Chapter 1

INTRODUCTION

The study of neural networks (NN) is one of the most rapidly expanding fields attracting people from a wide variety of disciplines. The study of neural networks is a field which cuts across many disciplines like philosophy, biology, psychology, mathematics, statistics, neuroscience, physics, engineering and even linguistics [Wasserman 1989]. These interwoven disciplines have made the study of neural networks unique. Neural networks bring together various subjects and disciplines in building intelligent systems.

1.1 What is a Neural Network?

The term neural network (NN) in this thesis refers to artificial neural network (ANN) which mimics biological neural systems.

There are several definitions as to what a neural network means: Maren defines neural networks as computational systems, either hardware or software, which mimic the computational abilities of biological systems by using simple *interconnected* artificial neurons [Maren *et al* 1990].

Hecht-Nielsen gives a rigorous definition of a neural network as "a *parallel*, distributed information processing structure consisting of processing elements which can possess a local memory and carry out localized information processing operations, *interconnected* together with unidirectional signal channels called connections. Each processing element has a single output connection which branches (fans out) into as many collateral connections as desired (each carrying the same signal - the processing element output signal). The processing element output signal can be of any mathematical type desired. All of the processing that goes on within each processing element must be completely local, i.e. must depend only upon the current values of the input signal arriving at the processing element via impinging connections and upon values stored in the processing element's local memory " [Hecht-Nielsen 1989].

A simpler definition of a neural network, given by Fausett, is that, a NN is an information processing system that has certain performance characteristics, such as adaptive learning, and parallel processing of information, in common with biological neural networks [Fausett 1994].

Haykins defines a neural network as a massively *parallel* distributed processor that has a natural propensity for storing experiential knowledge and making the knowledge available for use [Haykins, 1994].

A neural network can also be defined as a distributed computational system composed of a number of individual processing elements operating largely in *parallel*, *interconnected* according to some specific topology (architecture) and having the capability to self modify connection strengths and processing elements parameters [Rogas 1996].

From Müller and Reinhardt's view, a neural network model is defined as an algorithm for cognitive tasks, such as learning and optimization, which are in a loose sense based on concepts derived from research into the nature of the brain [Müller *et al* 1990].

From all these definitions, it can be deduced that

- A neural network is inspired by studies of the brain. Though, it would be wrong to say that a neural network duplicates brain functions, because the brain is highly complex and the actual *intelligence* exhibited by the most sophisticated neural network is well below the level of intelligence of any animal [Wasserman 1989].
- A neural network is made of several interconnected units similar to the neurons in the brain.
- A NN is an information processing system that operates in parallel.
- Signals are passed between units over connection links and each link has an associated weight.
- Artificial neurons are simple emulations of biological neurons. Artificial neurons receive information from other artificial neurons or the environment, perform a simple operation by applying functions on these input signals and pass the result to other neurons or the environment.
- Each unit applies an activation function (usually nonlinear) to the net input and determines the unit's output signal.

1.1.1 Characteristics Of A Neural Network?

A neural network is characterized by

- the architecture of the NN, which refers to the number of layers in the network, the number of neurons in the layer, and how these neurons are interconnected. Neural network types include single layer networks such as the Hopfield NN [Maren *et al* 1990], multilayer feedforward neural networks (MLNNs) such as backpropagation [Wasserman 1989]and recurrent NNs (RNNs) [Simpson 1990].

- The method of adjusting weights for each connection, referred to as the learning algorithm. Learning algorithms are divided into two main categories, namely supervised and unsupervised learning. Learning in supervised mode is done by comparing the network's output to the desired output, which is provided by the system or external teachers [Simpson 1990]. Learning in unsupervised mode, on the other hand, is by self organization. There is no target or desired output and hence no comparison to predetermined responses [Simpson 1990].

- The activation function used which can be linear, discrete functions such as the ramp function or continuous functions such as the sigmoid function.

The advantages of, and reasons for using neural networks rather than conventional methods of optimization, regression, classification and clustering are discussed in section 1.2.

1.2 Why Neural Networks?

The interest in neural networks is motivated by the desire to understand the brain, i.e. the principles on which the human brain works, to emulate some of the brain's strength and the wish to build machines that are capable of performing complex tasks for which the sequentially operating programmable computers are not well suited for.

Everyday observation shows that the brains of even animals of lower intelligence can perform tasks that are far beyond the range of even the largest and the fastest modern electronic computers. For example, dogs bark at human beings that are strangers while they are quite relaxed with human friends. Dogs can distinguish between foes and friends. No present day electronic computer has sufficient computational power to match this similar accomplishment. This accomplishment involves some need for the recognition of complex optical or acoustical patterns which are not determined by simple logical rules

[Müller *et al* 1990].

Neural networks are also used when data, on which conclusions are to based, is fuzzy. When the influential or informative patterns are subtle or hidden, a NN has the ability to discover patterns which are not clear, or unknown to the human researcher or standard statistical methods. An example is credit worthiness of loan applicants based on spending and payment history [Masters 1993]. NNs have been applied to data that exhibits significant unpredictable nonlinearity [Fausett 1994]. NNs have been adapted to predict future values not based on strictly defined models, and offer possibilities for solving problems that require pattern recognition, pattern mapping, dealing with noisy data and pattern completion [Masters 1993].

The advantages of NNs are summarized below:

1. A NN has the ability to learn.
2. Neural networks are robust to noise.
3. Neural networks work excellently for nonlinear data.
4. Because NN can learn to discriminate patterns based on examples and training, an elaborate a priori model is not needed neither is the probability function needed to be specified. The statistical distribution of the data used for training is not needed.

Specific areas where NNs have been applied include: pattern recognition and classification, adaptive control applications, financial analysis such as forecasting and credit assessment, database mining, function approximation and clustering [Fausett 1994, Masters 1993, Wasserman 1989, Towell *et al* 1993].

1.2.1 Features of Neural Networks

A very important feature of a neural network is an ability to adapt to changing environments, where learning is by examples. That is, the NN learns how to perform certain tasks by undergoing training with illustrative examples. Once trained, a NN can perform tasks without any external help, even if presented with distorted patterns [Beale *et al* 1990]. This feature makes NNs very appealing especially in application problems where little or no understanding of the problem is known, but where training data which reflects the characteristics of the problem is available. Neural networks can learn various things such as distinguishing a straight line from a convex curved line. The NN can discriminate between the lines once trained, even when the lines are shifted up or down, or even if the data is noisy.

Another feature of neural networks is the *parallel* architecture, which allows faster computation of some problems when the network is implemented on parallel digital computers, or when the network simulates parallelism. Electronic computers are designed to carry out one instruction after the other, extremely rapid whereas the brain work with slower units. A computer is a high speed, serial machine compared to the highly parallel nature of the brain. Computers therefore manage tasks such as counting (an essentially serial activity) which suit its design well, making the computer superior to the brain in such tasks. However, for highly parallel tasks such as vision or speech, computers perform badly. The brain is able to operate in parallel easily and thus is much faster than the computer in performing these tasks.

The approach of NNs in various applications is to capture the guiding principle that underlines the way the human brain solve problems and apply these principles to computer systems.

1.3 Background to Neural Networks

Neural networks have been motivated right from their inception by the fact that the brain computes in an entirely different way from the conventional von Neumann machines (computers) [Hassoun, 1995]. The brain is a highly complex, nonlinear and parallel information processing system. The brain has the capability of organizing neurons to perform certain tasks such as pattern recognition, speech recognition, pattern classification many times faster than the fastest digital computer in existence today.

The understanding of this neurobiology has allowed researchers to simulate neural behavior. This idea of simulating neural behavior dates back to the early 40's when one of the abstract models of a neuron was introduced by McCulloch and Pitts. They proposed a general theory of information processing based on networks of binary switches called neurons. These neurons were much simpler than their real biological counterparts. McCulloch and Pitts demonstrated that even simple types of neural networks could in principle, compute any arithmetic or logical function [Hecht-Nielsen 1989].

In 1949, Donalds Hebb proposed a learning rule that explained how a network of neurons learned. He used the learning rule to build a qualitative explanation of some experimental results. This bold step served to inspire many other researchers to pursue the same theme, which further laid the ground work for the advent of neural networks [Hecht-Nielsen 1989].

Rosenblatt invented the perceptron and its learning algorithm in 1958. The perceptron in its simplest form consists of two separate layers of neurons representing the input and output layers. An iterative algorithm for constructing synaptic coupling such that a specific input pattern is transformed into the desired output pattern was introduced. However, the perceptron had a serious shortcoming: it was only capable of solving classification problems that are linearly separable at the output layer [Fu 1994]. At the same time, Widrow and Hoff developed an important variation of perceptron learning known as the Widrow-Hoff rule [Fu 1994].

In the late 60's, Minsky and Papert caused research in NNs to be terminated with their results published in their landmark book called Peceptron [Hecht-Nielsen 1989]. Minsky and Papert pointed out the theoretical limitations of single layer neural network models. They proved that the perceptron cannot implement the exclusive or (XOR) logical function. The perceptron also had difficulty in learning other binary predicate functions. The implicit conclusion from their book was that essentially all neural networks suffer the same fatal flaw as the perceptron and they left the impression that neural network research was a dead end [Hecht-Nielsen 1989]. Due to this pessimistic work, research on neural network lapsed into an eclipse (a dark age for neural network research) for nearly two decades [Fu 1994].

Despite this, a few *faithful* researchers still continued their work on NNs and produced meaningful results during this period. For example, Anderson and Grossberg did important work on the psychological models [Hecht-Nielsen 1989]. Kohonen invented the self organising map (SOM), an associative memory model [Fu 1994].

In the early 80s, after two decades of obscurity, there was a renewed enthusiasm in the neural network field. A notable researcher who increased the visibility and respect for NN study is Hopfield. In 1982, Hopfield introduced the idea of energy minimization in physics to neural networks [Hopfield 1982, Fu 1994].

In the mid 80s, Rumelhart, Hinton and Williams developed a learning algorithm for multilayer networks called the backpropagation algorithm (BP) [Wasserman 1989]. This algorithm offered a powerful solution to training a multilayer neural network and hence counters the implicit conclusion of Minsky and Papert. Their development of multilayer feedforward networks was not restricted to linearly separable training sets. Along with a reasonably effective training algorithm for NNs, Rumelhart *et al* demonstrated that neural networks can provide real solutions to practical problems [Rumelhart *et al* 1986. Masters 1993]. Problems such as the XOR and lack of a general method of training a multilayer neural network, which had originally contributed to the demise of neural networks in the 60s, were overcome using the backpropagation algorithm and other techniques which followed

[Wasserman 1989]. It is interesting to note that Werbos had developed the idea of backpropagation in 1974 and also Parker in 1982 independently [Maren *et al* 1990].

A spectacular success of backpropagation is demonstrated by the NETTALK system developed by Sejnowski and Rosenberg in 1987. NETTALK is a system that converts English text into highly intelligible speech [Wasserman 1989]. The backpropagation algorithm is probably the most well known and widely used training algorithm [Maren *et al* 1990]. Much research effort was expended to improve backpropagation. The objective of this study is to further study methods to improve BP. Approaches and specific research to improve the performance of NNs using BP include finding optimal weight initialization [Wessels *et al* 1992], optimal learning rate and momentum [Yu *et al* 1997, Weir 1990], finding optimal architectures [Engelbrecht *et al* 1996. Hassibi *et al* 1994, Le Cun 1990, Karnin 1990, Sietsma *et al* 1988], using second order optimization techniques [Becker *et al* 1988], adaptive activation functions [Fletcher *et al* 1994, Engelbrecht *et al* 1995, Zurada 1992a] and active learning [Röbel 1994a, Zhang 1994, Engelbrecht *et al* 1999a].

A large number of neural networks are trained using the gradient descent optimization method in the supervised mode. In order to train the network successfully, the output of the network is made to approach the desired output by continually reducing the error between the network's output and the desired output. Training a NN is achieved by presenting the network with information to learn, which consists of a fixed set of input attributes and corresponding target outputs. The weights between the layers are then adjusted using an optimization algorithm, usually the gradient descent optimization, the error is computed and backpropagated from one layer to the previous layer. But presenting all the available data to the network can be problematic, especially when there are redundant data in the training set. The computational expense in terms of training time and the complexity can be unnecessarily high if all the data are used for training.

Studies have shown that selecting the most informative data, rather than training on

all the available data, improves, or at least maintains the generalization performance, as well as reduces the training cost, and the data needed for training [Engelbrecht *et al* 1998, Engelbrecht *et al* 1999a, Röbel 1994a, Zhang 1994]. Active learning refers to such selection of a subset of the available training data containing the most informative patterns for training. The concept of active learning is to efficiently select high utility patterns from available data for training the network. There are two approaches to active learning, namely incremental and selective learning.

This thesis focuses on the study of active learning as a method of improving performance of NNs on function approximation and time series problems. Section 1.4 discusses the objectives of this study.

1.4 Objective and Justification

The backpropagation learning algorithm played a vital role in the resurgence of interest in neural networks. Eversince, a lot of research effort has been concentrated on finding ways to improve the performance of backpropagation learning. Research has concentrated on finding the *optimal* size of networks, to make *optimal* use of training data, to *optimize* initial weights and learning parameters.

This thesis concentrates on methods to optimize the use of training data, i.e. *active learning*. A new selective algorithm for time series problems is proposed and is used as one of the selected active learning algorithms to be compared. A comparative study is carried out on three additional active learning algorithms. While many research efforts have concentrated on designing new active learning approaches, as well as other learning algorithms, an elaborate comparison of these approaches is still lacking and hence the motivation for this study. The four selected active learning algorithms are compared to each other with reference to their respective performances in terms of accuracy, computational complexity

and convergence characteristics.

Accuracy of a learning algorithm is how well a function is approximated by the network using the algorithm. The mean squared error (MSE) on the training set and the test set are used as the measure of accuracy. The training error is the error computed over all the patterns or data presented to a network for training, while the generalization error is the error computed over a set of patterns not used for training a network i.e. test set. A low generalization and training error is an indication of good approximation of the problem and a good performance of the network. However, a low training error and a large generalization error is an indication that the training set is overfitted. A MSE value close to zero shows a small error between the target and the output function. Computational complexity measures the cost of training the network. The cost is measured by the number of calculations made during training. The number of patterns selected for training is quite important because of the proportional relationship between computational cost and the number of patterns. The more patterns selected for training, the more calculations are made during training and thus, a higher training cost. Based on these criteria, a critique of the different algorithms as well as suggested future work are discussed.

The scope of this thesis is *multilayer feedforward neural networks*, focusing on function approximation and time series problems. Gradient descent is used as optimization method and sigmoid activation functions are used. A three layer neural network with one input layer, one hidden layer, and one output layer is used.

1.5 Outline

The rest of this thesis is organized as follows:

- Chapter 2 deals with learning in multilayer neural networks. A general introduction and a background study of multilayer neural networks are given. The architectures, learning algorithms and weight updating methods are discussed. The difficulties of training multilayer neural networks, as well as solutions to these difficulties are discussed.

- Active learning is discussed in chapter 3. The concept and the basis for active learning are examined. Results and simulations of the four selected active learning algorithms are presented.

- Chapter 4 concludes this thesis with observations and suggestions for future research.