Using Genetic Learning Neural Networks for Spatial Decision Making in GIS

Jiang Zhou and Daniel L. Civco

Abstract

Traditional approaches for suitability analysis in GIS are overlay and the more complicated multicriteria evaluation (MCE). Despite being widely used, these methods have at least three problems: (1) difficulties in handling spatial data possessing inaccuracy, multiple measurement scales, and factor interdependency; (2) requirements of prior knowledge in identifying criteria, assigning scores, determining criteria preference, and selecting aggregation functions; and (3) typically, an "unfriendly" user interface. To solve these problems, in this paper a neural network approach is presented. The neural network uses a genetic algorithm as its learning mechanism. A set of experiments revealed that the aforementioned difficulties are overcome by the evolutionary learning of neural networks. Our conclusion is that genetic learning neural networks can provide an alternative for and improvement over traditional suitability analysis methods in GIS.

Introduction

Suitability analysis usually requires making decisions among multiple factors. There are two methods commonly practiced to accomplish this task in GIS: a simple overlay and the more complicated multicriteria evaluation (MCE). Overlay can only combine deterministic digital map information to define areas simultaneously satisfying two or more specific criteria (Carver, 1991). Recently, the integration of MCE into GIS has attracted much attention (Carver, 1991; Heywood et al., 1995; Pereira et al., 1993; Jankowski, 1995). It also has been observed that there are many problems in these somewhat conventional methods. The problems fall into three different, but not independent, categories. First, it is well known that the spatial data in GIS usually have properties that are difficult to handle by traditional methods, such as inaccuracy, multiple measurement scales, and interdependency among factors. Second, traditional methods require prior knowledge to identify all relevant criteria, assign scores, determine the criterion preference, and select the aggregation function. Methodological uncertainty and error may appear in these procedures. Third, the interface of traditional methods usually makes it difficult for a user to be involved effectively in the decision making procedure. For example, a user may be asked to provide a set of values, such as criteria weights. These questions are usually cognitively demanding and far beyond his or her intuition. In many cases, it is possible for a user or an expert to judge the suitability of a site through the implicit gestalt method (Hopkins, 1977), but it is much more difficult to express explicitly that knowledge.

In this paper, we present a neural network approach to address the aforementioned problems. The learning algorithms for neural networks are genetic algorithms instead of the traditional gradient descent-based backpropagation. Through evolutionary learning from samples, a neural network adapts its connection weights to approximate the desired output. Then, a successfully trained neural network can accomplish the suitability analysis task. A set of experiments is presented. The results show that the difficulties in traditional methods are solved by evolutionary learning and the nonlinear mapping ability of the neural network.

Introduction to Traditional Methods

Overlay

The application of digital map overlay for the purpose of identifying suitable areas is a classic application of GIS. In raster GIS, for example in IDRISI (Eastman, 1995), a suitability map is produced from a series of Boolean images, where each image represents all areas meeting the criterion being depicted. These images are then combined using the overlay combination procedure to yield a final map that shows the sites meeting all the specified criteria. However, overlays have limitations when dealing with information of a nondeterministic nature (Carver, 1991).

Multicriteria Evaluation

The general model of MCE is shown in Figure 1. In the first step, a set of criteria or factors is selected according to the alternatives. For example, to locate areas suitable for buying houses. Heywood (1995) considered four factors - school location, roads, urban areas, and insurance. After all the relevant criteria have been identified, the criterion scores, which indicate the impacts of alternatives on each criterion, need to be determined. There exist many methods for this task (Hepner, 1984; Pereira et al., 1993). In many analyses, especially those utilizing quantitative and mixed sources of data, the criterion scores need to be standardized (Carver, 1991). A number of methods can be used for standardization, such as additivity constraint, ratio scale properties, and interval scale properties (Voogd, 1983). These criterion scores combined with the criteria importance are processed by aggregation functions to produce the final evaluation result. The most commonly used aggregation function may be weighted summation, in which the criterion importance is represented by a weight.

The methods for criterion importance determination and aggregation can be classified into compensatory and noncompensatory (Hwang *et al.*, 1981; Minch *et al.*, 1986; Jankowski, 1995). In a compensatory method, the high performance of an alternative achieved on one or more criteria can compensate for the weak performance of the same alternative on

The Laboratory for Earth Resources Information Systems, Department of Natural Resources Management and Engineering, The University of Connecticut, Storrs, CT 06269-4087.

Photogrammetric Engineering & Remote Sensing, Vol. 62, No. 11, November 1996, pp. 1287–1295.

^{0099-1112/96/6211-1287\$3.00/0} © 1996 American Society for Photogrammetry and Remote Sensing



other criteria. Weighted summation is such a method. Other compensatory methods include concordance analysis, analytical hierarchy process, and ideal point. Noncompensatory techniques are the stepwise reduction of the set of alternatives without trading off their deficiencies along some evaluation criteria for their strengths along other criteria.

Another promising method for suitability analysis and mapping is based on the Dempster-Schaefer's theory of evidence and fuzzy logic (Boham-Carter, 1994).

Problems with Traditional Methods

Difficulties in handling spatial data with inaccuracy, multiple measurement scales, and factor interdependency

Inaccuracy

It is widely recognized that spatial data sets may contain numerous errors and inaccuracies (Macdougall, 1975; Heywood *et al.*, 1995). The errors contained in a region map can be the error in the positioning of boundary (horizontal error) and the error of impurity. The lower limit to the accuracy of an overlay map consists of the sum of the positioning errors and the product of the purity of the constituent factor maps, plus errors which are introduced in the assembly of the final overlay (Macdougall, 1975).

Macdougall (1975) gave an illustration of the possibility that the purity problem alone may make the overlay maps differ little from a random map. If the impurities of each region of a factor map are independent from those of another factor map, the purity of the overlay region is equal to the product of the purity of the factor maps. Table 1 illustrates just how low these values can be for various combinations of map numbers and overlay purity.

Multiple Measurement Scales

There are four measurement scales, from lowest level to the highest level: nominal, ordinal, interval, and ratio (Hammond *et al.*, 1974). The nominal and ordinal data are discrete while the interval and ratio data are usually continuous. In many traditional methods, to process mixed measurement scale data, the continuous scaled data are first discretized. To conserve the continuity of data, a value function approach can be used (Hepner, 1984; Pereira *et al.*, 1993). However, the construction

of value functions is usually tedious and subjectivity is difficult to avoid.

Interdependency

Factor interdependency commonly exists in GIS (Hopkins, 1977). This may overemphasize some factors when using overlay or MCE because of the number of data sets which may have been derived from the same base data (Heywood et al., 1995). There are several ways to deal with factor interdependency. Some statistical techniques such as factor analysis can be used to generate independent factors, but limited experimental experience suggests that, because of the data requirements and difficulty of interpretation, using factor analysis is not worthwhile for most suitability analyses (Hopkins, 1977). In principle, nonlinear combination aggregation functions can handle interdependency among factors. However, because the required mathematical relationships for the full range of costs and impacts are not known, the nonlinear combination is generally insufficient by itself (Hopkins, 1977). Other methods that can handle interdependency of factors include gestalt methods, explicit identification of regions, and logical combination of factors (Hopkins, 1977) Gestalt and identification of regions methods are inapplicable when there are a large number of regions in the maps. The logical combination method can deal only with discrete data, and the combination rules are sometimes difficult to define.

Requirements of Explicit Knowledge to Identify all Relevant Criteria, Determine Score and Criteria Preferences, and Select an Aggregation Function

The first step of overlay analysis or MCE is to identify all relevant criteria without omission or redundancy. Sometimes this task is difficult (Janssen, 1992; Heywood *et al.*, 1995). When a user determines the criteria score, criteria preference, and aggregation function, the problem of "method uncertainty" usually surfaces. Carver's (1991) experiments show that different evaluation techniques can significantly affect the outcome of the suitable site search. He suggests that two or more methods should be applied to dilute the effect of technique bias. Heywood *et al.* (1995) used the MCE routines in SPANS and IDRISI to evaluate housing suitability and found the amount of agreement between the results of the two systems was 34.8 percent. The explicit requirement of prior knowledge also makes the user interface difficult.

The Problem of User Interface

The user interface of traditional methods makes it difficult for a user to be involved effectively in the decision making process. A user needs to be particularly familiar with the operational details of these techniques. In many cases, even if a user can judge the suitability of a site through the implicit gestalt method (Hopkins, 1997), it is much more difficult to express explicitly an expert's knowledge. It is found that some human knowledge is inexpressible in the form of rules and sometimes may not be understandable even though it can be expressed (Hoffman, 1987; Sui, 1993).

To address these problems in a traditional suitability analysis, in this paper we developed a neural network ap-

TABLE 1. SOME	POSSIBLE	OVERLAY	PURITIES	(MACDOUGALL,	1975)	
---------------	----------	---------	----------	--------------	-------	--

Average purity	Number of factor maps				
of factor maps	2	4	6	8	10
0.95	0.90	0.81	0.73	0.66	0.60
0.85	0.72	0.52	0.38	0.27	0.20
0.75	0.56	0.32	0.18	0.10	0.06
0.65	0.42	0.18	0.08	0.03	0.01
0.55	0.30	0.09	0.03	0.01	0.00



proach. The learning algorithm of the neural network is a genetic one instead of the traditional backpropagation algorithm. Through evolutionary learning from samples, a neural network adapts its connection weights to approximate the desired output. Then in the recall phase, a successfully trained neural network can be used to accomplish the suitability analysis task.

Introduction to Neural Networks and Genetic Algorithms

Neural Networks

The neural network model is derived from a simulation of the human brain. The basic computation unit in a neural network is a neuron. A neuron performs the simple weighted summation and nonlinear mapping (Figure 2), where w0 is a threshold and f is usually a sigmoid function: i.e.,

$$f(z) = 1/(1 + \exp(-z)).$$
 (1)

A neural network has many neurons. The way neurons connect determines the structure of a neural network. A type of neural network called multilayer perceptron (MP), or feedforward network, consists of a sequence of layers of neurons with full connections between successive layers. Two layers of MP have connections to the outside world: the input and output layer. There are one or more hidden layers between the input and output layer. Information sequentially passes through the input, hidden, and output layers. A feedforward network with one or more hidden layers can form any shape of decision boundaries or approximate any continuous function, given sufficient hidden neurons (De Viller *et al.*, 1992; Kreinovich *et al.*, 1993).

A neural network usually has two distinctive phases: learning and recall. Currently, the most popular learning algorithm for feedforward network is backpropagation (BP) (Rumelhart, 1986). We will refer to a feedforward network using a BP learning algorithm as a BP network in this paper. With the BP algorithm, a set of training samples with the desired output is required. It is a procedure which iteratively adjusts the weights of the connections in the gradient descent direction so as to minimize a measure of the difference between the actual output vector of the network and the desired output vector. The difference is usually measured by the error function:

$$E = \frac{1}{2} \sum_{c} \sum_{j} (y_{j,c} - d_{j,c})^2$$
(2)

where *c* is an index over cases (input-output pairs), *j* is an index over output units, *y* is the actual state of the output unit, and *d* is its desired state. The simplest version of gradient descent is to change each weight by an amount proportional to the accumulated $\partial E/\partial w$: i.e.,

$$\Delta w = -\eta \partial E / \partial w \tag{3}$$

where η is the learning rate.

A neural network's generalization ability, or the power to handle unseen data, is crucial. The generalization ability can be measured by a set of testing samples in the recall

phase. If a trained neural network generalizes well, it can be safely used to process the whole data set. BP networks have been successfully used for remote sensing image classifications (Hepner et al., 1990; Benediktsson et al., 1990; Civco, 1993) and GIS spatial analysis and modeling (Sui, 1993; Sui, 1994; Wang, 1992; Fischer et al., 1994). As reported by Sui (1994). Openshaw empirically tested a feed-forward neural network as the basis for representing a spatial interaction model contained within the journey-to-work data. Sui (1993) successfully integrated a three-layer backpropagation neural network with GIS for a development suitability analysis and found that a neural network can make a close approximation of experts' decisions without the explicit expression of experts' knowledge into "if-then" production rules. Wang (1992) used backpropagation neural networks to strengthen the spatial data modeling capabilities of GIS. Openshaw believes that neural computing has the potential to revolutionize many areas of urban and regional modeling by providing a general-purpose system modeling tool (Sui, 1994). Fischer and Gopal (1994) used a BP network for interregional telecommunication traffic analysis in Austria and found it outperformed the traditional regression approach.

However, there are some commonly recognized drawbacks with a BP network. The following are some of them:

- Learning is slow. This is due to the local minima and flat areas on the "error surface," which trap the gradient-based BP or make it perform slowly. It is also very costly to compute the gradient in BP because it needs the information to pass through the network twice: samples pass through the network and then the error passes backward;
- BP is very sensitive to the initial connection weights and parameters such as learning or momentum rates; and
- There is no model found in natural systems corresponding to the BP. It seems incompatible because neural networks are originally derived from simulating neural systems.

Genetic Algorithms

A genetic algorithm (GA) is a global search method simulating natural evolution (Holland, 1975; Goldberg, 1989). The evolution begins with a group of randomly initialized feasible solutions, or chromosomes in GA terms. These chromosomes compete to reproduce offsprings based on the Darwinian principle of survival of the fittest. Hopefully, after a number of generations of evolutions, the chromosomes remaining in the group are the optimal solutions.

A chromosome consists of a string of bits. There is a value of fitness associated with each chromosome. The objective of GA is to maximize the fitness function. A mechanism in GA called selection operation gives more chance of reproduction to the chromosomes with larger fitness. The selection may be based on the actual value of fitness (e.g., roulette wheel selection) or on the order of fitness (e.g., rank selection). The chromosomes reproduce offspring through two basic genetic operations: mutation and crossover. Mutation is bit flipping. For example, the following child chromosome is derived by flipping the bold bit <u>0</u> in parent chromosome to <u>1</u>.

Child: 01111001

Crossover divides two chromosomes at the same position and swaps portions as follows:

Parent 1: 011 11001	Child 1: 011 00000
Parent 2: <u>111</u> 00000	Child 2: <u>111</u> 11001

The general operation of GA is as follows:

(1) A population of chromosomes is randomly initialized.

(2) Each chromosome's fitness is evaluated.



- (3) A new population of chromosomes of the same size is generated in the following way:
 - (a) Through selection operation, the parent chromosomes are chosen from the current population.
 - (b) Parent chromosomes reproduce children by means of mutation, crossover, or simple duplication. Which way of reproduction is stochastically determined by the possibilities associated with them.
- (4) The old population is replaced with the new one.
- (5) If conditions of maximum of fitness or iterations are satisfied, stop; otherwise, return to step (2).

The global search ability of bit-representation genetic algorithm can be explained mathematically by Schemata Theorem. Genetic algorithms have been expanded to real numerical representation (Davis, 1991), where a chromosome is a string of real numbers. The crossover is the same as in bit string representation. The mutation is implemented as creeping, adding a small random number at a randomly selected position in the chromosome. The detailed discussion of the real numerical representation GA can be found in Qi and Palmieri (1994 a; 1994b).

In GA, successive populations of feasible solutions are generated in a stochastic manner and multiple solution trajectories proceed simultaneously, allowing interactions among them toward one or more regions of the search space (Qi *et al.*, 1994a). This is in contrast to the gradient-descent based BP method that follows one trajectory along gradient descent.

There are several advantages of using GA as the learning algorithm of a neural network:

- GA is not easily trapped in the local minima;
- The heavy burden of gradient computation of BP is avoided, because no gradient information is needed in GA;
- There is no tedious work of selecting parameters such as learning or momentum rates in BP; and
- The learning process is internally parallel.

To apply the GA to the learning problem of a neural network, two things need to be done: (1) selection of a fitness function, and (2) expression of a neural network in a chromosome form. Because GA is intended to maximize the fitness, we use -E, the negative of the error function (Equation 2), as the fitness function. In most applications reported in the literature, fitness function is positive. However, what really matters is the order of the fitness because we use the rank selection paradigm. In a feedforward network, each hidden or output neuron with its accepting connections usually forms the "functional block," which is sensitive to certain features. The crossover tends to conserve the shorter fragment in a chromosome while destroying the longer one. To reduce the probability of the beneficial function block being destroyed by crossover, the weights of connections entering the same hidden or output neuron are placed together in the chromosome. Figure 3 illustrates how a neural network is represented by a chromosome, actually a string of real numbers.

There are often a number of different solutions to any one neural network (Montana *et al.*, 1989). This can make GA slow to evolve (Whitley *et al.*, 1990). For example, assuming that to solve a problem three hidden neurons, each performing task A, B, and C, are needed, then Figure 4 shows how an ill-functional neural network is reproduced by beneficial parents through crossover. To overcome this difficulty, it is suggested that a much higher level of mutation and a small population of chromosomes be used (Montana *et al.*, 1989; Whitley *et al.*, 1990; Dominic *et al.*, 1991).

We have developed a set of neural network and genetic algorithm routines using C++. To accomplish the suitability analysis task, a complete software system was developed, which has modules including image processing, annotation, statistics, training sample selection, neural network construction and training, neural network database management, and data import/export, among other functions.

An Application of Genetic Learning Neural Network for Suitability Analysis

We used an exercise in the *IDRISI for Windows: Student Manual* (Eastman, 1995) as an example. The problem is to find all areas suitable for the location of a light manufacturing plant. The manufacturing company is primarily concerned that the site chosen be on fairly level ground (less than 2.5 degrees). The local town officials are concerned that no facility be close to (within 250 metres) any reservoirs. Additionally, only forested land is to be considered. To summarize, the following three criteria must be satisfied (for simplicity, the minimum area constraint is ignored):

- on land with a slope less than 2.5 degrees;
- outside a 250-metre buffer around reservoirs; and
- on land currently designated as forest.

Two maps available were a DEM (digital elevation map) and a land-use map, from which we could derive four maps using appropriate IDRISI functions. They are a continuous slope map (m_slope, Figure 5a), a binary map showing the area where the slope is less than 2.5 degrees (b_slope, Figure 5b), a binary map indicating outside the 250-metre buffers around reservoirs (b_dis, Figure 5c), and a binary forested area map (b_forest, Figure 5d). The larger slope is represented by the lighter grey in Figure 5a. All these images have the size of 72 by 86 pixels.

Using the Overlay/Multiply procedure within IDRISI, the



Figure 4. Problem caused by multiple solutions to a neural network. To solve the problem, three hidden neurons, each performing task A, B, and C, are needed. This figure illustrates how an ill-functional neural network is produced by two good neural networks through crossover operation.



resulting binary suitability map (b_result, Figure 5e), is derived by multiplying b_slope, b_dis, and b_forest (Figures 5b, 5c, and 5d). Obviously, all these criteria are considered equally important.

Five genetic learning artificial neural networks (GLANNs) were designed and tested. These neural networks had some common features. Because of the small size of our training samples, all the neural networks were small ones with only five hidden neurons. They had a single output neuron. Each network had several input neurons associated with the factors. The output neuron produced numerical values ranging from 0 to 1. The function of the neural network was to project multiple dimension criteria space into one dimension evaluation space. For the parameters of the genetic algorithms, there were 35 chromosomes in the population, and the probabilities of mutation and crossover were 0.3 and 0.7, respectively (we also tried other settings of population size and genetic operation possibilities, and found that GA was not sensitive to these parameters). The entire data set was used as the testing set and processed by the trained neural network in the recall phase. The results were compared with that derived from the conventional overlay in IDRISI (Figure 5e). Figure 6 illustrates the structures of these neural networks and how they work to perform the suitability analysis. These experiments are presented individually with some discussion.

Experiment 1. GLANN Learns from the Explicit Overlay Rules

The neural network had three input neurons (Figure 6a), associated with b_slope, b_dis, and b_forest (Figures 5b, 5c, and 5d). The neural network was trained with the eight combination rules as follows:

Input	Expected Output	Input	Expected Output	
1,1,1	1	0,1,1	0	

1,1,0	0	0,1,0	0
1,0,1	0	0,0,1	0
1,0,0	0	0,0,0	0

In the recall phase, the neural network can produce exactly the same result as that derived using conventional procedure in IDRISI (Figure 5e). The experiment shows that the traditional method of overlay or MCE could be replicated with a neural network. The rules for generating the suitability map are implicitly stored in the neural network as connection weights. Openshaw believes that, in both theory and practice, it is possible to develop neural networks equivalent to virtually all the existing spatial and non-spatial models for a wide range of applications (Sui, 1994).

For the following four experiments, 29 sample sites were selected — 15 for unsuitable and 14 for suitable — with the aid of the slope map (Figure 5b), distance map (Figure 5c), forest map (Figure 5d), and the suitability map derived from IDRISI (Figure 5e). With these sites of known suitability, we could extract 29 training samples for each experiment from the relevant factor maps.

Experiment 2. GLANN Learns from Samples

The structure of the neural network was the same as the first neural network (Figure 6a). The 29 training samples were extracted from b_slope, b_dis, and b_forest (Figures 5b, 5c, and 5d). We had no preconceived notions about the decision rules this time. However, a neural network extracted these rules through learning from the samples and implicitly expressing them in the form of connection weights. Learning from samples is one of the most prominent advantages of ANN over other static models. Some authors argue that the role of expert systems in socio-economic modeling will be limited because the static nature of the knowledge acquisi-



tion process in expert systems cannot reflect the dynamics of complex spatial interaction (Sui, 1994). Through learning from samples, ANNs can adapt themselves to the dynamic change of the environment.

Experiment 3. GLANN Automatically Discriminates the Criteria Relevant to the Problem

The neural network had four input neurons (Figure 6b). The 29 training samples were extracted from b_slope (Figure 5b), b_dis (Figure 5c), b_forest (Figure 5d), and a noise map (b_ noise, Figure 5f). The pixels on this noise map are randomly set to the value of 1 or 0 at equal possibility. In the recall phase, the neural network also produced exactly the same result as in Figure 5e. The irrelevant input (noise) did not affect the result. This shows that the neural network can automatically discriminate the relevant criteria to perform the task through evolutionary learning from samples.

Experiment 4. GLANN Handles Interdependency between Factors

The neural network had five input neurons (Figure 6c). In addition to b_slope (Figure 5b), b_dis (Figure 5c), b_forest (Figure 5d), two additional versions of the b_slope map were also used as redundant inputs. The 29 training samples were extracted from these five factor maps. The neural network also produced exactly the same result as in Figure 5e in the recall phase. The interdependent layers did not bias the results. In theory, a neural network can accomplish any nonlinear mapping, so it is possible to overcome interdependency among factors.

Experiment 5. GLANN Deals with Data with Multiple Measurement Scales and Inaccuracy

The neural network had three input neurons (Figure 6d). The 29 training samples are extracted from the continuous slope map, b__dis, and b__forest (Figures 5a, 5c, and 5d). No normalization was performed for the slope data. The output of the trained neural network was a continuous evaluation map (Figure 7a). The histogram of this evaluation map is given in Figure 7b, where the horizontal coordinate represents the evaluation value and the vertical coordinate is the frequency of the pixels at that value. We can select a threshold along the horizontal axis to control the total area of suitability. In our system, this is done through adjusting a scroller under the histogram and examining the change of the resulting map. The suitability binary map was the same as the b_result (Figure 5e) when the threshold ranges were approxi-



mately between 0.5 and 0.9 (Figure 7b). We selected 0.85 as the threshold. As mentioned before, a neural network can perform a projection from high dimensional criteria space into one dimension evaluation space, thereby facilitate the decision making process.

Figure 8a shows the curve of suitability values derived from the neural network. It is created by calculating the output of the neural network when fixing the input of both neurons associated with b_dis and b_forest (Figure 6d) to a value of 1 and continuously changing the input of the neuron associated with the slope layer. We could see that the output of the neural network is reasonable and somewhat "fuzzy." With the increasing of the slope, the suitability value continuously decreases from 1 to 0.

It is commonly recognized that slope data, like other spatial data, generally contain errors and inaccuracy. By simply adding a random error of a uniform distribution ranging from -0.5 to 0.5 degree to the slope, we got a suitability

curve for inaccurate slope as shown in Figure 8b using the same neural network. The inaccuracy in the measurement introduces some oscillation in the suitability value. Using the conventional overlay methods illustrated above, we also derived the suitability value curves for the perfect and noisy slope data shown in Figures 8c and 8d, respectively. Figure 8d indicates that the inaccuracy in slope causes the high vibration of the suitability value in the neighborhood of 2.5 degrees. From the above observations, we could conclude that the neural network is more tolerant to inaccuracy in data than traditional methods which involve the discretization of continuous data. The discretization of continuous data usually imposes an oversimplification of the problem, and is somewhat subjective and rigid. For example, it is arguable that 2.49 degrees is fairly level but 2.51 is not.

From this work, it can be seen that the neural network provides a promising approach to deal with the inaccuracy in geospatial data. If we can guarantee the accuracy of a small number of training samples, then the neural network may learn and generalize correctly and can be used to handle inaccurate or noisy data.

Pereira and Duckstein (1993) used a value function to process the original continuous data, and the continuity of the data could be conserved. However, the value function needs careful design. With a neural network, the original data could be directly accepted. A neural network could conduct an "implicit scaling" through learning from samples.

About the User Interface

Working with neural networks, the main task that the user needs to perform is to identify a set of samples (Figure 9). The selection of criteria need not be extremely refined. The neural network can automatically select the relevant factors to perform the task through evolutionary learning. It is permissible that there may exist interdependency between some factors. All these features were illustrated in the experiments described previously in this paper.

A prominent advantage of neural network approaches is that a user can focus on the problems themselves rather than on the details of the techniques. The user can utilize differ-



figure 8. (a) Suitability curve derived by NN for accurate slope data. (b) Suitability curve derived by NN for inaccurate slope data. (c) Suitability curve derived by overlay for accurate slope data. (d) Suitability curve derived by overlay for inaccurate slope data.



ent factor maps, remote sensing images, *in situ* data, and other ancillary maps or nongraphical data such as text, tabular, etc., and evaluate the sample sites using the expert knowledge, experience, intuition, creativity, and imagination.

Overlay, MCE, and neural network approaches should all permit user loop back. With neural networks, when errors occur in the final result, the user can easily identify and correct them. For example, in one experiment, we found the resulting suitable area derived from the neural network was the same as the area outside a buffer of 250 metres around the reservoirs (Figure 5c). A visual examination of the training samples showed that all the suitable samples were outside the buffer and all the unsuitable samples were within. It is understandable that the neural network generalized incorrectly. This error was corrected by adding several unsuitable points outside the buffer and training the neural network again.

With the traditional methods, we can begin to work only after we have acquired complete and explicit knowledge while, with neural networks, we can refine our results progressively by improving the quality of the training samples. If we have any new evidence about the suitability of a site, we could add it to the training samples. If we find an error in a training sample, we could remove or edit it. Then we need merely to retrain the neural network. So neural network methods are dynamic and compatible with the human cognitive process.

Problems with Neural Networks

The neural network has its own problems. The optimal structure of the neural network, i.e., how many hidden layers and how many neurons in each hidden layer, is still unclear for the specific task. Some general principles available for designing the structure are mainly based on experience. For example, De Villiers *et al.* (1992) suggest that a neural network with one hidden layer may be more preferable than one with two hidden layers in terms of learning speed and performance. One of the rules to estimate the number of hidden neurons is the geometric pyramid rule proposed by Masters (Wang, 1995). Genetic algorithms are also a promising approach to optimize the structure of neural networks. Some pioneering work has been done in this field (Whitley *et al.*, 1990; Koza *et al.*, 1991).

Another problem is "overfitting," which corresponds to some extent with the structural problem. When "overfitting" occurs, a neural network learns well from the training samples but performs poorly for unseen data. One method to overcome this problem is to use the fewest hidden neurons as possible to guarantee the most conservative generalization. A technique used by Fischer and Gopal (1994) to detect when overfitting occurs is called cross-validating. In their methods, a validation test set is used for the evaluation of the learning process in addition to the training and testing samples.

These problems did not surface in our experiments because of the simplicity of the problem. However, caution must be taken when applying neural network to larger, more complicated real world tasks.

Conclusions

Traditional methods of overlay and multicritera evaluation can be replicated or even replaced by genetic learning neural networks. Many difficulties in them are overcome by the evolutionary learning and nonlinear mapping ability of the neural network. A genetic learning neural network can process data of multiple measurement scales, continuous or discrete, and produce a somewhat fuzzy rather than rigid result. It provides a promising approach in that, if we can guarantee the accuracy of a small number of training samples, then the neural network may learn and generalize correctly and can handle inaccurate or noisy data. The selection of criteria need not be extremely refined because the neural network can automatically discriminate the factors relevant to the problem through learning. It is permissible that there may exist interdependency between factors. Much of the tedious work in traditional suitability analysis, i.e., the requirements of explicit knowledge to identify criteria, assign scores, determine criteria preference, and select an aggregation function are replaced by the evolutionary learning.

With a neural network, the user can focus on the problems themselves rather than the details of techniques. The main task of the user is to define a set of training samples for which a measurement of suitability is known. The user can utilize expert knowledge, experience, and creativity to evaluate the sample sites with the aid of related factor maps, remote sensing images, *in situ* data, other ancillary maps, or nongraphical data. Any of the traditional methods can also be used to evaluate the training samples. Compared with the traditional overlay and multicriteria evaluation methods, the neural network approach is more amenable to user feedback.

Although neural networks have structural and overfitting problems, they can provide an alternative for and improvement over traditional methods for suitability analysis in GIS.

Acknowledgment

The research upon which this paper is based was funded in part by the Storrs Agricultural Experiment Station under Project CONS00679, "Forest Landscape Ecosystem Assessment through Satellite Remote Sensing and Neural Processing." The authors extend their thanks to the three reviewers whose insightful opinions helped to refine this paper.

References

- Benediktsson, J.A., P.H. Swain, and O.K. Ersoy, 1990. Neural Network Approaches versus Statistical Methods in Classification of Multisource Remote Sensing Data, *IEEE Transactions on Geosci*ences and Remote Sensing, 28(4):540–551.
- Bonham-Carter, G.F., 1994. Geographic Information Systems for Geoscientists: Modeling with GIS, Pergamon, Oxford, New York.
- Carver, S.J., 1991. Integrating Multi-Criteria Evaluation into Geographic Information Systems, International Journal of Geographical Information Systems, 5(3):321–339.
- Civco, D.L., 1993. Artificial Neural Network for Land Cover and Mapping, International Journal of Geographical Information System, 7(2):173–186.
- Davis, L., 1991. Handbook of Genetic Algorithms, Van Nostrand Reinhold, New York.
- De Villers, J., and E. Barnard, 1992. Backpropagation Neural Nets with One and Two Hidden Layers, *IEEE Transactions on Neural Networks*, 4(1):136–141.

- Dominic, S., R. Das, D. Whitley, and C. Anderson, 1991. Genetic Reinforcement Learning for Neural Networks, Proc. IJCNN'91, Seattle, 2:71–76.
- Eastman, J. R., 1995. *Idrisi for Windows Student Manual*, Clark University, Worcester, Massachusetts.
- Fischer, M.M., and S. Gopal, 1994. Artificial Neural Networks: A New Approach to Modeling Interregional Telecommunication Flows, *Journal of Regional Science*, 34(4):503–527.
- Goldberg, D.E., 1989. Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Publishing Company, Reading, Massachusetts.
- Hammond, R., and P. McCullagh, 1974. *Quantitative Techniques in Geography: An Introduction*, Clarendon Press, Oxford.
- Hepner, G.F., 1984. Use of Value Functions as a Possible Suitability Scaling Procedure in Automated Composite Mapping, *Professional Geographer*, 36(4):468–472.
- Hepner, G.F., T.Logan, N. Ritter, and N. Bryant, 1990. Artificial Neural Network Classification Using a Minimal Training Set: Comparison to Conventional Supervised Classification, Photogrammetric Engineering & Remote Sensing, 56(4):469–73.
- Heywood, I., J. Oliver, and S. Tomlinson, 1995. Building an Exploratory Mulit-criteria Modeling Environment for Spatial Decision Support, Innovations in GIS, Taylor & Francis, pp. 127–136.
- Hoffman, R.R., 1987. The Problem of Extracting the Knowledge of Experts From the Perspective of Experimental Psychology, AI Magazine, 8(2):53–67.
- Holland, J.H., 1975. Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor.
- Hopkins, L., 1977. Methods of Generating Land Suitability Maps, Journal of American Institute of Planning, 43:386–400.
- Hwang, C.L., and K. Yoon, 1981. Multiple Attribute Decision Making Methods and Applications: A State of the Art Survey, Springer-Verlag, Berlin.
- Jankowski, P., 1995. Integrating Geographical Information Systems and Multiple Criteria Decision-Making Methods, International Journal of Geographical Information Systems, 9(3):251–273.
- Janssen, T., 1992. Multiobjective Decision Support for Environmental Management, Kluwer, Dordrecht.
- Koza, J.R., and J.P. Rice, 1991. Genetic Generation of Both the Weights and Architecture for a Neural Network, IJCNN'91, Seattle, (2):397–404.

- Kreinovich, V., and O. Sirisaengtaskin, 1993. 3-Layer Neural Networks Are Universal Approximators for Functionals and for Control Strategies, Neural, Parallel & Scientific Computations, 1(3):325–346.
- Macdougall, E.B., 1975. The Accuracy of Map Overlays, Landscape Planning, 2:23–30.
- Minch, R.P., and G.L. Sanders, 1986. Computerized Information Systems Supporting Multicriteria Decision Making, *Decision Science*, 17(3):395–413.
- Monata, D.J., and L. Davis, 1989. Training Feedforward Neural Network Using Genetic Algorithms, Proceedings of the Eleventh International Joint Conference on Artificial Intelligence, 1:762–767.
- Pereira, J.M.C., and L. Ducktein, 1993. A Multiple Criteria Decision-Making Approach to GIS-Based Land Suitability Evaluation, International Journal of Geographical Information Systems, 7(5): 407–424.
- Qi, X., and F. Palmieri, 1994a. Theoretical Analysis of Evolutionary Algorithms With an Infinite Population Size in Continuous Space Part I: Basic Properties of Selection and Mutation, *IEEE Transactions on Neural Network*, 5(1):102–119.
- —, 1994b. Theoretical Analysis of Evolutionary Algorithms With an Infinite Population Size in Continuous Space Part II: Analysis of the Diversification Role of Crossover, *IEEE Transactions on Neural Network*, 5(1):120–129.
- Rumelhart, D.E., and G.E. Hinton, 1986. Learning Representation by Back-propagation Errors, Nature, 323(6088):533–536.
- Sui, D.Z., 1993. Integrating Neural Networks with GIS for Spatial Decision Making, The Operational Geographer, 11(2):13–20.
- ——, 1994. Recently Applications of Neural Networks for Spatial Data Handling, Canadian Journal of Remote Sensing, 20(4):368– 379.
- Voogd, H., 1983. Multicriteria Evaluation for Urban and Regional Planning, London, Pion.
- Wang, F.J., 1992. Incorporating a Neural Network into GIS for Agricultural Land Suitability Analysis, GIS/LIS'92, (1):804–815.
- Wang, Y., 1995. Modularizing Backpropagation Neural Network for Multisource Spatial Data Modeling and Classification, Ph.D dissertation, The University of Connecticut, Storrs, Connecticut.
- Whitley, D., T. Starkweather, and C. Bogart, 1990. Genetic Algorithms and Neural Networks: Optimizing Connections and Connectivity, *Parallel Computing*, 14(3):347–361.

10 Things You Can Do at the ASPRS Website

- 1. Check out the GIS/LIS '96 page.
- 2. Apply for ASPRS Certification.
- 3. Contact Region or Division Directors, Committee Members...the ASPRS President!
- 4. Check out the PE&RS Guidelines for Authors.
- 5. Apply for ASPRS Scholarships and Fellowships.
- 6. Join, and learn about the benefits of ASPRS Membership.
- 7. Nominate someone for an ASPRS Award.
- 8. Check out and order ASPRS publications.
- 9. Ask and answer questions on the Discussion pages.
- Find exactly what you need by searching the Directory of the Mapping Sciences by company, product, service, and location.

