Chapter XVI
Artificial Neural Networks

Xiaojun Yang
Florida State University, USA

ABSTRACT

Artificial neural networks are increasingly being used to model complex, nonlinear phenomena. The purpose of this chapter is to review the fundamentals of artificial neural networks and their major applications in geoinformatics. It begins with a discussion on the basic structure of artificial neural networks with the focus on the multilayer perceptron networks given their robustness and popularity. This is followed by a review on the major applications of artificial neural networks in geoinformatics, including pattern recognition and image classification, hydrological modeling, and urban growth prediction. Finally, several areas are identified for further research in order to improve the success of artificial neural networks for problem solving in geoinformatics.

INTRODUCTION

An artificial neural network (commonly just neural network) is an interconnected assemblage of artificial neurons that uses a mathematical or computational model of theorized mind and brain activity, attempting to parallel and simulate the powerful capabilities for knowledge acquisition, recall, synthesis, and problem solving. It originated from the concept of artificial neuron introduced by McCulloch and Pitts in 1943. Over the past six decades, artificial neural networks have evolved from the preliminary development of artificial neuron, through the rediscovery and popularization of the back-propagation training algorithm, to the implementation of artificial neural networks using dedicated hardware. Theoretically, artificial neural networks are highly robust in data distribution, and can handle incomplete, noisy and ambiguous data. They are well suited for modeling complex, nonlinear phenomena ranging from financial management, hydrological modeling to natural hazard prediction. The purpose of the article is to introduce the basic structure of artificial neural networks, review their major applications in geoinformatics, and discuss future and emerging trends.

BACKGROUND

The basic structure of an artificial neural network involves a network of many interconnected neurons. These neurons are very simple processing elements that individually handle pieces of a big problem. A neuron computes an output using an activation function that considers the weighted sum of all its inputs. These activation functions can have many different types but the logistic sigmoid function is quite common:

$$f(x) = \frac{1}{1+e^{-x}}$$

where ***f(x)*** is the output of a neuron and ***x*** represents the weighted sum of inputs to a neuron. As suggested from Equation 1, the principles of computation at the neuron level are quite simple, and the power of neural computation relies upon the use of distributed, adaptive and nonlinear computing. The distributed computing environment is realized through the massive interconnected neurons that share the load of the overall processing task. The adaptive property is embedded with the network by adjusting the weights that interconnect the neurons during the training phase. The use of an activation function in each neuron introduces the nonlinear behavior to the network.

There are many different types of neural networks, but most can fall into one of the five major paradigms listed in Table 1. Each paradigm has advantages and disadvantages depending upon specific applications. A detailed discussion about these paradigms can be found elsewhere (e.g., Bishop, 1995; Rojas, 1996; Haykin, 1999; and Principe *et al.,* 2000). This article will concentrate upon multilayer perceptron networks due to their technological robustness and popularity (Bishop, 1995).

Figure 1 illustrates a simple multilayer perceptron neural network with a 4×5×4×1 structure. This is a typical feed-forward network that allows the connections between neurons to flow in one direction. Information flow starts from the neurons in the input layer, and then moves along weighted links to neurons in the hidden layers for processing. The weights are normally determined through training. Each neuron contains a nonlinear activation function that combines information from all neurons in the preceding layers.The output layer is a complex function of inputs and internal network transformations.

The topology of a neural network is critical for neural computing to solve problems with reasonable training time and performance. For any neural computing, training time is always the biggest bottleneck and thus, every effort is needed to make training effective and affordable. Training time is a function of the complexity of the network topology which is ultimately determined by the combination of hidden layers and neurons. A trade-off is needed to balance the processing purpose of the hidden layers and the training time needed. A network without a hidden layer is only able to solve a linear problem. To tackle a nonlinear problem, a reasonable number of hidden layers is needed. A network with one hidden layer has the power to approximate any function provided that the number of neurons and the training time are not constrained (Hornik, 1993). But in practice, many functions are difficult to approximate with one hidden layer and thus, Flood and Kartam (1994) suggested using two hidden layers as a starting point.