INTRODUCTION
Motivations

- Difficulty to automate "simple" tasks, example:
  - Recognize a handwritten letter
  - Learning from experience
  - Walking, grasping, . . . etc..
- The brain is a "computer" that computes differently:
  - "Parallel" processing of information
  - Capability to "(re-)organize" itself so as to perform any task
- Example, the *sonar* of a bat:
  - Distance, speed, size, azimuth and elevation of the target?
  - All this information from with a small brain!
  - What about sonar or radar machines?
  - How does the brain of the bat do it?

Motivations
(continued)

In summary, the brain has the following advantages over the computer

- Ability to derive meaning from complicated or imprecise data
- Learning and adaptive learning
- Self-organization
- Real-time processing of complex information at once
- Fault-tolerance
- Ability to store and process *experience*

An *Artificial Neural Network* is a machine (either software or hardware) that is designed to *model* the way in which the brain performs a particular task or function of interest. In order to achieve this we must *structure and behavior* of the brain.

Biological Neural Networks

The human nervous system can be viewed as a three-stage system

Stimulus → Receptors ⇄ Neural Net ⇄ Effectors → Response

- Receptors convert stimuli into electrical impulses

- Effectors convert electrical impulses into responses

- Brain receives, perceives information and make decisions. It has about 10^{11} neurons each connected to about 10^4 other neurons

- Biological neuron:

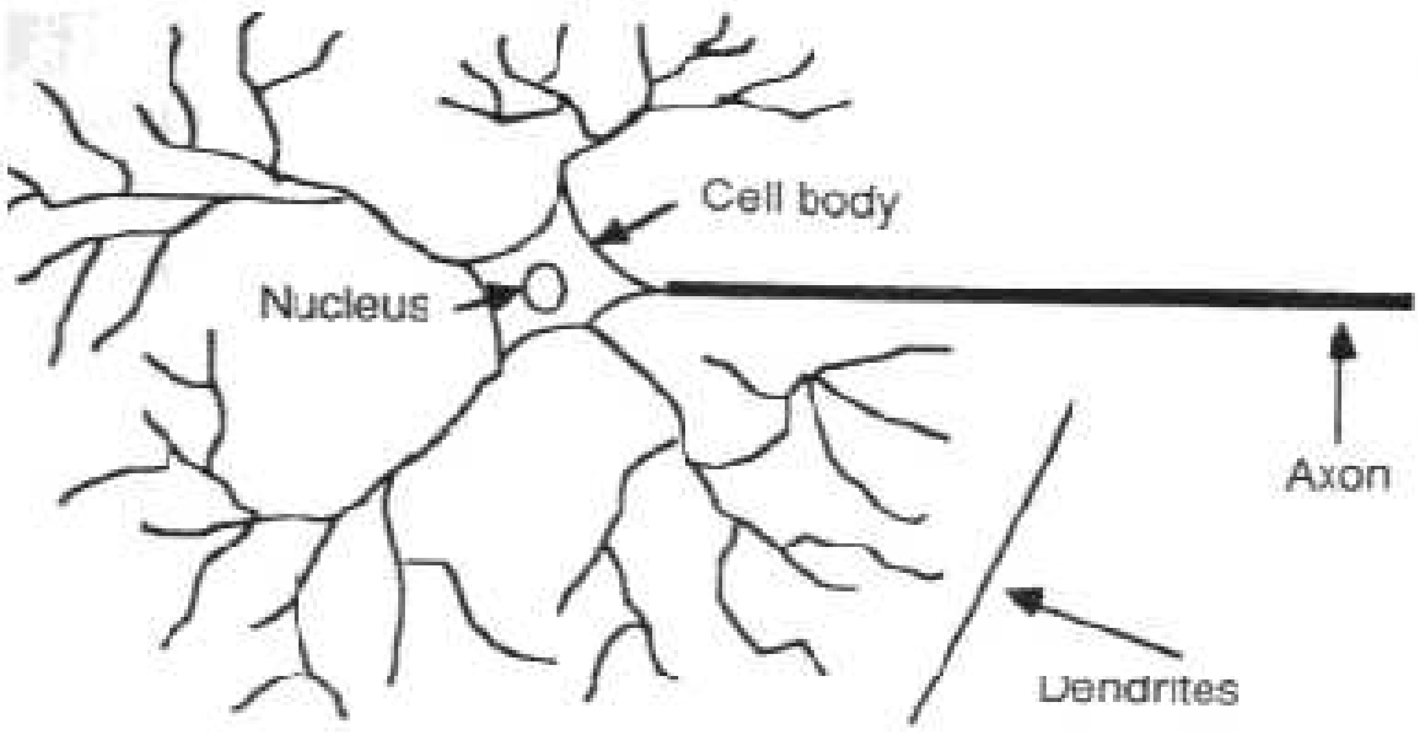
  - *Dendrites* (inputs)

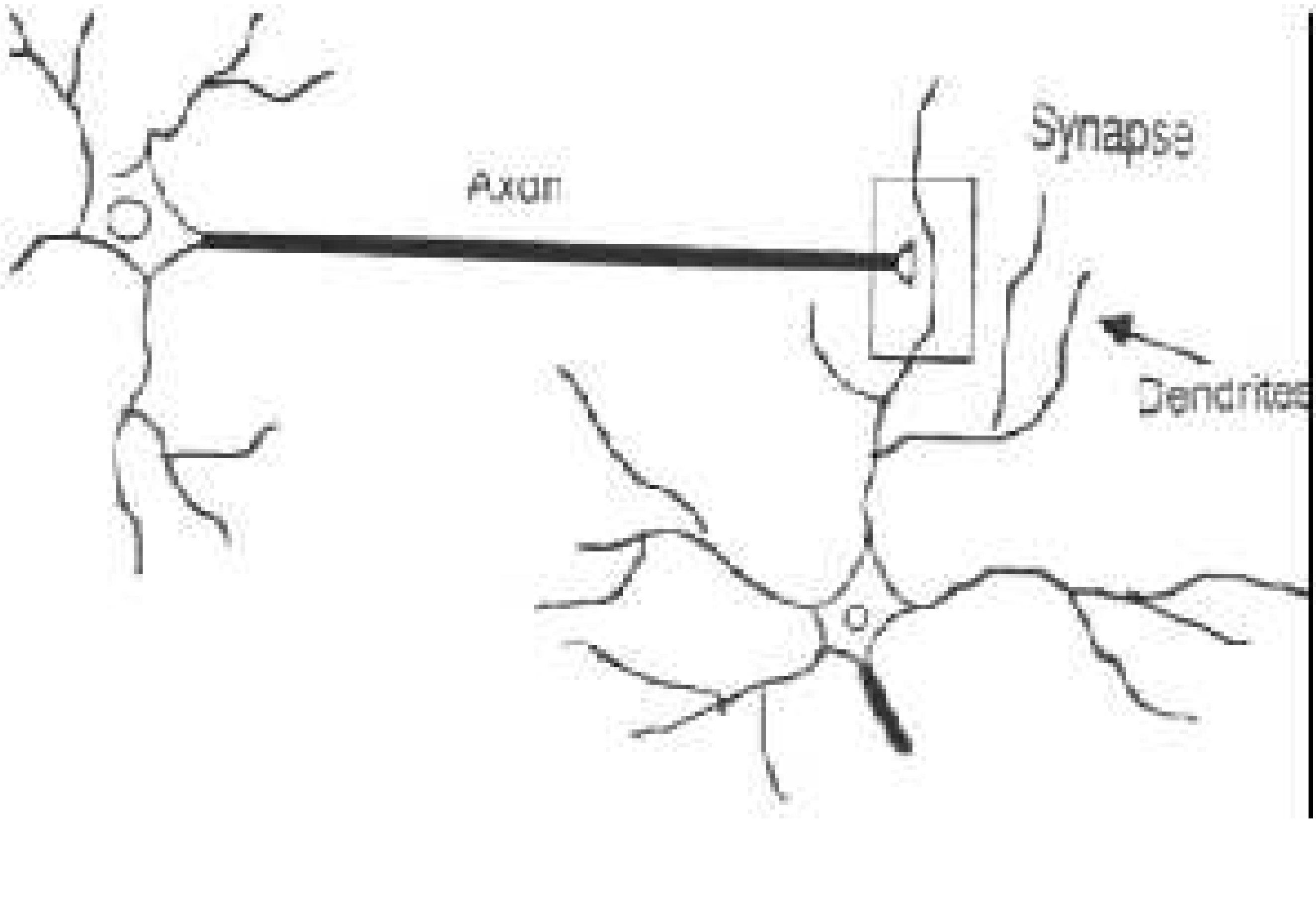
  - Cell body called *soma* (processor)

  - Tubular *axon* (outputs)

  - *Synapses* (connections between neurons)

Biological Neural Networks
(continued)





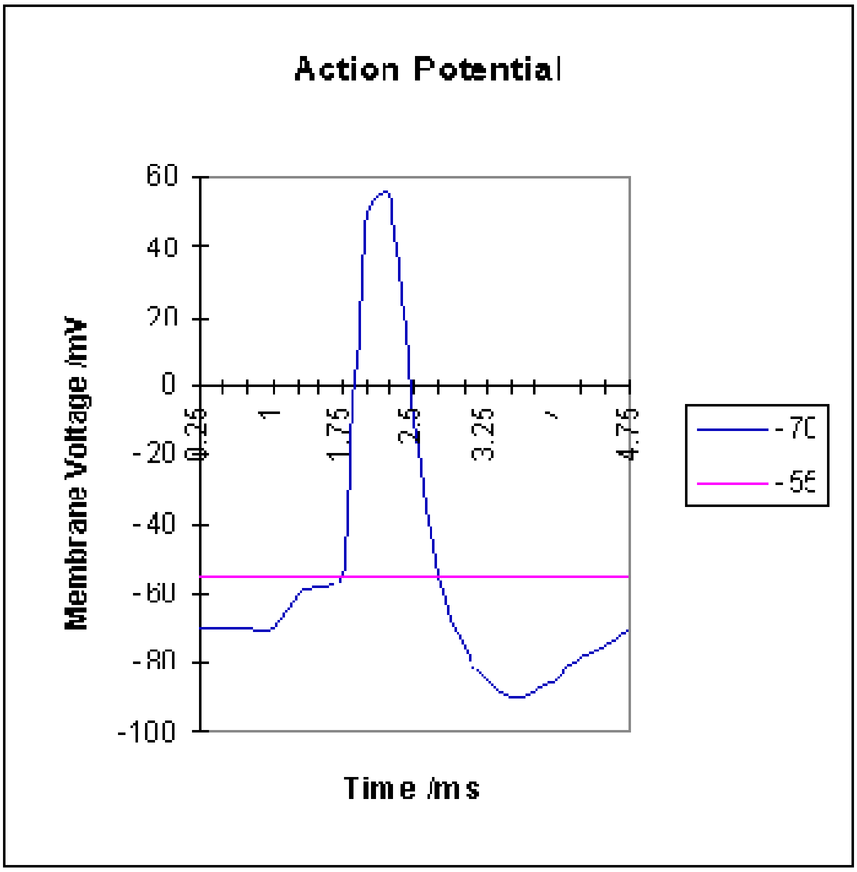
Biological Neural Networks
(continued)

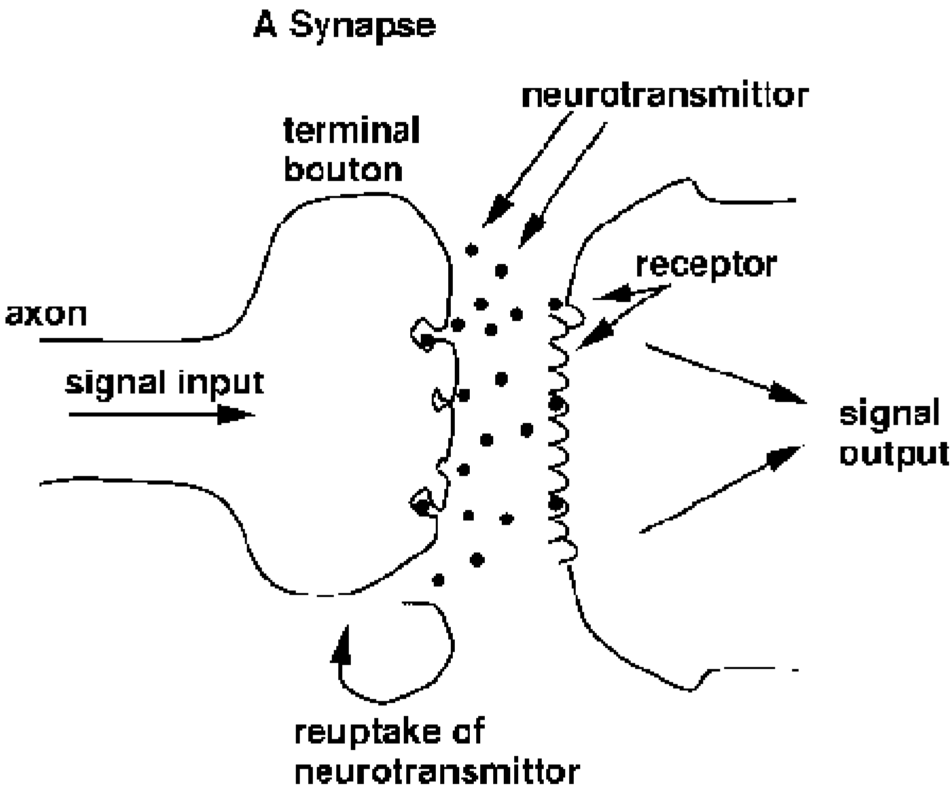
Signal Propagation

- A *resting potential* is maitained at about -65mv across the cell membrane
  - Membrane's inside is negatively charged with potassium ions
  - Membrane's outside is positively charged with sodium ions
- Soma generates an electric impulse
- An *action potential* is created
- Membrane permeability (i.e. polarity) changes
  - Sodium ions migrate to the outside
  - Potassium ions migrate to the inside

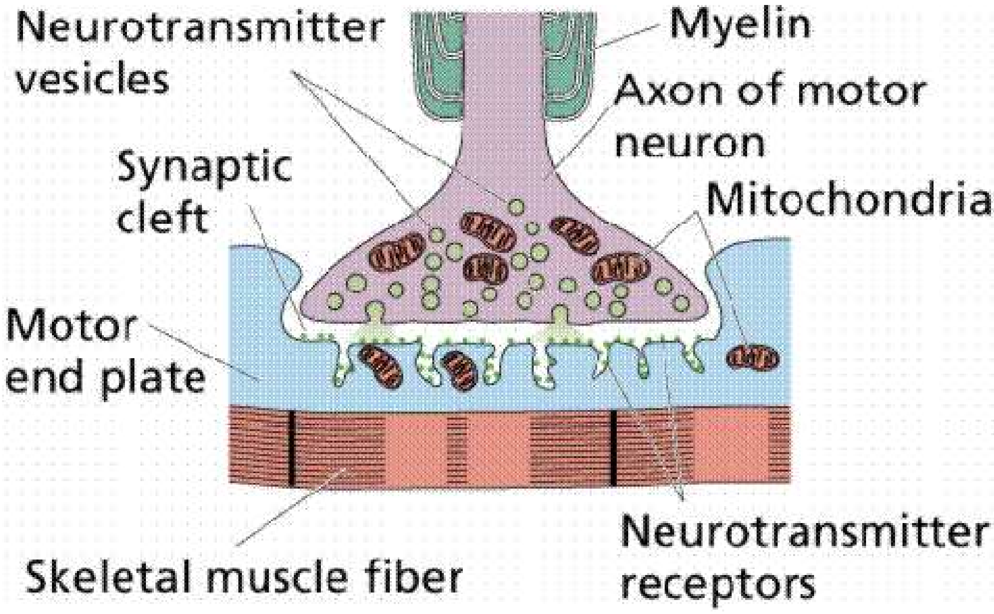
  (membrane attempt to restore its resting potential)
- The signal (action potential) propagates along the axon toward its end

Biological Neural Networks
(continued)





Biological Neural Networks
(continued)



Signal Transmission

- An action potential reaches the synapse
- Vesicles discharge their contents into the synaptic cleft
- Neurotransmitters diffuse accross the cleft and bind to receptors of post-synaptic neuron
- The neurotransmitters, depending on their qualities and quantities, may cause an action potential on the post-synaptic neuron

Biological Neural Networks
(continued)

Synapse

- Excitatory: signal is amplified
- Inhibitory: signal is weakened
- Magnitude of received signal depends on the synaptic *efficiency*

Post-synaptic neuron

- There is action potential when the neuron is sufficiently excited
- *Fires* if its net (i.e. total) excitation exceeds its inhibition by a critical amount called the neuron's *threshold*
- The neuron continues to firing at certain frequency if input(s) are remain(s) strong
- Output of neuron = firing frequency

Biological Neural Networks
(continued)

Synaptic Learning

- Brain learns by
  - altering synaptic efficiencies
  - creating new synapses
- Efficacy of a synapse can change as a result experience
- *Hebbs Postulate*: "...When an axon of a cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic changes take place in one or both cells such that A's efficiency as one of the cells firing B, is increased ..."
- Therefore knowledge (i.e. memory) is
  - stored at the synapses
  - represented in the form of synaptic efficiencies
  - acquired through a synaptic learning process

Brain vs. Computer
(non exhaustive)

Energy efficiency

- Brain: 10^{-16} joules
- Computer: 10^{-6} joules

Speed

- Brain: 10^{-3} seconds
- Computer: 10^{-9} seconds

Style of computation

- Brain: parallel, distributed
- Computer: serial, centralized

Memory

- Brain: content-addressable, distributed
- Computer: non-associative, centralized

Artificial Neural Networks

A artificial neural network is a massively parallel distributed processor made up of simple processing units, which has a natural propensity for storing experiential knowledge and making it available for use. It resembles the brain in two respects

1. Knowledge is acquired by the network from its environment through a learning process.

2. Interneuron connection strengths, known as synaptic weights, are used to store the acquired knowledge

Structure

- Set of nodes, called *neurons*

- Connectivity

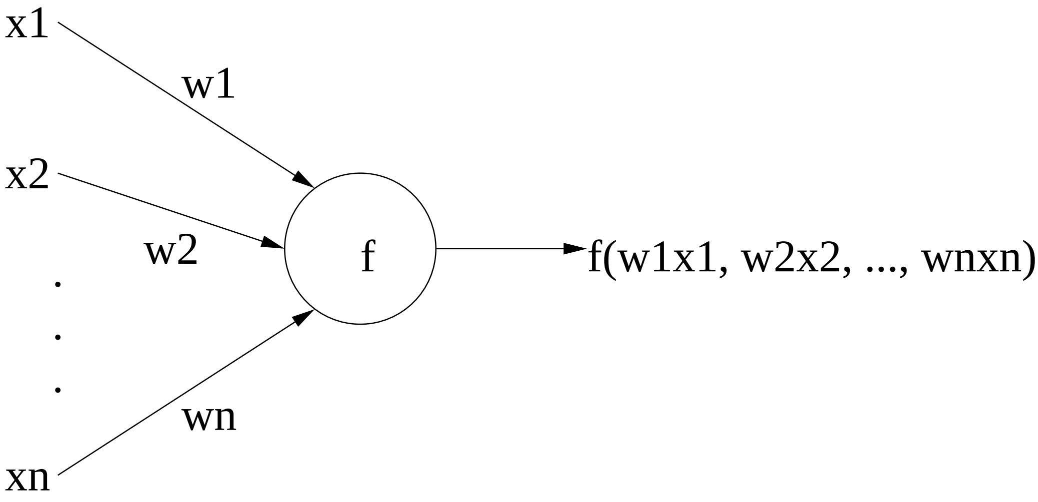
Behavior

- Learning algorithm: process that modifies the network's connectivity

- Functionality: function or task performed by the network

Artificial Neuron Models

General Neuron Model



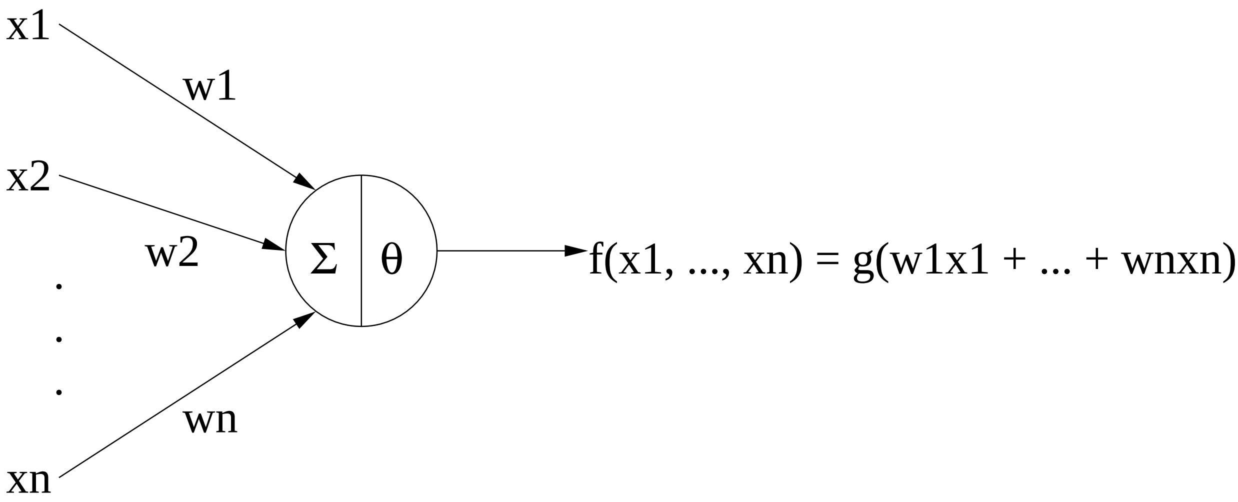
Where

- $\vec{x} = (x_1, x_2, \ldots, x_n)$: input vector
- $\vec{w} = (w_1, w_2, \ldots, w_n)$: weight vector
- $f(w_1x_1, w_2x_2, \ldots, w_nx_n)$: neuron's output
- f : function of the neuron

Biological Terminology	Artificial Terminology
Neuron	Node/Unit/Cell/Neurode
Synapse	Connection/Edge/Link
Synaptic Efficiency	Connection Weight
Firing Frequency	Node Output

Artificial Neuron Models
(continued)

Weighted Input Summation Model



Where

- $\vec{x} = (x_1, x_2, \ldots, x_n)$: input vector
- $\vec{w} = (w_1, w_2, \ldots, w_n)$: weight vector
- $\vec{w}\vec{x} = w_1 x_1 + \cdots + w_n x_n$: net input to neuron
- g : activation function
- $f(x_1, x_2, \ldots, x_n)$: output of the neuron
- f : function of the neuron
- θ : threshold level of the neuron

Types of Activation Function

- Constant function $g(\vec{w}\vec{x}) = c$

- Identity function $g(\vec{w}\vec{x}) = \vec{w}\vec{x}$

- Non-negative identity $g(\vec{w}\vec{x}) = \max(0, \vec{w}\vec{x})$

- Threshold function $g(\vec{w}\vec{x}) = \begin{cases} 0 & \text{if } \vec{w}\vec{x} < \theta \\ 1 & \text{if } \vec{w}\vec{x} \geq \theta \end{cases}$

- Ramp function $g(\vec{w}\vec{x}) = \begin{cases} 0 & \text{if } \vec{w}\vec{x} < \theta_1 \\ \frac{\vec{w}\vec{x} - \theta_1}{\theta_2 - \theta_1} & \text{if } \theta_1 \leq \vec{w}\vec{x} \leq \theta_2 \\ 1 & \text{if } \vec{w}\vec{x} > \theta_2 \end{cases}$

- Piecewise-linear function

- Sigmoid function

  - Logistic $g(\vec{w}\vec{x}) = \frac{1}{1+e^{-\lambda \cdot \vec{w}\vec{x}}}$

  - Hyperbolic tangent $g(\vec{w}\vec{x}) = \tanh(\lambda \cdot \vec{w}\vec{x})$

- Gaussian function $g(\vec{w}\vec{x}) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(\frac{\vec{w}\vec{x} - \mu}{\sigma})^2}$

Neural Network Architectures

- Architecture

  1. Topology

     – Physical connectivity of the nodes

     – Structure of the nodes

       * Net input

       * Activation function

     Topology determines the functions or tasks performed by the neural network

  2. Learning algorithm

     – Computation of the connection weights

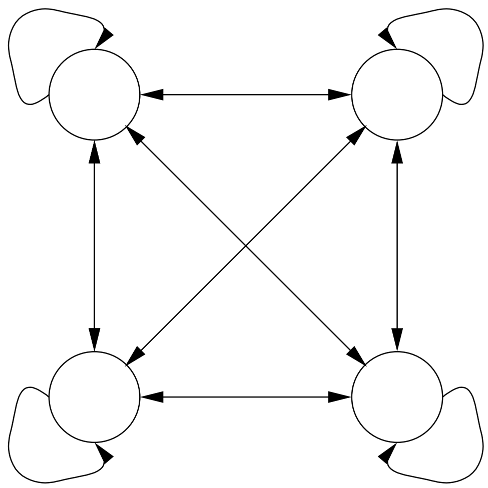
     – Same topology used for many tasks

     – Adaptive topology

     Learning determines the ability of the network to update its weights in order to perform a given task or function

Neural Network Architectures
(continued)

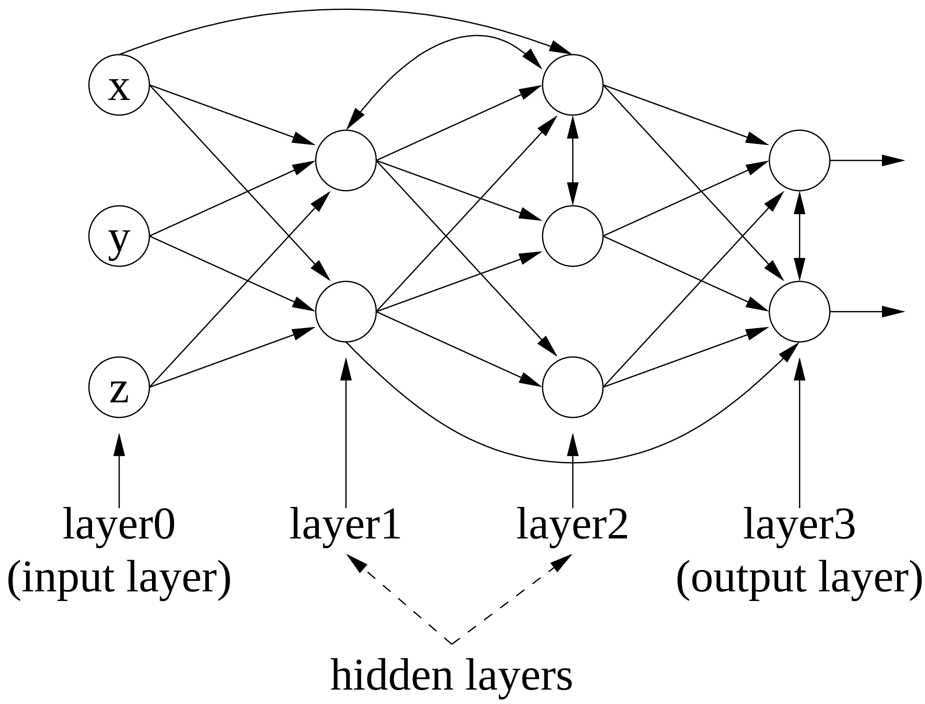
Fully connected networks



- Complete graph, most general architecture
- Symmetric or asymmetric
- Seldom used since n nodes $\rightarrow n^2$ weights
- Complex learning
- Biologically implausible

Neural Network Architectures
(continued)

Layered networks



- Possible intra-layer connections
- Possible self-connections
- Possible cycles
- Acyclic networks: No intra-layer connections
- Feedforward networks: Connections only from layer i to layer $i+1$
- Recurrent networks: Some feedback connections

Neural Network Learning

Learning is a process by which the free parameters of a neural network are adapted through a process of stimulation by the environment in which the network is embedded. The type of learning is determined by the manner in which the parameter changes take place, [Mendel and McLaren, 1970].

1. Neural network is *stimulated* by an environment

2. Neural network *undergoes changes* in its free parameters as a result of this simulation

3. Neural network *responds in a new way* to the environment because of the changes that have occured in its internal structure

- *Learning algorithm*: prescribed set of rules for the solution of a learning problem

- *Learning paradigm*: manner in which a neural network relates to its environment

Five Basic Learning Rules

Error-correction learning Minimize error $e(\vec{w}_t)$ between desired response $d(\vec{w})$ and actual response $a(\vec{w}_t)$. More formally, find $\vec{w}_t$ such that $e(\vec{w}_t) = d(\vec{w}_t) - a(\vec{w}_t) = 0$ or is as small as possible.

- Good objective function: ex: $\xi(\vec{w}_t) = \frac{1}{2}e^2(\vec{w}_t)$
- Basic rule: $\vec{w}_{t+1} = \vec{w}_t + \eta e(\vec{w}_t)\vec{x}$

Memory-based learning Minimize distance between past experience and new experience. Past experiences are explicitly stored in a memory of positive examples. Retrieves the experience that is closest to the new experience.

- Good distance function: ex: Euclidean distance
- Basic rule: *nearest neighbor rule*

Hebbian learning Basis of associative learning and memory. Increased activities between two connected neurons (or brain processes) correspond to increased association between two patterns.

- Basic rule: $\vec{w}_{t+1} = \vec{w}_t + \eta\vec{y}\vec{x}$

Five Basic Learning Rules
(continued)

Competitive learning For a given input, neurons compete to be *winners* with high level of activity (i.e. only a single neuron is active at any one time). Each node learns to specialize on ensembles of similar patterns.

- Good winning mechanism: ex: largest net input
- Basic rule: $\vec{w}_{t+1} = \vec{w}_t + \begin{cases} \eta(\vec{x} - \vec{w}_t) & \text{if winner} \\ 0 & \text{if loser} \end{cases}$

Learning Paradigms

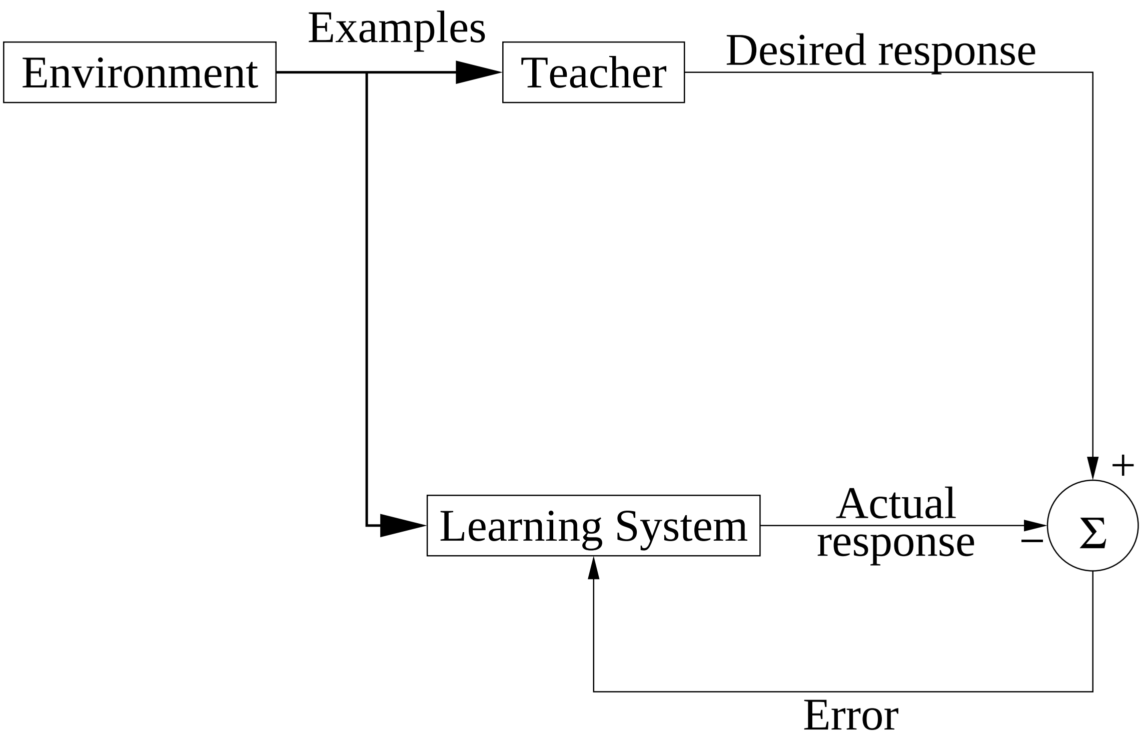
Credit assignment problem Who to blame for overall outcomes?

- Outcomes depend on internal decisions
- Relevant only in multicomponent system
- Operation of hidden neuron is important
- How to assign credit or blame for the action of hidden neurons?

Learning Paradigms
(continued)

Learning with a teacher (or supervised learning)

- Teacher has knowledge of the environment
- Knowledge representation: *input-output examples*
- Environment *unknown* to neural network
- Neural network learns from given example
- Network is to *emulate* teacher through training and error minimization



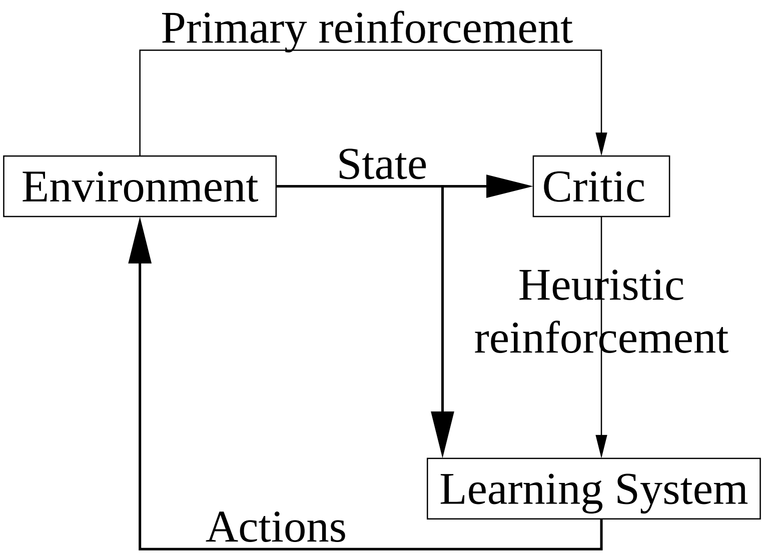
Learning Paradigms
(continued)

Learning without a teacher • No teacher oversees the learning process

• No *labelled* examples of task to learn

1 Reinforcement learning (or neurodynamic learning)

- A critic observes and reinforces the performance of the network
- Network learns to perform a task by discovering actions that lead to good performance
- Network performs a *temporal credit assignment*
- Network minimizes a *cost-to-go function*



Learning Paradigms
(continued)

Learning without a teacher (continued)

2 Unsupervised learning (or self-organized learning)

- No teacher, no critic
- Network discovers statistical regularities of the environment
- Network develops ability to form internal representations for encoding features of the inputs
- *Task-independent measure* of quality

Learning Tasks

- Pattern association
  - Autoassociation, Heteroassociation
- Pattern recognition
  - Classification
  - Clustering
- Function approximation
  - System identification
  - Inverse system
- Prediction
  - Generalization
- Control
- Filtering
- Beamforming

Knowledge Representation

Knowledge refers to stored information or models used by a person or machine to interpret, predict, and appropriately respond to the outside world, [Fischler and Firschein, 1987].

Characteristics

1. What information is actually made explicit
2. How information is physically encoded for subsequent use

Neural network learns a model of the world whose knowledge consists of

1. Known world state
2. Observations of the world
   - Inherently noisy
   - Subject to errors and imperfections
   - Provide *examples* for neural network

Knowledge Representation
(continued)

The set of examples represents knowledge about the environment

- Positive or negative
- Labelled or unlabelled

Design of neural network based on real-life data

Knowledge representation of environment is defined by the values of the free parameters of the neural networks

Importance of the network architecture

- Optimal structure for representation?
- Optimal structure for learning?
- Generalizability?

Internal Representation
(continued)

Rule 1 Similar inputs from similar classes should usually produce similar representations inside the network, and should therefore be classified as belonging to the same category

Rule 2 Items to be categorized as separate classes should be given widely different representations in the networks

Rule 3 If a particular feature is important, then there should be a large number of neurons involved in the representation of that item in the network

Rule 4 Prior information and invariances should be built into the design of the network, thereby simplifying the network design by not having to learn them