Artificial Neural Networks

Computational models inspired by the human brain:

- Massively parallel, distributed system, made up of simple processing units (neurons)

- Synaptic connection strengths among neurons are used to store the acquired knowledge.

- Knowledge is acquired by the network from its environment through a learning process
Applications of ANNs

ANNs have been widely used in various domains for:

- Pattern recognition
- Associative memory
- Function approximation
Artificial Neural Networks

Early ANN Models:
- Perceptron, ADALINE, Hopfield Network

Current Models:
- Multilayer feedforward networks (Multilayer perceptrons)
- Radial Basis Function networks
- Self Organizing Networks
- ...
Applications

Aerospace
- High performance aircraft autopilots, flight path simulations, aircraft control systems, autopilot enhancements, aircraft component simulations, aircraft component fault detectors

Automotive
- Automobile automatic guidance systems, warranty activity analyzers

Banking
- Check and other document readers, credit application evaluators

Defense
- Weapon steering, target tracking, object discrimination, facial recognition, new kinds of sensors, sonar, radar and image signal processing including data compression, feature extraction and noise suppression, signal/image identification

Electronics
- Code sequence prediction, integrated circuit chip layout, process control, chip failure analysis, machine vision, voice synthesis, nonlinear modeling
Applications

Financial
- Real estate appraisal, loan advisor, mortgage screening, corporate bond rating, credit line use analysis, portfolio trading program, corporate financial analysis, currency price prediction

Manufacturing
- Manufacturing process control, product design and analysis, process and machine diagnosis, real-time particle identification, visual quality inspection systems, beer testing, welding quality analysis, paper quality prediction, computer chip quality analysis, analysis of grinding operations, chemical product design analysis, machine maintenance analysis, project bidding, planning and management, dynamic modeling of chemical process systems

Medical
- Breast cancer cell analysis, EEG and ECG analysis, prosthesis design, optimization of transplant times, hospital expense reduction, hospital quality improvement, emergency room test advisement
Applications

Robotics
- Trajectory control, forklift robot, manipulator controllers, vision systems

Speech
- Speech recognition, speech compression, vowel classification, text to speech synthesis

Securities
- Market analysis, automatic bond rating, stock trading advisory systems

Telecommunications
- Image and data compression, automated information services, real-time translation of spoken language, customer payment processing systems

Transportation
- Truck brake diagnosis systems, vehicle scheduling, routing systems
Properties of ANNs

Learning from examples
  – labeled or unlabeled

Adaptivity
  – changing the connection strengths to learn things

Non-linearity
  – the non-linear activation functions are essential

Fault tolerance
  – if one of the neurons or connections is damaged, the whole network still works quite well
Properties of ANN Applications

They might be better alternatives than classical solutions for problems characterised by:

- Nonlinearities

- High dimensionality

- Noisy, complex, imprecise, imperfect and/or error prone sensor data

- A lack of a clearly stated mathematical solution or algorithm
Neural Networks Resources
Neural Networks Text Books

Main text books:
“Neural Networks: A Comprehensive Foundation”, S. Haykin (very good -theoretical)
“Pattern Recognition with Neural Networks”, C. Bishop (very good-more accessible)
“Neural Network Design” by Hagan, Demuth and Beale (introductory)

Books emphasizing the practical aspects:
“Neural Smithing”, Reeds and Marks
“Practical Neural Network Recipes in C++”’ T. Masters

Seminal Paper:
“Parallel Distributed Processing” Rumelhart and McClelland et al.

Other:
“Neural and Adaptive Systems”, J. Principe, N. Euliano, C. Lefebvre
Neural Networks Literature

**Review Articles:**


Neural Networks Literature

**Journals:**

- IEEE Transactions on NN
- Neural Networks
- Neural Computation
- Biological Cybernetics

...
Biological Inspirations
Biological Inspirations

Humans perform complex tasks like vision, motor control, or language understanding very well.

One way to build intelligent machines is to try to imitate the (organizational principles of) human brain.
Human Brain

• The brain is a highly complex, non-linear, and parallel computer, composed of some $10^{11}$ neurons that are densely connected ($\sim 10^4$ connection per neuron). *We have just begun to understand how the brain works...*

• A neuron is much slower ($10^{-3}\text{sec}$) compared to a silicon logic gate ($10^{-9}\text{sec}$), however the massive interconnection between neurons make up for the comparably slow rate.
  – Complex perceptual decisions are arrived at quickly (within a few hundred milliseconds)

• **100-Steps rule**: Since individual neurons operate in a few milliseconds, calculations do not involve more than about 100 serial steps and the information sent from one neuron to another is very small (a few bits)

• **Plasticity**: Some of the neural structure of the brain is present at birth, while other parts are developed through learning, especially in early stages of life, to adapt to the environment (new inputs).
Neuron Model and Network Architectures
Biological Neuron

- **dendrites**: nerve fibres carrying electrical signals to the cell
- **cell body**: computes a non-linear function of its inputs
- **axon**: single long fiber that carries the electrical signal from the cell body to other neurons
- **synapse**: the point of contact between the axon of one cell and the dendrite of another, regulating a chemical connection whose strength affects the input to the cell.
Biological Neuron

A variety of different neurons exist (motor neuron, on-center off-surround visual cells...), with different branching structures.

The connections of the network and the strengths of the individual synapses establish the function of the network.
Artificial Neuron Model

\[ x_0 = +1 \]

\[ \sum x_i w_{i0} = b_i \text{ : Bias} \]

Input
Synaptic Weights

Neuron\(^i\)
Activation function
Output

\[ f \]

\[ a_i \]
Bias

\[ a_i = f (n_i) = f \left( \sum_{j=1}^{\infty} w_{ij} x_j + b_i \right) \]

An artificial neuron:
- computes the \textit{weighted sum of its input} (called its net input)
- adds its bias
- passes this value through an activation function

We say that the neuron “\textit{fires}” (i.e. becomes active) if its outputs is above zero.
Bias

Bias can be incorporated as another weight clamped to a fixed input of +1.0

This extra free variable (bias) makes the neuron more powerful.

\[ a_i = f(n_i) = f(\sum w_{ij}x_j) = f(w_i \cdot x_j) \]

\[ j = 0 \]
Activation functions

Also called the squashing function as it limits the amplitude of the output of the neuron.

Many types of activations functions are used:

- linear: \( a = f(n) = n \)

- threshold: \( a = \{1 \text{ if } n \geq 0 \text{ (hardlimiting)} \}
\qquad 0 \text{ if } n < 0 \)

- sigmoid: \( a = 1/(1+e^{-n}) \)
Activation functions: threshold, linear, sigmoid

\[ a = \text{hardlim}(n) \]

Hard Limit Transfer Function

\[ a = \text{purelin}(n) \]

Linear Transfer Function

\[ a = \text{logsig}(n) \]

Log-Sigmoid Transfer Function
## Activation Functions

<table>
<thead>
<tr>
<th>Name</th>
<th>Input/Output Relation</th>
<th>Icon</th>
<th>MATLAB Function</th>
</tr>
</thead>
</table>
| Hard Limit           | \( a = 0 \quad n < 0 \)  
                      | \( a = 1 \quad n \geq 0 \) | ![Icon] | hardlim         |
| Symmetrical Hard Limit | \( a = -1 \quad n < 0 \)   
                      | \( a = +1 \quad n \geq 0 \) | ![Icon] | hardlims        |
| Linear               | \( a = n \)           | ![Icon] | purelin         |
| Saturating Linear    | \( a = 0 \quad n < 0 \)  
                      | \( a = n \quad 0 \leq n \leq 1 \)  
                      | \( a = 1 \quad n > 1 \) | ![Icon] | satlin          |
| Symmetric Saturating Linear | \( a = -1 \quad n < -1 \)        
                      | \( a = n \quad -1 \leq n \leq 1 \)  
                      | \( a = 1 \quad n > 1 \) | ![Icon] | satlins         |
| Log-Sigmoid          | \( a = \frac{1}{1 + e^{-n}} \) | ![Icon] | logsig          |
| Hyperbolic Tangent Sigmoid | \( a = \frac{e^n - e^{-n}}{e^n + e^{-n}} \) | ![Icon] | tansig          |
| Positive Linear      | \( a = 0 \quad n < 0 \)  
                      | \( a = n \quad 0 \leq n \) | ![Icon] | poslin          |
| Competitive          | \( a = 1 \) neuron with max \( n \)  
                      | \( a = 0 \) all other neurons | ![Icon] | compet          |
Artificial Neural Networks

A neural network is a massively parallel, distributed processor made up of simple processing units (artificial neurons).

It resembles the brain in two respects:

- Knowledge is acquired by the network from its environment through a learning process.
- Synaptic connection strengths among neurons are used to store the acquired knowledge.
Different Network Topologies

Single layer feed-forward networks
- Input layer projecting into the output layer
Different Network Topologies

Multi-layer feed-forward networks
- One or more hidden layers.
- Input projects only from previous layers onto a layer. 
  typically, only from one layer to the next

2-layer or
1-hidden layer
fully connected
network

Input layer  Hidden layer  Output layer
Different Network Topologies

Recurrent networks

- A network with feedback, where some of its inputs are connected to some of its outputs (discrete time).
How to Decide on a Network Topology?

- # of input nodes?
  - Number of features
- # of output nodes?
  - Suitable to encode the output representation
- transfer function?
  - Suitable to the problem
- # of hidden nodes?
  - Not exactly known
Multilayer Perceptron

Each layer may have different number of nodes and different activation functions

But commonly:

- **Same activation function within one layer**
  - Sigmoid activation function is used in the hidden units, and
  - sigmoid or linear activation functions are used in the output units depending on the problem (classification-sigmoid or function approximation-linear)