recognition

Neural Networks

Introduction

- Aim: To construct a computer capable of "human-like thinking"
- The human brain is composed primarily of neuron cells. They are the basic building blocks of the human brain.
- Artificial neural networks attempt to simulate the behavior of these cells.

Biological Neuron Cells



Biological Neuron Cells

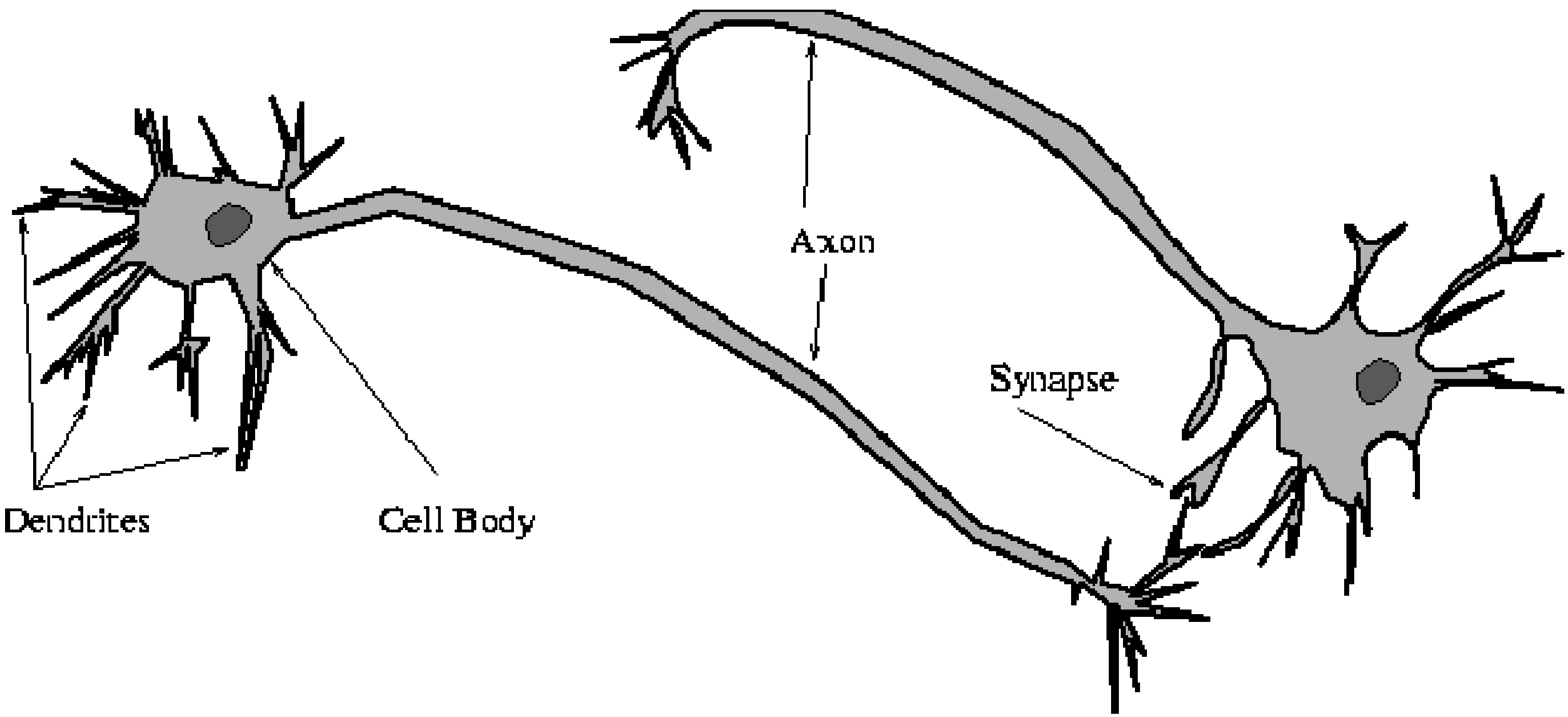
- A neuron accepts signals from dendrites.
- When a neuron accepts a signal, that neuron may fire.
- When a neuron fires, a signal is transmitted over the neuron’s axon.
- Ultimately, the signal will leave the neuron as it travels to the axon terminals.
- The signal is then transmitted to other neurons or nerves.

Decision in a Neuron

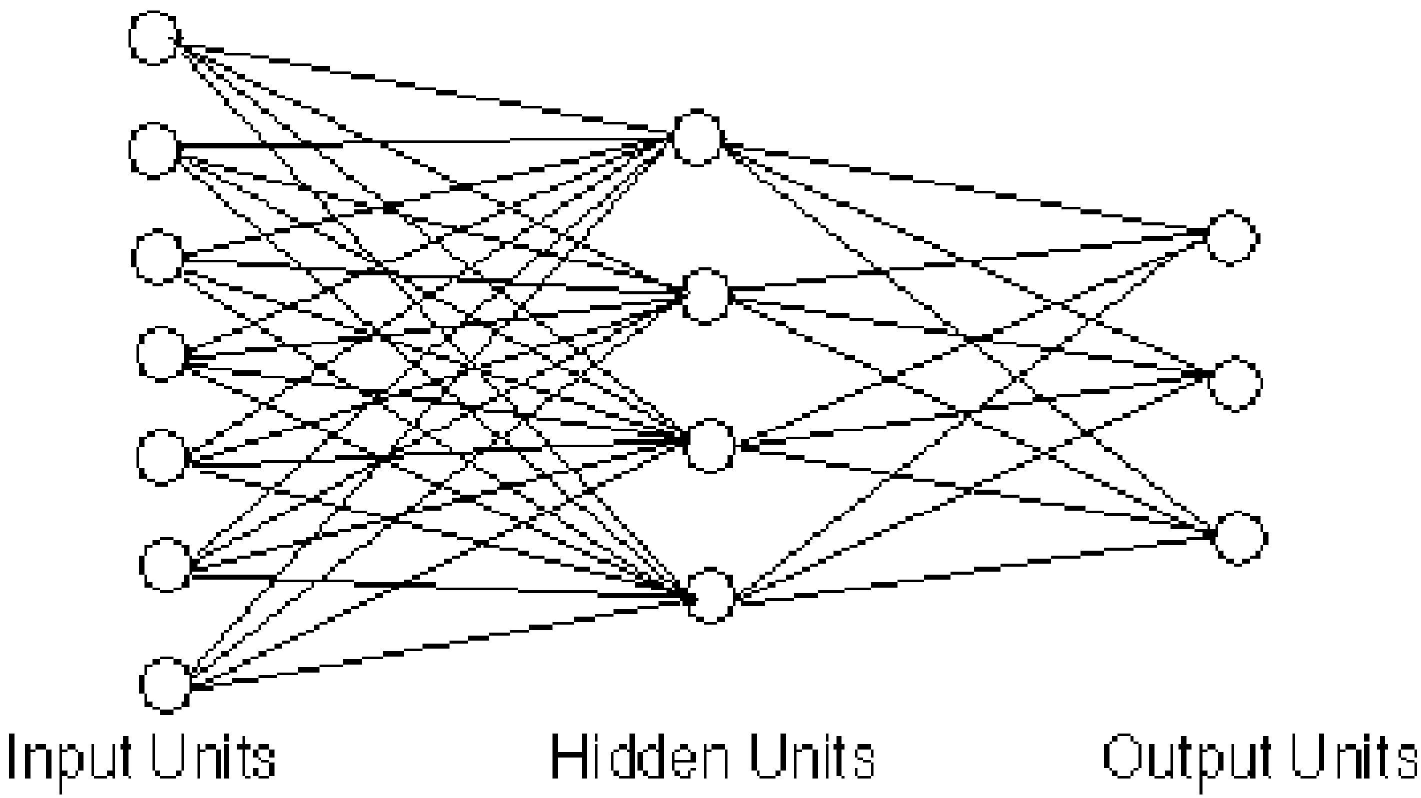
- A neuron makes a decision by **firing** or not firing.
- The decisions being made are extremely low-level decisions.
- Final decision is the result of many decisions made by **many** neurons.

How It Works?

- Neural networks are constructed of neurons that form layers.
- Input is presented to the layers of neurons. If the input to a neuron is within the range that the neuron has been trained for, then the neuron will fire.
- When a neuron fires, a signal is sent to the layers of neurons to which the firing neuron is connected.
- The connections between neurons are called synapses.



Axon
Synapse
Dendrites
Cell Body


Input Units
Hidden Units
Output Units

Basic Structures of Neural Networks

- The individual neurons that make up a neural network are interconnected through their synapses.
- These connections allow the neurons to signal each other as information is processed.
- Each connection is assigned a connection weight.
- If there is no connection between two neurons, then their connection weight is zero.
- These weights are what determine the output of the neural network; therefore, the connection weights form the memory of the neural network.
- Training is the process by which these connection weights are assigned.

Validating a Neural Network

- Validating a neural network allows us to determine if additional training is required.
- To correctly validate a neural network, validation data must be set aside that is completely separate from the training data.
- Once the network is properly trained, the validation group elements are used to validate the neural network.

Application in Problem Solving

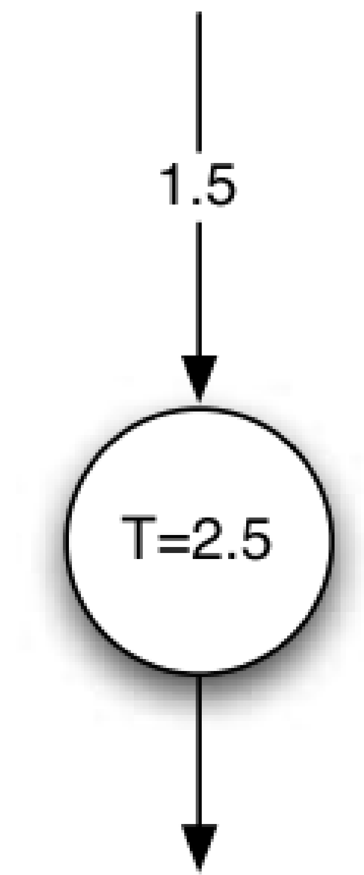
- Don’t Use Neural Network for:
  - Programs that are easily written out as flowcharts. If your program consists of well-defined steps, normal programming techniques will suffice.
  - One of the primary features of neural networks is their ability to learn. If the algorithm used to solve your problem is an unchanging business rule, there is no reason to use a neural network.
  - Finally, neural networks are often not suitable for problems in which you must know exactly how the solution was derived.

Using Neural Networks for Pattern Recognition

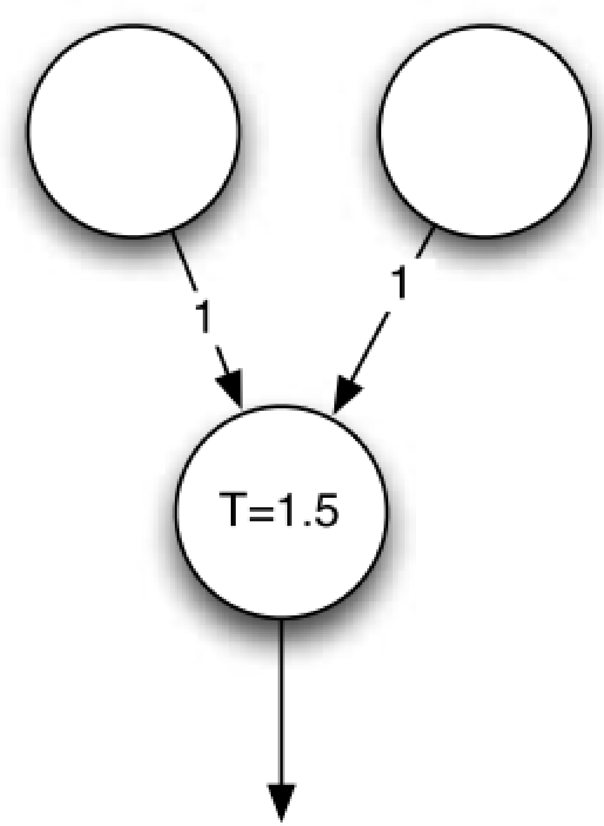
- In Pattern recognition applications (perhaps the most common use for neural networks), the neural network is presented a pattern. This could be an image, a sound, or any other data. The neural network then attempts to determine if the input data matches a pattern that it has been trained to recognize.
- A neural network trained for classification is designed to take input samples and classify them into groups.

Using a Simple Neural Network

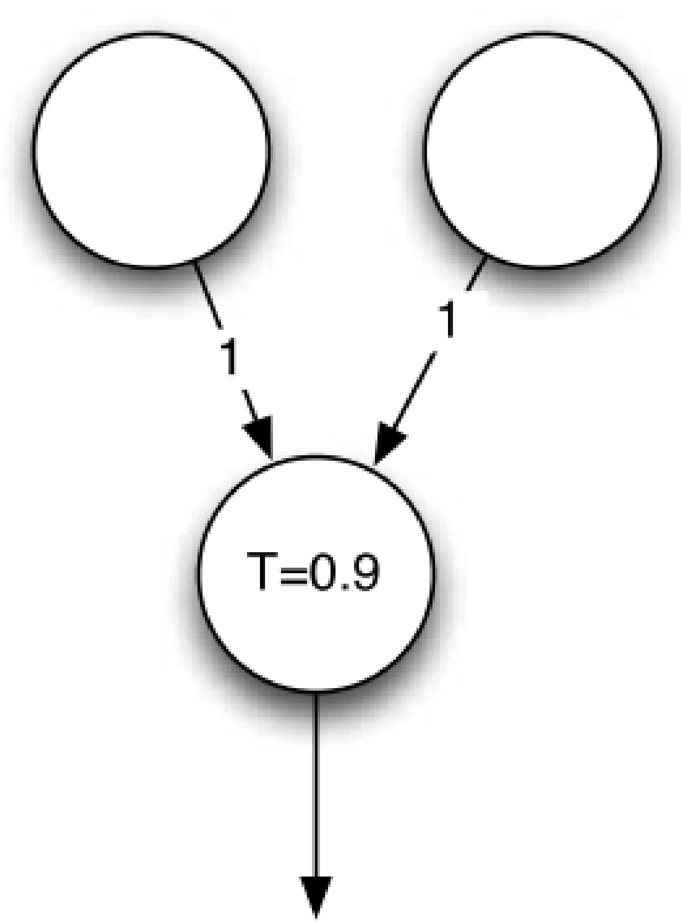
- Pass the input to output



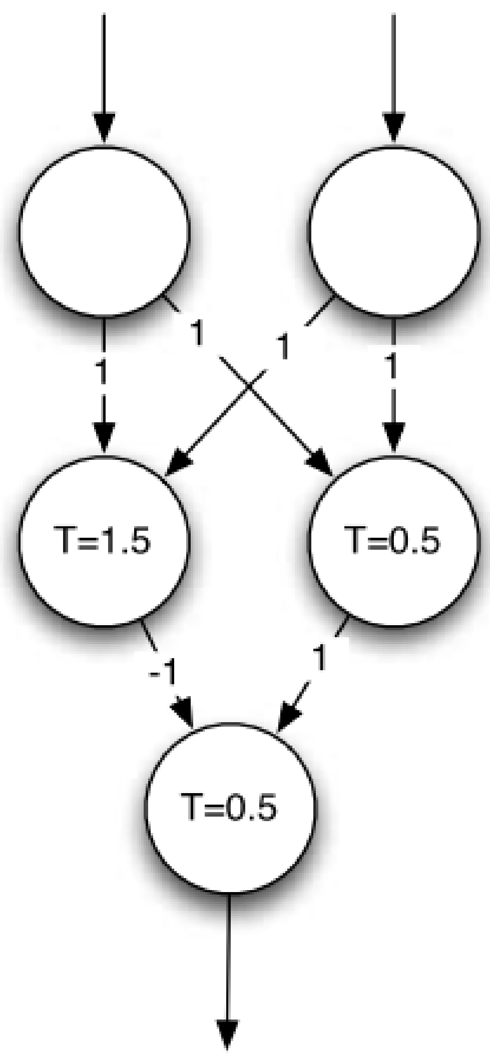
AND Operation



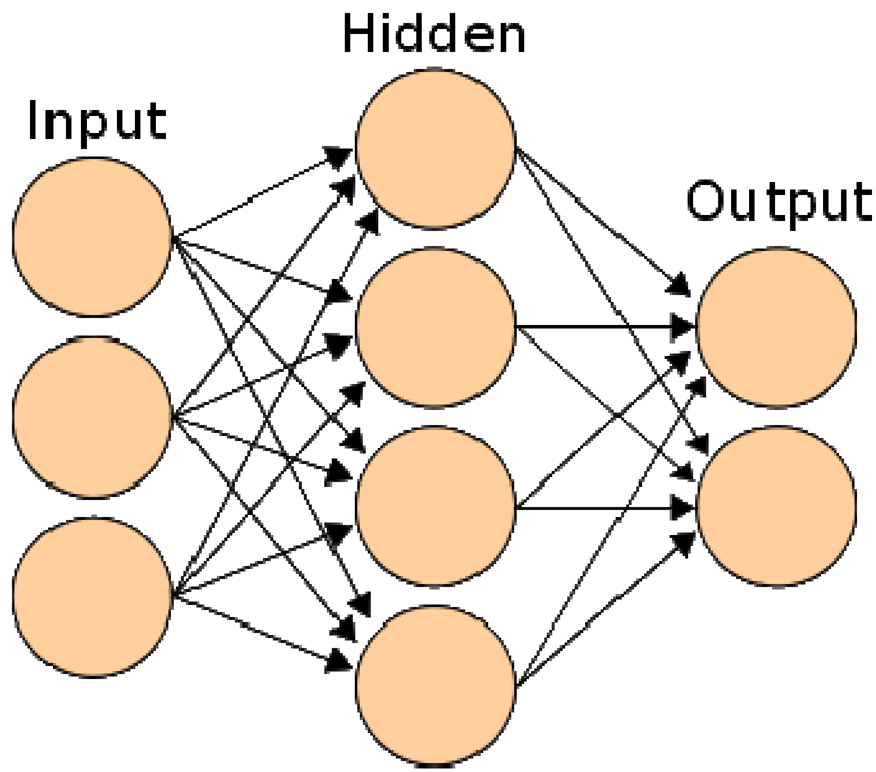
OR Operation



Neural Network for XOR Operation



Structure of a Neural Network



Neural Network Training

- Unsupervised training:
  - In the unsupervised training, the neural network is not provided with anticipated outputs.
- Supervised Training
  - The neural network has access to the anticipated outputs

Delta Training Rule

- Delta training rule is used with supervised learning
- The weights are initialized with small values
- The input is provided to the neural network
- The error is computed and the delta (update value) is found
- The weights are updated

Delta Training Rule

- The update value is given by:

  $\Delta w_{ij} = \mu(t_i - y_i)\, g(h_i)\, x_i$

  w_{ij} is the weight between neuron I and neuron j

  μ is the learning rate

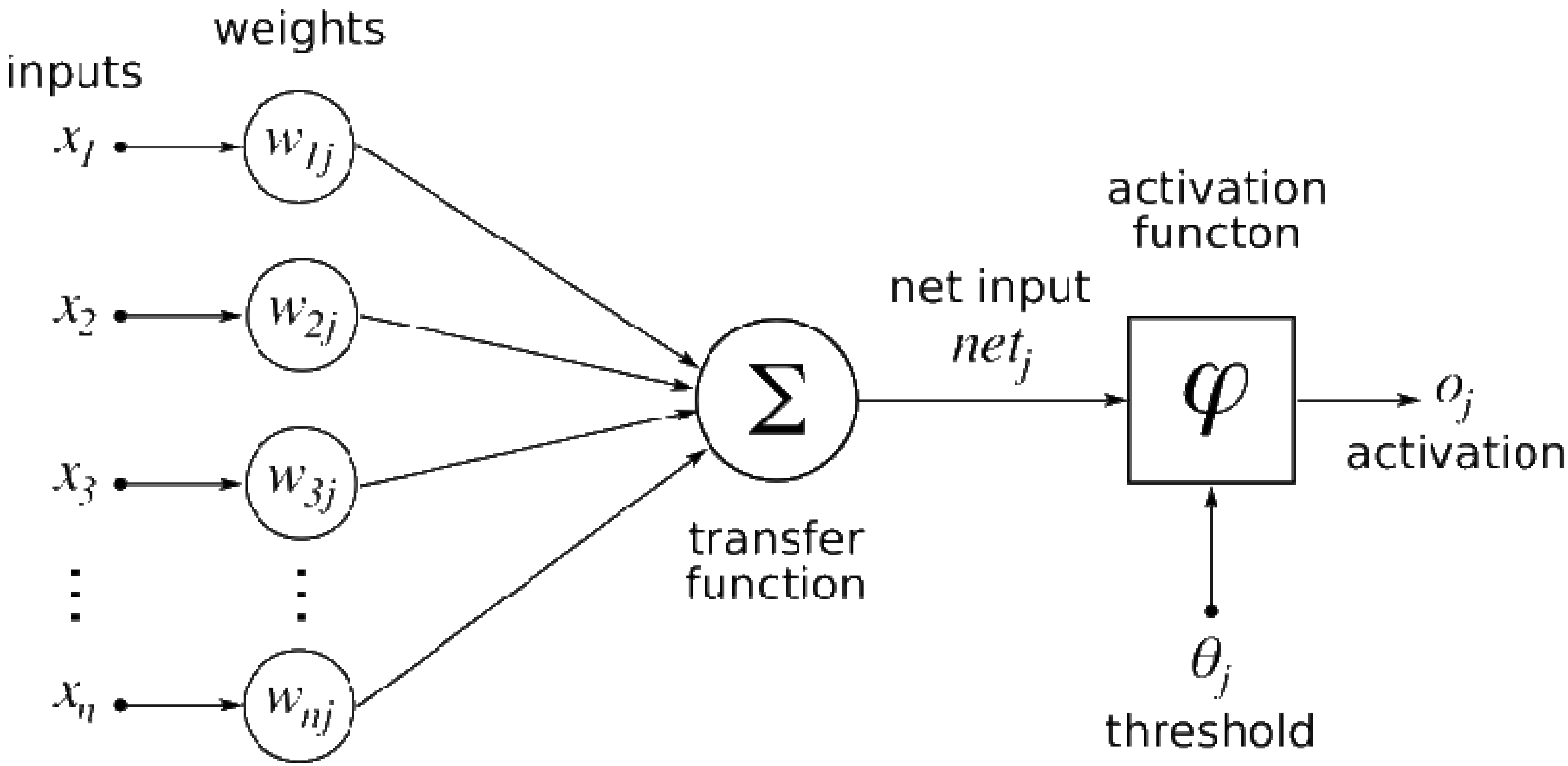
  t_i is the true value

  y_i is the output value

  $g(h_i)$ is the activation function

  x_i is the input value

Structure of a Neuron

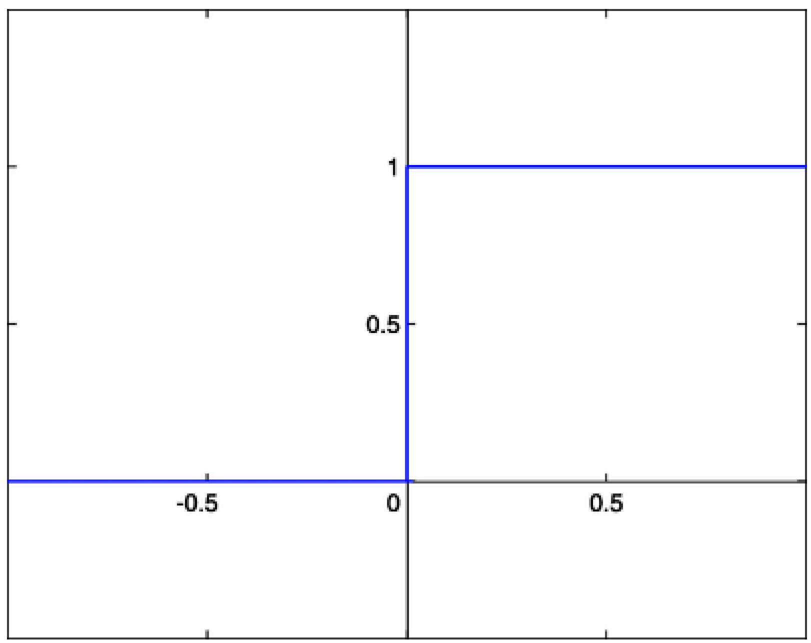


Activation Functions

- Four main activation functions are used with neural networks. These functions are:
  - Step Function
  - Ramp Function
  - Log-Sigmoid Function
  - Tan-Sigmoid Function

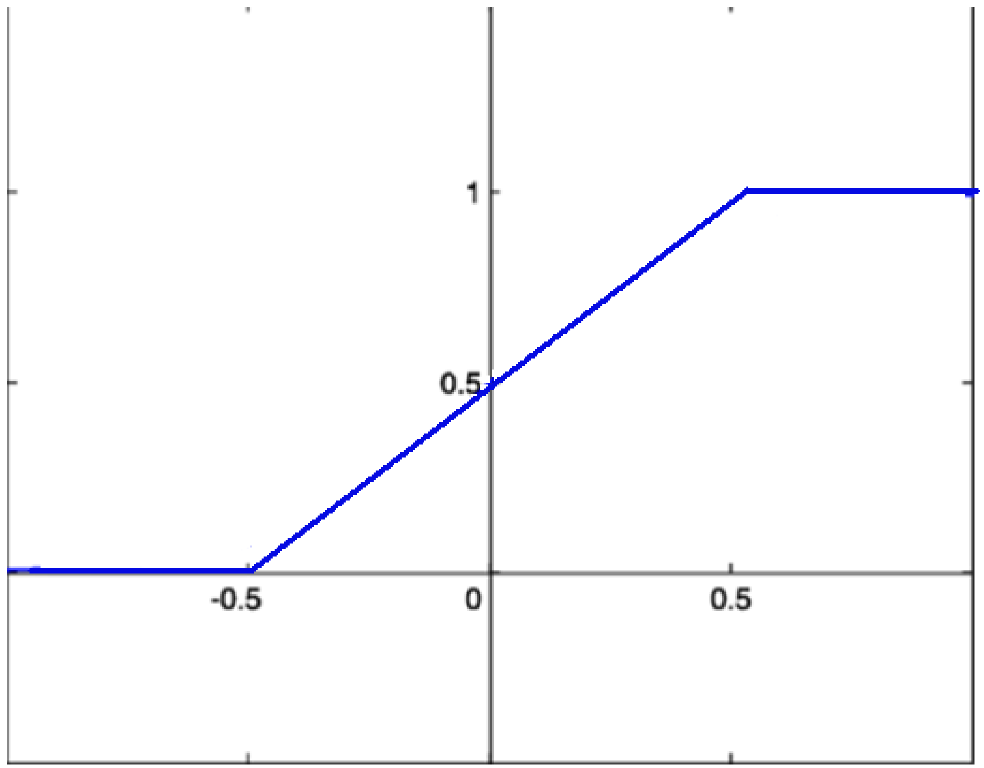
Step Function

- Useful for binary classification schemes



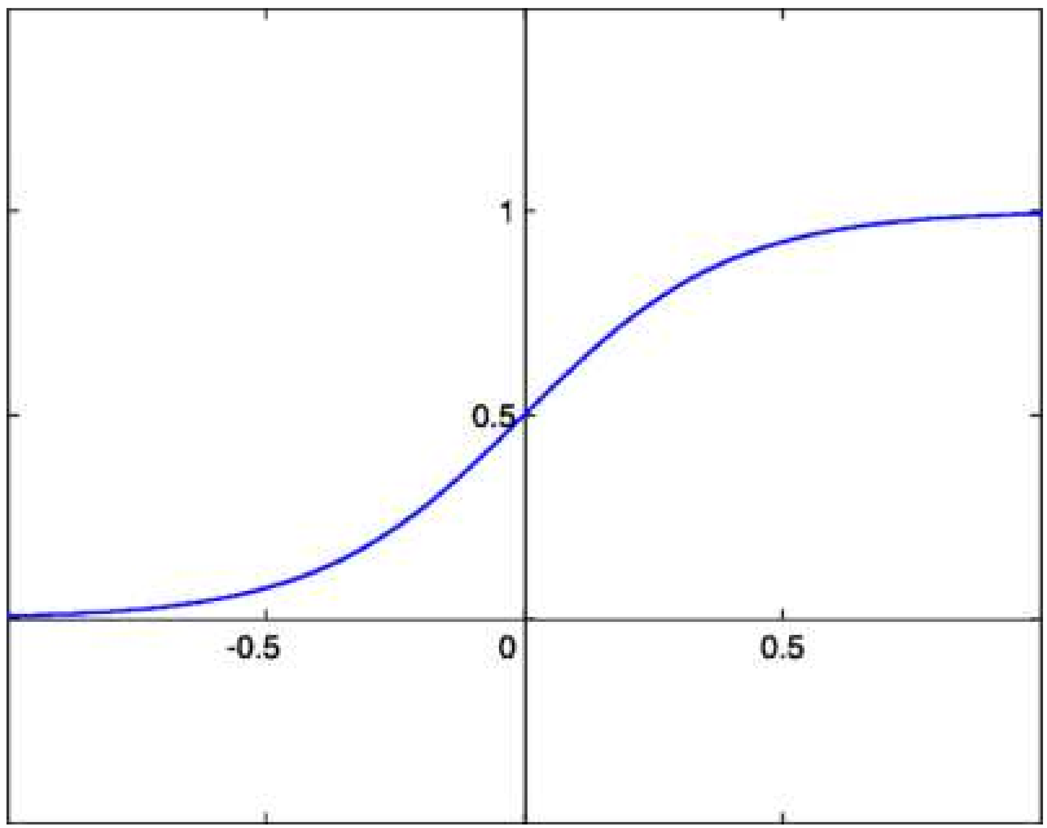
Ramp Function

- A line of positive slope may also be used to reflect the increase in firing rate that occurs as input current increases



Log-Sigmoid Function

- Is given by $\sigma(t) = \frac{1}{1 + e^{-\beta t}}$ where β is the slope parameter



Tan-Sigmoid Function

- Is given by $f(x) = \frac{e^{2x} - 1}{e^{2x} + 1}$ or

$$\sigma(t) = tanh(t) = \frac{c^t - c^{-t}}{e^t + e^{-t}}$$

- The derivative is given by

$$\frac{d\sigma(t)}{dt} = 1 - tanh^2(t) = sech^2(t) = 1 - \frac{(e^t - e^{-t})^2}{(e^t + e^{-t})^2}$$

Neural Network Design

- Input layer:
  - The input layer should represent the condition for which we are training the neural network.
  - Every input neuron should represent some independent variable that has an influence over the output of the neural network

Neural Network Design

- Output Layer:
  - To determine the number of neurons to use in output layer, we must consider the intended use of the neural network. If the neural network is to be used to classify items into groups, then it is often preferable to have one output neuron for each group that input items are to be assigned into.

Neural Network Design

- The Number of Hidden Layers

Number of Hidden Layers	Result
none	Only capable of representing linear separable functions or decisions.
1	Can approximate any function that contains a continuous mapping from one finite space to another.
2	Can represent an arbitrary decision boundary to arbitrary accuracy with rational activation functions and can approximate any smooth mapping to any accuracy.

Neural Network Design

- The Number of Neurons in the Hidden Layers?
- Some basic rules:
  - The number of hidden neurons should be between the size of the input layer and the size of the output layer.
  - The number of hidden neurons should be 2/3 the size of the input layer, plus the size of the output layer.
  - The number of hidden neurons should be less than twice the size of the input layer.
- Ultimately, the selection of an architecture for your neural network will come down to trial and error.

Feed-Forward and Back-Propagation Neural Networks

- Feed-forward describes how the neural network processes and recalls patterns.
- In a feed-forward neural network, neurons are only connected foreword.
- Back-propagation describes how the neural network is trained.
- Back-propagation is a form of supervised training.

Next…

- Training back-propagation neural networks
- Reducing the feature space