Neural Networks: Module 1 Review Notes

by Dr. Mark Fleischer

1.1 Administrative Issues

- Attendance/registration
- Overview of course
- Preparation for the Feedforward-Backpropagation algorithm
- Mathematical review on multivariate calculus, optimization

Assignments are due the following week after they are handed out. Each student will complete a project and deliver a presentation.

Textbook: *Neural Networks–A Systematic Introduction*, by Raul Rojas

Other Texts: *Neural Networks–A Comprehensive Foundation,* 2nd Edition by Haykin (Prentice-Hall, 2003). In addition, a collection of relevant papers and readings will be made available during the course.

Projects can involve designing an ANN, coding it up, and doing experiments with it. One can also download and use the JOONE libraries to develop their own networks (not asking to reinvent the wheel). Obviously, coding your own is very instructive, but so is learning how to use a library of objects to develop one's own "system". JOONE is available from: http://www.joone.org.

1.2 Course Overview

Neural networks are an attempt to mimic the behavior, in some respects, of the brain—a network of nerve tissues. Each nerve or neuron is considered a relatively simple device that when combined with other similar neurons becomes an exceedingly complex system capable of performing calculations and behaviors still considered difficult if not impossible for digital computers.

Neurons can be viewed as a simple system in that its behaviors and how it affects other neurons is simple. It is thus a system that can be modeled to some degree using methods from systems theory. The idea is that if such artificial but simple computational devices could be developed, then the possibility of mimicking complex behaviors is increased along with a host of useful applications, hence, the motivation for ANNs. Thus;

• Artificial neural networks have their roots in:

  - neuroscience
  - mathematics and statistics

- physics
- computer science
- engineering

• Application areas of neural networks:

- modeling
- time series processing
- signal processing
- pattern recognition
- automatic control

There are several advantages of trying to use ANNs:

- Comprised of very simple, computational devices.
- These elements are very fast (faster than digital computer elements *e.g.,* the flip-flop).
- They can be easily interconnected.
- They can be made adaptive. ANNs can be "trained" or learn from their "environment"—*i.e.,* supervised (with "teacher") and unsupervised (without "teacher") learning paradigms.
- They can be used in parallel fashion and exploit the computational aspects of non-linearities (what might these be? decisions…)

We are going to cover many of the most popular types of ANNs, how and why they work. This includes the FFBP, Hopfield nets, other recurrent nets such as Cellular Neural Nets and how these networks are used, trained and other relevant issues.

Does anyone know of a common usage of ANNs?

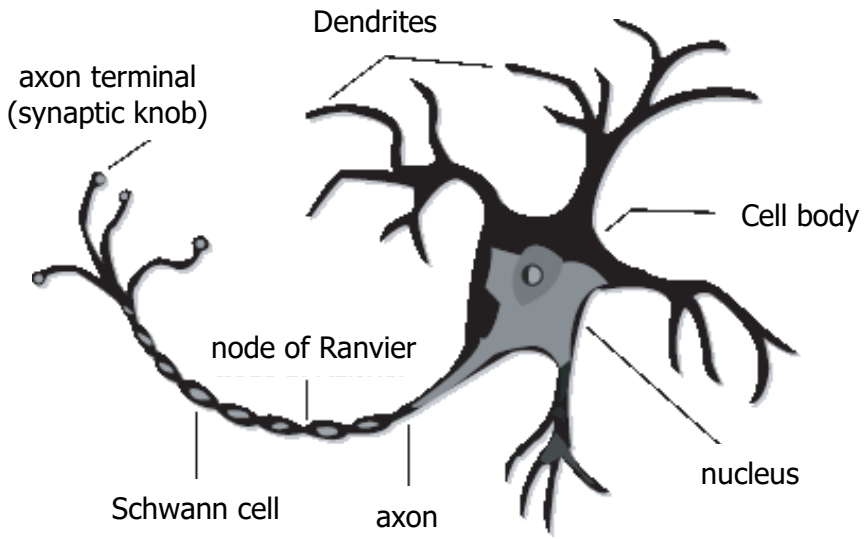
Before getting into the necessary mathematical review, let's take a look at how neurons work.

1.3 Basics of the Biological Neuron

The nerve body, the inputs at synapses in the dendritic tree, the axon transmits an electro-mechanical charge down its length as valves change as the charge moves down the axon much like a set of dominoes falls. These effects are caused by the so-called "ion-pump" and depends on various electrolytes in the form of sodium and potassium ions.

Once the charge reaches the terminal where the output synapses are located, it affects vesicles that contain neurotransmitters which are released into the space between the synapses of two connected neurons. These neuron-transmitters affect the neuron downstream by excitation or inhibition of the formation of an electric potential. This potential apparently decays over short intervals of time, hence enough input neurons must fire to trigger the firing of a neuron downstream and provided the inputs of other connected neurons sufficiently excite the neuron in question. In fact, some neurons at certain synapses are more prone

(trained) to provide higher excitation or inhibition signals depending on the resulting effects of a neuron firing. This provides a type of feedback.



1.5 Basic Neural Network Models

1.5.1 The Perceptron

A Long History—Neural nets have a surprisingly long history given the recent interest in the last decade. As early as 1943, McCullogh and Pitts (1943) described their formal neuron. See also Rosenblatt (1962). Minsky and Papert (1969) later developed their concept of *perceptrons* as small, simple computing devices that take weighted inputs and calculate a single output. Thus, neurons are seen as simple, signaling devices that are connected to other neurons. The earliest, simple, mathematical model of this type of process is based on the *perceptron* as shown in the figure:

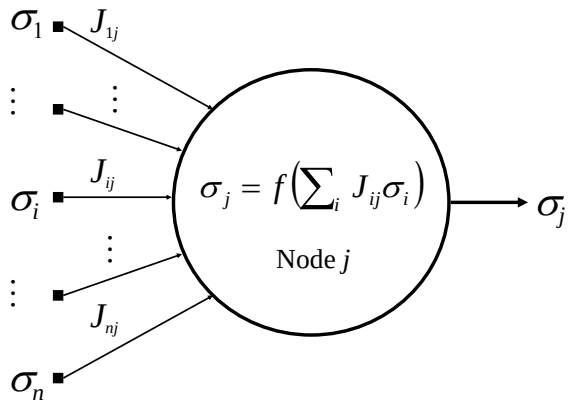


Figure 1: The basic *perceptron.*

The output function is some linear combination of the inputs weighted by the *synaptic efficacies* Jij . Thus, $\sigma_i = f\left(\sum_j J_{ij}\sigma_j\right)$ where *f* is some mapping onto 0-1. This simple structure gives the perceptron the capability to perform logic operations and evaluate the truth-value of any set of input propositions (Amit, 1989). Exhibit 1 illustrates the basic perceptron where neuron *i* produces a 0-1 output σ_i based on inputs from other similar neurons.

NN paradigm

- simple input-output device/abstraction.
- simple signals.
- simple computation.
- complex/involved mathematical capability

1.5.2 The Multiperceptron

The basic perceptron can be expanded to the multi- perceptron capable of producing *m* outputs as illustrated in Exhibit 2 (Amit, 1989). The multiperceptron thus maps n inputs onto *m* outputs and performs classifications or, depending on its uses, associations (Amit, 1989).

Quite a few neural nets employ concepts of feedback control in its dynamic behavior mode (this is in distinction to feedback used in a training mode such as the well-known feedforward, back-propagation training algorithm).

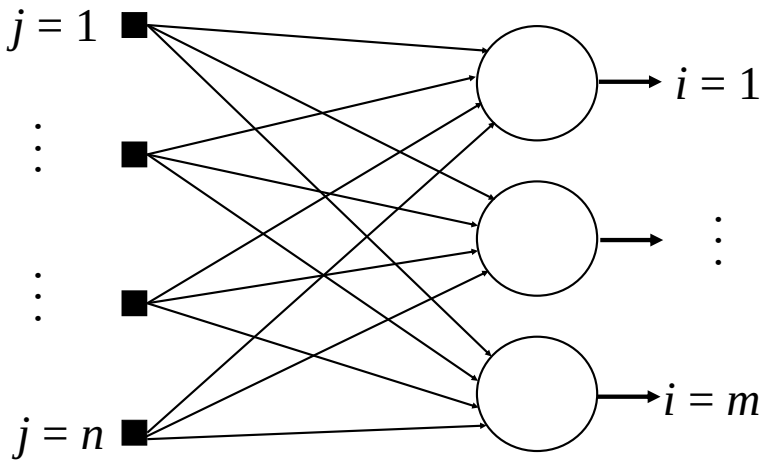


Figure 2: The Multi-perceptron.