AN ARTIFICIAL NEURAL NETWORK APPROACH FOR LAND COVER CLASSIFICATION

K.K. MOHANTY
Land Resources Division, Remote Sensing Applications Group, Space Applications Centre, Ahmedabad, Gujarat

Abstract *This paper presents a preliminary study on usage of Artificial Neural Network (ANN) for land cover classification using multispectral remotely sensed data. The ANN model used is a simple multi-layer perceptron trained through generalized delta learning rule. Performance of the ANN classifier has been assessed with reference to the conventional maximum likelihood (MXL) classifier. In the current study, it is found that the ANN classifier has a better classification accuracy than the MXL classifier.*

INTRODUCTION

Artificial Neural Network (ANN) is an attempt for simulation of human brain to achieve human-like performance in the field of information processing (Brunak and Lautrup, 1988). Human brain consists of nerve cells, called neurons, each of which is connected to a large number of other neurons. A neuron receives input or feedback from all other neurons that are connected to it at its input and generates a single output or response which again serves as input to many other neurons. The connections between different neurons may be of varying strength i.e. signal from one neuron to another is transferred with varying weightage. Response of brain to a given input is generated by this network partially through is genetically programmed structure, but mainly by its learning or past experience derived from execution of similar kind of inputs. Many attempts have been made in the recent years to establish ANN models in the field of image classification. (Bendiktsson et.al.,1990; Jonathan et.al., 1990 Bischof et.al., 1992). This paper reports an assessment of performance of an ANN model for land cover classification with special emphasis on its relative advantages over the conventional maximum likelihood (MXL) classifier.

OBJECTIVES

The objectives of the study are: (a) to develop a generalized algorithm and associated software for landcover classification using ANN and (b) to assess its performance in reference to the conventional maximum likelihood (MXL) classifier.

DATA USED

IRS-1A LISS-I image in four spectral bands each with a size of 512 pixel by 512 scan lines, extracted from a scene over Madhya Pradesh (Path/Row-27/52, date of acquisition 21-1-92) was selected for test classification.

ARTIFICIAL NEURAL NETWORK (ANN)

Based on the limited understanding of structure and organization of brain, a new tool variously known as ANN, connectionist model, parallel distributed processing model and neuromorphic system, has been formulated for information processing (Lippmann, 1987). The model consists of many simple non-linear (often analog) computational elements, called nodes or neurons, linked together through a dense interconnection among the individual nodes. Each of the nodes receives input signals from a large number of other nodes and generates a single output which again passes through many path ways to provide input to other nodes. The connections between different nodes are of varying strength or weightage. These connections are generally trained or adopted using a set of training data so as to map sets of input patterns to their corresponding output classes. An ANN model is characterized by its topology or organization, node characteristics and training or learning rules. Topology of an ANN model refers to the way in which different nodes in the network are arranged and linked together. A node or neuron is characterized by an internal threshold and a nonlinear transfer function e.g. sigmoid, hard-limiters, threshold logic elements etc. The simplest type of nodes sum up the inputs it receives and passes a part of the net input in excess of its internal threshold through a nonlinear transfer function. Training or learning rules define a set of initial weights and rules for modification of these weights. ANN models can be of both parametric (supervised) or nonparametric (unsupervised) types.

Perceptrons

Perceptrons are feed-forward type networks normally used under supervision as classifier or associative memory (Minsky and Papert, 1969). Single-layer perceptron model or two-layer associative network is the simplest kind of perceptron model consisting of a single input and an output layer of nodes with its input nodes directly connected to the output nodes (Fig.1).

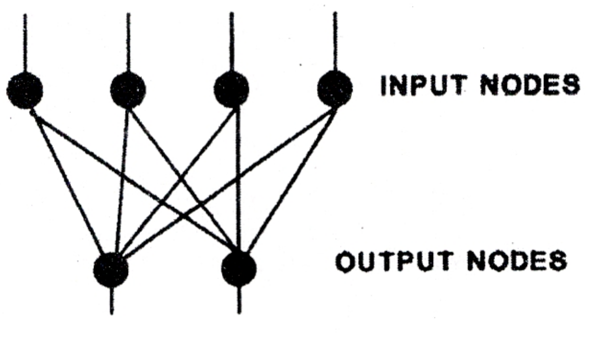

Fig. 1 Single layer perceptron model

Such models can map simple patterns, but fail when the similarity in pattern of individual classes as provided by the outside world differ widely. However, such problems can be solved using multi-layer perceptron models. A multi-layer perceptron model uses additional intermediate layers, called hidden layers, between the input and output layer (Fig. 2). The hidden layers in a multi-layer network generate an internal representation of the input pattern powerful enough to establish similarity between input and output pattern classes. Assuming the input layer in a multi-layer perceptron to lie at the top and the output layer at the bottom, a node in any of the layers can be connected with a neuron in layer lying below it. Any connection from a node to node lying in the same layer or in a higher-level is not allowed. However, a node can skip an intermediate layer to connect directly with a node in layer lying below.

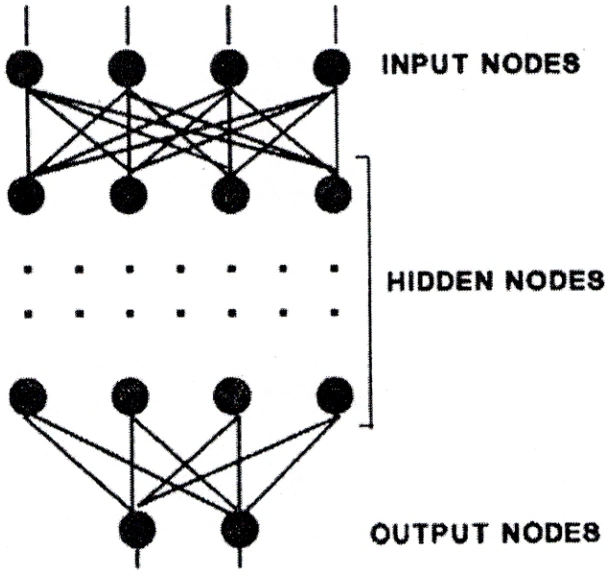

Fig. 2 Multi-layer perceptron model

Error Backpropagation Learning Rule

In parametric type networks, the model is fed with a set of examples or illustrations of the problem in the form of sets of input pattern and their respective target classes. These training data sets are repeatedly fed to the network either in a particular sequence or in a complete random fashion. In each iteration the model compares the computed output with the desired/target output and adjusts its model parameters so as to minimize the difference between the target and computed output. The iteration for modification of weights continue until all the weights get stabilized i.e. there is no more change in the output (in reality change is below a threshold) with additional iterations. In error back propagation method, while the signal or input moves from input layer, the error moves just in the reverse direction. The error is computed at the output nodes after each pass and the weights and thresholds are adjusted step by step from the output layer to the input layer so as to minimize the error.

Generalized Delta Rule

Delta rule or perceptron convergence procedure use in case of single-layer perceptron models achieves convergence by minimizing the mean of square of differences between the desired and actual output over all output units for all input-output pattern pairs (Minsky and Papert, 1969). Physically it cab be visualized as selecting the steepest descent in a weight space at a point whose height is equal to the error measure. The generalized delta learning rule (Rumelhart et. al., 1986) is a generalization of the delta rule to use it for training multi-layer perceptron models. In generalized delta learning scheme all weights and thresholds are first set to small random number. Starting with the output node the weights and thresholds are adjusted layer by layer working back towards the input layer as per the equation.

$$\Delta W_{ji} = \mu \delta_{pj} O_{pi} \tag{1}$$

where ΔW_{ji} is the change in weight for a connection from a hidden node i or an input to node j; O_{pi} is the output from a node i or an input fed to the network for a given pattern p; and μ is an adaptive gain term.

If j lies in the output layer, then

$$\delta_{pj} = (t_{pj} - o_{pj})\, f(net_{pj}) \tag{2}$$

where net_{pj} is the net input arriving at node j; t_{pj} is the target output at a node j for an input pattern p; and f is the transfer function used

If j lies in an internal hidden layer, then

$$\delta_{pj} = f(net_{pj})\, \Sigma \delta_{pk}\, W_{kj} \tag{3}$$

where k is over all nodes in layers above node j.

Perceptrons as Classifier

The classification of multispectral remote sensing data can be considered as a mapping from a multi-dimensional grey value space to a feature space. The input vector has a dimension equal to the number of classes involved in classification while that of the output vector is equal to the number of expected features. Figure3 shows a block diagram illustrating an ANN classifier model involving four spectral bands and five feature classes. The input vector whose coefficients are given by the four grey values are fed at their respective input channels. The output vector is measured at all output nodes. Each of the output nodes corresponds to particular feature class. The output node recording the highest reading indicates the class to which the input vector belongs. The basic classification problem using ANN involves following steps:

(i) Selection of a proper topology for the ANN model in terms of number of input units, output units, hidden layers, hidden nodes in individual hidden layers and a transfer function.

(ii) Generation of a set of training data set containing pairs of input-output vectors for all expected feature classes.

(iii) Training of the ANN model using the training data set iteratively until it achieves a specified degree of performance in classifying the training data set.

(iv) Classification of the actual images by feeding each of the input vectors across the network and finding out the node having maximum output.

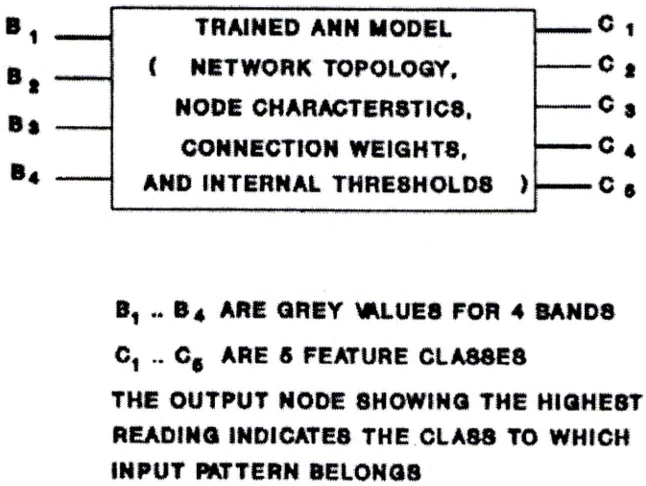

Fig.3 Block diagram illustrating ANN classifier model.

THE SOFTWARE

An algorithm and associated software in C language has been developed in SUN SPARC 4/280 system. The software is general in nature and can deal with any number of input units (spectral bands). Output units (feature classes) and hidden layers (minimum one hidden layer). Different number of hidden nodes can also be considered in different hidden layers.

Internal threshold values for the nodes have not been used in the present study. Initial values for all weights are set to random real numbers between 0 and 1. Subsequently these weights are trained iteratively through the generalized delta rule using the training data set. Sigmoid transfer function has been used to compute the outputs for all nodes. The output of a node j is given by

$$O_i = 1/(1 + \exp(-\Sigma\ W_{ij}\ x_i)) \tag{4}$$

where i is the locus of all nodes that are connected to the node j at its input; W_{ij} is the connection weight along a connection from a node i to a node j; and x_i is the output of the node i or an input connected to node j.

TEST CLASSIFICATION

On the basis of FCC, six classes (clear water, turbid water, healthy vegetation, poor vegetation and two waste land classes) were selected for classification. The coordinates of top left hand corner and bottom right hand corner or rectangular training sites were used to extract input vectors from all training sites. A modified training data set was formed using the input vectors so as to have an equal representation of all feature classes. The input data set is also rearranged so that network gets access to input vectors of all classes at an equal interval. The training is continued iteratively until the squared standard error (SSE) falls below a certain threshold. The averaged SSE is compared with a predefined threshold after each complete cycle of training. SSE is computed using the relation (Rumelhart et. al., 1986):

$$SSE = (1/2)\sum_p \sum_j \left(t_{pj} - O_{pj}\right)^2 \tag{5}$$

where j varies over all output units of an output vector, and p varies over all input-output vectors of the training data set.

Once the network is trained, the network is used to classify the images by feeding all input vectors across the network.

RESULTS AND DISCUSSION

In MXL classifier, classification of the training data sets showed that there is no confusion among the classes i.e. no commission or omission error. The same classification accuracy for training data set in case of ANN classifier could be achieved using one hidden layer and nine hidden neurons. The network was trained up to an averaged SSE<0.005. During classification, a threshold of 0.5 was set for actual output units it was found that for the training data set, the output at both the output units corresponding to vegetation classes may simultaneously exceed 0.5 for certain inputs belonging to vegetation –2 class. However, in such situations also the highest output value corresponds to the actual input class. An input vector for which all the output units shows readings less than the threshold or more than two of the output units shoot above the threshold is considered to be unclassified i.e. it cannot be classified as any of the classes for which the network is trained.

A large number of the pixels which are left unclassified by the MXL classifier could be correctly classified on the basis of the ANN classifier as is evident from the classified images and FCC. The confusion matrix (Table 1) generated using both classified images shows that most of the pixels left unclassified by MXL classifier have been correctly classified by the ANN classifier.

There are certain disagreements between the two classifier, particularly for the vegetation classes and wasteland classes e.g. some vegetation –2 classes for MXL have been classified as vegetation –1 by ANN. However a detailed inspection of the classified images on the basis of visual interpretation of FCC revealed that it is quite genuine i.e. ANN appears to be more correct.

Table 1 Confusion matrix generated using the ANN and MXL classified images

As Classified by MXL	As classified by ANN Classifier							MXL Result
	Clear water	Turbid water	Vegetation 1	Vegetation 2	Waste Land-1	Waste Land-2	Uncla-ssifed	
Clear water	5884 12.241	0 (0.0)	0 (0.0)	0 (0.0)	0 (0.0)	0 (0.0)	183 (0.07)	6067 (2.31)
Turbid water	0 (0.0)	131 (0.05)	0 (0.0)	0 (0.0)	0 (0.0)	0 (0.0)	2 (0.0)	133 (0.05)
Vegetation 1	0 (0.0)	0 (0.0)	4582 (1.75)	0 (0.0)	0 (0.0)	0 (0.0)	0 (0.0)	4582 (1.75)
Vegetation 2	0 (0.0)	0 (0.0)	3514 (1.34)	3857 (14.75)	0 (0.0)	0 (0.0)	7762 (2.96)	49933 (19.05)
Waste Land-1	0 (0.0)	0 (0.0)	0 (0.0)	0 (0.0)	90041 (34.35)	8550 (3.26)	11687 (4.45)	110278 (42.07)
Waste Land-2	0 (0.0)`	0 (0.0)	0 (0.0)	0 (0.0)	198 (0.07)	19142 (7.3)	105 (0.04)	19445 (7.42)
Uncla-ssifed	1195 (0.46)	424 (0.16)	1978 (0.76)	6759 (2.58)	23996 (9.15)	5610 (2.14)	31744 (12.11)	71706 (27.35)
ANN Result	7079 (2.7)	555 (0.21)	10074 (3.84)	45416 (17.32)	114235 (43.58)	33302 (12.7)	51483 (19.64)	190181 (72.55)

Total number of pixels = 262144. Figures in bracket represent the percent geographical coverage for the feature.

CONCLUSIONS

Present study shows that ANN classifier has got a better classification accuracy as it could classify the pixels left unclassified by MXL classifier. In case of MXL classifier, the distribution of input patterns for all classes are assumed to be strictly Gaussian. The ANN classifier is more robust than the MXL classifier as it does not need any assumption regarding the distribution of the input patterns. All statistical classifiers attempt to define the entire decision regions of the input pattern classes in terms of statistical measures i.e., the mean, standard deviation etc. derived from the training data set. However, ANN classifier tries to demarcate the decision boundary between the input pattern classes and hence may be able to separate more efficiently the closely spaced or overlapping pattern classes. With use of hidden layers, it is possible to model any complicated decision boundary.

Acknowledgement The author wishes to acknowledge Dr. Baldev Sahai, Group Director, RSAG/SAC, Dr. R.R. Navalgund, Head, LRD/RSAG/SAC and Dr. T.J. Majumdar, Sci./Engr. SF, LRE/SAC for their valuable suggestions and keen interest in the study. The author also wishes to thank Shri R.P. Singh and Shri. S.Dutta, Scientists/SAC for their help.

MXL classification programme provided by Shri S.A. Sharma, Sci./SAC, is also acknowledged.

REFERENCES

Bendiktsson, J.A, Swain P.H. and Ersay O.K. (1990) Neural network approach versus statistical method on classification of multisource remote sensing data, IEEE Transaction on Geoscience and Remote Sensing, 28(4), 540-552.

Bischof, H., Schneider, W. and Pinz., A.J. (1992) Multispectral classificatin of Landsat images using neural networks, IEEE Transactions on Geoscience and Remote Sensing, 30(3), 482-490.

Brunak, S. and Lautrup, B. (1988) Neural networks computer with intuition, World Scientific Publishing Co., Pte. Ltd.

Jonathan, L., Ronald, C.W., Sengupta, S.K. and Welch, R.M. (1990) A neural network approach to cloud classification, IEEE Transaction on Geoscience and Remote Sensing, 28(5), 846-855.

Lippmann, R.P. (1987) An introduction to computing with neural nets, IEEE ASSP Magazine, 4(2), 4-22.

Minsky, M.L. and Papert, S. (1969). Perceptrons, Cambridge, MA, MIT Press.

Rumelhart, D.E., Hinton, G.E. and Williams,R.J. (1986) Learning representation by back propagation error, Nature, 323(6088), 533-536