Introduction To Neural Networks

- Development of Neural Networks date back to the early 1940s. It experienced an upsurge in popularity in the late 1980s. This was a result of the discovery of new techniques and developments and general advances in computer hardware technology.
- Some NNs are models of biological neural networks and some are not, but historically, much of the inspiration for the field of NNs came from the desire to produce artificial systems capable of sophisticated, perhaps ``intelligent", computations similar to those that the human brain routinely performs, and thereby possibly to enhance our understanding of the human brain.
- Most NNs have some sort of “training" rule. In other words, NNs “learn" from examples (as children learn to recognize dogs from examples of dogs) and exhibit some capability for generalization beyond the training data.

Neural Network Techniques

- Computers have to be explicitly programmed
  - Analyze the problem to be solved.
  - Write the code in a programming language.
- Neural networks learn from examples
  - No requirement of an explicit description of the problem.
  - No need a programmer.
  - The neural computer to adapt itself during a training period, based on examples of similar problems even without a desired solution to each problem. After sufficient training the neural computer is able to relate the problem data to the solutions, inputs to outputs, and it is then able to offer a viable solution to a brand new problem.
  - Able to generalize or to handle incomplete data.

NNs vs Computers

Digital Computers

- Deductive Reasoning. We apply known rules to input data to produce output.
- Computation is centralized, synchronous, and serial.
- Memory is packetted, literally stored, and location addressable.
- Not fault tolerant. One transistor goes and it no longer works.
- Exact.
- Static connectivity.

- Applicable if well defined rules with precise input data.

Neural Networks

- Inductive Reasoning. Given input and output data (training examples), we construct the rules.
- Computation is collective, asynchronous, and parallel.
- Memory is distributed, internalized, and content addressable.
- Fault tolerant, redundancy, and sharing of responsibilities.
- Inexact.
- Dynamic connectivity.

- Applicable if rules are unknown or complicated, or if data is noisy or partial.

Evolution of Neural Networks

- Realized that the brain could solve many problems much easier than even the best computer
  - image recognition
  - speech recognition
  - pattern recognition

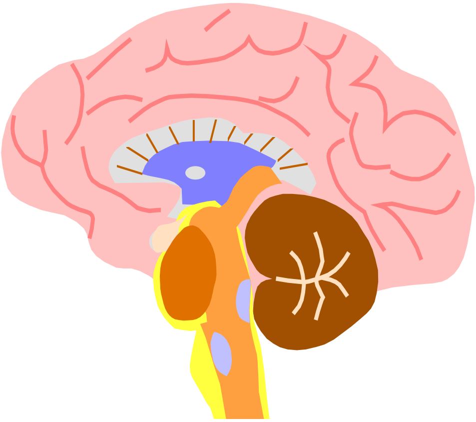
Very easy for the brain but very difficult for a computer

Evolution of Neural Networks

- Studied the brain
  - Each neuron in the brain has a relatively simple function
  - But - 10 billion of them (60 trillion connections)
  - Act together to create an incredible processing unit
  - The brain is trained by its environment
  - Learns by experience

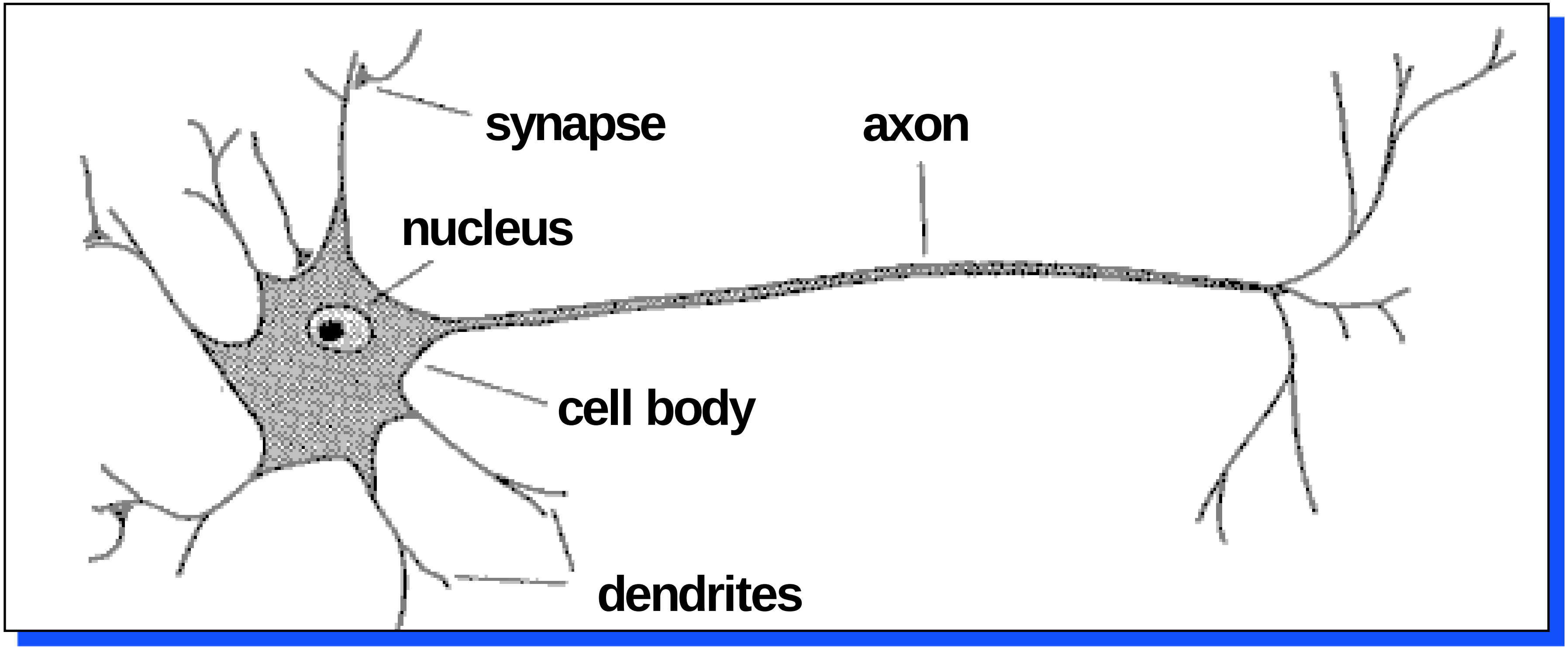
Compensates for problems by massive parallelism

The Biological Inspiration



- The brain has been extensively studied by scientists.
- Vast complexity prevents all but rudimentary understanding.
- Even the behaviour of an individual neuron is extremely complex
- Engineers modified the neural models to make them more useful
  - less like biology
  - kept much of the terminology

The Structure of Neurons



A neuron has a cell body, a branching input structure (the dendrite) and a branching output structure (the axon)

- Axons connect to dendrites via synapses.
- Electro-chemical signals are propagated from the dendritic input, through the cell body, and down the axon to other neurons

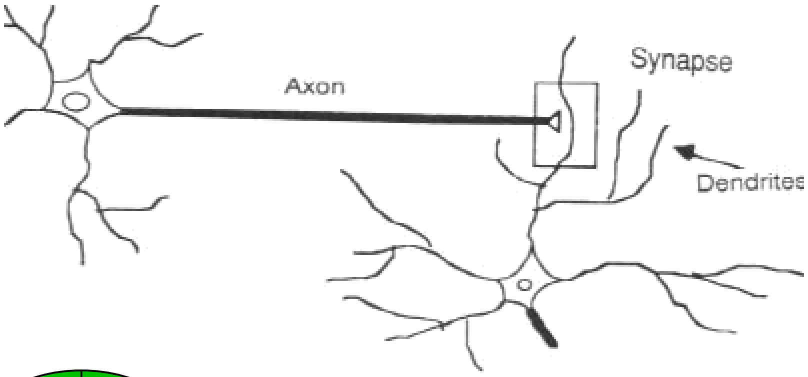
The Structure of Neurons

- A neuron only fires if its input signal exceeds a certain amount (threshold) in a short time period.

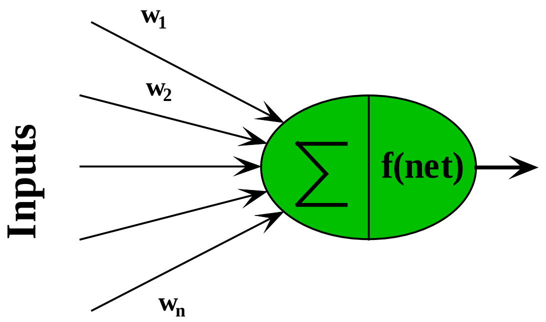
- Synapses vary in strength
  - Good connections allowing a large signal
  - Slight connections allow only a weak signal.
  - Synapses either:
    - Excitatory (stimulate)
    - Inhibitory (restrictive)

Biological Analogy

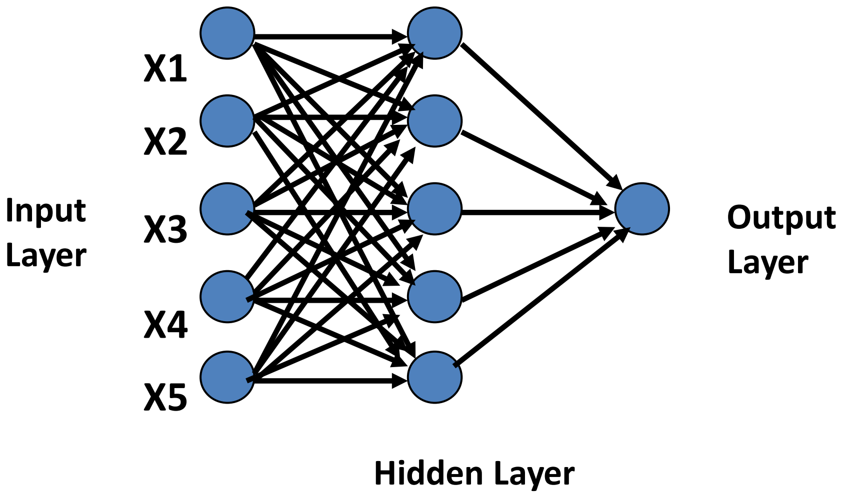
- Brain Neuron



- Artificial neuron (processing element)



- Set of processing elements (PEs) and connections (weights) with adjustable strengths

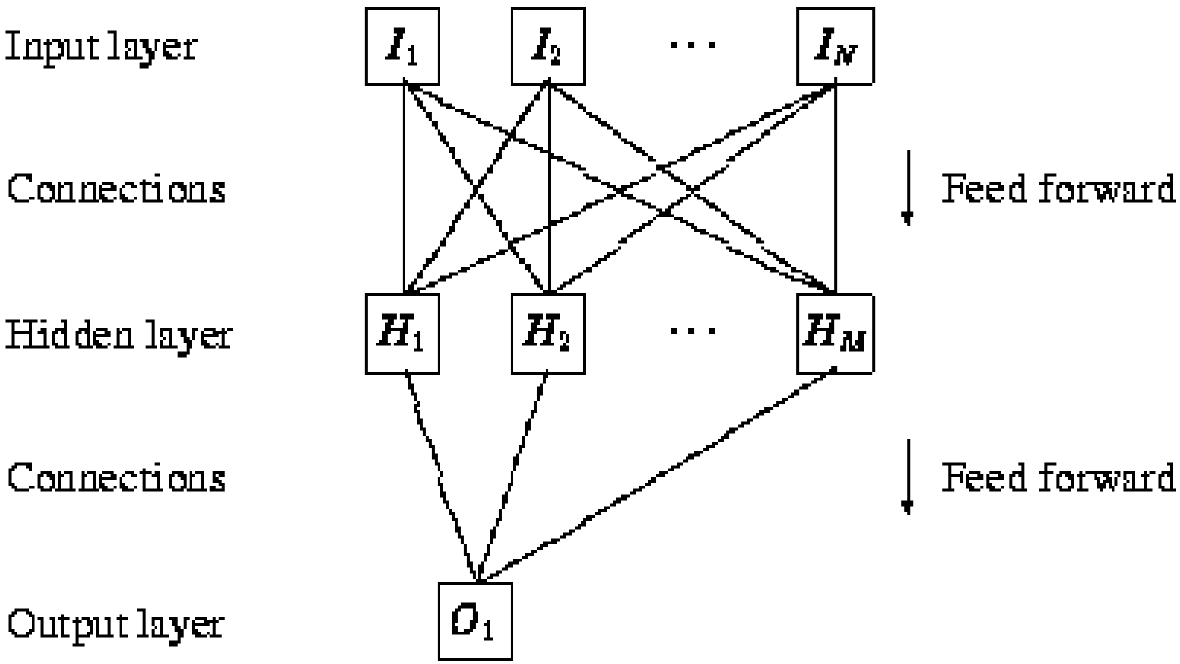


Benefits of Neural Networks

- Pattern recognition, learning, classification, generalization and abstraction, and interpretation of incomplete and noisy inputs
- Provide some human problem-solving characteristics
- Robust
- Fast, flexible and easy to maintain
- ***Powerful hybrid systems***

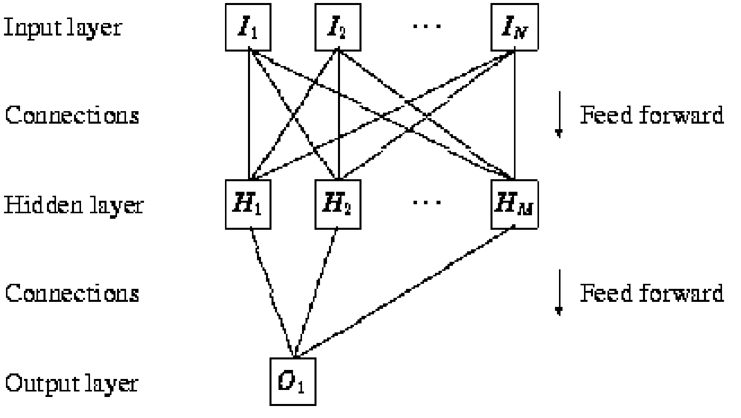
(Artificial) Neural networks (ANN)

- ANN architecture



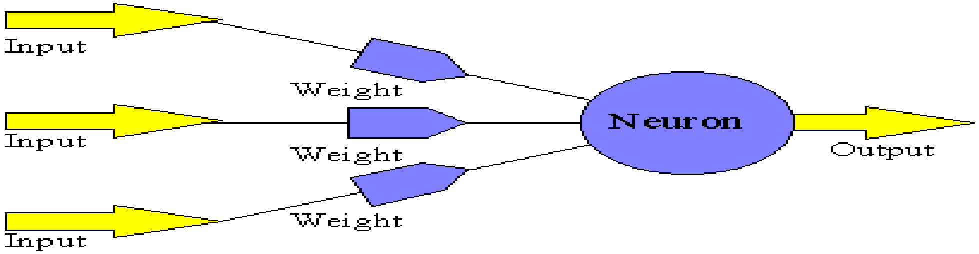
(Artificial) Neural networks (ANN)

- 'Neurons'
  - have 1 output but many inputs
  - Output is weighted sum of inputs
  - Threshold can be set
    - Gives non-linear response



The Key Elements of Neural Networks

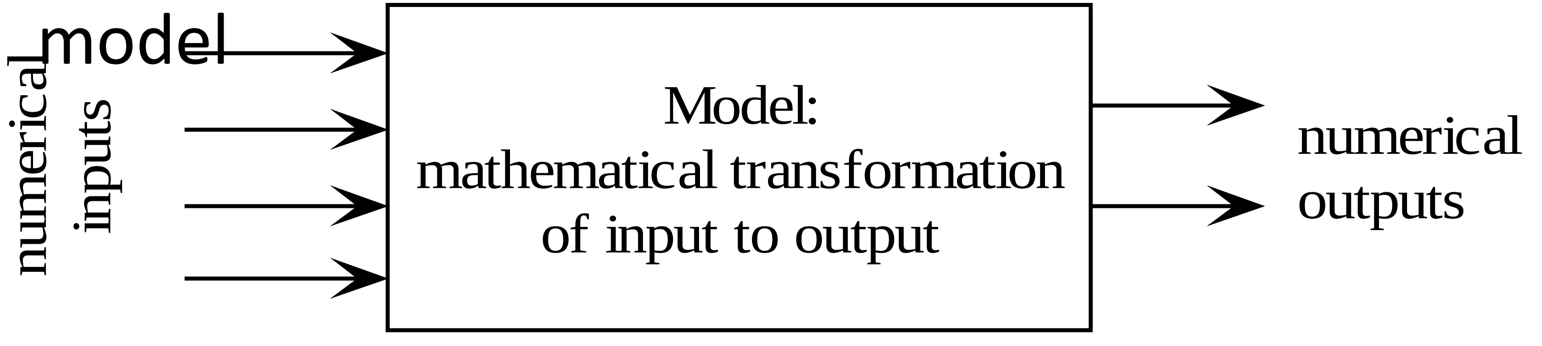
- Neural computing requires a number of neurons, to be connected together into a "neural network". Neurons are arranged in layers.



- Each neuron within the network is usually a simple processing unit which takes one or more inputs and produces an output. At each neuron, every input has an associated "weight" which modifies the strength of each input. The neuron simply adds together all the inputs and calculates an output to be passed on.

What is a Artificial Neural Network

- The neural network is:
  - model
  - nonlinear (output is a nonlinear combination of inputs)
  - input is numeric
  - output is numeric
  - pre- and post-processing completed separate from model



(Artificial) Neural networks (ANN)

- Training
  - Initialize weights for all neurons
  - Present input layer with e.g. spectral reflectance
  - Calculate outputs
  - Compare outputs with e.g. biophysical parameters
  - Update weights to attempt a match
  - Repeat until all examples presented

Training methods

- **Supervised learning**

  In supervised training, both the inputs and the outputs are provided. The network then processes the inputs and compares its resulting outputs against the desired outputs. Errors are then propagated back through the system, causing the system to adjust the weights which control the network. This process occurs over and over as the weights are continually tweaked. The set of data which enables the training is called the "training set." During the training of a network the same set of data is processed many times as the connection weights are ever refined.

  Example architectures : Multilayer perceptrons

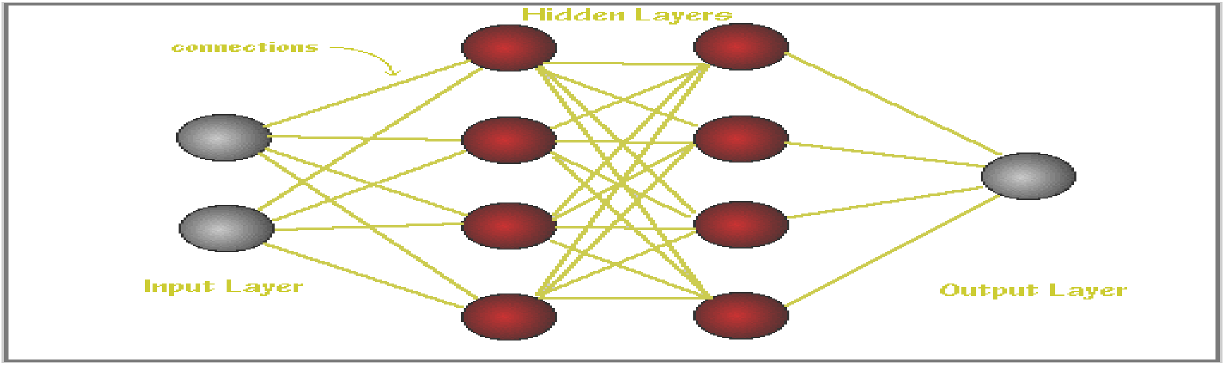
- **Unsupervised learning**

  In unsupervised training, the network is provided with inputs but not with desired outputs. The system itself must then decide what features it will use to group the input data. This is often referred to as self-organization or adaption. At the present time, unsupervised learning is not well understood.

  Example architectures : Kohonen, ART

Feedforword NNs

- The basic structure off a feedforward Neural Network

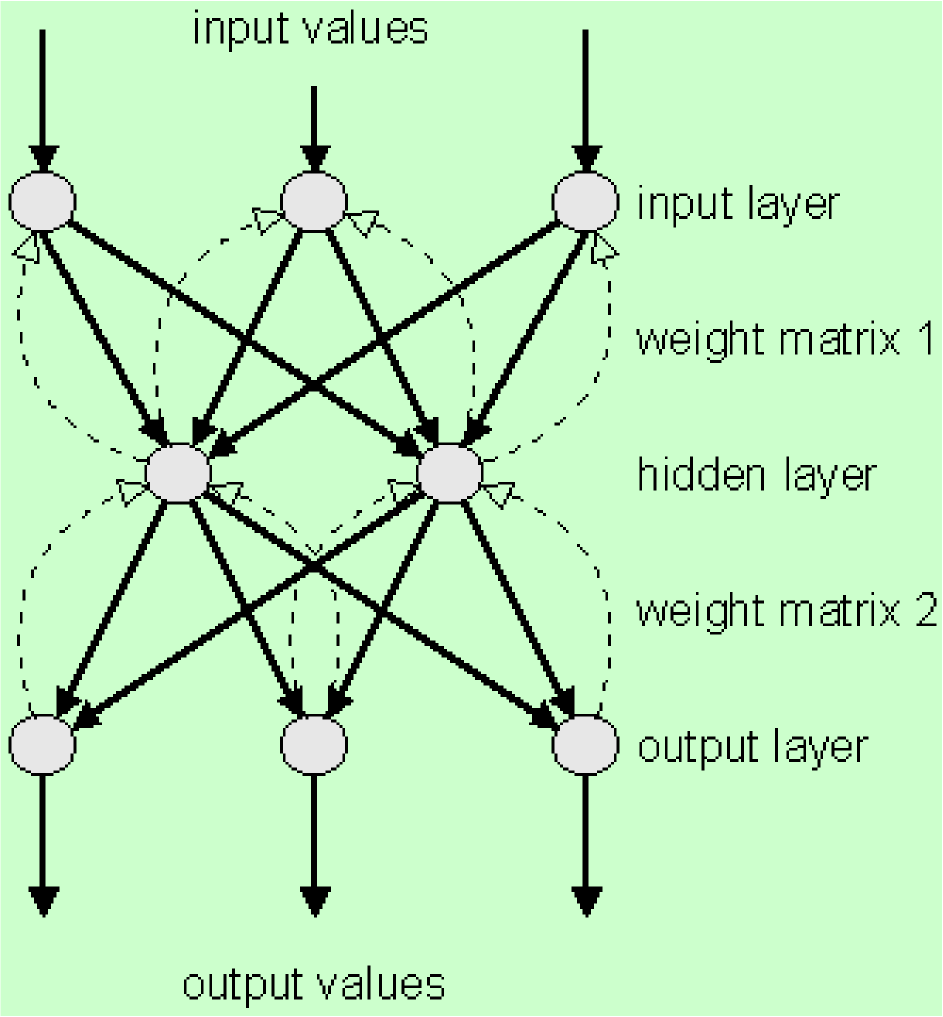


- The 'learning rule” modifies the weights according to the input patterns that it is presented with. In a sense, ANNs learn by example as do their biological counterparts.
- When the desired output are known we have supervised learning or learning with a teacher.

An overview of the backpropagation

1. A set of examples for training the network is assembled. Each case consists of a problem statement (which represents the input into the network) and the corresponding solution (which represents the desired output from the network).
2. The input data is entered into the network via the input layer.
3. Each neuron in the network processes the input data with the resultant values steadily "percolating" through the network, layer by layer, until a result is generated by the output layer.
4. The actual output of the network is compared to expected output for that particular input. This results in an *error value* which represents the discrepancy between given input and expected output. On the basis of this error value an of the connection weights in the network are gradually adjusted, working backwards from the output layer, through the hidden layer, and to the input layer, until the correct output is produced. Fine tuning the weights in this way has the effect of teaching the network how to produce the correct output for a particular input, i.e. the network *learns*.

Backpropagation Network



Neural Network Terminology

- <u>ANN</u> - artificial neural network
- <u>PE</u> - processing element (neuron)
- <u>Exemplar</u> - one individual set of input/output data
- <u>Epoch</u> - complete set of input/output data
- <u>Weight</u> - the adjustable parameter on each connection that scales the data passing through it

Types of Layers

- The input layer
  - Introduces input values into the network
  - No activation function or other processing

- The hidden layer(s)
  - Perform classification of features
  - Two hidden layers are sufficient to solve any problem
  - Features imply more layers may be better

- The output layer.
  - Functionally just like the hidden layers
  - Outputs are passed on to the world outside the neural network.