Neural Networks

MSE 2400 EaLiCaRA
Dr. Tom Way

Background

Neural Networks can be :

- **Biological** models
- **Artificial** models

Desire to produce artificial systems capable of sophisticated computations similar to the human brain.

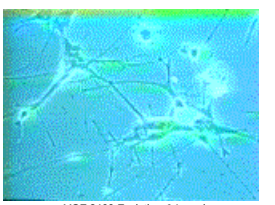
Biological analogy and some main ideas

- The brain is composed of a mass of interconnected neurons
  - each neuron is connected to many other neurons
- Neurons transmit signals to each other
- Whether a signal is transmitted is an all-or-nothing event (the electrical potential in the cell body of the neuron is thresholded)
- Whether a signal is sent, depends on the strength of the bond (synapse) between two neurons

How Does the Brain Work ?

NEURON

- The cell that performs information processing in the brain.
- Fundamental functional unit of all nervous system tissue.



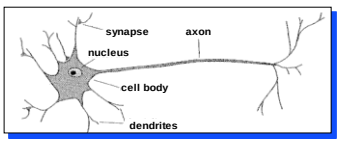
Brain vs. Digital Computers

- Computers require hundreds of cycles to simulate a firing of a neuron.
- The brain can fire all the neurons in a single step.
  ⇒ Parallelism
- Serial computers require billions of cycles to perform some tasks but the brain takes **less than a second.**
  e.g. Face Recognition

Comparison of Brain and computer

	Human	*Computer*
Processing Elements	100 Billion neurons	10 Million gates
Interconnects	1000 per neuron	A few
Cycles per sec	1000	500 Million
2X improvement	200,000 Years	2 Years

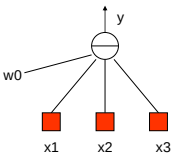
Biological neuron



- A neuron has
  - A branching input (dendrites)
  - A branching output (the axon)
- The information circulates from the dendrites to the axon via the cell body
- Axon connects to dendrites via synapses
  - Synapses vary in strength
  - Synapses may be excitatory or inhibitory

What is an artificial neuron ?

- Definition : Non linear, parameterized function with restricted output range



$$y = f\left(w_0 + \sum_{i=1}^{n-1} w_i x_i \right)$$

Learning in Neural Networks

- The procedure that consists in estimating the parameters of neurons so that the whole network can perform a specific task
- 2 types of learning
  - The supervised learning
  - The unsupervised learning
- The Learning process (supervised)
  - Present the network a number of inputs and their corresponding outputs
  - See how closely the actual outputs match the desired ones
  - Modify the parameters to better approximate the desired outputs

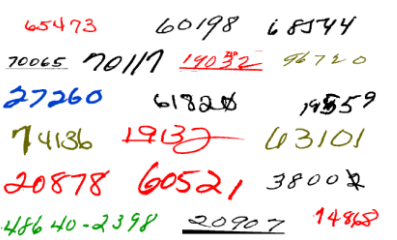
Supervised learning

- The desired response of the neural network in function of particular inputs is well known.
- A “Professor” may provide examples and teach the neural network how to fulfill a certain task

Unsupervised learning

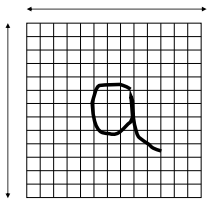
- Idea : group typical input data in function of resemblance criteria un-known a priori
- Data clustering
- No need of a professor
  - The network finds itself the correlations between the data
  - Examples of such networks :
    - Kohonen feature maps

Zipcode Example



Examples of handwritten postal codes
drawn from a database available from the US Postal service

Character recognition example



- Image 256x256 pixels
- 8 bits pixels values (grey level)

$2^{256\times256\times8} \approx 10^{158000}$ different images

- Necessary to extract features

Neural Networks

- A mathematical model to solve engineering problems
  - Group of highly connected neurons to realize compositions of non linear functions
- Tasks
  - Classification
  - Discrimination
  - Estimation
- 2 types of networks
  - Feed forward Neural Networks
  - Recurrent Neural Networks

Feed-forward Networks

- Arranged in ***layers***.

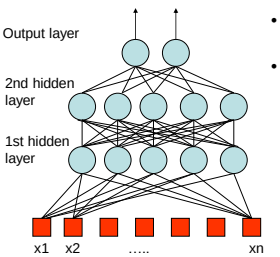
- Each unit is linked only in the unit in next layer.

- No units are linked between the same layer, back to the previous layer or skipping a layer.

- Computations can proceed uniformly from input to output units.

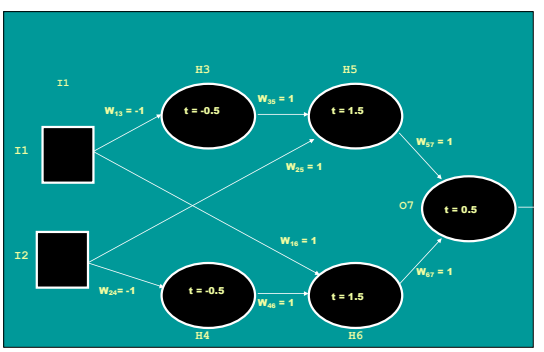
- No internal state exists.

Feed Forward Neural Networks



- Information is propagated from inputs to outputs
- Can pass through one or more hidden layers

Feed-Forward Example



Recurrent Network (1)

- The brain is not and cannot be a feed-forward network.

- Allows activation to be fed back to the previous unit.

- Internal state is stored in its activation level.

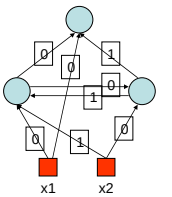
- Can become unstable

-Can oscillate.

Recurrent Network (2)

- May take long time to compute a stable output.
- Learning process is much more difficult.
- Can implement more complex designs.
- Can model certain systems with internal states.

Recurrent Neural Networks



- Can have arbitrary topologies
- Can model systems with internal states (dynamic ones)
- Delays are associated to a specific weight
- Training is more difficult
- Performance may be problematic
  - Stable Outputs may be more difficult to evaluate
  - Unexpected behavior (oscillation, chaos, …)

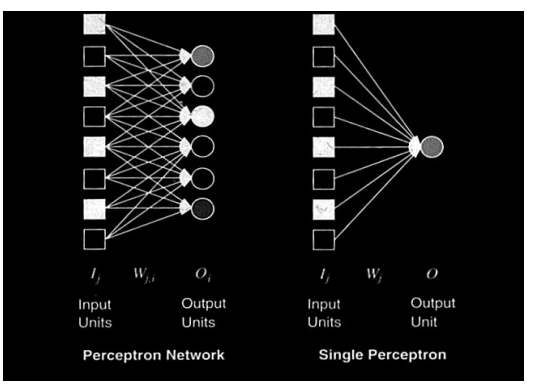
Multi-layer Networks and **Perceptrons**

- Have one or more layers of hidden units.
- With two possibly very large hidden layers, it is **possible to implement any function**.

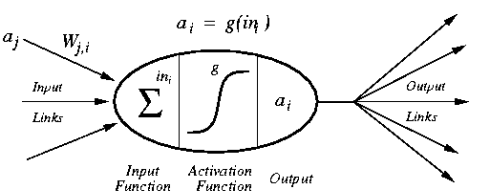
- Networks without hidden layer are called perceptrons.
- Perceptrons are very limited in what they can represent, but this makes their learning problem much simpler.

Perceptrons

- First studied in the late 1950s.
- Also known as Layered Feed-Forward Networks.
- The only efficient learning element at that time was for single-layered networks.
- Today, used as a synonym for a single-layer, feed-forward network.



Perceptrons



What can Perceptrons Represent?

Some complex Boolean function can be represented.

For example:
Majority function - will be covered in this lecture.

Perceptrons are limited in the Boolean functions they can represent.

Need for **hidden units**

- If there is one layer of enough hidden units, the input can be recoded (perhaps just memorized)
- This recoding allows any mapping to be represented
- Problem: how can the weights of the hidden units be trained?

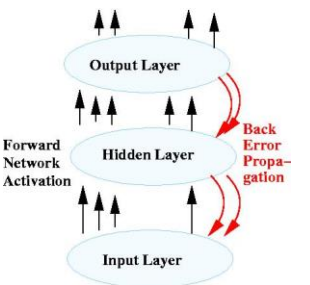
Backpropagation

- In 1969 a method for learning in multi-layer network, Backpropagation, was invented by Bryson and Ho.

- The Backpropagation algorithm is a sensible approach for dividing the contribution of each weight.

- Works basically the same as perceptrons

Backpropagation flow



Backpropagation Network training

- 1. Initialize network with random weights
- 2. For all training cases (called examples):
  - **a.** Present training inputs to network and calculate output
  - **b.** For all layers (starting with output layer, back to input layer):
    - i. Compare network output with correct output (error function)
    - ii. Adapt weights in current layer

This is what you want

Backpropagation Algorithm – Main Idea – error in hidden layers

The ideas of the algorithm can be summarized as follows :

1. Computes the **error term for the output units** using the observed error.

2. From output layer, repeat
   - propagating the error term back to the previous layer and
   - updating the weights between the two layers

   until the earliest hidden layer is reached.

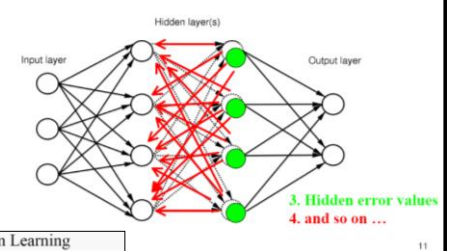
How many hidden layers?

- Usually just one (i.e., a 2-layer net)
- How many hidden units in the layer?
  - Too few ==> can't learn
  - Too many ==> poor generalization

How big a training set?

- Determine your target error rate, *e*
- Success rate is 1- *e*
- Typical training set approx. n/*e*, where n is the number of weights in the net
- Example:
  - *e* = 0.1, n = 80 weights
  - training set size 800
    trained until 95% correct training set classification
    should produce 90% correct classification
    on testing set (typical)

Backpropagation Learning



11

Backpropagation Learning

$$E_{out\,i} = d_{out\,i} - out_i$$

$$E_{total} = \sum_{i=0}^{numOut} E_{out\,i}^2$$

$$E_{hid\,i} = \sum_{k=1}^{numOut} E_{out\,k} \cdot w_{out\,ik}$$

$$diff_{hid\,i} = E_{hid\,i} \cdot (1 - o(n_{hid\,i})) \cdot o(n_{hid\,i})$$

Summary

- Neural network is a computational model that simulate some properties of the human brain.
- The connections and nature of units determine the behavior of a neural network.
- Perceptrons are feed-forward networks that can only represent linearly separable (very simple) functions.

Summary (cont'd)

- Given enough units, any function can be represented by Multi-layer feed-forward networks.
- Backpropagation learning works on multi-layer feed-forward networks.
- Neural Networks are widely used in developing artificial learning systems.