

A review of model designs

EUROPEAN COMMISSION
RESEARCH DIRECTORATE-GENERAL



PAEQANN

**Predicting Aquatic Ecosystem Quality using Artificial Neural Networks:
Impact of Environmental characteristics on the Structure of Aquatic
Communities (Algae, Benthic and Fish Fauna)**

DELIVERABLE 8
A REVIEW OF MODEL DESIGNS

P.F.M Verdonschot & R.C. Nijboer (eds.)

Contact Number : **EVK1-CT1999-00026**
Project Coordinator : **Professor Sovan LEK, CESAC-UPS-FR**
Project Home Page : **<http://quercus.cemes.fr/~paeqann>**

A review of model designs

P.F.M. Verdonschot & R.C. Nijboer (eds.)

Alterra-rapport 517

Alterra, Green World Research, Wageningen, 2002

ABSTRACT

Verdonschot P.F.M. & Nijboer R.C.(eds.), 2002. *A review of model designs*. Wageningen, Alterra, Green World Research.. Alterra-rapport 517. 92 pp. 13 figs.; 1 tables; 179 refs.

The PAEQANN project aims to review current ecological theories which can help identify suited models that predict community structure in aquatic ecosystems, to select and discuss appropriate models, depending on the type of target community (i.e. empirical vs. simulation models) and to examine how results add to ecological water management objectives. To reach these goals a number of classical statistical models, artificial neural networks and dynamic models are presented. An even higher number of techniques within these groups will be tested later on in the project. This report introduces all of them. The techniques are shortly introduced, their algorithms explained, and the advantages and disadvantages discussed.

Keywords: ecological model, statistical model, artificial neural network, structurally dynamic model, PAEQANN, community structure, aquatic ecosystem

ISSN 1566-7197

This report can be ordered by paying € 18,- into bank account number 36 70 54 612 in the name of Alterra, Wageningen, the Netherlands, with reference to rapport 517. This amount is inclusive of VAT and postage.

© 2002 Alterra, Green World Research,
P.O. Box 47, NL-6700 AA Wageningen (The Netherlands).
Phone: +31 317 474700; fax: +31 317 419000; e-mail: postkamer@alterra.wag-ur.nl

No part of this publication may be reproduced or published in any form or by any means, or stored in a data base or retrieval system, without the written permission of Alterra.

Alterra assumes no liability for any losses resulting from the use of this document.

Contents

Preface	11
Summary	13
1 Introduction	15
1.1 Background	15
1.2 Classical statistical models	16
1.3 Artificial Neural Networks	17
1.4 Dynamic models	18
2 Multinomial Logistic Regression (Paul Goedhart)	19
2.1 Introduction	19
2.2 Multinomial Logistic Regression algorithm	19
2.2.1 Algorithm and application	19
3 Informed Latent Class / Discriminant Model (Herbert Hoijtink, Paul Goedhart, Piet Verdonschot, Rebi Nijboer, Wies Akkermans & Cajo ter Braak)	23
3.1 Introduction	23
3.1.1 Background	23
3.1.2 Application of Bayesian models in aquatic ecology	23
3.1.3 Mixture Models Using A Priori Information and Discriminant Functions	24
3.1.4 Data needed	25
3.1.5 Questions involved	25
3.2 Informed Latent Class / Discriminant Model algorithms	26
3.3 Advantages and disadvantages	27
4 Generalized Linear Models (GLM) (Sovan Lek)	29
4.1 Introduction	29
4.1.1 The normal linear model	29
4.1.2 Why is linearity not enough?	29
4.1.3 Why is normality not enough?	29
4.2 Generalized Linear Models; properties and algorithms	30
4.2.1 Properties	30
4.2.2 GLM algorithms	31
5 Generalized Additive Models (GAM) (Sovan Lek)	33
5.1 Introduction	33
5.2 Smoothing methods and Generalized Additive Model algorithms	33
6 TREE Models (Sovan Lek)	37
6.1 Introduction	37
6.2 CART algorithm	38
6.3 Advantages and disadvantages	40
7 Partial least square regression (PLS) (Sovan Lek)	41
7.1 Introduction	41
7.2 Partial Least Square Regression algorithm	42
7.3 Advantages and disadvantages	44

8	Artificial Neural Networks (ANNs)	45
	8.1 Historic developments	45
9	Self-Organizing Maps (SOMs) (Jean Luc Giraudel)	47
	9.1 Introduction	47
	9.1.1 Development of SOMs	47
	9.1.2 The SOM concept	47
	9.2 Self-Organizing Map algorithm	48
10	Back Propagation Artificial Neural Network (ANNbp) (Muriel Gevrey)	51
	10.1 Introduction	51
	10.2 Back Propagation Artificial Neural Network algorithms	51
	10.2.1 Structure	51
	10.2.2 Algorithms	51
	10.2.3 Forward-propagating step	52
	10.2.4 Backward-propagating step	53
	10.2.5 Training the network	53
	10.2.6 Testing the network	54
	10.3 Advantages and disadvantages	54
11	Kernel-induced nonlinear model (KINM) (Sun Ruixiang)	55
	11.1 Introduction	55
	11.2 Kernel-induced nonlinear models algorithms	55
	11.2.1 SRM versus ERM	55
	11.2.2 KINM concept	56
	11.2.3 KINM algorithm	57
	11.3 Advantages and disadvantages	57
12	Counterpropagation network (Young-Seuk Park)	59
	12.1 Introduction	59
	12.2 Counterpropagation network algorithm	59
	12.3 Advantages and disadvantages	60
13	Competitive neural network in combination with Linear Vector Quantisation (Wies Akkermans)	61
	13.1 Introduction	61
	13.2 Competitive neural network and Linear Vector Quantisation algorithms	61
	13.2.1 Competitive ANNs	61
	13.2.2 Linear Vector Quantisation Network	62
	13.3 Advantages and disadvantages	64
14	Structurally Dynamic Models (Sven Jørgensen)	65
	14.1 Introduction	65
	14.2 Structurally dynamic models	66
15	Use of models in the PAEQANN project	75
	15.1 Classical statistical models in PAEQANN	75
	15.1.1 GLM, GAM and CART	75
	15.1.2 Bayesian Models	75
	15.2 Artificial Neural Networks in PAEQANN	75
	15.2.1 MultiLayer Perceptrons	75
	15.2.2 Sensitivity analysis	76
	15.2.3 Self-Organizing Maps	77
	15.2.4 Artificial Neural Networks Backpropagation	77

15.2.5	Counterpropagation	77
15.2.6	Future developments of ANNs	78
15.3	Structurally dynamic models in PAEQANN	78
16	References	81

Preface

The landmass on which we live is an integral part of our water catchment. Any human activity will inevitably have some consequences on the availability and composition of freshwater. These consequences are becoming increasingly important and detectable. The problem has to be addressed at the EU-level, as frequently, decisions made have inter-regional and international impacts, and must therefore be co-ordinated. In a number of Member States, the availability of water resources depends on the activities of other upstream countries. The demand for freshwater in Europe, as well as in the world, is increasing. There is an upward pressure on European water demand for public supplies (drinking water, recreation, etc.), for industry, and for irrigated agriculture. The ecological impacts of different uses are complex, and currently not always predictable. Our program should help planners in their decisions on different water management options for human use.

Water, of course, is not only relevant as a resource, exploited for human activities, but it is also relevant to aquatic ecosystems and to their quality. Preservation and/or restoration of the ecological quality of these ecosystems, have a major social impact, as it has been stressed in several Community actions.

It is important to stress that the development of models for the prediction of aquatic ecosystem quality in the framework of a European consortium is not merely a scientific exercise, but a social one as well. Water management and environmental policies of different countries in the European Union share a common base (e.g. Urban Waste Water Treatment Directive 91/271/EEC, Integrated Pollution Prevention and Control Directive 96/61/EEC, etc.) thus policies will be become more and more integrated in the near future, the models will aid throughout this process.

The main feature of the PAEQANN project is to provide a unified, common set of tools for i) checking the river ecology status, and ii) predicting environmental impacts of management action on a European scale. This project will not only provide a significant improvement of our knowledge about the ecological applications of classical statistical techniques, dynamic models, and artificial neural networks and other artificial intelligence techniques, but also a set of predictive tools that will be easily applied to real management scenarios. At the end of the project, the results of the research activities will be readily disseminated and the products will be made available to a wide spectrum of potential end-users.

In the first phase of the project, the PAEQANN partners focus their work in two directions: i) to produce the database, and ii) to develop the modelling methodology. This report deals with the second item. In the PAEQANN project, river ecosystem integrity will be assessed using the relationship between environmental impacts and organism groups, i.e. community structure as depending upon the environmental variables. The Classical statistical, Artificial Neural Network and Dynamic models developed can be used as predictive tools, and can also be used to explain and understand the complex relationships between variables. These tools could be applied in other river networks throughout Europe. They will be simple, easy to handle and applicable to stream management and stream policy-making. The

PAEQANN partners focus their work to improve the knowledge on these methods with several algorithms, e.g. backpropagation, self-organising maps, goal function, Bayesian function, etc. All of these techniques are presented in this report.

Summary

The complexity of the ecological systems often results in complex relations between the biological and abiotic variables, justifying the uses of multiple modelling techniques. This use of models is based on different statistical and simulation techniques, designed to predict community structure from environmental variables.

The PAEQANN project tackles this challenge and aims to:

1. to review current ecological theories which can help identify suited models that predict community structure in aquatic ecosystems
2. to select and discuss appropriate models, depending on the type of target community (i.e. empirical vs. simulation models)
3. to examine how results add to ecological water management objectives

PAEQANN uses the following approaches:

Classical statistical models; Multinomial Logistic Regression ,Informed Latent Class / Discriminant Models, Generalised Linear Models (GLM), Generalized additive models (GAM), TREE Model, Partial least square regression (PLS)

Classical statistical approaches mostly follow three steps:

- (1) water samples are clustered into groups on the basis of the biological data and
- (2) the groups are related to the environment data, for example by discriminant analysis.
- (3) the reverse process is used whereby regression techniques use environmental variables to predict the biological communities

Artificial Neural Networks; Self-Organizing Map (SOM), Back Propagation Artificial Neural Network (ANNbp), Kernel-induced nonlinear models (KINM), Counterpropagation network, Competitive neural network in combination with Linear Vector Quantisation

Artificial Neural Networks (ANNs) are powerful computational tools that can be used for classification, pattern recognition, empirical modeling and for many other tasks. Even though most of these tasks can also be performed by conventional statistical or mathematical methods, ANNs often provide a more effective way to deal with problems that are difficult, if not intractable, for traditional computation. Several kinds of ANNs have been developed during the last 10-15 years, but two main categories can be easily recognized, depending on the way ANNs “learn”:

- in *supervised* learning, there is a "teacher" who in the learning phase "tells" the ANN how well it performs or what the correct behaviour would have been;
- in *unsupervised* learning the ANN autonomously analyses the properties of the data set and learns to reflect these properties in its output.

Dynamic models; Structural dynamic models

The present models have rigid structures and a fixed set of parameters, reflecting that no changes or replacements of the components are possible.

The type of models that can account for the change in species composition as

well as for the ability of the species to change their properties, that is to adapt to the prevailing conditions imposed on the species, are called structural dynamic models. The name indicates that the models are able to capture structural changes.

The last chapter summarizes all different techniques to be tested in in the PAEQANN project.

1 Introduction

1.1 Background

Ecological communities are the expression of complex biological processes (reproduction, nutrition, rest, interspecific relationships, et cetera) and abiotic processes (nutrient cycling, discharge regimes, erozion, et cetera), both being expressed on various scales of time and space. To analyze all these processes, i.e. to include and understand the relationship which exist in the community, and to characterize their relationships with environmental parameters, their degree of importance, and their structuring, requires the observation of variables related to the operation of the system.

The complexity of the ecological systems often results in complex relations between the biological and abiotic variables, justifying the uses of multiple modelling techniques. This use of models is based on different statistical and simulation techniques, designed to predict community structure from environmental variables. The PAEQANN project is designed to tackle this challenge. The main objectives of the project are:

1. to review current ecological theories which can help identify suited models that predict community structure in aquatic ecosystems
2. to select and discuss appropriate models, depending on the type of target community (i.e. empirical vs. simulation models)
3. to examine how results add to ecological water management objectives

Ecological water management aims to contribute to the value of aquatic ecosystems. Such management requires the understanding how these ecosystems function, and thus how communities are related to the environment. To learn about the community-environment relationships, data-analytical approaches are explored. Within the PAEQANN project the following approaches are used:

Classical statistical models

- Multinomial Logistic Regression
- Informed Latent Class / Discriminant Models
- Generalised Linear Models (GLM)
- Generalized additive models (GAM)
- TREE Model
- Partial least square regression (PLS)

Artificial Neural Networks

- Self-Organizing Map (SOM)
- Back Propagation Artificial Neural Network (ANNbp)
- Kernel-induced nonlinear models (KINM)
- Counterpropagation network
- Competitive neural network in combination with Linear Vector Quantisation

Dynamic models

- Structural dynamic models

1.2 Classical statistical models

Classical statistical approaches mostly follow three steps:

1. water samples are clustered into groups on the basis of the biological data and
2. the groups are related to the environment data, for example by discriminant analysis.
3. the reverse process is used whereby regression techniques use environmental variables to predict the biological communities

These three steps are part of the PAEQANN project. Multinomial logistic regression (chapter 2) is an improvement over normal discriminant analysis to carry out step 2. In the traditional approach the environmental data play no part in the cluster analysis of step 1. This may be unfortunate. If the environmental data already show distinct groups as a result of water chemistry processes, it is a shame not to use this information. There is also a statistical reason. If for example, discriminant analysis is used in step 2 of the analysis, it can occur that a water sample is misclassified on the basis of the environmental data but on further inspection happens to be a borderline case in the cluster analysis. One would then rather reclassify such a sample and iterate the two steps. A popular rival method for studying community-environment relationships is to use ordination rather than cluster analysis in step 1. By ordination the biological data are reduced to continuous gradients rather than to group. But groups have a simplicity that helps to communicate the results to water managers. Groups, when well described in a typology, can get meaning and become real as if they already existed. The cenotypes of Verdonschot (1990) are just one example.

For the above reasons within the PAEQANN project the potential benefits of other model-based methods, like Bayesian latent class analysis (chapter 3), are investigated. Generalized linear models (GLMs) are used to perform regression modelling for non-normal data with a minimum of extra complication compared with normal linear regression (chapter 4). In the statistical analysis of data and observational studies, the identification and adjustment for prognostic factors is an important component. A valid comparison of different treatments requires the appropriate adjustment for relevant prognostic factors. The failure to consider important prognostic variables, particularly in observational studies, can lead to errors in estimating treatment differences. In addition, incorrect modelling of prognostic factors can result in the failure to identify nonlinear trends or threshold effects.

Chapter 5 describes flexible statistical methods that may be used to identify and characterise the effect of potential prognostic factors on an outcome variable. These methods are called "generalized additive models", and extend the traditional linear statistical model. They can be applied in any setting where a linear or generalized linear model is typically used. These settings include standard continuous response regression, categorical or ordered categorical response data, count data, survival data and time series.

Tree-based methods involve dividing the observations into groups that differ with respect to the variable of interest. A tree-based procedure automatically chooses the grouping that results in homogeneous groups that have the largest difference in proportion of the variable of interest. The tree-based method first divides the observations into two groups. The next step is to subdivide each of the groups based on another characteristic. The process of subdividing is separate for each of the

groups. This is an elegant way of handling interactions that can become complicated in traditional linear models. When the process of subdivision is complete, the result is a classification rule that can be viewed as a tree. For each of the subdivisions, the proportion of the variable of interest can be used to predict the effect of that variable. The structure of the tree gives insight into which characteristics are relevant. There are several tree-based methods that differ with respect to the types of variables allowed, the way groups are chosen, and the way groups are split (chapter 6). The most common methods are Classification and Regression Trees (CART) and Chi-Squared Automated Interaction Detection (CHAID).

Partial least squares (PLS) is a method for constructing predictive models when the factors are many and highly collinear. The general idea of PLS is to try to extract from many factors a few underlying or latent factors, accounting for as much of the manifest factor variation as possible while modelling the responses well. The overall goal is to use the factor to predict the responses in the population (chapter 7).

1.3 Artificial Neural Networks

Artificial Neural Networks (ANNs) are powerful computational tools that can be used for classification, pattern recognition, empirical modeling and for many other tasks. Even though most of these tasks can also be performed by conventional statistical or mathematical methods, ANNs often provide a more effective way to deal with problems that are difficult, if not intractable, for traditional computation.

In fact, while traditional computation is based on the *a priori* selection of suitable functions or algorithms, ANNs are able to adjust their inner structure to provide optimal solutions, given enough data and a proper initialisation. Thus, if appropriate inputs are applied to an ANN, it can acquire knowledge from the environment, mimicking the functioning of a brain, and users can later recall this knowledge.

ANNs lie in a sort of machine learning middle ground, somewhere between engineering and artificial intelligence (Zurada, 1992). They use mathematical techniques, such as mean-square error minimization, but they also rely on heuristic methods, since very often there is no theoretical background to support decisions about ANNs implementation.

Several kinds of ANNs have been developed during the last 10-15 years, but two main categories can be easily recognized, depending on the way ANNs "learn":

- in *supervised* learning, there is a "teacher" who in the learning phase "tells" the ANN how well it performs or what the correct behaviour would have been;
- in *unsupervised* learning the ANN autonomously analyses the properties of the data set and learns to reflect these properties in its output.

In the PAEQANN project both categories of ANNs will be used, with special attention to Self Organizing Maps for unsupervised learning (chapter 9), and Multilayer Perception ANNs for supervised learning (chapter 10).

Kernel-induced nonlinear models (KINMs) are developed in recent years and they can be used to perform nonlinear modelling in complex engineering problems, such as ecological research (chapter 12). They extract the most informative information from the real-world data sets and establish the nonlinear model by utilizing those partial informative data points in the high dimensional kernel-induced space.

In chapter 13 unsupervised Competitive Artificial Neural Networks (CANNs) and the supervised LVQ network are described. Applying these two networks in combination may give a powerful tool for fast classification of future observations.

1.4 Dynamic models

The present models have rigid structures and a fixed set of parameters, reflecting that no changes or replacements of the components are possible. The type of models that can account for the change in species composition as well as for the ability of the species to change their properties, that is to adapt to the prevailing conditions imposed on the species, are called structural dynamic models. The name indicates that the models are able to capture structural changes. Chapter 13 deals with these fifth generation of ecological models. The generation label indicates that these models are radically different from previous modeling approaches and can do more, namely describe changes in species composition.

This type of models was developed by use of biomass as goal function by Pr. Milan Straskraba in the late seventies and later by use of the theoretically more correct, exergy, in the mid-eighties by S.E. Jørgensen (see Jørgensen, 1997). Exergy measures biomass and information, thus more developed organisms, will contribute more to the exergy per weight unit than less developed organisms. The idea behind this type of model is that the goal function describes the development direction of the considered ecosystem. It can therefore be applied to describe how organisms will adapt to the currently changing conditions and how - if the adaptation process is not sufficient - the present organisms will be replaced by other and better fitted organisms with other properties.

This type of model will therefore currently change the parameter values according to how the organisms will change their properties in real ecosystems according to the currently changed conditions. This type of models using exergy as goal function has already been applied successfully in (Jørgensen, 1997). It should also be possible by this model type to give the properties of the species present in a river, lake or coastal area, provided that we know the conditions. From the properties it will probably in most cases be possible to describe the species that have these properties.

CLASSICAL STATISTICAL MODELS

2 Multinomial Logistic Regression (Paul Goedhart)

2.1 Introduction

A two-step approach is often used to build a model for the prediction of macrofaunal community composition from environmental data. In the first step the water samples are clustered into groups on the basis of the macrofauna data alone; these groups are called cenotypes. This is generally a so-called hard clustering in which every water sample is assigned to one cenotype only. In the second step the cenotypes are related to the environmental data. Two important methods to establish this relationship are discriminant analysis and multinomial logistic regression. Discriminant analysis assumes that the environmental data follow a multivariate normal distribution and that is often not the case, for example for dichotomous or nominal environmental variables. Furthermore for the multinomial logistic regression model the whole range of methods and techniques for the generalised linear model is available, such as selection of environmental variables. A third advantage of multinomial logistic regression is that it directly models the probability of occurrence of each cenotype as a function of the environmental data, while in discriminant analysis this is a by-product of the analysis. Multinomial logistic regression is therefore the method of choice to relate cenotypes to environmental data. The results can be used to predict the occurrence of species from environmental variables.

2.2 Multinomial Logistic Regression algorithm

2.2.1 Algorithm and application

Multinomial logistic regression is a well-established technique and is a direct extension of ordinary logistic regression which itself is a special case of a generalised linear model. Suppose for simplicity that there are only two cenotypes 1 and 2. Suppose further that y_1 and y_2 are indicator variables for cenotypes 1 and 2 respectively; i.e. y_1 is 1 when a water sample is classified as cenotype 1 and 0 otherwise, and similarly for y_2 . And finally, π_1 and π_2 are the probabilities of occurrence of cenotypes 1 and 2. It is then natural to assume that y_1 follows a binomial distribution with probability π_1 of "success", i.e. $y_1 \sim \text{Binomial}(1; \pi_1)$. It follows directly from $y_1 + y_2 = 1$ and $\pi_1 + \pi_2 = 1$ that $y_2 \sim \text{Binomial}(1; \pi_2)$. Ordinary logistic regression can then be used to relate the probabilities of occurrence π_1 to the environmental variables x_1, \dots, x_p :

$$\log(\pi_1/(1-\pi_1)) = \log(\pi_1/\pi_2) = \beta_0 + \beta_{1x_1} + \dots + \beta_{px_p}$$

With k genotypes there are k indicator variables y_1, \dots, y_k and k probabilities of occurrence π_1, \dots, π_k . The indicator variables then follow a multinomial distribution which is a direct generalisation of the binomial distribution, i.e. $(y_1, \dots, y_k) \sim \text{Multinomial}(1; (\pi_1, \dots, \pi_k))$. The relationship between the k probabilities and the p environmental variables is now modeled by means of $(k-1)$ linear predictors:

$$\begin{aligned} \log(\pi_1/\pi_k) &= \beta_{10} + \beta_{11x_1} + \dots + \beta_{1px_p} \\ \log(\pi_2/\pi_k) &= \beta_{20} + \beta_{21x_1} + \dots + \beta_{2px_p} \\ &\vdots \quad \quad \quad \vdots \quad \quad \quad \vdots \quad \quad \quad \vdots \\ \log(\pi_{k-1}/\pi_k) &= \beta_{k-1,0} + \beta_{k-1,1x_1} + \dots + \beta_{k-1,px_p} \end{aligned}$$

This completes the specification of the multinomial logistic regression model. There are $(k-1)(p+1)$ parameters in total, which must be estimated from the data. The model can be fitted by means of software for generalised linear models by employing the equivalence between the multinomial likelihood and the Poisson likelihood. In figure 2.1 an example is given of a multinomial logistic model with 4 genotypes and only 1 environmental variable. Note that the sum of the 4 probabilities is 1 for every value of the environmental variable. Further information about multinomial logistic regression can be found in McCullagh & Nelder (1989) and in Hosmer & Lemeshow (1989).

There are in general a lot of environmental variables, 20 or more is not uncommon, and they are frequently correlated. Using all variables in a model then yields unstable estimators of the regression coefficients and thus poor predictions. Some form of selection of environmental variables is therefore necessary. Forward selection, backward elimination or stepwise regression result in only one model and alternative models, with an equivalent or even better fit, are easily overlooked. A preferable method is to fit all possible regression models and to evaluate these according to some criterion such as the deviance. However the fitting of all possible regression models is very computer intensive, especially for multinomial logistic models. A practical approach is to perform iterative model selection. Firstly, the variables are subdivided in a few groups, and model selection is performed within each group. The best predictors from each group are then combined in a new model selection step, which yields a few best variables. With these few variables fixed in the model, the remaining variables are again subdivided in a few groups and the next iteration starts. This eventually results in a number of best candidate models. The predictive power of these candidate models can be assessed by means of leave-one-out, also called cross validation. In the first leave-one-out step, the first observation is temporarily deleted from the data, and the model is fitted to the remaining observations. This model is then used to calculate a so-called leave-one-out prediction for the first observation. In the same way leave-one-out predictions are obtained for all observations, by subsequently removing them from the data. The mean leave-one-out probability of predicting the correct genotype can then be used as a criterion for choosing among the candidate models.

Another approach, which has been proven beneficial for the prediction of genotypes, is what is called hierarchical modelling. This approach assumes that the effect of certain environmental variables is more or less the same for groups of genotypes.

Instead of estimating this effect for every individual cenotype, it is now estimated for groups of cenotypes. In this way a reduction of the number of estimated parameters can be accomplished. In this approach the cenotypes are first classified into a small number of groups, in such a way that cenotypes within a group are comparable. Then separate multinomial logistic models are fitted to the grouping of the cenotypes, and to the cenotypes within each group. These models can then be used to predict the probability of occurrence of groups, and the probability of occurrence of cenotypes within groups. These probabilities need then be multiplied to obtain the probability of occurrence of cenotypes. Suppose for example that there are two groups (1 and 2), group 1 consists of three cenotypes (A, B and C), and group 2 consists of two cenotypes (D and E). Suppose further that for a set of environmental variables the predicted probability of occurrence of group 1 and 2 is 0.8 and 0.2 respectively. Suppose furthermore that the predicted probability for A, B and C within group 1 equals 0.9, 0.1 and 0.0, and that the predicted probability for D and E within group 2 equals 0.4 and 0.6. The probability of occurrence of cenotypes A - E is then as follows

$$\begin{aligned} \text{A: } & 0.8 \times 0.9 = 0.72 & \text{D: } & 0.2 \times 0.4 = 0.08 \\ \text{B: } & 0.8 \times 0.1 = 0.08 & \text{E: } & 0.2 \times 0.6 = 0.12 \\ \text{C: } & 0.8 \times 0.0 = 0.00 & & \end{aligned}$$

The sum of the probabilities over cenotypes A - E is then again 1.0.

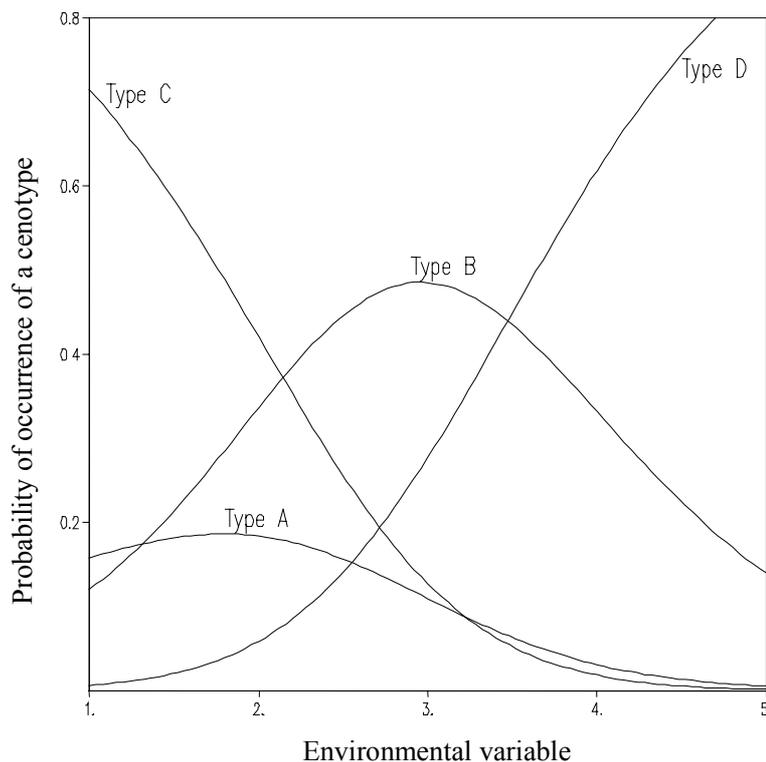


Figure 2.1 The probability of occurrence of 4 cenotype A, B, C and D as a function of an environmental variable. The sum of the 4 probabilities is 1 for any value of the environmental variable.

3 Informed Latent Class / Discriminant Model (Herbert Hoijtink, Paul Goedhart, Piet Verdonschot, Rebi Nijboer, Wies Akkermans & Cajo ter Braak)

3.1 Introduction

3.1.1 Background

Bayesian models are based on the principles of Bayes Rule or Bayes Theorem. It defines the formalism of updating a belief about a hypothesis (or a-priori probability) in the light of a new evidence (e.g. new data). The updated probability is called the posterior probability. The distinctive feature of Bayesian models is the explicit consideration of probability. So it is a strong method to increase the knowledge about a certain system of the real world by the integrative analysis of probabilities of models and observation data.

The basic applications in Bayesian reasoning were extended for application in time series as well as for cases of interdependent probabilities. Extension of the basic method is the integration of Markov chain theory, Metropolis Hastings algorithm, Monte Carlo methods, information theory, and spatial analysis. Integrated methods are for example the Gibbs sampler, Markov chain Monte Carlo techniques (MCMC), Bayesian maximum entropy, and Bayesian kriging. This family of methods is used in a wide field of disciplines, e.g. medicine, astrophysics, economy as well as in ecology. The integration of Bayesian principles with other methods supported the development of complex Bayesian models in Bayesian networks (BN). In hierarchical Bayesian networks the hierarchical influences of parameters with different probability functions can be modelled. Bayesian Belief Networks (BBN, also known as Belief Networks, Causal Probabilistic Networks, Causal Nets, Graphical Probability Networks, Probabilistic Cause-Effect Models, and Probabilistic Networks) are building the bridge to Artificial Intelligence (AI) by the possibility to integrate expert knowledge into the model. Advantages of BBNs are the ability to represent and manipulate complex models, and the possibility for event prediction based on partial or uncertain data.

3.1.2 Application of Bayesian models in aquatic ecology

A common application of Bayesian models is stock assessment especially in fish ecology and fish management. An application of special practical interest is the determination of stock assessment for regulation of fish catches and related topics, with a predominance of basic Bayesian models (Adkison & Peterman, 1996; Cow-Rogers, 1997; Hammond, 1997; Hilborn & Liermann, 1998; Jon *et al.*, 2000; Kinas, 1996; McAllister & Ianelli, 1997; McAllister & Kirkwood, 1998; McAllister & Kirkwood, 1998; McAllister & Pikitch, 1997; Myers *et al.*, 1999; Newman, 1997; Ogle *et al.*, 1996; Pella *et al.*, 1998; Peterman *et al.*, 1999; Punt & Hilborn, 1997; Punt

& Walker, 1998; Robb & Peterman, 1998; Smith & Punt, 1998). A lower number of publications in fish management is related to advanced Bayesian models (Chen & Fournier, 1999; Chen *et al.*, 2000; Helu *et al.*, 2000; Kuikka *et al.*, 1999; Lee *et al.*, 1996; Lee & Rieman, 1997; Liermann & Hilborn, 1997; Meyer & Millar, 1999; Patterson, 1999; Vignaux *et al.*, 1998).

The improvement of Bayesian models by addition of other methods effected an increase in applications outside of fish ecology. Basic Bayesian models are applied for risk and decision analysis (Qian *et al.*, 2000; Steinberg *et al.*, 1996; Varis, 1997). A focus of complex Bayesian models is classification and diagnosis of water quality (Trigg *et al.*, 2000; Varis & Kuikka, 1997; Walley & Dzeroski, 1996; Walley & Fontama, 1997, 2000). Other applications are related to environmental reconstruction (Vasko *et al.*, 2000) and models about heterogeneous populations (Billheimer *et al.*, 1997; Carpenter *et al.*, 1996; Cottingham & Schindler, 2000; Lamon & Clyde, 2000; Mau *et al.*, 1999; Meyer & Millar, 1999; Pinelalloul *et al.*, 1995).

3.1.3 Mixture Models Using A Priori Information and Discriminant Functions

Mixture models (Titterton *et al.*, 1985) assume that each sample is a member of one of a finite number of classes, and use the data to estimate the parameters of the model. Within each class a specific group of taxa is linked to a specific environment. The latter implies that knowledge of the environment enables a researcher to predict the group of taxa. Previous research (amongst others Verdonshot & Goedhart, 2000) indicates that the mixture assumption is reasonable and worthwhile pursuing. This approach can be seen as the counterpart of ANNs. ANNs use the data to construct a model with which the number of each taxon can be predicted using environmental variables.

Traditionally mixture models are analysed using maximum likelihood. The latter increases in difficulty with the complexity of the mixture model. It may be impossible to determine if the likelihood has actually been maximised; confidence intervals and standard errors for the model parameters are unobtainable, it is hard to determine the number classes that have to be used in the mixture, and, the distribution of goodness of fit tests is unknown. Using Bayesian computational statistics these problems can almost completely be avoided. See Hoijtink & Molenaar (1997), Hoijtink (1998, 2001) for an application of this approach to mixture models where the observed variables are dichotomous.

The use of Bayesian computational procedures for the analysis of the data with mixture models does not solve the problem that the number of parameters in the model is rather big compared to the amount of data (the sample size). The consequence is that the predictive validity of the model will probably be rather small if this problem is ignored. Two measures will be taken to control this problem:

Some of the taxa are more similar (i.e. have a higher probability of living in the same environment) than others. The similarity among the taxa will be quantified into a number of variables that will be used to determine the parameters of a hyperprior for each of the taxa. The result is that similar taxa will receive similar parameter-values. The latter will increase the predictive validity of the model.

The environmental variables are correlated. The latter implies that not all environmental variables are needed to make a distinction among the mixture components. Instead of estimating parameters for each environmental variable, the parameters of a (much smaller) number of discriminant-functions will be estimated. The latter will also increase the predictive validity of the model.

A pilot project indicated that the approach described above is rather promising. In the sequel of the project software for mixture models using a priori information and discriminant functions will be developed, tested and implemented.

3.1.4 Data needed

Informed Latent Class / Discriminant Models can use different type of data to answer ecologically relevant questions. A potential use is described below, first introducing the data features as well as the research questions involved.

The data consist of $N = 664$ samples obtained from, for example, streams and ponds. For each sample two sets of variables are recorded: the truncated logarithm of the number (subsequently called the prevalence) of each of $S=854$ taxa present in the sample; and, the value of $M=53$ environmental variables like temperature, oxygen content and current velocity. The measurement of the environmental variables is relatively straightforward. The prevalence of the taxa is obtained by placing a net in the water, retraction of the net and subsequent manual identification and count of the number of each taxon present in the sample.

Each of the 854 taxa is characterised using $K=6$ variables representing the existing knowledge with respect to each of the taxa. Among these variables are the watertype that a taxon usually lives in and the usual habitat of a taxon. These variables constitute prior information with respect to the taxa. The statistical model presented in paragraph 3.3 will use this information in order to obtain more stable estimates of the parameters of each taxon.

3.1.5 Questions involved

The research question consists of two parts:

- How many water-classes Q can be distinguished using the taxa and the environmental variables, and, what is the label/characterisation that should be attached to each of these classes.
- How well can class-membership be predicted using only the environmental variables i.e. can the population of taxa be predicted from the environment.

A cenotype is a group of individual waterbodies that are relatively homogeneous with respect to taxon composition and environmental characteristics. The probability that the taxon composition can be predicted from the environment increases if the cenotypes are rather distinct. Since the number of variables in the data ($854+53$) is rather large in comparison to the sample size (664), there is room for errors. It may be difficult to correctly estimate the number of classes. The labelling / characterisation of the cenotypes may be wrong (local maxima of the likelihood of the model that is used); and, there is room for capitalization on chance i.e. the

“correlation” between taxon composition and environment may overestimated leading to an overestimation of the predictive validity of environment. Two measures will be taken to reduce the room for errors; the taxon parameters will be estimated using the prior knowledge with respect to the taxa; and, using discriminant functions the parameter space of the environmental variables will be reduced. Both measures will be described in the next paragraph.

3.2 Informed Latent Class / Discriminant Model algorithms

To save space, the presentation of the model will be rather informal. The interested reader is referred to the full technical report that will be finished in the spring of 2002. The likelihood of the model is:

$$L = \prod_{i=1}^N \left[\sum_{q=1}^Q \pi_q \prod_{s=1}^S g(Y_{is} | \theta_i = q) \prod_{m=1}^M b(x_{im} | \theta_i = q) \right], \quad (1)$$

in which y_{is} contains the prevalence of taxa s in water sample i , x_{im} contains the scores on environmental variable m , θ_i denotes the class membership of water sample i , and π_q the proportion of water samples belonging to class q . The density $g(\cdot)$ of the environmental variables is Poisson with parameter λ_{qs} denoting the prevalence of taxa s in class q . The density $b(\cdot)$ of the environmental variables can be normal, Bernouille, binomial or multinomial depending on the scale on which the variable is measured. Each density is (at least) characterised by a parameter μ_{qm} representing the centre of environmental variable m in class q .

Prior information will be specified for each parameter in the model. Usually relative vague priors will be used. An exception are the parameters on the taxa. A normal hyperprior will be assigned to the logarithm of each Poisson parameter, each with expectation

$$\alpha_{q0} + \sum_k \alpha_{qk} \tilde{z}_{ks}, \quad (2)$$

in which \tilde{z}_{ks} denotes the score of taxa s on the k -th variable containing prior knowledge with respect to the taxa and α_{qk} the hyperparameter associated with the k -th variable in class q . The variance of the hyperprior is σ_{qs}^2 . The use of these hyperpriors has two consequences:

1. Taxa with similar scores on the K prior information variables will also have similar preferences of the parameter λ_{qs} in each of the latent classes.
2. The number of parameters to be estimated for the taxa is strongly reduced. Instead of a preference for each taxon, only the hyperparameters have to be estimated. This should have a favourable effect on the problems described at the end of the previous paragraph.

Within each of Q latent classes, each of the M environmental variables has a distribution containing a location parameter μ_{qm} . This implies that $Q \times M$ location parameters have to be estimated. It is very likely that not all M environmental variables are needed to make a distinction between the q latent classes. Stated otherwise, it is very likely that a limited number of discriminant functions is sufficient. Let v_q denote the set of location parameters for the q -th class. Then the

implication of using, for example, three discriminant functions to distinguish the Q latent classes is that

$$v_q = v_1 + \beta_{q2}(v_1 - v_2) + \beta_{q3}(v_1 - v_3) + \beta_{q4}(v_1 - v_4) \text{ for } q = 5, \dots, Q. \quad (3)$$

This reduces the number of parameters to be estimated from $Q \times M$ to $4 \times M + (Q - (D+1)) \times 3$, which should have a favourable effect on the problems described at the end of the previous paragraph.

The parameters of the informed latent class/discriminant model can be estimated using a procedure based on the Gibbs sampler (Gelman, Carlin, Stern & Rubin, 1995; Zeger & Karim, 1991). The selection of the number of classes Q and the number of discriminant functions will be based on Bayes factors (Kass & Raftery, 1995).

Once the model described in this paragraph is calibrated i.e. the number of classes Q is determined and each of the classes is labelled/characterised using the λ_{qs} 's and the μ_{qm} 's an answer to the first research question is obtained. To obtain an answer to the second research question, two classifications have to be compared: an assignment of the samples to latent classes using both the taxon preferences and the environmental variables; and, an assignment using only the environmental variables. The higher the agreement between both classifications, the higher the predictive validity of the model as defined in the second research question.

3.3 Advantages and disadvantages

The model described in the previous paragraph has two features, which distinguishes it from other approaches (e.g. cluster analyses or artificial neural networks). Both could be used in this context: it deals explicitly with the fact that the number of variables exceeds the sample size; and, it is a *statistical* model which implies that it can be used to make inferences with respect to the unknown population from which the sample is obtained. To clarify the latter, both cluster analysis and neural nets are models that describe the structure in the data without reference to a population from which the data are obtained.

4 Generalized Linear Models (GLM) (Sovan Lek)

4.1 Introduction

Generalised linear models (GLMs) are used to do regression modelling for non-normal data with a minimum of extra complication compared with normal linear regression. GLMs are flexible enough to include a wide range of common situations, but at the same time allow most of the familiar ideas of normal linear regression to carry over. The essay by Firth (1991) gives a good introduction to GLMs; the comprehensive reference is McCullagh & Nelder (1989).

4.1.1 The normal linear model

Let y be a vector of observations, and let X be a matrix of covariates. The usual multiple regression model takes the form:

$$\mu = X\beta \quad (1)$$

where $\mu = E(y)$ and β is a vector of regression coefficients. Typically we assume that the y_i are normal and independent with standard deviation σ , so that we estimate β by minimizing the sum of squares:

$$(y - \mu)^T(y - \mu)$$

4.1.2 Why is linearity not enough?

The most important and common case is that in which the y_i and μ_i are bounded. For example, if y represents the amount of some physical substance then we may have $y > 0$ and $\mu > 0$. On the other hand if y is binary, $y = 1$ if an animal survives and $y = 0$ if it does not, then $0 < \mu < 1$. The linear model (1) is inadequate in these cases because complicated and unnatural constraints on β would be required to make sure that μ stays in its range. Generalized linear models instead assume a *link linear* relationship:

$$g(\mu) = X\beta \quad (2)$$

where $g()$ is some known monotonic function which acts pointwise on μ . Typically $g()$ is used to transform μ to a scale which is unconstrained. For example we might use $g(\mu) = \log(\mu)$ if $\mu_i > 0$ or $g(\mu) = \log[\mu / (1-\mu)]$ if $0 < \mu_i < 1$.

4.1.3 Why is normality not enough?

In some situations, typically cases when σ is small, the normal approximation to the distribution of y is accurate. More typically, responses are not normal.

If y is bounded then the variance of y must depend on its mean. Specifically if μ is close to a boundary for y then the $\text{var}(y)$ must also be small. For example, if $y > 0$, then we must have $\text{var}(y) \rightarrow 0$ as $\mu \rightarrow 0$. For this reason strictly positive data almost always shows increasing variability with increased size. If $0 < y < 1$, then $\text{var}(y) \rightarrow 0$ as $\mu \rightarrow 0$ or $\mu \rightarrow 1$. For this reason, generalized linear models assume that:

$$\text{var}(y) = \phi V(\mu) \quad (3)$$

where $V()$ is some known variance function appropriate for the data at hand.

We therefore estimate the nonlinear regression equation (2) weighting the observations inversely according to the variance functions $V(\mu)$. This weighting procedure turns out to be exactly equivalent to maximum likelihood estimation when the observations actually come from an exponential family distribution.

4.2 Generalized Linear Models; properties and algorithms

4.2.1 Properties

Generalised linear models, or GLMs, (see Dobson (1983) or McCullagh & Nelder (1994)) refer to a wide class of statistical models including log-linear models, analysis of variance, probit analysis, logistic regression and standard multiple regression. As a result, GLMs have many applications. The GLMs were first given a firm theoretical and computational framework by Nelder & Wedderburn (1972), who assumed distributions in the exponential family. Such distributions have the form:

$$f_Y(y, \theta, \phi) = \exp\left\{\frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi)\right\} \quad (4)$$

for known functions $a(\cdot)$, $b(\cdot)$ and $c(\cdot)$. The function $a(\phi)$ is often of the form $a(\phi) = \phi/w_i$, where w_i are known prior weights θ is referred to as the *canonical parameter*, and ϕ as the *dispersion parameter*. The covariates, \mathbf{X} , are linked to the expected values through a monotonic differentiable *link function*, $g(\cdot)$, such that:

$$\eta = g(\mu) = X\beta \quad (5)$$

in which β is to be estimated. The link function can be chosen independently of the error distribution. A commonly used link function is the *canonical link function*, $\eta = b(\theta)$. Distributions such as the normal (Gaussian), gamma, inverse Gaussian, Poisson and binomial distributions are members of the exponential family.

It can be shown that $E(Y) = \mu = b'(\theta)$ and $\text{var}(Y) = b''(\theta)a(\phi)$. The first component of the variance, $b''(\theta)$, depends on the canonical parameter and therefore the mean. It is usually referred to as the *variance function* and written as $V(\mu)$. The second component, $a(\phi)$, depends only on θ and is independent of the canonical parameter. For example, the normal distribution has $b(\theta) = \theta^2/2$ and $\phi = \sigma^2$, so $E(Y) = \mu = \theta$ and $\text{var}(Y) = \sigma^2$ with a variance function of $V(\mu) = 1$.

The *unit deviance* is defined as:

$$d_i(y_i, \mu_i) = 2 \int_{\mu_i}^{y_i} \frac{y_i - u}{V(u)} du \quad (6)$$

and the total deviance, a measure of the distance between y and μ , is defined as:

$$D(y, \mu) = \sum_{i=1}^n w_i d(y_i, \mu_i) \quad (7)$$

where the summation is over the data and w_i are the prior weights. The quantity $D(y, \mu)/\phi$ is called the scaled deviance. For the normal distribution, the deviance is equivalent to the residual sum-of-squares:

$$\sum_{i=1}^n (y_i - \mu_i)^2.$$

4.2.2 GLM algorithms

The algorithms for fitting generalised linear models are robust and well established (see Nelder & Wedderburn (1972) and McCullagh & Nelder (1994)). The maximum likelihood estimates of β can be obtained using iterative least-squares. Given $\hat{\mu}_0$, initial values of the linear predictor $\hat{\eta}_0$ are found from the link function, and the adjusted dependent variable, z_0 , can be formed as:

$$z_0 = \hat{\eta}_0 + (y - \hat{\mu}_0) \left(\frac{d\hat{\eta}}{d\hat{\mu}} \right)_0 \quad (8)$$

where the derivative is evaluated at the initial values. The quadratic weights, W_0 , are defined so that:

$$\frac{1}{W_0} = \left(\frac{d\hat{\eta}}{d\hat{\mu}} \right)_0^2 V_0 \quad (9)$$

where V_0 is the variance function evaluated at the initial values. z_0 is then regressed onto the covariates, \mathbf{X} , with weights W_0 to produce new parameter estimates, $\hat{\beta}_1$. These are then used to form new estimates $\hat{\mu}_1$ and $\hat{\eta}_1$, and iterations are repeated until changes are sufficiently small. Starting values are obtained directly from the data, using $\mu_0 = y$ with occasional refinements in some cases (for example, to avoid evaluating $\log(0)$ when fitting a log-linear model with zero counts).

By default, the scale parameter is estimated by the mean deviance, $\frac{1}{n} \sum_i D(y, \mu)$, in the case of the normal, gamma and inverse Gaussian distributions. The default value of the scale parameter for binomial and Poisson distributions is one. In the case of over- or under-dispersion, it can be estimated by the mean deviance.

5 Generalized Additive Models (GAM) (Sovan Lek)

5.1 Introduction

One of the most commonly used statistical models in ecological research is multiple linear regression for the quantitative dependant data and the logistic regression model for binary data. The logistic model is used as a specific illustration of a generalized additive model (GAM). Logistic regression (and many other techniques) model the effects of prognostic factors x_j in terms of a linear predictor of the form $\sum x_j \beta_j$, where the β_j are parameters. The generalized additive model replaces $\sum x_j \beta_j$ with $\sum f_j(x_j)$ where f_j is a unspecified ("non-parametric") function. This function is estimated in a flexible manner using a scatterplot smoother (e.g. LOESS). The estimated function $\hat{f}_j(x_j)$ can reveal possible nonlinearities in the effect of the x_j .

Hasti & Tibshirani (1990) is an excellent reference of GAM, and Chambers & Hastie (1992) cover GAM and LOESS.

5.2 Smoothing methods and Generalized Additive Model algorithms

First some background is given on the methodology, and then the details of the logistic regression model and its generalization are discussed. Some related developments are discussed in the last paragraph.

The building block of the generalized additive model algorithm is the scatterplot smoother. First, scatterplot smoothing is described in a simple setting, and then it is indicated how it is used in generalized additive modelling.

Suppose that we have a scatterplot of points (x_i, y_i) like that shown in figure 5.1.

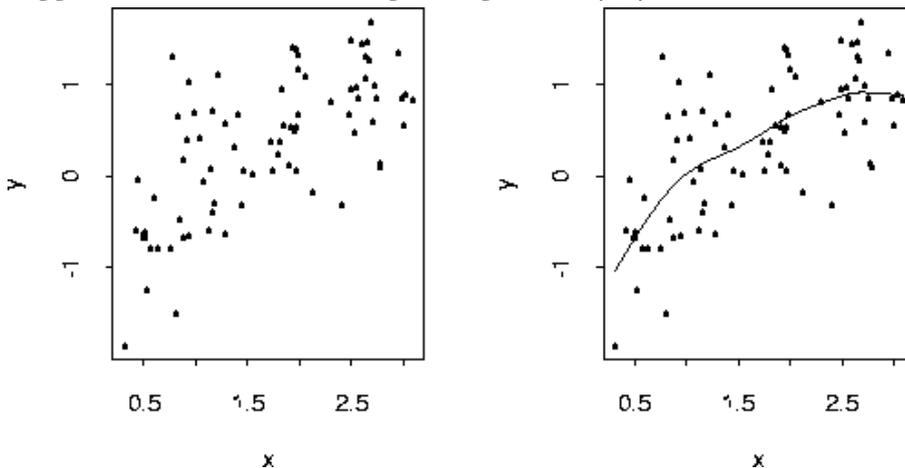


Figure 5.1 Left panel shows a fictitious scatterplot of an outcome measure y plotted against a prognostic factor x . In the right panel, a scatterplot smooth has been added to describe the trend of y on x .

Here y is a response or outcome variable, and x is a prognostic factor. We wish to fit a smooth curve $f(x)$ that summarizes the dependence of y on x . If we were to find the curve that simply minimizes $\sum (y_i - f(x_i))^2$, the result would be an interpolating curve that would not be smooth at all.

The cubic spline smoother imposes smoothness on $f(x)$. We seek the function $f(x)$ that minimizes:

$$\sum (y_i - f(x_i))^2 + \lambda \int f''(x)^2 dx \quad (1)$$

Notice that $\int f''(x)^2$ measures the “wiggleness” of the function f : linear f s have $\int f''(x)^2 = 0$, while non-linear f s produce values bigger than zero. λ is a non-negative smoothing parameter that must be chosen by the data analyst. It governs the trade-off between the goodness of fit to the data and (as measured by $\sum (y_i - f(x_i))^2$) and wiggleness of the function. Larger values of λ force f to be smoother.

For any value of λ , the solution to (10) is a cubic spline, i.e., a piecewise cubic polynomial with pieces joined at the unique observed values of x in the dataset. Fast and stable numerical procedures are available for computation of the fitted curve. The right part of figure 5.1 shows a cubic spline fit to the data.

What value of λ was used in figure 5.1? In fact, to express the desired smoothness of f in terms of λ is inconvenient. As the meaning of λ depends on the units of the prognostic factor x . Instead, it is possible to define an "effective number of parameters" or "degrees of freedom" of a cubic spline smoother, and then use a numerical search to determine the value of λ to yield this number. In figure 5.1 we chose the effective number of parameters to be 5. Roughly speaking, this means that the complexity of the curve is about the same as a polynomial regression of degrees 5. However, the cubic spline smoother "spreads out" in a more even manner its parameters, and hence is much more flexible than polynomial regression. Note that the degrees of freedom of a smoother need not be an integer.

The above discussion tells how to fit a curve to a single prognostic factor. With multiple prognostic factors, if x_{ij} denotes the value of the j^{th} prognostic factor for the i^{th} observation, we fit the additive model:

$$\hat{y}_i \approx \sum_j f_j(x_{ij}) \quad (2)$$

A criterion like (10) can be specified for this problem, and a simple iterative procedure exists for estimating the EMBEDDED Equation.3 s. A cubic spline smoother to the outcome EMBEDDED Equation.3 as a function of x_{ik} is applied, for each prognostic factor in turn. The process is continues until the estimates \hat{f}_j stabilize. This procedure is known as "backfitting" and the resulting fit is analogous to multiple regression for linear models.

When generalized additive models are fit to binary response data (and in many other settings), the appropriate error criterion is a penalized log likelihood or a penalized log partial-likelihood. To maximize it, the backfitting procedure is used in conjunction with a maximum likelihood or maximum partial likelihood algorithm. The usual Newton-Raphson routine for maximizing log-likelihoods in these models can be cast in an IRLS (iteratively reweighted least squares) form. This involves a

repeated weighted linear regression of a constructed response variable on the covariates: each regression yields a new value of the parameter estimate, which gives a new constructed variable, and the process is iterated. In the generalized additive model, the weighted linear regression is simply replaced by a weighted backfitting algorithm. The detail of algorithm can be found in Hastie & Tibshirani 1990.

6 TREE Models (Sovan Lek)

6.1 Introduction

Tree-based methods involve dividing the observations into groups that differ with respect to the variable of interest. For example, suppose one wants to know which passengers on the Titanic were most likely to survive the ship's sinking, and what characteristics were associated with survival. In this case, the variable of interest is survival. The passengers could be divided into groups based on age, sex, and class, and look at the proportion surviving in each group. A tree-based procedure automatically chooses the grouping that results in homogeneous groups that have the largest difference in proportion surviving.

In the Titanic example, the tree-based method first divided the observations into men and women. The next step is to subdivide each of the groups based on another characteristic. Men were divided into adults and children, while women were divided into groups based on class. Notice that the process of subdividing is separate for each of the groups. This is an elegant way of handling interactions that can become complicated in traditional linear models.

When the process of subdivision is complete, the result is a classification rule that can be viewed as a tree. For each of the subdivisions, the proportion surviving can be used to predict survival for members of that group. The structure of the tree gives insight into which characteristics of the passengers are related to survival. A tree for Titanic survival, with the proportion surviving in each subgroup, is given in figure 6.1.

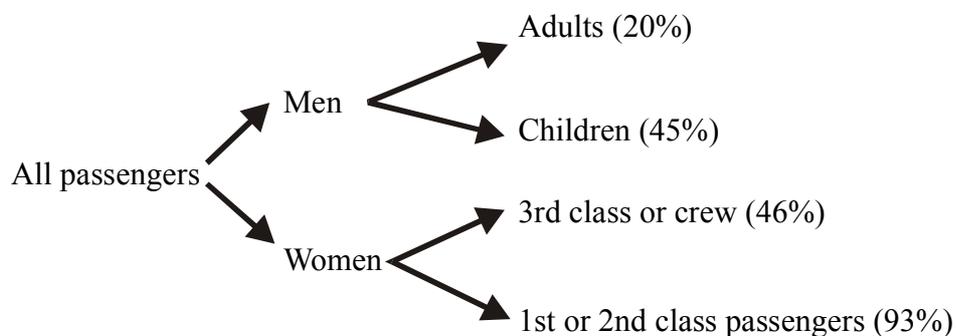


Figure 6.1 A tree for Titanic survival, with the proportion surviving in each subgroup.

There are several tree-based methods that differ with respect to the types of variables allowed, the way groups are chosen, and the way groups are split. The most common methods are Classification and Regression Trees (CART) and Chi-Squared Automated Interaction Detection (CHAID). CART and similar methods allow the response and grouping variables to be either categorical or continuous. CART methods are implemented in SYSTAT version 7, in S-Plus, and in SAS. CHAID and similar methods require the response variable to be categorical. CHAID methods are available in an SPSS add-on module and in the SAS macro %TREEDISC.

The best description of regression trees theory can be found in Breiman *et al.* (1984).

6.2 CART algorithm

The main steps of the CART method extensively developed by Breiman *et al.* (1984) are used and used to grow a regression tree model.

Let a learning sample L be a set of n couples of observations $(y_1, \mathbf{x}_1), \dots, (y_n, \mathbf{x}_n)$ where $\mathbf{x}_i \in R^p$ and $y_i \in R$, $i = 1, \dots, n$. The objective of a regression tree method is to predict values of the response variable $y = (y_1, \dots, y_p)^t$ explained by the set of the

other p predictor variables $\mathbf{X}_{n \times p} = (\mathbf{x}_1, \dots, \mathbf{x}_n)^t$. The sample $L = \{(y_i, \mathbf{x}_i), i = 1, \dots, n\}$ is partitioned by a sequence of binary splits, into a set of terminal nodes. Splits are formed by means of successive inequality conditions on the predictor variables. At each split and starting to the root node, these conditions make the couples (y, \mathbf{x}) falling in successive intermediate nodes to finally move into a terminal node t . At each terminal node t , a predicted value of y is estimated by the average of the $n(t)$ values of y associated to that terminal node,

$$\mu(\mathbf{x}_i) = \frac{1}{n(t)} \sum_{(y_i, \mathbf{x}_i) \in t} y_i = \bar{y}(t). \text{ The final structure is displayed as a binary tree } T.$$

Thus, the tree building procedure with the CART method revolves around three steps:

- selection of the best binary split at each node,
- decision either to declare a node as terminal or to continue the splitting procedure,
- assignment of the estimator $\mu(\mathbf{x}) = \bar{y}(t)$ to the predicted variable.

The first step in tree construction is to determine best binary splits of L into smaller and smaller exclusive subsets. At any node t , the split selection is carried out so that the data are more and more homogeneous in every of the descendant subsets, this according to the squared deviations of the response variable from the estimated average $\bar{y}(t)$, called the deviance $D(t)$ and given by:

$$D(t) = \sum_{(y_i, \mathbf{x}_i) \in t} (y_i - \bar{y}(t))^2$$

When a split is performed, two subsets of observations t_L and t_R are obtained. The deviance of the response variable is respectively given on both sides of the split by :

$$D(t_L) = \sum_{(y_i, \mathbf{x}_i) \in t_L} (y_i - \bar{y}(t_L))^2 \text{ and } D(t_R) = \sum_{(y_i, \mathbf{x}_i) \in t_R} (y_i - \bar{y}(t_R))^2$$

in which: $\bar{y}(t_L), \bar{y}(t_R)$ are the means estimated at the nodes t_L and t_R .

Among the set S of possible binary partitions, the best split s^* of a node t is the split which most decreases the deviance value $D(t)$ *i.e.* the one that maximizes the difference between deviance before and after that split. For any split s of t in t_L and t_R , the best split s^* is selected, so that it verifies:

$$\Delta D(s, t) = D(t) - [D(t_L) + D(t_R)]$$

$$\Delta D(s^*, t) = \max_{s \in S} \Delta D(s, t)$$

The two subsets satisfying this rule will generate two new nodes according to a given condition on a predictor. Then, they will be divided one after the other to define a second level of partition. The splitting procedure is reiterated until the decision to stop splitting. A binary structured tree is then obtained with several levels of splits (figure 6.1).

Theoretically, a tree could have as many terminal nodes as there are observations. Consequently, a node is declared to be suitable for splitting if :

- the deviance at this node is greater than a given fraction of the root node deviance,
- the number of observations at this node is greater than a given percentage of the total sample size.

This process results in a large tree with a great number of terminal nodes. As the number of terminal nodes increases, the tree structure tends to be determined from peculiarities in the learning sample: the model used for prediction is not easy to interpret and it must be simplified. However, the more splitting steps are performed, the more precise are the estimations of y . A compromise between the tree size and the error given by the model used for prediction must be found.

Pruning a tree allows a decrease of the size of a tree T_{\max} build on the sample L by successively removing nodes. A pruned subtree T of T_{\max} is a simplified tree that has the same root node but less terminal nodes than T_{\max} . This is denoted by $T \leq T_{\max}$, the set of current terminal nodes of T is denoted by \tilde{T} and the number of terminal nodes in T by $|\tilde{T}|$. The pruning process consists in finding the best compromise between the number of terminal nodes and the mean deviance of the predicted variable y between each terminal group.

For any subtree $T \leq T_{\max}$, let the penalized deviance $D_\alpha(T)$ be the following linear combination:

$$D_\alpha(T) = D(T) + \alpha |\tilde{T}|$$

with: - $D(T) = \sum_{t \in \tilde{T}} \sum_{(y_i, x_i) \in t} (y_i - \bar{y}(t))^2$, the total deviance of a tree T ,

- $|\tilde{T}|$, the complexity of T *i.e.* the number of terminal nodes in T ,

- $\alpha \geq 0$, the cost parameter which weights the number of terminal nodes.

Now, for each value of α , the subtree $T(\alpha) \leq T_{\max}$ which minimizes $D_\alpha(T)$ must be found, *i.e.*

$$D_\alpha(T(\alpha)) = \min_{T \leq T_{\max}} D_\alpha(T)$$

Although α values may vary continuously, the minimal error-complexity trees grown on L result in a decreasing finite sequence $T_1 > T_2 > \dots > \{t_1\}$ of k pruned trees such that $T_1 \leq T_{\max}$ and $\{t_1\}$ is the root node. Thus, at each T_k is associated an α_k , a threshold value such that $\alpha_k \leq \alpha \leq \alpha_{k+1}$, with $\alpha_1 = 0$ and $T(\alpha) = T(\alpha_k) = T_k$, $k \geq 1$ (Breiman *et al.*, 1984).

The standard error committed by each tree of the sequence $\{T_k\}$ could be estimated considering $D(T_k)$ as a criterion of prediction error. But, using the sample that was used to build the sequence of pruned subtrees would lead to an unrealistic low estimation of the total deviance. By construction, the largest tree T_1 would be selected as the optimum-sized tree *i.e.* the tree with the lowest standard error. To determine the best-sized tree, a better estimate of the deviance $D(T_k)$ must be found. Cross-validation method is a solution to limit the over-optimistic bias of the estimation of the total standard error (Breiman *et al.*, 1984; Efron, 1983).

6.3 Advantages and disadvantages

Tree-based methods have several attractive properties when compared to traditional methods. They provide a simple rule for classification or prediction of observations, they handle interactions among variables in a straightforward way, they can easily handle a large number of predictor variables, and they do not require assumptions about the distribution of the data. However, tree-based methods do not conform to the usual hypothesis-testing framework.

7 Partial least square regression (PLS) (Sovan Lek)

7.1 Introduction

In such so-called *soft science* applications, the researcher is faced with many variables and ill-understood relationships, and the object is merely to construct a good predictive model. For example, ecological communities are often used to estimate its biodiversity and related to its structure; the amount of different environmental variables characteristic for the studied site. In this case, the factors are the measurements that comprise the environmental data; they can number in the dozens or hundreds but are likely to be highly collinear. The responses are community amounts that the researcher wants to predict in the future samples.

Partial least squares (PLS) is a method for constructing predictive models when the factors are many and highly collinear. Note that the emphasis is on predicting the responses and not necessarily on trying to understand the underlying relationship between the variables. For example, PLS is not usually appropriate for screening out factors that have a negligible effect on the response. However, when prediction is the goal and there is no practical need to limit the number of measured factors, PLS can be a useful tool. The general idea of PLS is to try to extract from many factors a few underlying or latent factors, accounting for as much of the manifest factor variation as possible while modelling the responses well. Figure 7.1 gives a schematic outline of the method. The overall goal is to use the factor to predict the responses in the population. This is achieved indirectly by extracting latent variable T and U from sampled factors and responses, respectively. The extract factors T (also referred to as X -scores) are used to predict the Y -scores U , and then the predicted Y -scores are used to construct predictions for responses. For the detail of PLS methods, readers can refer to Geladi & Kowalski (1986), Höskuldsson (1988), and Helland (1990)

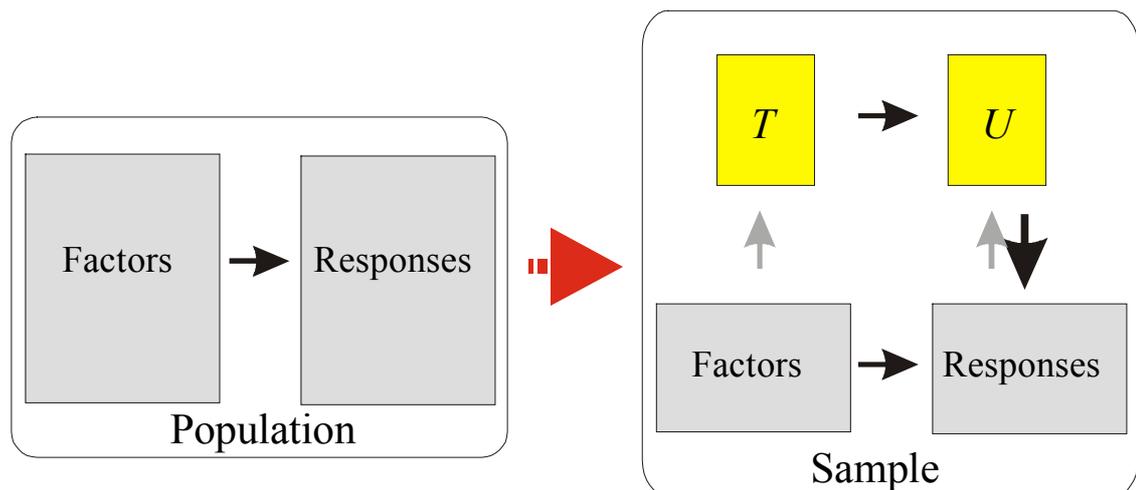


Figure 7.1 Indirect modelling.

7.2 Partial Least Square Regression algorithm

PLS is a regression technique to solve the linear model in a stepwise fashion, including every predictor variable in the model. PLS bears some resemblance to principal component regression (PCR) in that PCR also creates an orthogonal set of variables. However, in PCR the orthogonal variable extraction is independent of the target variables and a subsequent MR step is needed to relate target and explanatory variables, while in PLS the orthogonal set of variables is constrained to maximize directly the communality of the predictor and response variable blocks. PLS applies this constraint by using *NIPALS* (Non-linear Iterative Partial Least Squares) rather than diagonalization, to extract factors.

Denote the block of explanatory variables as: $X = \{x_{ij}\}$, $i=1,n$ $j=1,p$, where n is the number of rows (compounds) and p is the number of explanatory columns.

The block of target variables is denoted as: $Y = \{y_{ik}\}$ $i=1,n$ $k=1,r$, where r is the number of target data (dependent variables).

There is an optional associated weight vector: $W = \{w_i\}$ $i=1,n$

PLS calculates an orthogonal set of explanatory variables that are linear combinations of the original p variables. In each step an iterative procedure calculates the weights of the linear combinations in both blocks and a new set of latent variables.

1. start with the first response variable as latent variable of the response block
 $v_{i,a} = y_{i,1}$, $i=1,n$

2. weights of the linear combination in the first block $c_{j,a} = \sum_{i=1,n} w_i x_{i,j} v_{i,a}$

3. normalize the weight vector to length one $c_{j,a} = \frac{c_{j,a}}{\sqrt{\left(\sum_{i=1,p} c_{i,a}^2\right)}}$

4. latent variable of the predictor block $u_{i,a} = \sum_{j=1,p} x_{i,j} c_{j,a}$

5. weights of the linear combination in the response block $d_{k,a} = \sum_{i=1,n} w_i y_{i,k} u_{i,a}$

6. normalize the weight vector to length one $d_{k,a} = \frac{d_{k,a}}{\sqrt{\sum_{i=1,r} d_{i,a}^2}}$

7. latent variable of the response block $v'_{i,a} = \sum_{k=1,r} y_{i,k} d_{k,a}$

8. test for convergence if $\sum_{i=1,n} (v_{i,a} - v'_{i,a})^2 > \epsilon$ goto 2

9. inner relationship coefficient ρ $\rho_a = \frac{\sum_{i=1,n} w_i u_{i,a} v_{i,a}}{\sum_{i=1,n} w_i u_{i,a}^2}$

10. loading of the predictor block (projection on the model $\rho^* u$)

$$b_{j,a} = \frac{\sum_{i=1,n} w_i x_{i,j} \rho_a u_{i,a}}{\sum_{i=1,n} w_i (\rho_a u_{i,a})^2}$$

11. residuals in the block of explanatory variables $x_{i,j} = x_{i,j} - b_{j,a} \rho_a u_{i,a}$

12. residuals in the block of target variables $y_{i,k} = y_{i,k} - d_{k,a} \rho_a u_{i,a}$

This procedure is reiterated until $a=A$, i.e., until the specified number of components have been obtained, each time starting from the residual matrices of the previous step. If all possible components are calculated, i.e., $A=p$, PLS gives the conventional least squares regression equation, here called a full model.

Crossvalidation is used to determine the number of components that yields an optimally predictive model (refer to the [crossvalidation procedure](#)). Usually a model with less than the maximum number of components ($A < p$) gives a better crossvalidated sum of squared residuals than the full model. The model with optimal number of components has a higher standard error of fit than a full-rank model, but is generally more reliable for prediction. Thus the crossvalidated PLS model is usually less subject to errors of over-specification than is a regression model.

Bootstrap can be used to calculate confidence intervals around the model parameters (refer to the [bootstrapping procedure](#)). The number of crossvalidation groups and bootstrap samples is user-specified.

Other useful PLS statistics include:

The residual e_{ij} for the descriptor value X_{ij} from the j^{th} column of the i^{th} row:

$$e_{ij} = X_{ij} - \sum_{a=1,A} u_{i,a} \cdot \rho_a \cdot b_{ja}$$

The Modelling Power function MP : $MP_{(j)} = 1 - \sum_{i=1,n} \frac{e_{ij}^2}{(s_j^2 \cdot n)}$, where: $s_j^2 =$ variance of the j^{th} column

The (X-)Outlier Statistic function OS : $OS_{(i)} = \sum_{j=1,p} \frac{e_{ij}^2}{s_j^2 \cdot (1 - MP_{(j)} \cdot p)}$

The Discriminant Power function, $DP(j)$: $DP_{(j)} = \sum_{a=1,A} c_{ja}^2$

7.3 Advantages and disadvantages

Research in ecological sciences sometime involves controllable and/or easy-to-measure variables (*factors*) to explain, regulate, or predict the behaviour of other variables (*response*). When the factors are few in number, are not significantly redundant (collinear) and have a well-understood relationship to the responses, then multiple linear regression (MLR) can be a good way to turn data into information. However, if any of these three conditions breaks down, PLS can be more efficient and appropriate than MLR.

ARTIFICIAL NEURAL NETWORKS

8 Artificial Neural Networks (ANNs)

8.1 Historic developments

Over the last ten years, the use of Artificial Neural Networks (ANNs) in ecological modelling is discussed (Colasanti, 1991), which resulted in different applications (e.g. Adams *et al.*, Albiol *et al.*, 1995; Balls *et al.*, 1996; Baran *et al.*, 1996; Brey *et al.*, 1996; Chon *et al.*, 1996; French & Recknagel, 1994; Komatsu *et al.*, 1994; Lek *et al.*, 1995, 1996a, 1996b, 1996c; Recknagel *et al.*, 1997; Scardi, 1996; Spitz *et al.*, 1996). Typical ecological applications of ANNs include amongst others:

- pattern recognition and classification in taxonomy (e.g. Nakano *et al.*, 1991; Simpson *et al.*, 1992, 1993; Boddy *et al.*, 2000),
- remote sensing (e.g. Carpenter *et al.*, 1999; Civco, 1993; Gross *et al.*, 1999; Keiner & Yan, 1998; Kimes *et al.*, 1996; Mann & Benwell, 1996),
- GIS data analysis (e.g. Hilbert & Ostendorf, in press; Silveira *et al.*, 1996).
- empirical models of ecological processes (Aoki & Komatsu, 1997; Aoki *et al.*, 1999; Barciela *et al.*, 1999; Brey *et al.*, 1996; Brey & Gerdes, 1998; Brosse *et al.*, 1999; Lae *et al.*, 1999; Mastrotillo *et al.*, 1997, 1998; Recknagel *et al.*, 1997; Scardi, 1996, 2000; Scardi & Harding, 1999),
- tools for predicting community structure or population characteristics (Aussem & Hill, 1999; Baran *et al.*, 1996; Guegan *et al.*, 1998; Giske *et al.*, 1998; Lek *et al.*, 1996a, 1996; Scardi *et al.*, in press; Schleiter *et al.*, 1999; Wagner *et al.*, 2000)
- water management (e.g. Kastens & Featherstone, 1996),
- time series analysis and prediction (e.g. Recknagel, 1997; Chon *et al.*, in press),
- ecosystem dynamics (e.g. with recurrent back-propagation algorithms by Pineda, 1987; Chon *et al.*, in press; Jeong *et al.*, in press).

As for hydrological applications, although parametric statistical protocols and deterministic models have been the traditional approaches in forecasting water quality variables in streams, many recent efforts have shown that, when explicit information of hydrological subprocesses is not available, ANNs can then be more effective (Zhu *et al.*, 1994; Maier & Dandy, 2000).

A comprehensive overview of ANN applications in ecology and evolution is compiled by Lek & Guegan (2000).

9 Self-Organizing Maps (SOMs) (Jean Luc Giraudel)

9.1 Introduction

9.1.1 Development of SOMs

The Self Organizing Map (SOM) algorithm has been proposed by Kohonen in the early eighties (Kohonen, 1982). Since that time, SOMs have been used in a number of different applications. As for ecological ones, SOMs have been used to reveal the relationships between ecological communities, for instance to describe patterns in communities (Chon *et al.*, 1996), to analyse community data (Foody, 1999) or to model micro-satellite data (Giraudel *et al.*, 2000).

SOMs are among the most well known neural networks with unsupervised learning rules. They perform a topology-preserving projection of the data space onto a regular two-dimensional space and can be used to effectively visualise clusters (Kohonen, 1995). Consequently, SOMs are ideal for ecological ordination (Jongman *et al.* 1995) Already used in different areas, SOMs were able to recognize clusters in datasets where other statistical algorithms failed to produce meaningful clusters (e.g. Cho, 1997). With the U-Matrix method, SOMs can be used for clustering without prior knowledge of the number or size of the clusters and then for studying Multivariate Time Series (Ultsch, 1999).

The detection and the visualisation of clusters are very straightforward and often outperform the results obtained by classical classification methods. A drawback of the SOM algorithm is that the size and the shape of the map have to be fixed in advance. Growing self-organising networks have been proposed in order to deal with this problem (Villmann & Bauer, 1998), but this approach remains still to be applied to ecological data.

9.1.2 The SOM concept

The Kohonen Self-Organizing Map (SOM) is one of the most popular unsupervised artificial neural network (Kohonen, 1995), it performs a topology-preserving projection of the data space onto a regular two-dimensional space. SOM shares with the conventional ordination methods the basic idea of displaying a high-dimensional dataset in a lower dimensional space (usually a 2-dimensional space). This method is recommended for use in an exploratory approach for data sets in which unexpected structures might be found. The complete description of this method can be found in Kohonen (1995).

Let $\{X_1; \dots; X_p\}$ be p vectors of R^n . With SOM, these p vectors will be projected in a non-linear way onto a rectangular grid laid out on a hexagonal lattice with S hexagons: the Kohonen map (figure 9.1).

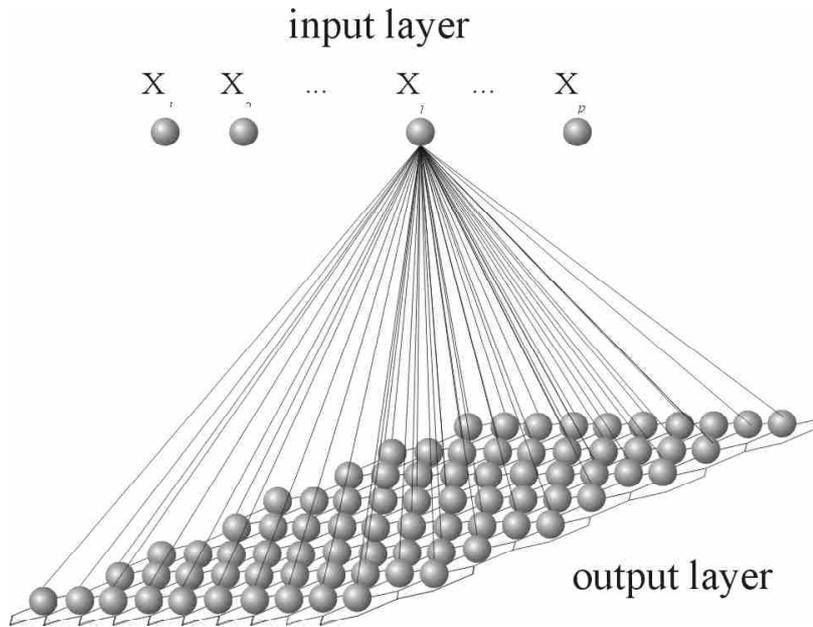


Figure 9.1 A two-dimensional Self-Organizing Map. Each sphere symbolises each neuron in the input layer and in the output layer (Kohonen map).

The goal of the SOM algorithm consists in putting the dataset on the map preserving the neighbourhood, so the similar vectors should be mapped close together on the grid. For this purpose, in each hexagon, a neuron will be considered. These neurons $(m_k)_{1 \leq k \leq S}$ are in fact *reference vectors* of R^n with components $(w_{ik})_{1 \leq i \leq n, 1 \leq k \leq S}$ to be computed.

The modifications of the reference vectors are made through an *Artificial Neural Network* (ANN). Modelling the human brain working, ANN has a learning ability: the components (w_{ik}) of each reference vector are computed during a training phase. The modifications of each (w_{ik}) take place by iterative adjustments. The Kohonen neural network consists of two layers: the first one (input layer) is connected to each vector of the dataset, the second one (output layer) forms a two-dimensional array of nodes (figure 9.1). In the output layer, the units of the grid (reference vectors) give a representation of the distribution of the data set in an ordered way. For learning, only input units are used, no expected-output data is given to the system: we are referring to unsupervised learning.

9.2 Self-Organizing Map algorithm

- Step 1: Epoch $t=0$, the reference vectors $(m_k)_{1 \leq k \leq S}$ are initialised with random samples drawn from the input dataset.
- Step 2: A sample vector $X_j = (x_{ij})_{1 \leq i \leq n}$ is randomly chosen as an input unit.
- Step 3: The distances between X_j and each reference vector are computed.

- Step 4: The virtual unit m_c closest to the input X_j is chosen as the winning neuron. m_c is called the Best Matching Unit (BMU).
- Step 5: The reference vectors $(m_k)_{k \leq S}$ are updated with the rule:

$$w_{ik}(t+1) = w_{ik}(t) + h_{ck}(t)[x_{ij}(t) - w_{ik}(t)] \quad (1)$$
- Step 6: Increase time t to $t+1$. If $t < t_{\max}$ then go to step 2 else stop the training.

In step 5, in the equation (1), the function $h_{ck}(t)$ is called the neighbourhood function and plays a very central role. During the learning process, the BMU defined in step 4 is not the only updated unit. In the grid, a neighbourhood is defined around the BMU and all units within this neighbourhood are updated. Several choices can be made for the definition of the neighbourhood function. For instance, the neighbourhood can be written in terms of the Gaussian function:

$$h_c(t) = \alpha(t) \cdot \exp\left(-\frac{\|r_k - r_c\|}{2\sigma^2(t)}\right) \quad (2)$$

- $\|r_k - r_c\|$ is the Euclidean distance on the map between the winning unit m_c and each reference vector m_k .
- σ is a decreasing function of time which defines the width of the part of the map affected by the learning.
- α is the "learning-rate factor", it is a decreasing function of the time. σ and α both converge towards 0.

The learning is broken down into two parts:

the *ordering phase*: during this phase, the virtual stations are widely modified in a large neighbourhood of the Best Matching Unit. So, this occurs with large values for α and σ .

the *tuning phase*: when this second phase takes place, only the virtual units adjacent of the Best Matching Unit are modified. This phase is much longer than the former one and α is decreasing very slowly towards 0.

When the learning process is finished, The sample vectors can be mapped. For this purpose, the BMU is computed for each sample vector and this one can be represented in the corresponding hexagon.

10 Back Propagation Artificial Neural Network (ANNbp) (Muriel Gevrey)

10.1 Introduction

The back propagation neural networks, also called multilayer feed-forward neural networks or multilayer perceptron, are very popular and are used for a wide variety of problems more than other types of neural networks. The ANNbp is based on the supervised procedure, i.e., the network is built with a dataset where the outputs are known. ANNbp is a powerful system, often capable of modelling complex relationships between variables. For a given input, one can predict an output.

10.2 Back Propagation Artificial Neural Network algorithms

10.2.1 Structure

ANNbp is a layered feed-forward neural network, in which the non-linear elements (neurons) are arranged in successive layers, and the information flows unidirectionally, from input layer to output layer, through the hidden layer(s) (figure 10.1). As can be seen in this figure, neurons from one layer are connected to all neurons in the adjacent layer, but no lateral connection between neurons within one layer, or feedback connection are possible. The number of input and output neurons depends on the number of explanatory and explained variables, respectively. The hidden layer(s) is (are) an important parameter in the network.

10.2.2 Algorithms

ANNbp learning and update procedure is based on a relatively simple concept: if the network gives the wrong answer, then the weights are corrected so that the error lessens, so future responses of the network are more likely to be correct. The conceptual basis of the back propagation algorithm was presented to a wide readership by Rumelhart *et al.* (1986).

In a training phase a set of input/target pattern pairs is used for training, which is presented to the network many times. After training is stopped, the performance of the network is tested. The ANNbp learning algorithm involves a forward-propagating step followed by a backward-propagating step.

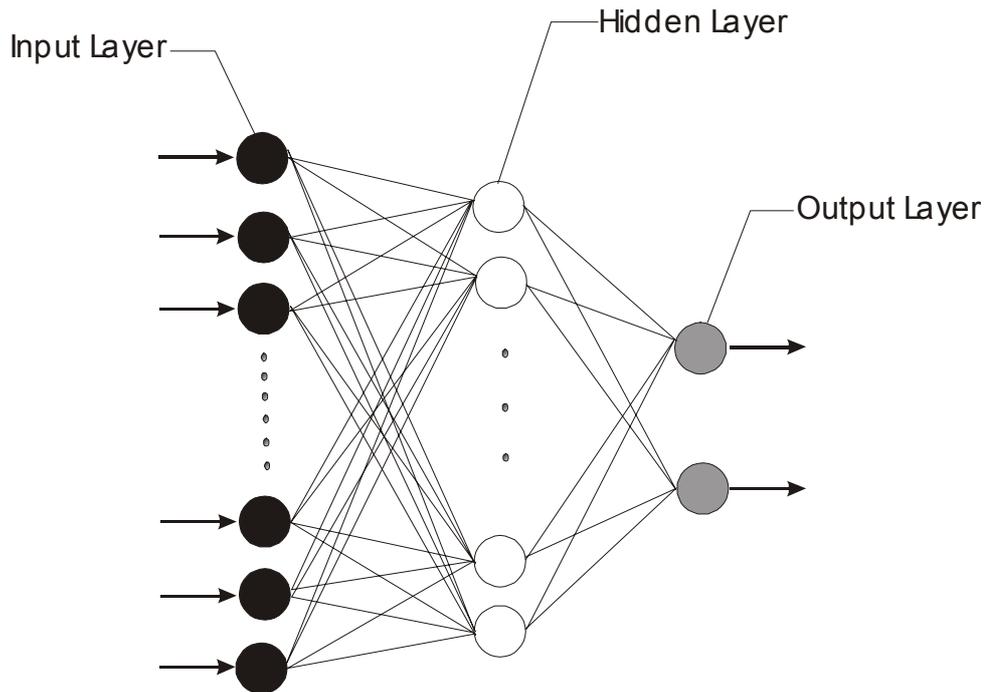


Figure 10.1 Schematic illustration of a three-layered feed-forward neural network, with one input layer, one hidden layer and one output layer.

10.2.3 Forward-propagating step

Like a real neuron, the artificial neuron has many inputs, but only a single output, which can stimulate many other neurons in the network. The neurons are numbered, for example the one neuron in figure 10.2 is called j .

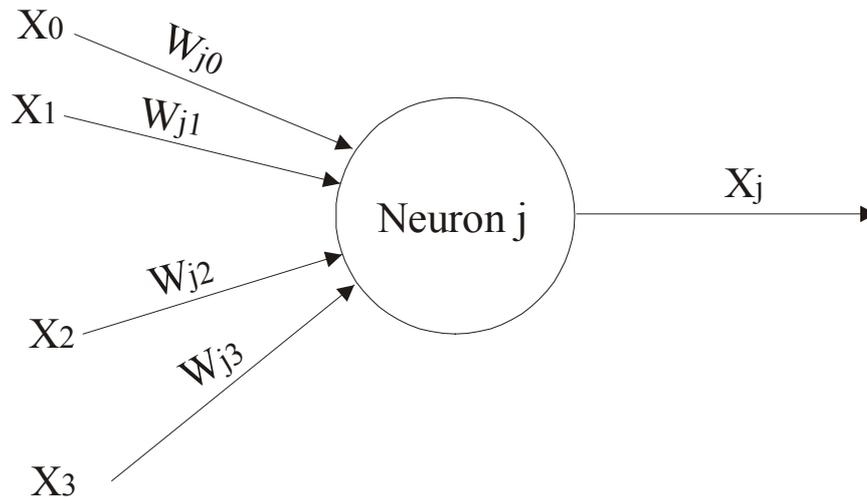


Figure 10.2 Basic processing neuron in a network. Each input connection value (x_i) is associated with a weight (w_{ji}).

The input the j th neuron receives from the i th neurons is indicated as x . Each connection to the j th neuron is associated to a quantity called weight. The weight on

the connection from the i th neuron to the j th neuron is denoted w_{ji} . An input connection may be excitatory (positive weight) or inhibitory (negative weight). A net input (called activation) for each neuron is the sum of all its input values multiplied by their corresponding connection weights, expressed by the formula:

$$a_j = \sum_i w_{ji} x_i + B_j$$

where i is the total number of neurons in the previous layer, B_j is a bias term, which influences the horizontal offset of the function. The bias B_j may be treated as the weight from the supplementary input unit, which has a fixed output value of 1. Once the activation of the neuron is calculated, we can determine the output value by applying a transfer function:

$$x_j = f(a_j)$$

We can use many different transfer functions, e.g. linear function, a threshold function, a sigmoid function, etc. The sigmoid function is often used in ecology, its formula is:

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

The weight plays an important role in propagation of the signal in the network. They establish a link between an input pattern and the associated output pattern, i.e. they contain the knowledge of the neuronal network about the problem/solution relationship.

10.2.4 Backward-propagating step

The backward-propagating step begins with the comparison of the network output pattern to the target value, when the difference (or error) is calculated. The backward-propagating step then calculates error values and changes the incoming weights, starting with the output layer and moving backward through the successive hidden layers.

The error signal associated with each processing unit indicates the amount of error associated with that unit. This parameter is used during the weight-correction procedure, while learning is taking place. A large value for the error signal indicates a large correction should be made to the incoming weights; its sign reflects the direction in which the weights should be changed. The adjustment of weight depends on three factors: the error value of the target unit, the output value for the source unit and the learning rate. The learning rate commonly between 0 and 1, determine the rate of learning of the network.

10.2.5 Training the network

Before starting the training, the connection weights are set to small random values. Next the input patterns are applied to the network to obtain the output. The differences between the output calculations and the target expected are used to modify the weights. One complete calculation is called an epoch or iteration. This process is repeated until a suitable level of error is achieved. Using a parameter

called momentum, chosen generally between 0 and 1 allow getting out of a local minimum.

10.2.6 Testing the network

A testing set of data serves to assess the performance of the network after training is complete. The input patterns are fed into the network and the desired output patterns compared with those given by the neural network. The agreement or the disagreement of these two sets gives an indication of the performance of the neural network model. If it is possible, the best solution is to divide the data set in the aim to use two different sets of data, one for the training and the testing stage and the second one to validate the model (Mastrorillo *et al.* 1998). Different partitioning procedures existed according to the size of the available dataset: k-fold cross-validation or hold-out (Utans & Moody 1991; Efron & Tibshirani 1995; Kohavi & Wolpert 1996; Friedman 1997), leave-one-out (Efron, 1983; Kohavi 1995).

The network can be overtrained or overfitted, that is it loses its capacity to generalize. Three parameters are responsible of this phenomenon: the number of epochs, the number of hidden layers and the numbers of neurons in each hidden layer. It is very important to determine the appropriate numbers of these elements in ANNbp modelling.

10.3 Advantages and disadvantages

ANNs can be regarded as an extension of the many conventional techniques for understanding complex data, and they have been developed over several decades. Feed-forward artificial neural networks are powerful to perform non-linear pattern discrimination. They are powerful tools and popular especially in pattern recognition and other decision-making, forecasting and signal processing and in the modelling of complex non-linear systems by fitting the network to non-linear data. Those are major advantages of using ANNbp in ecology where the relationships in the data sets are always non-linear and complex.

However, sometimes they do not show high performance abilities. These failures are due to inadequate training, inappropriate architecture for the used data set, and non-separability of the feature data (inappropriate data...). These problems underline the necessity of model calibration for its successful use. A drawback of ANNbp is that the gradient training encounters multiple local minima. Another one comes from the model type. Actually, ANNbp is black-box type model that means that it is not possible to interpret the phenomenons that occur inside the network. The complexity of the model comparing to the classical one as the regression models can also be seen as a disadvantage.

11 Kernel-induced nonlinear model (KINM) (Sun Ruixiang)

11.1 Introduction

Kernel-induced nonlinear models (KINMs) are derived from the principle of statistical learning theory (Vapnik, 1995). They have attracted more and more attention since the 1990's and have been applied in many areas due to the attractive features and the promising performances. As we know, multiple linear regression (MLR) or its generalized version is widely used for most of ecological problems. In order to deal with the nonlinearity, artificial neural networks (ANNs) are introduced and make a great progress toward to more accurate models (Lek, 2000). Nevertheless, there is still some weakness in ANN, such as the local minimum of the error surface, the difficulties with generalisation. Therefore, it is very significant to find an alternative or robust method to empirical data modelling in ecological area.

11.2 Kernel-induced nonlinear models algorithms

11.2.1 SRM versus ERM

The structural risk minimization (SRM) principle has been shown to be superior to the most commonly used empirical risk minimization (ERM) principle employed in ANN. Let (\mathbf{x}, \mathbf{y}) be a vector of observations, and f be a function to be estimated or modelled from the observations. The usual ERM inspired model minimized the risk (error),

$$R[f] = \int_{X \times Y} (y - f(\mathbf{x}))^2 P(\mathbf{x}, y) d\mathbf{x} dy \quad (1)$$

As in most general cases, the probabilistic density function $P(\mathbf{x}, y)$ is unknown, the approximated alternative version, according to ERM principle, is

$$R_{emp}[f] = \frac{1}{l} \sum_{i=1}^l (y_i - f(\mathbf{x}_i))^2 \quad (2)$$

in which l is the number of the observations.

This approximation makes sense only if,

$$\lim_{l \rightarrow \infty} R_{emp}[f] = R[f], \quad (3)$$

which is true from the law of large numbers.

In SRM, the following bound holds with probability $1 - \delta$,

$$R[f] \leq R_{emp}[f] + \sqrt{\frac{h(\ln \frac{2l}{h} + 1) - \ln(\frac{\delta}{4})}{l}} \quad (4)$$

in which h is the VC dimension, a scalar value to measure the capacity of a set of functions. SRM minimises the right side of the inequality (4).

11.2.2 KINM concept

The main idea behind KINMs is to map the initial data space to the high dimensional feature space by a nonlinear kernel function chosen a priori, also called kernel-induced space. In that space, a linear regression is possible to be implemented, which corresponds to a nonlinear regression function in the initial data space.

One of the most important concepts in KINMs is the loss function (L), which determines how to penalize the function according to the amounts of the error (ϵ) between the desired and the practical values. Figure 11.1(a) is the commonly used quadratic loss function and (b) is the linear loss function. In KINMs, in order to obtain the sparse distribution of the “support vectors” (explained later), a new type extended from the linear one is developed as shown in Figure 11.1(c), the so-called ϵ -insensitive loss function.

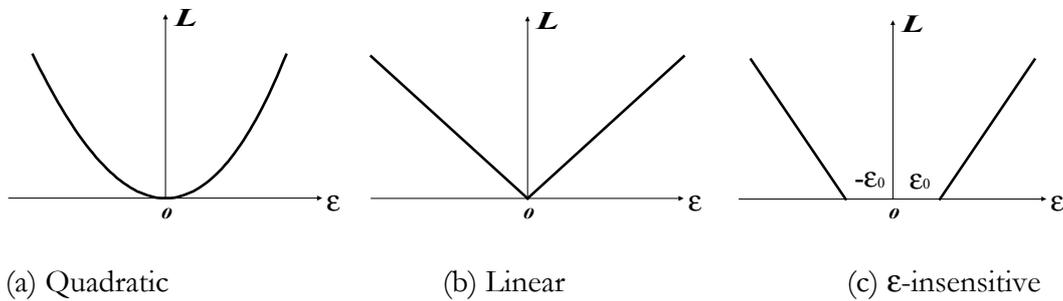


Figure 11.1 Three types of Loss Functions.

The form of the ϵ -insensitive loss function is:

$$L(y, \epsilon) = \begin{cases} 0 & |f(\mathbf{x}) - y| < \epsilon_0 \\ |f(\mathbf{x}) - y| - \epsilon & \text{others} \end{cases} \quad (5)$$

The classical linear regression problem is to find the weighted vector \mathbf{w} and scalar b for the data set $(\mathbf{x}_i, y_i), i = 1, 2, \dots, l, \mathbf{x} \in R^n, y \in R$, so that:

$$f(\mathbf{x}) = (\mathbf{w} \cdot \mathbf{x}) + b \quad (6)$$

meets the minimum of the quadratic loss function in the data set.

The primal problem of KINMs can be defined as follows:

$$\begin{aligned} \text{Min} \quad & \|\mathbf{w}\|^2 + C \sum_{i=1}^l (\xi_i^2 + \hat{\xi}_i^2), \\ \text{Subject to} \quad & (\langle \mathbf{w} \cdot \mathbf{x}_i \rangle + b) - y_i \leq \epsilon_0 + \xi_i, i = 1, 2, \dots, l \quad (7) \\ & y_i - (\langle \mathbf{w} \cdot \mathbf{x}_i \rangle + b) \leq \epsilon_0 + \hat{\xi}_i, i = 1, 2, \dots, l \\ & \xi_i, \hat{\xi}_i \geq 0, i = 1, 2, \dots, l \end{aligned}$$

where $\xi_i, \hat{\xi}_i, i = 1, 2, \dots, l$ are the slack variables; C is the parameter to measure the trade-off between complexity and losses.

According to the Lagrange multiplier method, the optimization problem (7) can be solved as a convex programming which has a global minimum. Vapnik (1995) and Gunn (1998) provide more details on both the theoretical and the practical aspects of KINMs.

11.2.3 KINM algorithm

The primal problem of KINMs, (7) can be transformed to the dual one:

$$\begin{aligned} \text{Max } \mathbf{W}(\alpha, \alpha^*) &= -\frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) K(x_i, x_j) + \sum_{i=1}^l \alpha_i^* (y_i - \varepsilon_0) - \alpha_i (y_i + \varepsilon_0) \\ \text{Subject to } & \mathbf{0} \leq \alpha_i \leq C, i=1,2,\dots,l \quad (8) \\ & \mathbf{0} \leq \alpha_i^* \leq C, i=1,2,\dots,l \\ & \sum_{i=1}^l (\alpha_i^* - \alpha_i) = 0 \end{aligned}$$

where $\alpha_i^*, \alpha_i, i=1,2,\dots,l$ are 2/Lagrange multipliers and $K()$ is the kernel function.

After obtaining the multipliers $\alpha_i^*, \alpha_i, i=1,2,\dots,l$ by solving (8), the regression function is then given by:

$$f(x) = \sum_{SVs} (\bar{\alpha}_i - \bar{\alpha}_i^*) K(x_i, x) + \bar{b} \quad (9)$$

Many literature gives the commonly used kernel functions. Generally speaking, any symmetric function, which meets the Mercer condition (Vapnik,1995) can be a kernel function. Among them, Gaussian radial basis function and polynomial function are two examples.

Gaussian radial basis function

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{w}\right) \quad (10)$$

where w is the width constant ($w > 0$).

Polynomial function

$$K(x_i, x_j) = (x_i \cdot x_j)^d \quad (11)$$

where d is the exponent constant, $d \in \mathbf{N}^+$.

The ε -insensitive loss function (5) makes the solutions of problem (8) be sparse, i.e. only a few of the $\alpha_i^*, \alpha_i, i=1,2,\dots,l$ are not zeros. The data points which correspond to the nonzero (larger than zero) are called ‘‘Support Vectors (SVs)’’. This leads to the capacity of KINMs does not depend on the dimension of the data space because KINMs only take use of the most informative data points, ‘‘Support Vectors’’. The optimisation problem (8), in essence, is a convex programming, which has a global maximum. All these afford the superiority to KINMs, compared with ANN and other regression methods.

11.3 Advantages and disadvantages

The advantages of KINMs, in comparison with ANNs, are due to the implementattion of the structural risk minimization (SRM) principle. This principle has been shown to be superior to the most commonly used empirical risk minimization (ERM) principle employed in ANN. SRM minimizes an upper bound on the VC dimension, i.e. the generalisation error, while ERM minimizes the error

on the training data, usually in a quadratic fashion. This difference equips the power of KINMs when modelling the nonlinear complex problems.

12 Counterpropagation network (Young-Seuk Park)

12.1 Introduction

The counterpropagation network (CPN), proposed by Hecht-Nielsen (1987), is a combined network of the two artificial neural networks: the Kohonen self-organizing map (SOM) (Kohonen, 1982) and the Grossberg outstar (Grossberg, 1982). The name counterpropagation derived from the initial presentation of this network as a five-layered network with data flowing inward from both sides (figure 12.1). There is literally a counterflow of data through the network. The network is designed to approximate a continuous function $f: \mathcal{A} \subseteq \mathbb{R}^m \rightarrow \mathcal{B} \subseteq \mathbb{R}^n$ defined on a data set \mathcal{A} , and serves as a statistically optimal self-programming lookup-table (Hecht-Nielsen, 1987).

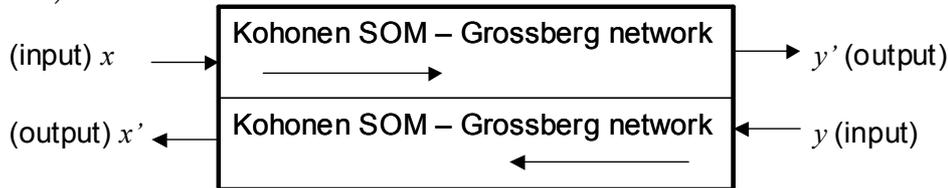


Figure 12.1 Structure of a full counterpropagation network.

12.2 Counterpropagation network algorithm

The full network works best if the inverse function f^{-1} exists and is continuous. It is assumed that the \mathbf{x} and \mathbf{y} vectors are drawn from \mathcal{A} and \mathcal{B} , respectively. During the training, (\mathbf{x}, \mathbf{y}) of f (where $\mathbf{y} = f(\mathbf{x})$) are presented to the network from both sides (figure 12.1). These \mathbf{x} and \mathbf{y} vectors then propagate through the network in a counterflow manner to yield output vectors \mathbf{x}' and \mathbf{y}' which are intended to be approximations of \mathbf{x} and \mathbf{y} , respectively (Hecht-Nielsen, 1987; Lin & Lee, 1996). Although this is an accurate picture of the network, it is complex; thus a simplified forward-only CPN is preferred with no loss of accuracy (Hecht-Nielsen, 1987) (figure 12.2).

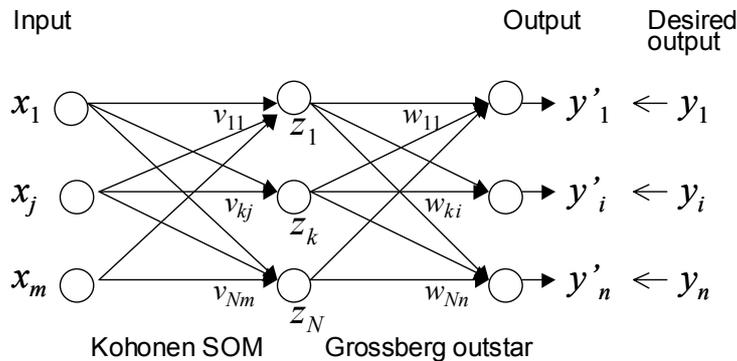


Figure 12.2 Structure of a forward-only counterpropagation network

As with any neural network, there are a few steps that must be performed to test a network's performance. Initially the data for \mathbf{x} (explanatory variables) and \mathbf{y} (dependent variables) are given to the SOM layer and the Grossberg layer, respectively. For the CPN the learning process occurs in two phases. First, the SOM layer must be trained. The learning process is carried out in an unsupervised mode to follow the general SOM learning rules. After learning the SOM layer, the Grossberg layer should be trained. This is carried out in a supervised mode. The weight vector \mathbf{w} is updated according to the Grossberg outstar learning rule:

$$\mathbf{w} = \mathbf{w} + \exists(\mathbf{y} - \mathbf{w})\mathbf{z}$$

where \exists is learning rate, and \mathbf{z} is assigned to 1 for winner neuron of the SOM layer while set to 0 for all other neurons. By repeating this process until the weight difference become sufficiently small, the effective information characterising relations of the two variable sets are preserved in the weights of the network. Finally, the trained CPN functions exactly as an optimal self-programming lookup-table.

12.3 Advantages and disadvantages

Strength and weakness of CPN depend on the geometry and probabilities of the inputs and outputs. If that information is known or estimated, this can be a very good approach. Another advantage is speed of convergence. Compared to other mapping networks, it typically requires fewer training steps to achieve its best performance. This kind of approach shows how combining different network architectures, unsupervised network and supervised network, can produce good results for certain classes of patterns. Finally, the CPN can provide the clustering and predicting values at the same model. However there are also some limitations as follows; a large database is required to train the network, and the predicted values by the network are dependent on the number of groups in Kohonen SOM layer.

13 Competitive neural network in combination with Linear Vector Quantisation (Wies Akkermans)

13.1 Introduction

Artificial Neural Networks (ANNs) are non-linear models. Two main types of ANNs can be distinguished: supervised and unsupervised ones. Unsupervised networks can be used for clustering a set of units or patterns, i.e. for discovering 'groups' when no explicit information about group membership is present in the data. Supervised networks on the other hand are applicable when the goal is to predict an observed dependent variable from a set of predictors, that is, they are models for regression and classification. One of the advantages of neural nets over more classical techniques such as k-means clustering or ordinary least squares regression is that no assumptions have to be made regarding the distribution of variables. Furthermore, being nonlinear, these methods can be used to detect and describe a host of nonlinear relationships.

In this paragraph the unsupervised Competitive Artificial Neural Networks (CANNs) and the supervised LVQ network will be described. Applying these two networks in combination may give a powerful tool for fast classification of future observations. An example is given in paragraph 12.4.

13.2 Competitive neural network and Linear Vector Quantisation algorithms

13.2.1 Competitive ANNs

A competitive neural network consists of two layers, an input layer and an output layer. The input layer consists of M input variables x_1, x_2, \dots, x_M ; the output layer consists of one neuron for each cluster that will be formed. The researcher has to decide on the number of clusters, say K . When there are K clusters, the goal of a Competitive ANN is to find K 'prototype vectors' that can be used to describe these clusters. Starting from K initial prototypes (either randomly chosen, or obtained through for example K-means clustering), for each input pattern in turn the distance to each prototype is calculated. The prototype it is closest to, is then adjusted somewhat into the direction of this input pattern. This process is iterated until the change in the adjustments becomes smaller than a predetermined threshold. Mathematically, provided both vectors have been scaled to the same (unit) length, the pattern and prototype that are closest also have the largest inner product. Now denote the prototype vectors by \mathbf{p} and place them into the rows of an $K \times M$ matrix \mathbf{P} . Then $\mathbf{a} = \mathbf{P}\mathbf{x}$ gives the inner products of all K prototypes with \mathbf{x} , as the k -th element of \mathbf{a} is given by $a_k = \sum_{m=1}^M p_{km} x_m$. Hence, the prototype matrix thus formed also acts as a weight matrix. Now let prototype i be the one having the largest inner product

with \mathbf{x} , or alternatively, the one being closest to \mathbf{x} . The rows of the prototype (or weight) matrix \mathbf{P} are then updated using the so-called instar rule:

$$\mathbf{p}^{(k;t)} = \mathbf{p}^{(k;t-1)} + \alpha (\mathbf{x} - \mathbf{p}^{(k;t-1)}) \text{ if } k = i ;$$

$$\mathbf{p}^{(k;t)} = \mathbf{p}^{(k;t-1)} \text{ for all other } k.$$

in which the superscript k refers to the k -th prototype, i.e. the k -th row of \mathbf{P} , and the superscripts t and $t-1$ to cycles in the iteration. The parameter α has to be chosen by the investigator. This updating rule has the effect that the prototype closest to the input vector is moved a little into the direction of \mathbf{x} , and all other prototypes remain unchanged.

Starting from an initial unit length prototype matrix $\mathbf{P}^{(0)}$, the algorithm for the Competitive ANN is to iterate the following steps until convergence:

1. Select the next input vector \mathbf{x} , and scale it to have unit length.
2. Identify the prototype closest to the input vector. This can be done either by calculating the inner product of all prototypes with \mathbf{x} ; or by calculating the distance between \mathbf{x} and each prototype vector.
3. Update the prototypes according to the instar rule described above. For more see e.g. Haykin (1999) or Hagan et al (1996).
4. Compare with the previous value. If the difference is smaller than tolerance stop; otherwise go back to 1.

A competitive ANN is schematically represented in figure 13.1.

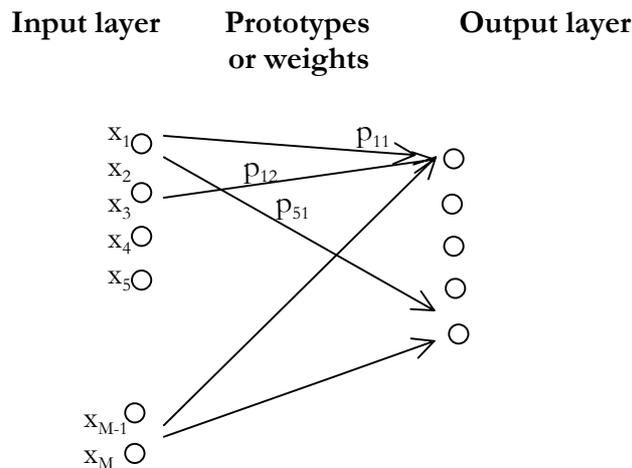


Figure 13.1 Schematisation of a competitive neural net (All neurons in the output layer should be connected to all neurons in the input layer).

13.2.2 Linear Vector Quantisation Network

The second network discussed in this paragraph is the Linear Vector Quantisation network. This is a network for classification, that is, it can be trained to learn the relation between a set of input variables x_1, x_2, \dots, x_M and a qualitative output variable Y . In ordinary statistical language: its parameters are iteratively estimated such as to minimise the prediction errors. The variable Y is represented as a dummy vector, that

is, if pattern \mathbf{x} belongs to class i , $1 \leq i \leq K$, then Y consists of a ‘one’ on position i , and of zeros everywhere else. Figure 13.2 gives a schematic description of an LVQ network. The circles represent ‘nodes’ in the network; the squares are indicators for the target variable which the network aims to reproduce. Each square represents one class or group. In addition to the input and output layers, this network has a third, so-called hidden layer. The hidden layer identifies, as it were, a number of ‘subclasses’ for each class. This feature of the network allows it to have classes formed from non-convex input regions. The number of hidden neurons has to be determined by the researcher.

The first layer in the network is a competitive layer; it is similar to above, except that here the prototypes and input patterns are not scaled to unit length. Again, let prototype i be the prototype closest to \mathbf{x} . The second layer merely combines the subclasses into broader classes. The weights of this second layer are not changed during the estimation process. Recall that in this case the ‘true’ class membership of \mathbf{x} is known, i.e. the output variable Y has been observed. Hence, it is possible to use this information in the updating of the weights: check if the predicted class i is equal to the target class $Y(\mathbf{x})$. If Y has been correctly predicted, the prototype vector i is moved a little into the direction of \mathbf{x} . This is similar to the procedure with the Competitive ANN above. However, if Y has been incorrectly predicted, now the prototype vector i is moved a little *away from* \mathbf{x} :

$$\begin{aligned} \mathbf{p}^{(k;t)} &= \mathbf{p}^{(k;t-1)} + \alpha (\mathbf{x} - \mathbf{p}^{(k;t-1)}) \text{ if } k = i \text{ and } i \text{ is the correct class;} \\ \mathbf{p}^{(k;t)} &= \mathbf{p}^{(k;t-1)} - \alpha (\mathbf{x} - \mathbf{p}^{(k;t-1)}) \text{ if } k = i \text{ and } i \text{ is an incorrect class;} \\ \mathbf{p}^{(k;t)} &= \mathbf{p}^{(k;t-1)} \text{ for all other } k. \end{aligned}$$

This could be called an extended instar rule. More details again can be found in Haykin (1999) and Hagan *et al.*(1996).

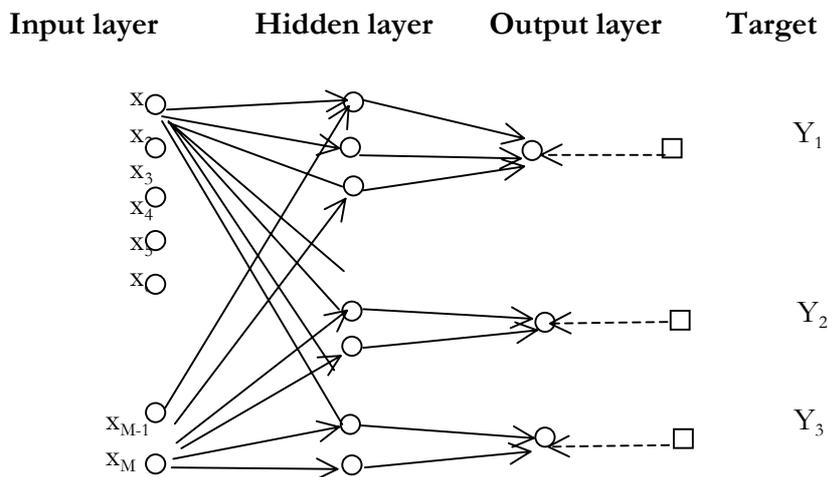


Figure 13.2 Schematisation of an LVQ network (The input layer should be fully connected to the hidden layer).

Summarising the algorithm for the LVQ: start from an initial prototype matrix $P^{(0)}$ and iterate the following steps until convergence:

1. Select the next input vector \mathbf{x} .
2. Identify the prototype closest to the input vector.
3. From this prototype, predict Y .
4. Update the prototypes according to the extended instar rule described above.
5. Compare with the previous value. If smaller than tolerance stop; otherwise go back to 1.

13.3 Advantages and disadvantages

Competitive ANNs can be used to discover clusters of natural sites. As an example, think of clustering aquatic sites on the basis of environmental characteristics (width, pH) and biotic data (e.g. abundance of benthic fauna). Once a satisfactory clustering has been achieved, an LVQ can then be trained to learn the features of the different clusters from only a subset of the input variables, for example from only variables that are easy to measure. Then future observations could be assigned to one of the clusters using this LVQ network, that is, using only easy-to-measure variables.

DYNAMIC MODELS

14 Structurally Dynamic Models (Sven Jørgensen)

14.1 Introduction

If we follow the general modeling procedure for dynamic bio-geo-chemical models, we will attain a model that describes the processes in the focal ecosystem, but the *parameters* will represent the properties of the state variables as they are in the ecosystem *during* the examination period. They are not necessarily valid for another period, because it is known that an ecosystem can regulate, modify and change them, if needed as response to the change in the prevailing conditions, determined by the forcing functions and the interrelations between the state variables. It is necessary, however, to introduce parameters (properties) that can change according to changing forcing functions and general conditions for the state variables (components) to optimize continuously the ability of the system to move away from thermodynamic equilibrium. So, it is hypothesized that adaptations and shifts in species composition can be accounted for in a model by a current change of parameters according to an ecological goal function. The idea is currently to test if a change of the most crucial parameters produces a higher goal function of the system and, if that is the case, to use that set of parameters.

The type of models that can account for the change in species composition as well as for the ability of the species to change their properties, that is to adapt to the prevailing conditions imposed on the species, are sometimes called structurally dynamic models.

It could be argued that the ability of ecosystems to replace present species with other better fitted species, can be considered by construction of models that encompass *all* actual species for the entire period that the model attempts to cover. This approach has, however, two essential disadvantages. The model becomes first of all very complex, as it will contain many state variables for each *trophic level*. It implies that the model will contain many more parameters that have to be calibrated and validated and this will introduce a high uncertainty to the model and will render the application of the model very case specific. In addition, the model will still be rigid and not give the model the property of the ecosystems of having continuously changing parameters even without changing the species composition.

Changes of the structure can be described by the introduction of a goal function. Exergy is used as a goal function as it describes the distance of the ecosystem (described by the model) from thermodynamic equilibrium = the sum of biomass and information. Figure 14.1 shows how exergy calculations are used to examine currently the properties of the species that are best survivors under the currently changed conditions. With a selected frequency it is examined whether the exergy of the model could be greater by a 10% change of the parameters = the properties of the organisms

included in the model. If that is the case the properties are changed accordingly. When the change is limited to 10% and a certain selected frequency for instance once a week, it is because the structurally dynamic changes require a certain time, determined by the rate of the biological processes.

It is to a certain extent possible to relate the properties to the size, which is beneficial because that facilitates the optimisation of the goal function, as only a few parameters, for instance the average size of the three trophic layers, have to be varied when the optimum has to be determined.

Structurally dynamic models have been applied in 11 cases (Jørgensen, 1997, and Jørgensen and Fath, 2001). It was used to explain the failure and success of biomanipulation (Jørgensen and de Bernardi, 1998) and the intermediate disturbance hypothesis (Jørgensen and Padišak, 1997) to mention a few important case studies. In addition has been used to explain succession of phytoplankton species (Jørgensen and Padišak, 1997) and actually observed structural changes in several other cases, resulting from changes in the forcing function of the focal ecosystem.

14.2 Structurally dynamic models

Several goal functions have been proposed as shown in table 14.1, but only very few models, that account for change in species composition or for the ability of the species to change their properties within some limits, have been developed. Bossel (1992) uses what he calls six basic *orientors* or requirements to develop a system model, which can describe the system performance properly. The six orientors are:

1. *Existence*. The system environment must not exhibit any conditions, which may move the state variables out of its safe range.
2. *Efficiency*. The exergy gained from the environment should exceed over time the exergy expenditure.
3. *Freedom of action*. The system reacts to the inputs (forcing functions) with a certain variability.
4. *Security*. The system has to cope with the different threats to its security requirement with appropriate but different measures. These measures either aim at internal changes in the system itself or at particular changes in the forcing functions (external environment).
5. *Adaptability*. If a system cannot escape the threatening influences of its environment, the one remaining possibility consists in changing the system itself to cope better with the environmental impacts.
6. *Consideration of other systems*. A system must respond to the behavior of other systems. The fact that these other systems may be of importance to a particular system may have to be considered with this requirement.

Bossel (1992) applies maximization of a benefit or satisfaction index based upon balancing weighted surplus *orientor* satisfactions on a common satisfaction scale. The approach is used to select the model structure of continuous dynamic systems and is able to account for the ecological structural properties.. The approach seems very promising, but has unfortunately not been applied to ecological systems except in three cases.

Table 14.1 Goal functions proposed.

Proposed for	Objective function	Reference
Several systems	Maximum useful power or energy flow	Lotka (1924), Odum and Pinkerton (1955)
Several systems	Minimum entropy	Glansdorff and Prigogine (1971)
Networks	Maximum ascendancy	Ulanowicz (1980)
Several systems	Maximum exergy	Mejer and Jørgensen (1979)
Ecological systems	Maximum persistent organic matter	Whittaker and Woodwell (1971) O'Neill <i>et al.</i> (1975)
Ecological systems	Maximum biomass	Margalef (1968), Straskraba (1979)
Economic systems	Maximum profit	Various authors

Straskraba (1979) uses a maximization of biomass as the governing principle (table 14.1). The model computes the biomass and adjusts one or more selected parameters to achieve the maximum biomass at every instance. The model has a routine that computes the biomass for all possible combinations of parameters within a given realistic range. The combination that gives the maximum biomass is selected for the next time step and so on.

Exergy has been used most widely as a goal function in ecological models, and a few of the available case studies will be presented and discussed below in this paragraph. Exergy has two pronounced advantages as goal function compared with entropy and maximum power (table 14.1): It is defined far from thermodynamic equilibrium and it is related to the state variables, which are easily determined or measured. As exergy is not a generally used thermodynamic function, we need, however, first to present this concept.

Exergy expresses energy with a built-in measure of quality like energy. Exergy accounts natural resources (Eriksson *et al.*, 1976) and can be considered as fuel for any system that converts energy and matter in a metabolic process (Schrödinger 1944). Ecosystems consume energy, and an exergy flow through the system is necessary to keep the system functioning. Exergy measures the distance from the “inorganic soup” in energy terms, as will be further explained below.

Exergy, Ex , is defined by the following equation:

$$Ex = T_0 * NE = T_0 * I = T_0 * (S_{eq} - S) \quad (1)$$

in which T_0 is the temperature of the environment, I is the thermodynamic information, defined as NE , NE is the negentropy of system, i.e., $(S_{eq} - S)$ = the difference between the entropy for the system at thermodynamic equilibrium and the entropy at the present state.

It can be shown (Evans, 1969) that exergy differences can be reduced to differences of other, better known, thermodynamic potentials, which may facilitate the computations of exergy in some relevant cases.

As can be seen the exergy of the system measures the contrast - it is the difference in free energy if there is no difference in pressure, as may be assumed for an ecosystem - against the surrounding environment. If the system is in equilibrium with the surrounding environment the exergy is zero.

Since the only way to move systems away from equilibrium is to perform work on them, and since the available work in a system is a measure of the ability, we have to distinguish between the system and its environment or thermodynamic equilibrium

alias the inorganic soup. Therefore it is reasonable to use the available work, i.e., the exergy, as a measure of the distance from thermodynamic equilibrium.

Let us turn to the translation of *Darwin's theory* into thermodynamics, applying exergy as the basic concept. Survival implies maintenance of the biomass, and growth means increase of biomass. It costs exergy to construct biomass and biomass therefore possesses exergy, which is transferable to support other exergy (energy) processes. Survival and growth can therefore be measured by use of the thermodynamic concept exergy, which may be understood as *the free energy relative to the environment*; see equations 1.

Darwin's theory may therefore be reformulated in thermodynamic terms as follows: **The prevailing conditions of an ecosystem steadily change and the system will continuously select the species and thereby the processes that can contribute most to the maintenance or even growth of the exergy of the system.**

Ecosystems are open systems, and receive an inflow of solar energy. It carries low entropy, while the radiation from the ecosystem carries high *entropy*.

If the power of the *solar radiation* is W and the average temperature of the system is T_1 , then the *exergy gain* per unit of time, ΔEx is (Erikson *et al.* 1976):

$$\Delta Ex = T_1 * W (1/T_0 - 1/T_2) \quad (2)$$

where T_0 is the temperature of the environment and T_2 is the temperature of the sun. This exergy flow can be used to construct and maintain structure far away from equilibrium.

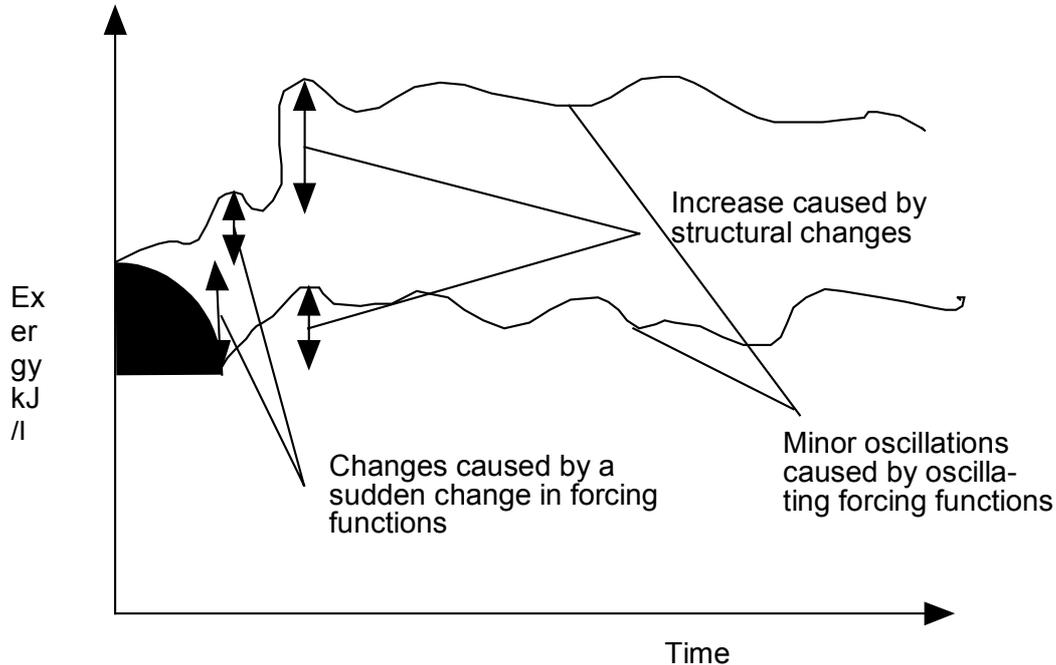


Figure 14.1 Exergy response to increased and decreased nutrient concentration.

Notice that the thermodynamic translation of Darwin's theory requires that populations have the properties of reproduction, inheritance and variation. The

selection of the species that contribute most to the exergy of the system under the prevailing conditions requires that there are enough individuals with different properties that a selection can take place - it means that the reproduction and the variation must be high and that once a change has taken place due to better fitness it can be conveyed to the next generation.

Notice also that the change in exergy is not necessarily ≥ 0 , it depends on the changes of the resources of the ecosystem. The proposition claims, however, that the ecosystem attempts to reach the highest possible exergy level under the given circumstances and with the available genetic pool ready for this attempt (Jørgensen and Mejer, 1977 and 1979). Compare figure 14.1, where the reactions of exergy for a lake ecosystem to an increase and a decrease in nutrient concentrations are shown. It is not possible to measure exergy directly - but it is possible to compute it, if the composition of the ecosystem is known. Mejer and Jørgensen (1979) have shown by the use of thermodynamics that the following equation is valid for the components of an ecosystem:

$$Ex = \sum_{i=1}^{i=n} RT (C_i * \ln (C_i/C_{eq,i}) - (C_i - C_{eq,i})) \quad (3)$$

where R is the gas constant, T the temperature of the environment (Kelvin), while C_i represents the i 'th component expressed in a suitable unit, e.g., for phytoplankton in a lake C_i could be milligrams of a focal nutrient in the phytoplankton per liter of lake water, $C_{eq,i}$ is the concentration of the i 'th component at thermodynamic equilibrium, which can be found in Morowitz (1968) and n is the number of components. $C_{eq,i}$ is, of course, a very small concentration of organic components, corresponding to the probability of forming a complex organic compound in an inorganic soup (at thermodynamic equilibrium). Morowitz (1968) has calculated this probability and found that for proteins, carbohydrates and fats the concentration is about 10^{-86} $\mu\text{g/l}$, which may be used as the concentration at thermodynamic equilibrium.

The idea of the new generation of models presented here is to find continuously a new set of parameters (limited for practical reasons to the most crucial, i.e., sensitive parameters) that is better fitted for the prevailing conditions of the ecosystem. "Fitted" is defined in the Darwinian sense by the ability of the species to survive and grow, which may be measured by the use of exergy (see Jørgensen, 1982a, 1986, 1988a and 1990a, Jørgensen and Mejer, 1977 and 1979 and Mejer and Jørgensen, 1979). Figure 14.2 shows the proposed modeling procedure, which has been applied in the cases presented below.

Exergy has previously been tested as a "goal function" for ecosystem development; see for instance Jørgensen (1986), Jørgensen and Mejer (1979) and Herendeen (1989). However in all these cases the model applied did not include the "elasticity" of the system, obtained by use of variable parameters, and therefore the models did not reflect real ecosystem properties. A realistic test of the exergy principle would require the application of *variable parameters*.

Exergy is defined as the work the system can perform when it is brought into equilibrium with the environment or another well-defined reference state. If we presume a reference environment for a system at thermodynamic equilibrium, meaning that all the components are: (1) inorganic, (2) at the highest possible oxidation state signifying that all free energy has been utilized to do work, and (3) homogeneously distributed in the system, meaning no gradients, then exergy

becomes an expression for the biomass (physical structure) and the information (embodied in the complex biochemical composition of the cells, determined by the genes. Temperature and pressure differences between systems and their reference environments are anyhow small contribution to the overall exergy and for present purposes can be ignored.

The idea is therefore to compute an *exergy index* based entirely on chemical energy: $\sum_i (\mu_c - \mu_{c,O}) N_i$, where i is the number of exergy-contributing compounds, c , and μ_c is the chemical potential relative to that at a reference inorganic state, $\mu_{c,O}$. Our (chemical) exergy index for a system will be taken with reference to the same system at the same temperature and pressure, but in the form of an inorganic soup without life, biological structure, information, or organic molecules.

As $(\mu_c - \mu_{c,O})$ can be found from the definition of the chemical potential, replacing activities by concentrations we obtain the following expression for *chemical exergy*:

$$Ex = RT \sum_{i=0}^n c_i \ln c_i / c_{i,eq} \quad [ML^2T^{-2}] \quad (4)$$

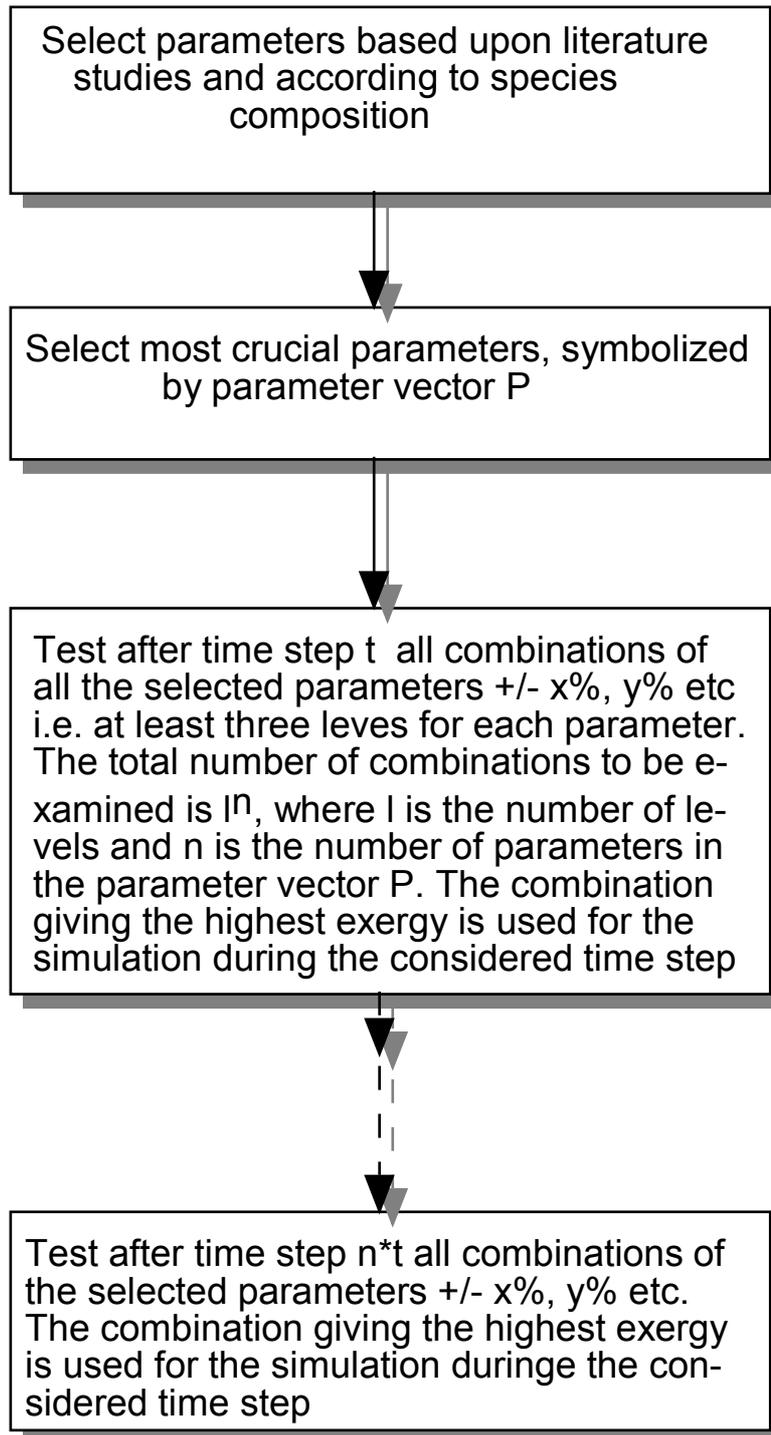


Figure 14.2 The procedure used for the development of structurally dynamic models.

R is the gas constant, T is the temperature of the environment and system (Figure 14.4), c_i is the concentration of the i'th component expressed in suitable units, $c_{i,eq}$ is the concentration of the i'th component at thermodynamic equilibrium, and n is the number of components. The quantity $c_{i,eq}$ represents a very small, but nonzero, concentration (except for $i = 0$, which is considered to cover the inorganic compounds), corresponding to a very low probability of forming complex organic compounds spontaneously in an inorganic soup at thermodynamic equilibrium. It can be shown (Jørgensen, 1997), that an ecologically useful exergy index can be computed based on concentrations of chemical components, c_i , multiplied by *weighting factors*, β_i , reflecting the exergy contents of the various components due to their chemical energy and the information embodied in DNA:

$$Ex = \sum_{i=0}^n \beta_i c_i \quad (5)$$

Values for β_i based on detritus exergy equivalents are available for a number of different species and taxonomic groups. The unit, detritus exergy equivalents expressed in g/l, can be converted to kJ/l by multiplication by 18.7, which corresponds approximately to the average energy content of 1 g detritus (Morowitz, 1968). The index $i = 0$ for constituents covers inorganic components, but in most cases these will be neglected as contributions from detritus and living biota are much higher due to extremely low concentrations of these components in the reference system. Our exergy index therefore accounts for the chemical energy in organic matter plus the information embodied in living organisms. It is measured from the extremely small probabilities of forming living components spontaneously from inorganic matter. The weighting factors, β_i , - see table 14.2 - may be considered as quality factors reflecting the extent to which different taxa contribute to overall exergy.

Based on numbers of *information genes* and the *exergy* content of organic matter in the various organisms, compared with the exergy content of detritus (about 18 kJ/g). Further detail see Jørgensen (1997).

Table 14.2 *Approximate numbers of non-repetitive genes.*

Number of Organisms	<i>information genes</i>	<i>Conversion factor*</i>
Detritus (reference)	0	1
Minimal cells	470	2.7
Bacteria	600	3.0
Algae	850	3.9
Yeast	2000	6.4
Fungi	3000	10.2
Sponges	9000	30
Molds	9500	32
Trees	10000–30000	30–87
Jellyfish	10000	30
Worms	10500	35
Insects	10000–15000	30–46
Zooplankton	10000–15000	30–46
Fishes	100000–120000	300–370
Amphibians	120000	370
Birds	120000	390
Reptiles	130000	400
Mammals	140000	430
Humans	250000	740

APPLICATIONS

15 Use of models in the PAEQANN project

15.1 Classical statistical models in PAEQANN

15.1.1 GLM, GAM and CART

The complexity of the ecological systems often results in complex relations between the variables, justifying the uses of multiple modelling techniques. PAEQANN, amongst others, aims to apply three statistical techniques: GLM (Generalised Linear Models), GAM (Generalised Additive Models) and CART (Classification And Regression Tree) methods. The three models (GLM, GAM and CART) will be used to predict and to explain the community structures. The predictive powers and the roles of explanatory variables will be studied.

15.1.2 Bayesian Models

The Bayesian models will be developed whereby three types of input data will be used: a species matrix, an environmental variables matrix and a so called z-matrix. The latter is composed of autecological knowledge of the experts. This knowledge is added to learn the model.

15.2 Artificial Neural Networks in PAEQANN

15.2.1 MultiLayer Perceptrons

Most of the neural networks that will be used in the PAEQANN project to develop predictive models will be MultiLayer Perceptrons (MLPs). These ANNs are by far the most commonly used, not only in ecological applications but also in many different fields. They are usually trained by means of the error back-propagation training algorithm, that at first was presented by Webos (1974), then independently developed by Parker (1982), and finally presented to a wider readership by Rumelhart *et al.* (1986). When properly trained, these ANNs are able to approximate virtually every multivariable function, provided that enough data are available for their training and that the ANN structure is adequate (Hornik *et al.*, 1989).

Even though more efficient training algorithms are available, the theoretical benefits of those algorithms are usually not relevant as far as ecological applications are concerned. In fact, the accuracy of the results that can be obtained in ecological applications of ANNs is usually limited by the amount of the available data and by

their quality: ecological variables are usually expensive, time-consuming and difficult to measure and they are often affected by large sampling and measurement errors. Therefore, the selection of the training algorithm or the fine tuning of the training parameters are usually less relevant in comparison to other kind of applications, where data are more abundant and reliable.

15.2.2 Sensitivity analysis

On the other hand, ecologists want to understand the functioning of the models as well as the role the predictive variables play. ANNs are usually regarded as “black box” models, unable to provide information about the relative importance of the various predictive variables on the dependent variables. Even if the intrinsic predictive power of a variable (i.e. the role it plays in the underlying ecological processes) is a very important feature that should be taken into account in the design phase of a model, it is also necessary to evaluate the way predictive variables affect the behaviour of the models. Therefore, several authors focused on the analysis of the sensitivity of ANN with respect to the in- and output. Many of these authors have used sensitivity analysis methods by determining those input variables;

- that contribute to the predictive power (El-Keib & Ma, 1995; Engelbrecht *et al.*, 1995; Guo & Uhrig, 1992; Hsu *et al.*, 1995; Kim *et al.*, 2000; Liong *et al.*, 2000a; Liong *et al.*, 2000b; Maier & Dandy, 1998; van Wijk & Bouten, 1999; Yao *et al.*, 1998; Zurada *et al.*, 1994),
- that contribute to the underlying causal relationships (Balls *et al.*, 1996; Cai *et al.*, 1999; Chow & Yee, 1991; Dimopoulos *et al.*, 1999; Dimopoulos *et al.*, 1995; Garson, 1991; Goh, 1995; Hammitt & Bartlett, 1995; Lek *et al.*, 1996a,b; Ozesmi & Ozesmi, 1999; Scardi & Harding, 1999; Seginer, 1997; Sung, 1998; Viktor *et al.*, 1995; Yao *et al.*, 1998),

Different methods are available, e.g.;

- The stepwise method; a frequently used method which adds or rejects an input variable and in each step evaluates the output results (Maier *et al.*, 1998; Sung, 1998) by ranking the most significant input variables according to their effect on the output.
- Methods with which the connection weights of the input variables are partitioned (Garson, 1991; Goh, 1995).
- The method of varying one input variable while keeping the others constant (Lek *et al.*, 1996a; Lek *et al.*, 1996b; Ozesmi & Ozesmi, 1999). With this method response profiles are obtained that summarise the effect of each predictive variable (i.e. of each input variable) on the output.
- Methods based on random perturbation of each input variable and analysing the effect on the output (Scardi & Harding, 1999).
- Methods based on partial derivatives computation (Dimopoulos *et al.*, 1995,1999; Engelbrecht *et al.*, 1995; Zurada *et al.*, 1994), which classifies the input variables and indicates how each input variable affects the output.

In PAEQANN project some of the sensitivity analysis methods will be applied to study the contribution of the available environmental parameters on the different studied species.

15.2.3 Self-Organizing Maps

In ecological applications, like in the PAEQANN project, SOMs enable both the visualisation of the sample units and the visualisation of species abundances. Thus, they provide a visual way to detect structures in ecological communities and can be recommended for exploratory approaches in which unexpected structures might be found. The SOM algorithm seems fully usable in ecology; it can perfectly complement classical techniques for exploring data and for achieving community ordination. A promising approach may be to combine SOM with other neural techniques. For instance:

- learning vector quantification is a supervised extension of the Kohonen algorithm and can be used for classification problems or to add some data after the training of the SOM (Kohonen, 2000);
- supervised learning combined with SOM can be applied to a Radial Basis Function network (Shwenker *et al.*, 2000);
- fuzzy logic may provide a new view over the ability of the SOM (Baraldi & Parmiggiani, 1997).

15.2.4 Artificial Neural Networks Backpropagation

ANNbp will be used principally to forecast some questions in the PAEQANN project. Actually, in our project we try to predict the community structure of European freshwater using environmental features. Firstly, the prediction of species richness or of groups would be obtained. The association of different method then would be used to reach our goals.

15.2.5 Counterpropagation

In ecological modelling, the CPN can be applied from two points of view based on its structure. First, the possibility of full counterpropagation structure can be considered. In this case the network has a property of counterflow of information in both directions. Thus, both input and output variables can be predicted. Based on this idea biological attributes can be predicted with environmental variables, and at the same time also environmental variables can be predicted with biological attributes.

Another possibility is use of a simplified forward-only network. In this network the characteristics of the network which combines unsupervised and supervised learning algorithms can be used. Thus, the network can pattern sampling sites, interpret relationships among environmental variables, biological attributes and sampling sites as well as relationships among input variables, and can predict biological attribute for assessment of target ecosystem in a network. So, the CPN can be used as a tool for assessing ecological status and predicting water quality of target ecosystems.

15.2.6 Future developments of ANNs

The most promising developments in ANN applications to ecological modelling will be certainly related to hybridisation with other modelling paradigms (Recknagel & Lek, in press; See & Openshaw, 2000). As some preliminary results pointed out, fuzzy neural networks (Kosko, 1992) as well as cellular neural networks (Wolfram, 1984) and evolutionary neural networks (Yao & Liu, 1997) will provide new opportunities to push the limit of the explanatory capacity of ANNs for ecological applications.

Moreover, ANNs can also be used to merge different models (and data sets) into a single “metamodel” (Scardi *et al.*, in press) or, from an opposite perspective, as a basis for rule extraction algorithms (e.g. Andrews *et al.*, 1995).

In most papers dealing with ANN applications, sufficient details about ANN theory are usually provided, as well as information about the case study considered and the results obtained, but the modelling process (e.g. preliminary data analysis and pre-processing) is generally described poorly (Maier & Dandy, 2000). Moreover, in order to achieve significant advances in ANN applications, ANNs should be regarded as alternatives to more traditional approaches rather than as “...a remedy for all existing computational failings” (Flood & Kartam, 1994).

15.3 Structurally dynamic models in PAEQANN

This model type could probably work together with ANN as an overlying rule, that could make ANNs more causal and maybe also increase the accuracy of ANN based models. Structurally dynamic models are based upon a description of the processes that are important for the model objectives. The change in structure is based on a current test of the best survivors, i.e., the species that have the properties that give most growth for the entire ecosystem under the prevailing conditions. This type of model is therefore based to a high extent on causality opposite ANN, which in its basic form is a black box model.

The use of exergy calculations to vary continuously the parameters has only been used successfully in eleven cases of ecological modelling: 1-2) two shallow lakes in Denmark, 3) The Lagoon of Venice, Italy, 4) Roskilde Fjord (Denmark), 5) Mondego Estuary (Portugal), 6-7) two population dynamic models, 8) Lake Annone, Italy, 9) Lake Balaton, Hungary, 10) to explain the success and failure of biomanipulation, 11) to explain the selection of Darwin Finches. In all the cases a model was calibrated in accordance with the normal procedure and the validation was carried out successfully by the application of the structurally dynamic approach, including a validation of the observed structural changes.

Structurally dynamic changes take however also place during the calibration period, for instance succession in lakes of phytoplankton and zooplankton species. It has therefore been decided to test in the PAEQANN project the structurally dynamic approach in the calibration of models of lakes. Two case studies have been tested, Lake Glumsø in Denmark and Lake Mogan in Turkey, where good data were available. The two cases were successfully terminated in December 01 and two papers have been submitted to Ecological Modelling. The two cases show clearly

that the calibration is improved considerably by use of the structurally dynamic approach. The idea is to transfer this calibration method to the use of ANN that will take place during the spring 02.

16 References

- Adams S.M., Jaworska J.S. & Ham K.D., 1996. Influence of ecological factors on the relationship between MFO induction and fish growth: bridging the gap using neural networks. *Mar. Environ. Res.*, 42(1-4): p. 197-201.
- Adkison M.D. & Peterman R.M., 1996. Results of Bayesian methods depend on details of implementation: An example of estimating salmon escapement goals. *Fish. Res.*, 25(2), 155-170.
- Albiol J., Campmajo C., Casas C. & Poch M., 1995. Biomass estimation in plant cell cultures: a neural network approach. *Biotechn. Progr.*, 11, 88–92.
- Andrews R., Diederich J. & Tickle A.B., 1995. A survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowl. Based Syst.*, 8, 373–389.
- Aoki I., Komatsu T. & Hwang, K., 1999. Prediction of response of zooplankton biomass to climatic and oceanic changes. *Ecol. Model.*, 120 (2-3): 261-270.
- Aoki I. & Komatsu T., 1997. Analysis and prediction of the fluctuation of the sardine abundance using a neural network. *Oceanol. Acta*, 20(1): 81-88.
- Aussem A. & Hill D., 1999. Wedding connectionist and algorithmic modelling towards forecasting *Caulerpa taxifolia* development in the north-western Mediterranean sea. *Ecol. Model.*, 120 (2-3): 225-236.
- Balls G.R., Palmer Brown D. & Sanders G.E., 1996. Investigating microclimatic influences on ozone injury in clover (*Trifolium subterraneum*) using artificial neural networks. *New Phytol.*, 132, 271-280.
- Baraldi A. & Parmiggiani F., 1997. Fuzzy combination of Kohonen's and ART neural network models to detect statistical regularities in a random sequence of multi-valued input patterns. In: Proceedings of ICNN'97, International Conference on Neural Networks, volume I, pages 281--286. IEEE Service Center, Piscataway, NJ.
- Baran P., Lek S., Delacoste M. & Belaud A., 1996. Stochastic models that predict trout population density or biomass on a mesohabitat scale. *Hydrobiologia*, 337(1-3): p. 1-9.
- Barciela R. M., Garcia E. & Fernandez E., 1999. Modelling primary production in a coastal embayment affected upwelling using dynamic ecosystem models and artificial neural networks. *Ecol. Model.*, 120, 199-211.
- Boddy L., Morris C. W., Wilkins M. F., Luan Al-Haddad G. A., Tarran R. R., Jonker P. & Burkill H., 2000. Identification of 72 phytoplankton species by radial basis function neural network analysis of flow cytometric data. *Mar. Ecol. Prog. Ser.*, 195: 47-59.
- Billheimer D., Cardoso T., Freeman E., Guttorp P., Ko H.W. & Silkey M., 1997. Natural viability of benthic species composition in the Delaware Bay. *Environ. Ecol. Stat.*, V4, 95-115.
- Breiman L., Friedman J., Olshen R. & Stone C.J. (1984) Classification and regression trees. Chapman & Hall, New York.
- Brey T. & Gerdes D., 1998. High Antarctic macrobenthic community production. *J. Exp. Mar. Biol. Ecol.*, 231: 191–200.

- Brey T., Jarre-Teichmann A. & Borlich O., 1996. Artificial neural network versus multiple linear regression: predicting P/B ratios from empirical data. *Mar. Ecol. Prog. Ser.*, 140: 251-256.
- Brosse S., Guegan J., Tourenq J. & Lek S., 1999. The use of artificial neural networks to assess fish abundance and spatial occupancy in the littoral zone of a mesotrophic lake. *Ecol. Model.*, 120 (2-3): 299-311.
- Cai J., Cottis R. A. & Lyon S. B., 1999. Phenomenological modelling of atmospheric corrosion using an artificial neural network. *Corros. Sci.*, 41, 1001-1030.
- Carpenter G. A., Gopal S., Macomber S., Martens S., Woodcock C.E. & Franklin J. 1999. A neural network method for efficient vegetation mapping. *Remote Sens. Environ.*, 70: 326-338.
- Carpenter S.R., Kitchell J.F., Cottingham K.L., Schindler D.E., Christensen D.L., Post D.M. & Voichick N., 1996. Chlorophyll variability, nutrient input, and grazing: Evidence from whole-lake experiments. *Ecology*, 77(3), 725-735.
- Chambers, J. & Hastie, T. (1991) *Statistical Models in S*, Wadsworth/Brooks Cole, Pacific Grove.
- Chen Y. & Fournier D., 1999. Impacts of atypical data on Bayesian inference and robust Bayesian approach in fisheries. *Can. J. Fish. Aquat. Sci.*, 59(9), 1525-1533.
- Chen Y., Breen P.A. & Andrew N.L., 2000. Impacts of outliers and mis-specification of priors on Bayesian fisheries-stock assessment. *Can. J. Fish. Aquat. Sci.*, 57(11), 2293-2305.
- Cho S.B. 1997. Self-organizing map with dynamical node-splitting: Application to handwritten digit recognition. *Neural Comput.*, 9, 1345-1355.
- Chon T.S., Kwak I.S., Park Y.-S., Kim T.-H. & Kim Y.-S., in press. Patterning and the short-term prediction of community dynamics of benthic macroinvertebrates in urbanized streams with implementation of biological and environmental factors in a recurrent artificial neural network. *Ecol. Model.*
- Chon T.S., Park Y.S., Moon K.H. & Cha E.Y., 1996. Patterning communities by using an artificial neural network. *Ecol. Model.*, 90: 69-78.
- Chow M. Y. & Yee S. O., 1991. Robustness test of an incipient fault detector Artificial Neural Network. *IEEE*, 73-78.
- Civco D.L. 1993. Artificial neural networks for land cover classification and mapping. *International Journal of Geographic Information Systems*, 7(2):173-186.
- Colasanti R.L., 1991. Discussions of the possible use of neural network algorithms. *Ecol. Model.* Binary 3, 13-15.
- Cottingham K.L. & Schindler D.E., 2000. Effects of grazer community structure on phytoplankton response to nutrient pulses. *Ecology*, 81(1), 183-200.
- Cow-Rogers S., 1997. Inseason forecasting of skeena River sockeye run size using Bayesian probability theory. *Canadian Report of Fisheries and Aquatic Sciences*, 0(2412), 1-43.
- Dimopoulos I., Chronopoulos J., Chronopoulou Sereli A. & Lek S., 1999. Neural network models to study relationships between lead concentration in grasses and permanent urban descriptors in Athens city (Greece). *Ecol. Model.*, 120 (2-3): 157-165.

- Dimopoulos Y., Bourret P. & Lek S., 1995. Use of some sensitivity criteria for choosing networks with good generalization ability. *Neural Process. Lett.*, 2: 1-4.
- Dobson A.J., 1983. An introduction to statistical modelling. Chapman & Hall.
- Efron B., 1983. "Estimating the error rate of a prediction rule: improvement on cross-validation." *Journal American Statistical Association* 78: 316-330.
- Efron B. and R. J. Tibshirani (1995). "Cross-validation and the bootstrap: estimating the error rate of the prediction rule." Rep. Tech. Univ. Toronto.
- El-Keib A. A. & Ma X., 1995. Application of Artificial Neural Networks in Voltage Stability Assessment. *IEEE Transactions on Power Systems*, 10: 1890-1896.
- Engelbrecht A. P., Cloete I. & Zurada J. M., 1995. Determining the significance of input parameters using sensitivity analysis, From natural to artificial neural computation. Springer-Verlag, Malaga-Torremolinos, Spain.
- Firth D., 1991. Generalized linear models. In Hinkley *et al.* Statistical Theory and Modelling. London, Chapman & Hall.
- Flood I. & Kartam N., 1994. Neural networks in civil engineering. I: Principles and understanding. *J. Comput. Civil Eng.*, 8 (2): 131-148.
- Foody G. M., 1999. Applications of the self-organizing feature map neural-network in community data-analysis. *Ecol. Model.*, 120 (2-3): 97-107.
- French M. & Recknagel F., 1994. Modeling algal blooms in freshwaters using artificial neural networks, in Computer Techniques in Environmental Studies. Vol. 2. Environmental Systems, P.e. Zannetti, Editor. Computational Mechanics Publications: Southampton, Boston. p. 87-94.
- Friedman J. H. 1997. "On bias, variance, o/1-loss and the curse-of-dimensionality." *Data Mining and knowledge discovery* 1: 55-77.
- Garson G.D., 1991. Interpreting neural network connection weights. *Artificial Intelligence Expert*, 6, 47-51.
- Geladi P. & Kowalski B.R., 1986. Partial Least Squares Regression: A tutorial, *Anal. Chim. Acta* 185: 1-17
- Gelman, A., Carlin, J.B., Stern, H.S. and Rubin, D.B.(1995). *Bayesian Data Analysis*. London: Chapman and Hall.
- Giraudel J.L., Aurelle D. & Lek S., 2000. Application of the self-organizing mapping and fuzzy clustering microsatellite data: how to detect genetic structure in brown trout (*Salmo trutta*) populations. In Lek, S. & Gueguan, J.F., editors, {Artificial Neuronal Networks: application to ecology and evolution}, *Environmental Science*, pages 187-201, Springer-Verlag.
- Giske J., Huse G. & Fiksen O., 1998. Modelling spatial dynamics of fish. *Rev. Fish. Biol. Fish.* 8, 57-91.
- Goh A. T. C., 1995. Back-propagation neural networks for modeling complex systems. *Art. Intell. Eng.*, 9: 143-151.
- Gross L., Thiria S. & Frouin R., 1999. Applying artificial neural network methodology to ocean color remote sensing. *Ecol. Model.*, 120 (2-3): 237-246.
- Grossberg, S. 1982. *Studies of Mind and Brain: Neural Principals of Learning, Perception, Development, Cognition, and Motor Control*. Reidel Press, Boston.
- Guegan J.F., Lek S. & Oberdorff T., 1998. Energy availability and habitat heterogeneity predict global riverine fish diversity. *Nature*, 391: 382-384.

- Gunn S., 1998. Support vector machines for classification and regression. ISIS Technical Report, Image-Speech & Intelligent Systems Group, University of Southampton.
- Guo Z. & Uhrig R.E., 1992. Using modular neural networks to monitor assident conditions in nuclear. SPIE, *Applications of Artificial Neural Networks*, III: 1709, 505-516.
- Hammitt A.M. & Bartlett E.B., 1995. Determining Functional Relationships from trained Neural Networks. *Math. Comput. Model.*, 22: 83-103.
- Hammond T., 1997. A Bayesian interpretation of target strength data from the Grand Banks. *Can. J. Fish. Aquat. Sci.*, 54(10), 2323-2333.
- Hagan, M.T., H.B. Demuth, & M.Beale, 1996. Neural Network Design. PWS Publishing Company, Boston.
- Hastie, T. & Tibshirani, R., 1990. Generalized Additive Models, Chapman and Hall.
- Haykin, S., 1999. Neural Networks: A Comprehensive Foundation (2nd edition). Prentice Hall, New Jersey
- Hecht-Nielsen, R. 1987. Counter propagation networks. Proc. of the Int. Conf. On Neural Networks, II,19-32, IEEE Press, New York, June 1987.
- Heiland I., 1990. Partial Least Squares Regression and Statistical Models, Scandinavian Journal of Statistics, 17: 97-114
- Helu S.L., Sampson D.B. & Yin Y., 2000. Application of statistical model selection criterion to the Stock Synthesis assessment program. *Can. J. Fish. Aquat. Sci.*, 57(9): 1784-1793.
- Hilbert D.W. & Ostendorf B., in press. The utility of artificial neural networks for modelling the distribution of vegetation in past, present and future climates. *Ecol. Model.*
- Hilborn R. & Liermann M., 1998. Standing on the shoulders of giants: Learning from experience in fisheries. *Rev. Fish boil. Fisher.*, 8(3), 273-283.
- Hojtink, H. & Molenaar I.W., 1997. A multidimensional item response model: Constrained latent class analysis using the Gibbs sampler and posterior predictive checks. *Psychometrika*, 62: 171-190.
- Hojtink, H., 1998. Constrained latent class analysis using the Gibbs sampler and posterior predictive p-values: Applications to educational testing. *Statistica Sinica*, 8: 691-712.
- Hojtink, H., 2001. Confirmatory latent class analysis: Model selection using Bayes factors and (pseudo) likelihood ratio statistics. *Multivariate Behavioral Research*.
- Hornik K & Stinchcombe M, White H, 1989 Multilayer feedforward networks are universal approximators. *Neural Networks* 2: 359-366
- Höskuldsson A., 1988. PLS Regression Methods. *J. Chemometrics*, 2: 211-228
- Hosmer, D.W. and Lemeshow, S. 1989. Applied logistic regression. Wiley. New York.
- Hsu C. T., Tzeng Y. M., Chen C. S. & Cho M. Y., 1995. Distribution feeder loss analysis by using an artificial neural network. *Electr. Pow. Syst. Res.* 34, 85-90.
- Jeong K.-S., Joo G.-J., Kim H.-W., Ha K. & Recknagel F., in press. Predicting dynamics of phytoplankton in the flow regulated Nakdong River (South Korea) by means of recurrent artificial neural networks. *Ecol. Model.*

- Jon T., Cass A. & Richards L.J., 2000. A Bayesian decision analysis to set escapement goals for Fraser River sockeye salmon (*Oncorhynchus nerka*). *Can. J. Fish. Aquat. Sci.*, 57(5), 962-979.
- Jongman, R.H.G., ter Braak, C.J.F. & van Tongeren, O.F.R. (1987). Data analysis in community and landscape ecology. Pudoc Wageningen 1987.
- Jørgensen S.E. & Padisak J., 1996. Does the intermediate disturbance hypothesis comply with thermodynamics? *Hydrobiologia* 323, 9-21.
- Jørgensen S.E., 1997. Integration of Ecosystem Theories: A Pattern. Kluwer Academic Publ. Dordrecht, 388 pp.
- Jørgensen S.E. & de Bernardi D., 1998. The Use of structural dynamic models to explain successes and failures of biomanipulation. *Hydrobiologia* 359, 1-12 .
- Jørgensen S.E. & Fath B., 2001. Growth and Development. Submitted to *American Naturalist*.
- Kass, R.E., and Raftery, A.E. (1995). Bayes factors. *Journal of the American Statistical Association*, 90, 773-795.
- Kastens T.L. & Featherstone A.M., 1996. Feedforward back-propagation neural networks in prediction of farmer risk preference. *Am. J. Agri. Econ.* 78, 400–415.
- Keiner L.E. & Yan X.-H., 1998. A neural network model for estimating sea surface chlorophyll and sediments from Thematic Mapper imagery. *Remote Sens. Environ.*, 66: 153–165.
- Kim S.H., Yoon C. & Kim B.J., 2000. Structural Monitoring System Based on Sensitivity Analysis and a Neural Network. *Computer-aided Civil and Infrastructure Engineering*, 155: 309-318.
- Kimes D.S., Holben B.N., Nickeson J.E., & McKee W.A., 1996. Extracting forest age in a Pacific Northwest forest from Thematic Mapped and topographic data. *Remote Sens. Environ.*, 56: 133-140.
- Kinas P.G., 1996. Bayesian fishery stock assessment and decision making using adaptive importance sampling. *Can. J. Fish. Aquat. Sci.*, 53(2): 414-423.
- Kohavi, R. (1995). A study of cross-validation and bootstrap for estimation and model selection, Morgan Kaufmann Publishers.
- Kohonen, T., 1982. Self-organized formation of topologically correct feature maps. *Biol. Cybern.*, 43: 59-69.
- Kohonen T., 1982. Analysis of a simple self-organizing process. *Biol. Cyb.*, 44 (2): 135-140.
- Kohonen, T., 1984. Self-Organizing Maps, volume 30 of Springer Series in Information Science, Springer, Berlin.
- Kohonen T., 1995. Self-Organizing Maps, volume 30 of Springer Series in Information Sciences. Springer, Berlin, Heidelberg. (Second Extended Edition 1997).
- Kohonen T., 2000. An unsupervised learning method that produces organized representation from real information. In Malmgren, H., Boga, M., & Niklasson, L., editors, Artificial Neural Networks in Medicine and Biology, Proceedings of the ANNIMAB-1 Conference, Göteborg, Sweden, 13--16 May 2000, pages 45-53.
- Komatsu T., Aoki I., Mitani I. & Ishii T., 1994, Prediction of the catch of Japanese sardine larvae in Sagami Bay using a neural network. *Fish. Sci.*, 60: 385-391.

- Kosko B., 1992. *Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence*. Prentice-Hall, Englewood Cliffs.
- Kuikka S., Hilden M., Gislason H., Hansson S., Sparholt H. & Varis O., 1999. Modeling environmentally driven uncertainties in Baltic cod (*Gadus morhua*) management by Bayesian influence diagrams. *Can. J. Fish. Aquat. Sci.*, 59(4): 629-641.
- Lae R., Lek S. & Moreau J., 1999. Predicting fish yield of African lakes using neural networks. *Ecol. Model.* 120, 325-335.
- Lamon E.C. & Clyde M.A., 2000: Accounting for model uncertainty in prediction of chlorophyll a in Lake Okeechobee. *Journal of Agricultural Biological and Environmental Statistics*, 5(3): 297-322.
- Lee D., Shepard B., Sanborn B. & Ulmer L., 1996. Assessing extinction risks for westslope cutthroat trout in the upper Missouri River Basin using BayVAM model. *Bulletin of the Ecological Society of America*, 77: 258.
- Lee D.A. & Rieman B.E., 1997. Population viability assessment of salmonids by using probabilistic networks. *North American Journal of Fisheries Management*, 17(4): 1144-1157.
- Lek S. & Guegan J-F [eds.], 2000. *Artificial Neuronal Networks: Application to Ecology and Evolution*, Springer-Verlag, Berlin/Heidelberg.
- Lek S., Belaud A., Dimopoulos I., Lauga J. & Moreau J., 1995. Improved estimation, using neural networks, of the food consumption of fish populations. *Mar. Freshwater Res.*, 46: p. 1229-1236.
- Lek S., Belaud A., Baran P., Dimopoulos I. & Delacoste M., 1996a. Role of some environmental variables in trout abundance models using neural networks. *Aquat. Living Resour.*, 9: 23-29.
- Lek S., Delacoste M., Baran P., Dimopoulos I., Lauga J. & Aulagnier S., 1996b. Application of neural networks to modelling nonlinear relationships in ecology. *Ecol. Model.*, 90: 39-52.
- Lek S., Dimopoulos I. & Fabre A., 1996c. Predicting phosphorus concentration and phosphorus load from watershed characteristics using backpropagation neural networks. *Acta Oecologica*, 17(1): 43-53.
- Liermann M. & Hilborn R., 1997. Depensation in fish stocks: A hierarchic Bayesian meta-analysis. *Can. J. Fish. Aquat. Sci.*, 54(9): 1976-1984.
- Lin, C.-T. and C.S.G. Lee. 1996. *Neural Fuzzy Systems*. Prentice Hall PTR, Upper Saddle River, New Jersey, 797 pp.
- Liong S. Y., Lim W. H., Kojiri T. & Hori T., 2000a. Advance flood forecasting for flood stricken Bangladesh with a fuzzy reasoning method. *Hydrolog. Process.*, 14: 431-448.
- Liong S. Y., Lim W. H. & Paudyal G. N., 2000b. River Stage Forecasting in Bangladesh: Neural Network Approach. *J. Comput. Civil Eng.*, 14: 1-8.
- Maier H. R. & Dandy G. C., 1998. The effect of internal parameters and geometry on the performance of back-propagation neural networks: an empirical study. *Environ. Modell. Softw.*, 13: 193-209.
- Maier H. R. & Dandy G. C., 2000. Neural networks for the prediction and forecasting of water resources variables: A review of modelling issues and applications. *Environ. Modell. Softw.*, 15: 101-124.

- Mann S. & Benwell G.L., 1996. The integration of ecological, neural and spatial modelling for monitoring and prediction for semi-arid landscapes. *Comput. Geosci.*, 22(9): 1003-1012.
- Mastrorillo, S., F. Dauba, *et al.* (1998). "Predicting local fish species richness in the Garonne River basin." *Comptes Rendus de l'Académie des Sciences Série III Sciences de la Vie* 321(5): 423-428.
- McCullagh P. & Nelder J.A. (1989) Generalized linear models. Second edition. London, Chapman & Hall.
- Nelder J.A. & Wedderburn R.W.M. (1972) Generalized additive models. *Journal of Royal Statistical Society, A*, 135: 370-384.
- Mastrorillo S., Dauba F., Oberdorff T., Guegan J.F. & Lek S., 1998. Predicting local fish species richness in the Garonne River basin. *Comptes Rendus de l'Académie des Sciences, Serie III Sciences de la Vie*, 321: 423-428.
- Mastrorillo S., Lek S. & Dauba F., 1997. Predicting the abundance of minnow *Phoxinus phoxinus* (Cyprinidae) in the river Ariege (France) using artificial neural networks. *Aquat. Living Resour.*, 10(3): 169-176.
- Mau B., Newton M.A., Larget B., 1999. Bayesian phylogenetic inference via Markov chain Monte Carlo methods. *Biometrics*, 55(1), 1-12.
- McAllister M.K. & Ianelli J.N., 1997. Bayesian stock assessment using catch-age data and the sampling importance resample algorithm. *Can. J. Fish. Aquat. Sci.*, 54(2): 284-300.
- McAllister M.K. & Kirkwood G.P., 1998. Bayesian stock assessment: A review and example application using the logistic model. *ICES J. Mar. Sci.*, 55(6): 1031-1060.
- McAllister M.K. & Kirkwood G.P., 1998. Using Bayesian decision analysis to help achieve a precautionary approach for developing fisheries. *Can. J. Fish. Aquat. Sci.*, 55(12): 2642-2661.
- McAllister M.K. & Pikitch E.K., 1997. A Bayesian approach to choosing a design for surveying fishery resources: Application to the eastern Bering Sea trawl survey. *Can. J. Fish. Aquat. Sci.*, 54(2): 301-311.
- Meyer R. & Millar R., 1999. Bayesian stock assessment using a state-space implementation of the delay difference model. *Can. J. Fish. Aquat. Sci.*, 59(1): 37-52.
- Myers R.A., Bowen K.G. & Barrowman N.J., 1999. Maximum reproductive rate of fish at low population sizes. *Can. J. Fish. Aquat. Sci.*, 56(12): 2404-2419.
- Nakano, H., *et al.*, 1991. Identification of plankton using a neural network with a function of unknown species detection. *Rep. Measurement Res.*, Group, IM-91-30: 47-56.
- Newman K., 1997. Bayesian averaging of generalized linear models for passive integrated transponder tag recoveries from salmonids in the Snake River. *North American Journal of Fisheries Management*, 17(2): 362-377.
- Ogle D.H., Pruitt R.C., Spangler G.R. & Cyterski M.J., 1996. A Bayesian approach to assigning probabilities to fish ages determined from temporal signatures in growth increments. *Can. J. Fish. Aquat. Sci.*, 53(8): 1788-1794.
- Ozesmi S. L. & Ozesmi U., 1999. An artificial neural network approach to spatial habitat modelling with interspecific interaction. *Ecol. Model.*, 116: 15-31.

- Parker D.B., 1982. Learning logic. Invention report S81-64, File 1, Office of Technology Licensing, Stanford University.
- Patterson K.R., 1999. Evaluating uncertainty in harvest control law catches using Bayesian Markov chain Monte Carlo virtual population analysis with adaptive rejection sampling and including structural uncertainty. *Can. J. Fish. Aquat. Sci.*, 56(2): 208-221.
- Pella J.J., Masusa M.M. & Chen D.G., 1998. Forecast methods for inseason management of the southeast Alaska chinook salmon troll fishery. In, Funk f., Quinn T.J., Heifetz J., Ianelli J.N., Powers J., Schweigert J.F., Sullivan P., Zhang C.I., editors. Lowell Wakefield Fisheries Symposium Series; Fishery stock assessment models. Alaska Sea Grant College Program, Fairbanks.
- Peterman R.M., Peters C.N., Robb C.A. & Frederick S.W., 1999. Bayesian decision analysis and uncertainty in fisheries management. In Pitcher T.J., Hart P.J., Pauly D., editors. Reverting fisheries management. Kluwer Academic Publishers, Dordrecht .
- Pineda F., 1987. Generalization of backpropagation to recurrent neural networks. *Phys. Rev. Lett.*, 19 (59): 2229-2232.
- Pinelalloul B., Niyonsenga T. & Legendre P., 1995. Spatial and environmental components of fresh-water zooplankton structure. *Ecoscience*, V2: 1-19.
- Punt A.E. & Hilborn R., 1997. Fisheries stock assessment and decision analysis: the Bayesian approach. *Rev. Fish Biol. Fisher.*, 7(1): 35-63.
- Punt A.E. & Walker T.I., 1998. Stock assessment and risk analysis for the school shark (*Galeorhinus galeus*) of southern Australia. *Mar. Freshwater Res.*, 49(7): 719-731.
- Qian S.S., Lavine M. & Stow C.A., 2000. Univariate Bayesian nonparametric binary regression with application in environmental management. *Environ. Ecol. Stat.*, 7(1): 77-91.
- Recknagel F. & Lek S., in press. State of the art of ecological modelling by artificial neural networks. *Ecol. Model.*
- Recknagel F., 1997. ANNA - Artificial Neural Network model predicting blooms and succession of blue-green Algae. *Hydrobiologia*, 349: 47-57.
- Recknagel F., French M., Harkonen P. & Yabunaka K., 1997. Artificial neural network approach for modelling and prediction of algal blooms. *Ecol. Model.*, 96(1-3): 11-28.
- Robb C.A. & Peterman R., 1998. Application of Bayesian decision analysis to management of a sockeye salmon (*Oncorhynchus nerca*) fishery. *Can. J. Fish. Aquat. Sci.*, 55(1): 86-98.
- Rumelhart D.E., Hinton G.E. & Williams G.E., 1986. Learning representations by back-propagating errors. *Nature*, 323: 533-536
- Scardi M., in press. Advances in neural network modeling of phytoplankton primary production. *Ecol. Model.*
- Scardi M., Lek S., Lim P., Di Dato P. & Oberdorff T., in press. Artificial neural networks as a tool for predicting fish community composition in rivers. Proceedings of the Third World Congress of Nonlinear Analysts (WCNA-2000), Catania (Italy), July 19-26, 2000. *Journal of Nonlinear Analysis*.

- Scardi M. & Harding L. W., 1999. Developing an empirical model of phytoplankton primary production: a neural network case study. *Ecol. Model.*, 120 (2-3): 213-223.
- Scardi M., 1996. Artificial neural networks as empirical models of phytoplankton production. *Mar. Ecol. Prog. Ser.*, 139: 289-299.
- Scardi M., 2000. Neuronal network models of phytoplankton primary production. p. 115-129. In: Lek S, Guegan J-F [eds.], *Artificial Neuronal Networks: Application to Ecology and Evolution*, Springer-Verlag, Berlin/Heidelberg.
- Schleiter I.M., Borhardt D., Wagner R., Dapper T., Schmidt K., Schmidt H., & Werner H., 1999. Modelling water quality, bioindication and population dynamics in lotic ecosystems using neural networks. *Ecol. Model.*, 120 (2-3): 271-286.
- Schwenker F., Kestler Hans~A., & Palm G., 2000. Combination of supervised and unsupervised learning for radial-basis-function networks. In Bothe, H. and Rojas, R., editors, {Proceeding of the ICSC Symposia on Neural Computation (NC'2000) May 23-26, 2000 in Berlin, Germany}. University of Ulm, D-89069 Ulm, Germany, ICSC Academic Press.
- See L. & Openshaw S., 2000. A hybrid multi-model approach to river level forecasting. *Hydrol. Sci. J.*, 45 (4): 523-536
- Seginer I., 1997. Some artificial neural network applications to greenhouse environmental control. *Comput. Electron. Agr.*, 18: 167-186.
- Silveira V.F., Khator S.K. & Barcia R.M., 1996. An information management system for forecasting environmental change. *Computers ind. Engng*, 31 (1-2): 289-292.
- Simpson R., Williams R., Ellis R. & Culverhouse, P.F., 1992. Biological pattern recognition by neural networks. *Mar. Ecol. Prog. ser.*, 79 (3): 303-308.
- Simpson R., Culverhouse P.F., Williams R. & Ellis R., 1993. Classification of Dinophyceae by artificial neural networks. *Dev. Mar. Biol.*, 3: 183-190.
- Smith A.D.M. & Punt A.E., 1998. Stock assessment of gemfish (*Rexea solandri*) in eastern Australia using maximum likelihood and Bayesian methods. In, Funk f., Quinn T.J., Heifetz J., Ianelli J.N., Powers J., Schweigert J.F., Sullivan P., Zhang C.I., editors. Lowell Wakefield Fisheries Symposium Series; Fishery stock assessment models. Alaska Sea Grant College Program, Fairbanks.
- Spitz F., Lek S. & Dimopoulos I., 1996. Neural network models to predict penetration of wild boar into cultivated fields. *Journal of Biological Systems*, 4: 433-444.
- Steinberg L.J., Reckhow K.H. & Wolpert R.L., 1996. Bayesian model for fate and transport of polychlorinated biphenyl in upper Hudson River. *Journal of Environmental Engineering*, 122 (5): 341-349.
- Sung A. H., 1998. Ranking importance of input parameters of neural networks. *Expert Syst. Appl.* 15, 405-411.
- Titterton, D.M., Smith, A.F.M. & Makov, U.E., 1985. *Statistical Analysis of Finite Mixture Distributions*. New York: Wiley.
- Trigg D.J., Walley W.J. & Ormerod S.J., 2000. A prototype Bayesian belief network for the diagnosis of acidification in Welsh rivers. In Ibarra-Berastegi G., Brebbia C., Zanetti P., editors. Proceedings of the ENVIROSOFT 2000. Development and Application of Computer Techniques to Environmental Studies, Bilbao.

- Ultsch A., 1999. Data mining and knowledge discovery with emergent self-organizing feature maps for multivariate time series. In Oja, E. & Kaski, S., editors, Kohonen Maps, pages 33-45. Elsevier, Amsterdam.
- Utans, J. and J. E. Moody (1991). "Selecting neural network architectures via the prediction risk: application to corporate bond rating prediction." Proceedings of the first International Conference on Artificial Intelligence Applications on Wall Street, IEEE Computer Society Press, Los Alamitos.
- van Wijk M. T. & Bouten W., 1999. Water and carbon fluxes above European coniferous forests modelled with artificial neural networks. *Ecol. Model.*, 120: 181-197.
- Vapnik V., 1995. The nature of statistical learning theory. Springer, New York.
- Varis O. & Kuikka S., 1997. Joint use of multiple environmental assessment models by a Bayesian meta-model: The Baltic salmon case. *Ecol. Model.*, 102: 341-351.
- Varis O., 1997. Bayesian decision analysis for environmental and resource management. *Environmental Modelling & Software with Environmental data News*, 12: 177-185.
- Vasko K., Toivonen H.T.T. & Korhola A., 2000. A Bayesian multinomial Gaussian response model for organism-based environmental reconstruction. *J. Paleolimnol.*, 24(3): 243-250.
- Verdonschot P.F.M. & Goedhart P.W. (2000): RISTORI. Effecten van ingrepen in het waterbeheer op aquatische levensgemeenschappen. Cenotypenbenadering, fase 2: Verfijning van het prototype. Alterra, Wageningen. 172: 1- 109.
- Vignaux M., Vignaux G.A., Lizamore S. & Gresham D., 1998. Fine-scale mapping of fish distribution from commercial catch and effort data using maximum entropy tomography. *Can. J. Fish. Aquat. Sci.*, 55(5): 1220-1227.
- Viktor H. L., Engelbrecht A. P. & Cloete I., 1995. Reduction of Symbolic Rules from Artificial Neural Networks using Sensitivity Analysis. IEEE: 1788-1793.
- Villmann T. & Bauer H.U., 1998. Applications of the growing self-organizing map. *Neurocomputing*, 21 (1): 91-100.
- Wagner R., Dapper T. & Schmidt H.H., 2000. The influence of environmental variables on the abundance of aquatic insects: a comparison of ordination and artificial neural networks. *Hydrobiologia*, 422: 143-152.
- Walley W.J. & Dzeroski S., 1996. Biological Monitoring: A Comparison Between Bayesian, Neural and Machine Learning Methods of Water Quality Classification. In Environmental Software Systems, London, Chapman and Hall, pp. 229-240.
- Walley W.J. & Fontama V.N., 1997. Bio-Monitoring of Rivers: An AI Approach to Data Interpretation. *Toxicology & Ecotoxicology News*, 4(6): 182-184.
- Walley W.J. & Fontama V.N., 2000. New Approaches to River Quality Classification Based Upon Artificial Intelligence. In Assessing the Biological Quality of Fresh Waters Ambleside/GB. Freshwater Biological Association, pp. 263-280.
- Webos P., 1974. Beyond regression: new tools for prediction and analysis in the behavioral sciences. Thesis, Harvard University.
- Wolfram S., 1984. Cellular Automata. *Physical D*, 10, 1-35.

- Yao J., Teng N., Ph H.L. & Tan C. L., 1998. Forecasting and analysis of marketing data using neural networks. *Journal of Information Science and Engineering*, 14: 843-862.
- Yao X. & Liu Y., 1997. A new evolutionary system for evolving artificial neural networks. *IEEE T. Neural Networ.*, 8 (3): 694-713.
- Zegers, S.L. and Karim, M.R. (1991). Generalized linear models with random effects: a Gibbs sampling approach. *Journal of the American Statistical Association*, 86, 79-86.
- Zhu M.L., Fujita M. & Hashimoto N. 1994. Application of neural networks to runoff prediction. In: Stochastic and statistical methods in hydrology & environmental engineering, ed. Hipel, K.W. *et al.* Kluwer Academic Publishers, Norwell, Mass.
- Zurada J.M., 1992. Introduction to Artificial Neural Systems. West Publishing Co., St. Paul, MN, USA. 785 pp.
- Zurada J. M., Malinowski A. & Cloete I., 1994. Sensitivity Analysis for Minimization of Input Data Dimension for Feedforward Neural Network. In "ISCAS'94. IEEE International Symposium on circuits and Systems", Vol. 6, pp. 447-450, London.

