Doerr, Martin and Apostolis Sarris (eds) 2003. The Digital Heritage of Archaeology. CAA2002. Computer Applications and Quantitative Methods in Archaeology. Proceedings of the 30th CAA Conference, Heraklion, Crete, April 2002.

Archaeological Predictive Modelling in Intelligent Network Structures

Benjamin Ducke

Project Archäoprognose Brandenburg,
State Service for Heritage Management in Brandenburg and Archaeological State Museum,
Wünsdorfer Platz 4-5
15838 Wünsdorf, Germany
ducke@inf.fu-berlin.de

Abstract. The statistical prediction of archaeological site locations requires advanced methods that can model non-linear, discontinuous functions. While traditional statistical techniques can satisfy these requirements to some extent, recent progress in artificial intelligence research has made better tools available: artificial neural networks in particular stand out as efficient, easy-to-use tools with a broad range of possible applications. This paper discusses design, organisation and use of artificial neural networks in archaeological predictive mapping tasks.

Key words: archaeological predictive modelling, site location prediction, connectionist systems, artificial neural networks, classification.

1 Introduction

The work presented in this paper is part of an ongoing project that aims to create an efficient predictive modelling toolkit for archaeological resource management in the state of Brandenburg in north-eastern Germany (an area covering about 300.000 sqkm; for more details, refer to the contribution of U. Münch in this volume and the project home page: http://www.uni-bamberg.de/~ba5vf99/index.html; also Kunow and Müller 2001).

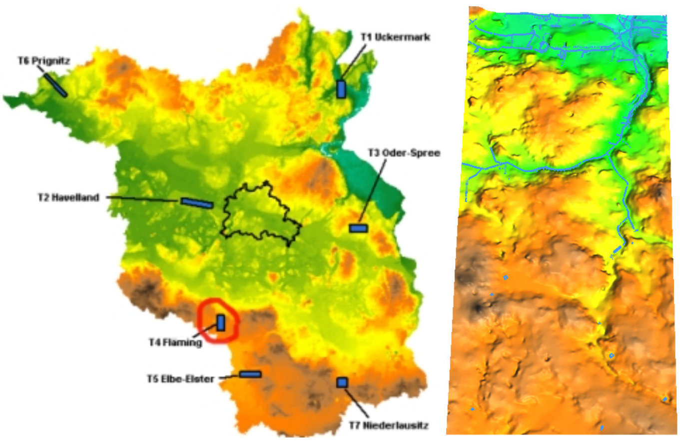


Fig. 1. *Brandenburg in north-eastern Germany (left). The black outline indicates the city limits of Berlin. The sample area (right) that provided the data for this paper is circled in red. Its dimensions are 4x9 km.*

Archaeological predictive modelling (APM) tries to develop and establish methods suitable for predicting potential locations of archaeological sites in geographic space utilising regression and classification techniques, spatial statistics and heuristic approaches. This paper outlines yet another, sophisticated approach to APM on a landscape scale based on GIS, geo-archaeological data sets and the capabilities of advanced symbolic processing. Geographic information systems (GIS) are an essential part of APM because they provide the tools for storage and analysis of spatial variables as well as the interface between geo-archaeological information and the advanced numerical processing underlying all archaeological predictive models.

Obviously, all large scale aproaches to APM that are based on geo-archaeological information (such as Hudak et al. 2002, Zeidler 2001) must assume that settlement patterns are strongly determined by environmental characteristics (attributes) such as terrain geometry (height, slope and aspect angle), distance from surface water, ground water level, soil texture, soil quality etc. The attractiveness of different locations for prehistoric settlements is revealed by patterns in the spatial distribution of archaeological sites in surveyed areas. The specific nature and importance of location attributes may vary according to the type of landscape and archaeological culture under study. The area under study in this paper is part of the European Lowlands, a landscape that experienced its last phase of geomorphological shaping by glacial processes about 12000 years ago. Throughout the Holocene period, its environmental setup was defined by broad, gentle slopes and dense stream networks that allowed for an indiscriminate use of land and nearly constant settlement patterns from the Neolithic to the Iron Age on a large scale (Bork et al. 1998). For this type of landscape, a predictive model may treat sites of various prehistoric periods indifferently.

2 Artificial Neural Networks

An ANN (Artificial Neural Network) is essentially a mathematical model that aims to capture the basic mechanisms of information processing in biological systems such as the human brain (Rojas 1996). A wide variety of ANN architectures have been implemented as computer programs such as SNNS (Stuttgart Neural Network Simulator), a freely available software package for use on Unix and Windows operating systems (the SNNS programs and accompanying documentation have been used extensively for obtaining the results presented in this paper: http://www-ra.informatik.uni-tuebingen.de/SNNS). ANNs, being connectionist systems, are composed of many primitive units (equivalent to the neurons in the human brain, Fig. 2) that work in parallel and are linked via directed connections. In analogy to the dendrites and axons of biological neurons, each unit of an ANN has one or more in-going connections and only one out-going connection. The main principle of information processing in a neural network is the distribution of activation patterns across these connections.

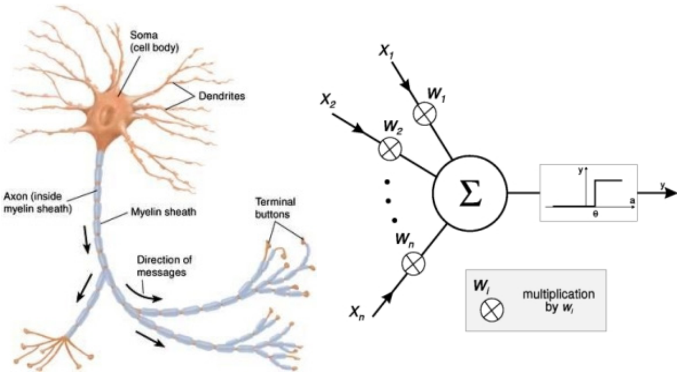


Fig. 2. *Scheme of a biological neuron and a primitive unit of an artificial neural network.*

In analogy to the way that chemical transmitters transport signals across neurons in the human brain, a mathematical function (learning algorithm) controls the "transport" of numerical values through the connections of an ANN. If the strength of a signal arriving in a neuron exceeds a certain threshold-value, the neuron will itself become active and "fire", i.e. pass on the signal through its outgoing connection. In the case of an ANN, this process is modelled by primitive units which individually and independently form a weighted sum of incoming signal values, process them through a simple activation function and pass the output on to another unit. According to simple neuro-psychological models, this mechanism is sufficient to enable biological organisms to achieve highly complex cognitive tasks. In fact, the high performance of biological cognitive systems in tasks involving classification and pattern recognition has been one of the biggest motivations for the development of ANNs.

ANNs are being applied across a broad range of problem domains, in areas such as marketing, medicine, engineering, geology, physics and archaeology – anywhere that there are problems of prediction, classification or control. One of the reasons for their popularity is without a doubt the fact that ANNs are easy to use because, like their biological counterparts, they learn by example (White 1989). Although one needs to have some prior knowledge of how to select and prepare data, select and design an appropriate ANN, and how to interpret the results, the level of user knowledge needed to successfully apply ANNs is generally much lower than would be the case using more traditional statistical methods. Using an artificial neural network for classification is a procedure that consists of three basic steps (train, test, use).

3 Artificial Neural Networks as Predictive Models

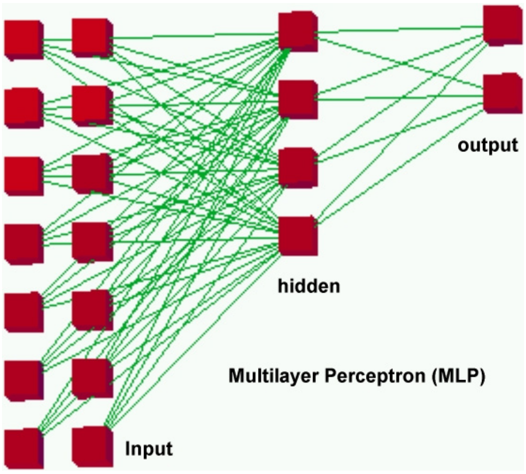
For a basic understanding of the role of ANNs in APM, it is sufficient to think of an ANN as a statistical black box system that maps input cases (geographic locations) to output classes (e.g. “low”, “medium” or “high” site probabilities). For an in-depth understanding of an ANN's statistical properties, many technical details have to be known, most of which can only be discussed superficially in this brief paper (an excellent, comprehensive overview of the applicability and statistical properties of the most popular ANN architectures is available online as part of the StatSoft Electronic Textbook at http://www.statsoftinc.com/textbook/stathome.html). From a general statistical perspective, ANNs can be used to replace traditional regression and classification methods with a more intuitively appealing design. Since the statistical problems of APM have already been tackled in an number of ways (for methodical reviews see e.g. Kohler and Parker 1986, Kvamme 1990, Leusen 1996) the application of ANNs for APM could be considered just another blind use of technology. However, the specific value of ANNs for APM lies in their ability to efficiently handle the most severe problems that archaeological data usually suffers from:

Missing or incomplete data Archaeological data is generally imperfect in that some information may not be available at the same level of quality for all objects under study or even missing completely. The data processed by an ANN does not have to be perfect because missing and incomplete data is approximated as part of the training process (see next section).

Noisy data More often than not, archaeological data is distorted by influences of a statistically random nature (noise). The spatial distribution of archaeological sites, although primarily driven by environmental factors, might include political, social and religious motivations that originally had an influence on landscape usage by prehistoric settlers. Since these motivations remain blurred and the strength of their effect on site distributions cannot be quantified, there is no way to analytically compensate for them. However, ANNs, due to their non-analytical nature, behave robustly even if the input is noisy. In fact, with noise rising fast, the quality of the output degrades only slowly ("graceful degradation").

Non-linear relationships The probability that an archaeological site exists in a given location can be considered a function of that location's attributes. APM can then be defined as the task of finding the function that most accurately maps environmental parameters to find probabilities. Although an optimal function will in practice never be found, even a moderately accurate mapping function can be very complex and difficult to find in an analytical way. By learning from input cases alone, ANNs are capable of modelling extremely complex functions (ANNs themselves can be viewed as functions that map input cases to output cases). Although there are other ways to model non-linear relationships, ANNs are equalled by no other statistical tool in their ability to model non-linear functions with large numbers of variables. Classification with an ANN involves creating, training, testing and finally using the network. Each of these steps will be described briefly in the following subsections.

3.1 Designing an Artificial Neural Network



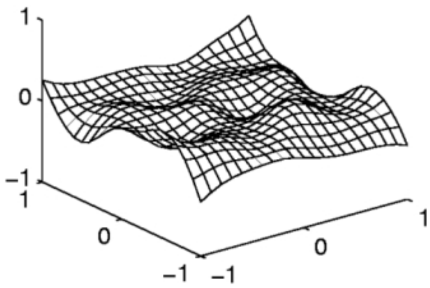


Fig. 3. *A simple multilayer perceptron network (top) and the error surface of a two-dimensional input vector (bottom).*

A peculiar property of ANN-based classification compared to traditional statistics is that there is no standardised way to take the user from unstructured input data to classified output data. Instead, the architecture of an ANN is always bound to the specific data (and the underlying distribution function) that constitutes the basis of the task at hand. The first step, therefore, is to find an individual ANN design. To illustrate the basic concepts involved, a very simple ANN architecture will suffice. The ANN shown in Fig. 3, a so-called "multilayer perceptron network", is one of the most basic architectures (Baum 1988). It was designed to achieve a simple classification of input locations as either "site present" or "no site present".

Apart from its general architecture, an ANN's statistical properties are defined by the layout of its units and connections. In most cases, an ANN consists of at least four components:

1. An input layer with as many inputs as there are predictor variables (location attributes). This is the layer where the data to be learned and later classified is fed into the network.
2. A so-called "hidden" layer with, in this simple example, four units. These hidden units are essentially responsible for determining the form of the mathematical function that the network represents. The number of hidden units is determined experimentally by adding or removing hidden units and checking how well the net performs afterwards. If one adds more hidden units, the function approximated by the network will fit the training data more perfectly. However, if the fit is too perfect, the network will lose its ability to generalise. This means that classification performance for unknown cases (those not presented to the network during training) will degrade. This condition is called *over-learning*. If the number of hidden units is too small, *under-learning* occurs and the network will never achieve a good performance.
3. An output layer with just two units. These correspond to the two possible cases "site present" and "no site present". The output units have to functions: during training, they are set to predefined values by the user (*supervised learning*; see next subsection). During classification, the network uses them to store classification results for an input case.
4. The final component is a full set of connections between the units. It is in these connections that the net stores its actual "knowledge". For this purpose, there is a weight attached to each connection that represents its activation strength. The weights change during the learning process. Setting them to their optimal values is a mathematical challenge addressed by a number of different learning algorithms, the most well-known (although in practice not the most useful) of which is the back propagation algorithm (Rumelhart, Hinton, Williams 1987).

There is a vast number of ANN architectures and each of them allows for further variations in the number of units and layers, the number of connections between the units and the mathematical functions that control the signal passing. However, in practice only a few architectures are useful. Again, this paper cannot explain every technical aspect that one has to be aware of when trying to find the optimal network architecture, layout and learning function. A very helpful resource is the Neural Network FAQ (ftp://ftp.sas.com/pub/neural/FAQ.html).

Designing an ANN involves experimenting with a large number of different networks, training each one a number of times and observing individual performances. However, following the standard scientific motto that, all else being equal, a simple model is always preferable, one should always select a smaller, simpler network in preference to a larger one with a negligible improvement in performance. Recent advances in software development also include end user software packages that automate this process (e.g. http://www.statsoftinc.com/stat_nn.html).

Fig. 4 shows the classification result of the simple design described above. Although it was able to achieve a rough classification, the overall picture gained from its output is rather noisy. Also, limiting the predictive model to just two output cases ("site present", "no site present"), might not be what one expects of a useful model.

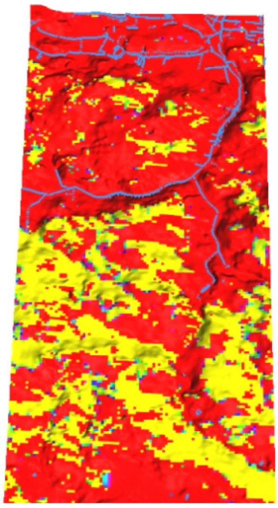


Fig. 4. Classification output for the sample area using a multilayer perceptron network, superimposed on a digital elevation model. Red areas are those classified as "site present", yellow as "no site present".

3.2 Training an Artificial Neural Network

As was stated earlier, an ANN's knowledge (its ability to correctly classify objects) is stored in the weights of its connections. The weights are modified by training methods that replace traditional classification and regression techniques with "best match" conclusions. Training, being the second step in ANN-based classification, adjusts the weights of the connections in order to get the desired behaviour. The site attributes used to train the net are organised as patterns (vectors). For each individual geographic location, a corresponding pattern exists that contains all descriptive attributes (height, slope, aspect angle, soil texture...).

The training process itself is one of supervised learning: patterns representing locations for which the archaeological classification ("site present" or "no site present") is already known (the training pattern set) are repeatedly presented to the ANN which then arranges the weights attached to its connections to map the input data (stored in the input units) to the given output classes (values of the output units). The degree of freedom the ANN has for finding a good mapping function depends largely on the number of units in the hidden layer in between the input and output layers. Pattern by pattern, the ANN then "learns" to recognise the characteristics of site locations and can later identify them in new, unclassified input patterns. To ensure correct training, the standard procedure is to split up the entire training pattern set into three separate parts:

1. The actual training set used to train the ANN. The patterns that constitute this set may be available in the form of survey files, heritage management archives etc.
2. A validation set for testing the performance of the ANN on patterns not in the training set and to find the point at which to stop the training process. Just as an ANN's optimal design depends on the statistical properties of the data to be classified, its knowledge is bound to the input data. ANN performance will therefore only be acceptable if the training process is near optimal. The ANN's error rate (the ratio of correct to incorrect classifications) can always be lowered by training the same patterns many times. The danger is, however, that the ANN gets to highly tuned to the training patterns and loses its ability to generalise. This is another type of over-learning in which training only minimises the error function of the training data instead of the true error function which is sensitive to the complete data set. Therefore, training progress is repeatedly checked using the patterns in the validation set. As training progresses, the training error drops, and (providing training is minimising the true error function) the validation error drops, too. However, if the validation error stops dropping, or starts to rise, this indicates that the network is starting to over fit the data, and training should cease.
3. A test set for finally checking the overall performance of the net with patterns the net has never "seen" during learning.

Because three separate sets have to be supplied with patterns, it is preferable to have a large number of patterns available for training the net. The number of patterns needed also depends on the number of attributes used to describe each location. As a general rule, at least several hundred patterns are needed for moderately complex classification tasks.

The objective of ANN training is to find a set of weights and thresholds that minimise the prediction error. A peculiarity about neural networks is that the error surface of the modelled function is very complex. Its dimensionality directly depends on the number of input units. For a very simple two-dimensional input pattern, the error function can be visualised as a three-dimensional plane (Fig. 3, bottom). In this case, the minimum error can obviously be found in the deepest pit of the surface. Finding this global minimum is essentially the role of the learning algorithm that controls the actual training process by calculating and updating connection weights. In order to increase network performance, the learning algorithm has to descend into the deepest pit of the error surface without getting trapped in a local minimum. This is difficult, because the error surface, due to its complexity, cannot be scanned globally but must be traversed step by step with only limited information about the parts that can immediately be reached in the next training step. Therefore, one can never be sure that training has stopped at an optimal point. The only thing that helps is to train several times in order to avoid getting trapped in a local minimum. For training the ANN, a huge number of different learning algorithms exists and one possibly needs to experiment a bit for optimal results.

3.3 Using an Artificial Neural Network for APM

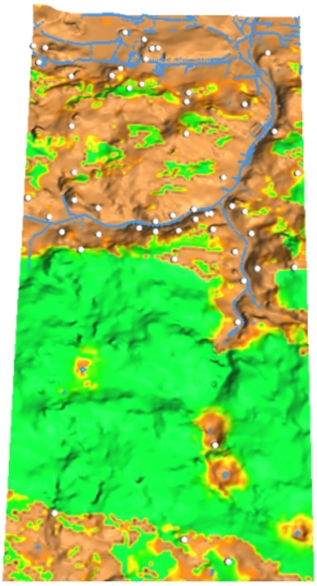


Fig. 5. *Classification output for the sample area using a probabilistic neural network, superimposed on a digital elevation model. White dots represent archaeological site locations. Site location probabilities are colour-coded with dark brown representing highest and light green representing lowest probabilities.*

The training process has essentially created a software expert that is able to discriminate between locations with potential sites and those without. In other words: the ANN has learned to model the unknown function which relates the input variables to the output variables, and can subsequently be used to make predictions for cases where the output is not known. A question that arises immediately is how to interpret the ANN's actual output values. Obviously, they could be interpreted as a degree of confidence in the classification. Formal mathematical proves (e.g. Richard and Lippmann 1991) show that an optimally trained ANN's outputs can be interpreted as true posterior probabilities (the ANN can then be considered a bayesian probability model). This is also generally true for probabilistic neural networks (PNNs), a special ANN architecture (MacKay 1992; Paas 1990). PNNs are essentially the ANN-based version of a statistical method called kernel density estimation, which uses kernel functions – simple functions which are added together, positioned at known data points – to approximate a sampled distribution (see Parzen, 1962; Williams and Rasmussen 1996). Functionally seen, a PNN is able to estimate the true probability function underlying the input data. PNNs are a powerful concept and among the most popular ANN architectures.

So far, it has been assumed that some knowledge about locations classified as both "site present" and "no site present" was available prior to the ANN-based classification. However, in most real world applications, there is only knowledge about the positive indication of site presence. Site absence is seldom recorded systematically and being sure about a site's absence is even more problematic than being sure about a site's presence. In the most general case, one simply wants to make predictions based on some sort of measure of environmental similarity. What is required in this case is an approach to APM that takes an "unbiased" look at the archaeological record by determining data structure exclusively from the data itself. The environmental features represented by the location attributes can then be correlated with the frequency of sites in different environmental feature sets to gain an estimation of archaeological potential for a specific landscape (Fig. 5). The ANN architecture appropriate for this is the self-organising feature map (SOM) as proposed by Kohonen (1982). SOMs are capable of unsupervised learning which means that no training patterns have to be provided. In this respect, SOMs share some properties with exploratory statistical techniques such as clustering analysis but are unique in that they preserve topological information in the data. The classification presented in the next section was achieved using Ainet (http://www.ainet-sp.si/), a public domain software that combines properties of both PNNs and SOMs in a mathematically efficient manner and also offers a graphical user interface for data entry.

4 Performance of ANN-Based APM

Fig. 5 shows the classification output of a PNN. The overall picture is quite homogeneous and thus easy to interpret. Archaeological site locations show a strong preference for valley-type environments and moderate slopes. For comparison, an archaeological predictive model for the same geographic area was processed using a more traditional statistical prediction technique. Using a standard clustering analysis, locations were grouped (clustered) into patches of similar environments according to their attributes. In this way, each group (cluster) represents a particular type of environment. After analysing the distribution of archaeological sites over these environment types, an estimation of archaeological potential could be gained for the rest of the area. A side-by-side visual comparison of PNN and clustering analysis outputs (Fig. 6) uncovers a few differences: Apparently, the clustering analysis tends to form connected patches with a certain degree of internal segmentation. The ANN, owing to its greater modelling power, has found a distribution that includes more unconnected areas (e.g. a patch at the bottom which the clustering analysis left out) and produced a "smooth" overall picture. Validation with split data sets has shown that both models produce acceptable results with about 75-85% of the validation data being correctly predicted. A more convincing but also much more costly field validation of these and other predictive models in altogether seven sampling areas (T1-7, Fig. 1) is still in progress but preliminary results seem to support the assumed validatiy of a geo-archaeological data-driven approach.

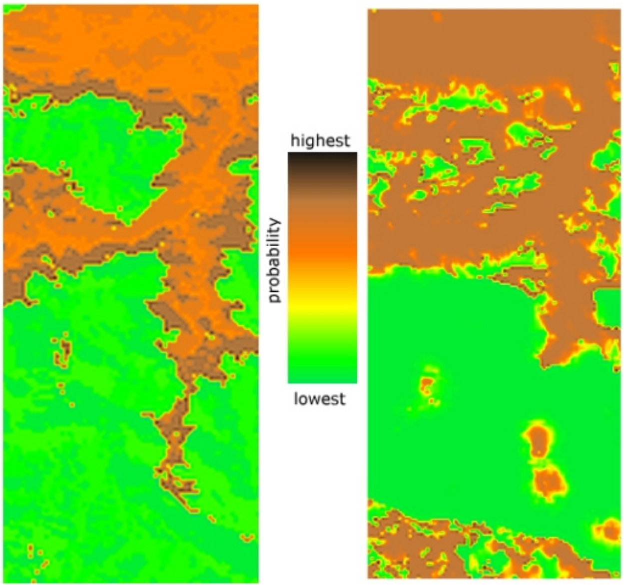


Fig. 6. *Comparison of classification outputs from clustering analysis (left) and probabilistic neural network (right).*

There is one more aspect of great practical implications that has not been stressed yet: one of the biggest practical challenges in APM is the amount of data that has to be dealt with. The sample area shown in Fig. 1 (right) is 4x9 km in size. A model of site distribution using a resolution of 10 m generates 360.000 locations for input as classification patterns. The total input size is thus 360.000 multiplied by the number of location attributes. Input size is often a problem, because statistical techniques tend to have high demands on computing resources. Traditional statistics often include complex calculations which have to be carried out in a predefined order because each intermediate result depends on the preceding operation. In the case of clustering analysis, the input data needs to be sorted and since a set of data points cannot be sorted by considering isolated sub sets, all data has to be held in memory at the same time. In this way hardware limits are quickly reached. It is important to point out that this is not an issue of current PC technology but rather a fundamental mathematical problem (complexity), that many useful methods in traditional statistics will always suffer from, regardless of advances in technology.

ANNs, on the other hand, process information much more efficiently. An ANN may consist of a lot of units, but each single one of them works in a very primitive way: calculations consist of simple signal passing and basic arithmetic operations (sum up all the input signals, process them through some simple activation function and output the result) which can be processed in any order. For the PNN to calculate its output, a PC with 128 MB of main memory was completely sufficient. The fact that such high-quality results can be achieved on inexpensive PCs relieves under-financed heritage management services of a great financial burden and makes high resolution APM for large areas possible.

References

BAUM, E., 1988. On the Capabilities of Multilayer Perceptrons. Journal Of Complexity 4, 193-215.

BORK ET AL., 1998. Bork H.-R., Bork H., Dalchow C., Faust B., Piorr, H.-P. and Schatz, T., Landschaftsentwicklung in Mittel-europa. Wirkungen des Menschen auf Landschaften (Gotha, Stuttgart 1998).

HUDAK ET AL., 2002. Hudak G. J, Hobbs E., Brooks A., Sersland, C.A and Phillips, C. (eds.): A Predictive Model of Precontact Ar-chaeological Site Location for the State of Minnesota. Final Re- port 2002 (Department of Transportation, Minnesota 2002).

KOHLER, T. A. and PARKER, S. C., 1986. Predictive models for ar- chaeological resource locations, in M. B. Schiffer (ed.): Advan-ces in Archaeological Method and Theory 9 (New York 1986), 397-452.

KOHONEN, T., 1982. Self-organized formation of topologically correct feature maps. Biological Cybernetics, 43, 59-69.

KUNOW, J. and MÜLLER, J., 2001. Archäoprognose im Land Bran-denburg. "Die Rekonstruktion ur- und frügeschichtlichen Sied-lungsverhaltens und anthropogener Landschaftsgestaltung", in M. Aufleger, D. Karg, J. Kunow, A. Mikoleietz, R. Paschke, P. Woidt (eds.): Denkmalpflege im Land Brandenburg 1990-2000 (Worms 2001), 612-614.

KVAMME, K. L. 1990. The fundamental principles and practice of predictive archaeological modeling, in A. Voorrips (ed.): Mathe-matics and Information Science in Archaeology: A Flexible Frame-work. Studies in Modern Archaeology 3 (Bonn 1990), 257-295.

LEUSEN, P. M. van, 1996. GIS and Locational Modeling in Dutch Archaeology: A Review of Current Approaches, in Maschner, H. D. G. Maschner (eds.): New Methods, Old Problems: Geo- graphic Information Systems in Modern Archaeological Re-search. Occasional Paper No. 23, Center for Archaeological Investigations, Southern Illinois University at Carbondale (Carbondale 1996), 177-197.

MACKAY, D. J. C., 1992. A Practical Bayesian Framework for Back-propagation Networks. Neural Computation 4(3), 448-472.

PAAS, G.,1990. Probabilistic Reasoning and Probabilistic Neural Networks. Proc. IPMU '90, 6-8.

PARZEN, E., 1962. On estimation of a probability density function and mode. Annals of Mathematical Statistics 33, 1065-1076.

RICHARD, M. D. and LIPPMANN, R. P., 1991. Neural Network Classifiers Estimate a posteriori Probabilities. Neural Computa-tion, Vol. 3(4), 461-483.

ROJAS, R., 1996. Neural Networks - A Systematic Introduction. (Berlin, New York 1996).

RUMELHART, D.E., HINTON, G.E., WILLIAMS, R. J., 1987. Learning internal representations by error propagation, in Rumel- hart, D.E. and McClelland, J.L. (eds.), Parallel Distributed Pro- cessing: Explorations in the Microstructure of Cognition, Vol. 1, 318-362.

WHITE, H., 1989. Learning in Artificial neural networks: A Statisti-cal perspective. Neural Computation, Vol. 1, 425-464.

WILLIAMS, C. K. I. and RASMUSSEN, C. E., 1996. Gaussian Pro-cesses for Regression. In Touretzky, D. S., Mozer, M. C. and Hasselmo, M. E. (eds.), Advances in Neural Information Process-ing Systems 8, MIT Press.

ZEIDLER, J.A., 2001. Dynamic Modeling of Landscape Evolution and Archaeological Site Distributions: A Three Dimensional Approach (Fort Collins 2001).