ELSEVIER

Behavioural Processes 40 (1997) 35–43

BEHAVIOURAL PROCESSES

Artificial neural networks as a classification method in the behavioural sciences

David Reby [a], Sovan Lek [b], Ioannis Dimopoulos [c], Jean Joachim [a], Jacques Lauga [c], Stéphane Aulagnier [a]

[a] *Institut de Recherche sur les Grands Mammifères, I.N.R.A., B.P. 27, 31326 Castanet-Tolosan Cedex France*
[b] *CNRS UMR 5576 CESAC, Bat 4R3, Université Paul Sabatier, 118 route de Narbonne, 31062 Toulouse Cedex France*
[c] *CNRS UMR 5552 LET, Bat 4R3, Université Paul Sabatier, 118 route de Narbonne, 31062 Toulouse Cedex France*

Received 8 January 1996; received in revised form 8 August 1996; accepted 6 November 1996

Abstract

The classification and recognition of individual characteristics and behaviours constitute a preliminary step and is an important objective in the behavioural sciences. Current statistical methods do not always give satisfactory results. To improve performance in this area, we present a methodology based on one of the principles of artificial neural networks: the backpropagation gradient. After summarizing the theoretical construction of the model, we describe how to parameterize a neural network using the example of the individual recognition of vocalizations of four fallow deer (*Dama dama*). With 100% recognition and 90% prediction success, the results are very promising. © 1997 Elsevier Science B.V.

Keywords: Mammal; Deer; Vocalization; Neural network; Classification; Modelling

1. Introduction

In behavioural sciences, as well as in most biological sciences, statistical analyses using traditional algorithms do not always lead to a satisfactory solution, particularly in classification analysis. Current classification methods rely on parametric or non-parametric multivariate analyses: discriminant analysis, cluster analyses, etc. (e.g. Sparling and Williams, 1978; Martindale, 1980; Figueredo et al., 1992; Le Pape and Chevalet, 1992; Terhune et al., 1993). These methods are often rather inefficient when the data are nonlinearly distributed, even after variable transformation. Therefore, here we propose a classification method, based on the principles of artificial neural networks (NN).

During the eighties, the use of the NN developed explosively in the areas of word recognition (Maravall et al., 1991; Gemello and Mana, 1991;

PII S0376-6357(96)00766-8

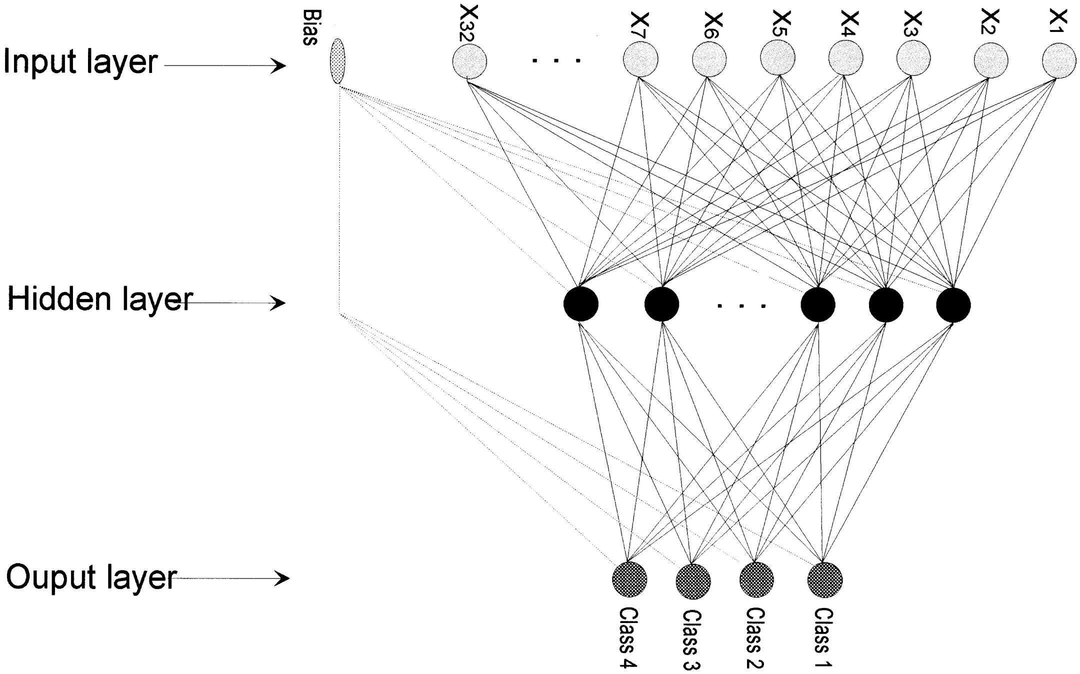

Fig. 1. Structure of a neural network as used in the experiments.

Waibel et al., 1989; Lefebvre et al., 1990), and character or image recognition (Belliustin et al., 1991; Van Allen et al., 1990; Tirakis et al., 1990; Omatu et al., 1990; Fukushima and Wake, 1990; Iwata et al., 1990). Yet, only a few applications were related to the biological sciences. In hydrobiology models, compared to multiple regression, NN clearly improved prediction performance (Lek et al., 1994, 1995). For classification purposes, NN have been used for the analysis of protein structure (Qian and Sejnowski, 1988; Andreassen et al., 1990), the classification of seaweeds (Smits et al., 1992), and the recognition of impulsive noises in marine mammals (Nicolas et al., 1989). In this paper, NN are used to discriminate the vocalizations of four male fallow deer, *Dama dama* (*Mammalia*, *Cervidae*), during the rutting period.

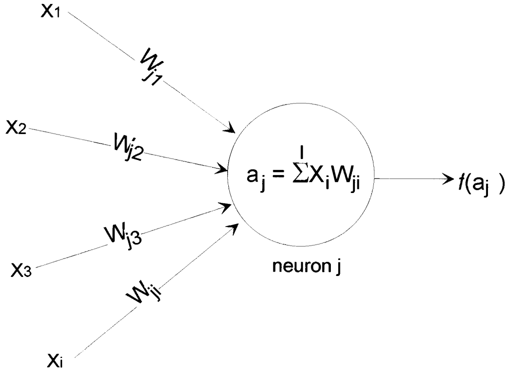

Fig. 2. Detail of one neuron (see text for parameter identification).

2. Theory of neural networks

Artificial neural networks were initially developed according to the elementary principle of the operation of the (human) neural system. Since then, a very large variety of networks have been constructed. All are composed of units (neurons), and connections between them, which together determine the behaviour of the network. The choice of the network type depends on the problem to be solved; the backpropagation gradient network is the most frequently used (Rumelhart et al., 1986; Kosko, 1992). This network consists of three or more neuron layers: one input layer, one output layer and at least one hidden layer. In most cases, a network with only one hidden layer is used to restrict calculation time, especially when the results obtained are satisfactory. All the neurons of each layer (except the neurons of the last one) are connected by an axon to each neuron of the next layer (Fig. 1).

2.1. Signal propagation

The input layer comprises n neurons that code for the n pieces of input signal ($X_1 \ldots X_n$) of the network (independent variables). The number of neurons of the hidden layer is chosen empirically by the user. Finally, the output layer comprises k neurons for the k classes (dependent variables). Each connection between two neurons is associated with a weight factor (random value between -0.3 and $+0.3$ at first); this weight is modified by successive iterations during the training of the network according to input and output data. In the input layer, the state of each neuron is determined by the input variable; the other neurons (hidden layer and output layer) evaluate the state of the signal from the previous layer (Fig. 2) as:

$$a_j = \sum_{i=1}^{I} X_i W_{ji}$$

where a_j is the net input of neuron j; X_i is the output value of neuron i of the previous layer; W_{ji} is the weight factor of the connection between neuron i and neuron j.

The activity of neurons is usually determined via a sigmoid function:

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

Thus, weight factors represent the response of the NN to the problem being faced.

2.2. Training the network

The backpropagation technique is akin to supervised learning as the network is trained with the expected reply/replies. Each iteration modifies the connection weights in order to minimize the error of the reply (expected value-estimated value). Adjustment of the weights, layer by layer, is calculated from the output layer back to the input layer. This correction is made by:

$$\Delta W_{ji} = \eta \delta_j f(a_i)$$

where ΔW_{ji} is the adjustment of weight between neuron j and neuron i from the previous layer; $f(a_i)$ is the output of neuron i, η is the learning rate, and δ_j depends on the layer. For the output layer, δ_j is:

$$\delta_j = (Y_j - \hat{Y}_j) f'_i(a_j)$$

where Y_j is the expected value ('observed value') and $\hat{Y}j$ is the current output value ('estimated value') of neuron j. For the hidden layer, δ_j is:

$$\delta_j = f'_j(a_j) \sum_{k=1}^{K} \delta_k W_{kj}$$

where K is the number of neurons in the next layer. The learning rate plays an important role in training. When this rate is low, the convergence of the weight to an optimum is very slow, when the rate is too high, the network can oscillate, or more seriously it can get stuck in a local minimum (Gallant, 1993). To reduce these problems, a momentum term α is used and ΔW_{ji} becomes:

$$\Delta W_{ji} = \eta \delta_j f(a_i) + \alpha \Delta W_{ji}^{\text{Prev}}$$

where $\Delta W_{ji}^{\text{Prev}}$ denotes the correction in the previous iteration. In our study, initially $\alpha = 0.7$ and $\eta = 0.01$, then they are modified according to the importance of the error by the following algorithm:

```
If present_error > previous_error * 1.04
then          $\eta = \eta * 0.75$,
              $\alpha = 0$,
else          $\eta = \eta * 1.05$,
              $\alpha = 0.95$,
EndIf
```

The training, performed on a representative data set, runs until the sum squared of errors (SSE) is minimized:

$$\text{SSE} = \frac{1}{2} \sum_{p=1}^{P} \sum_{j=1}^{N} (Y_{pj} - \hat{Y}_{pj})^2$$

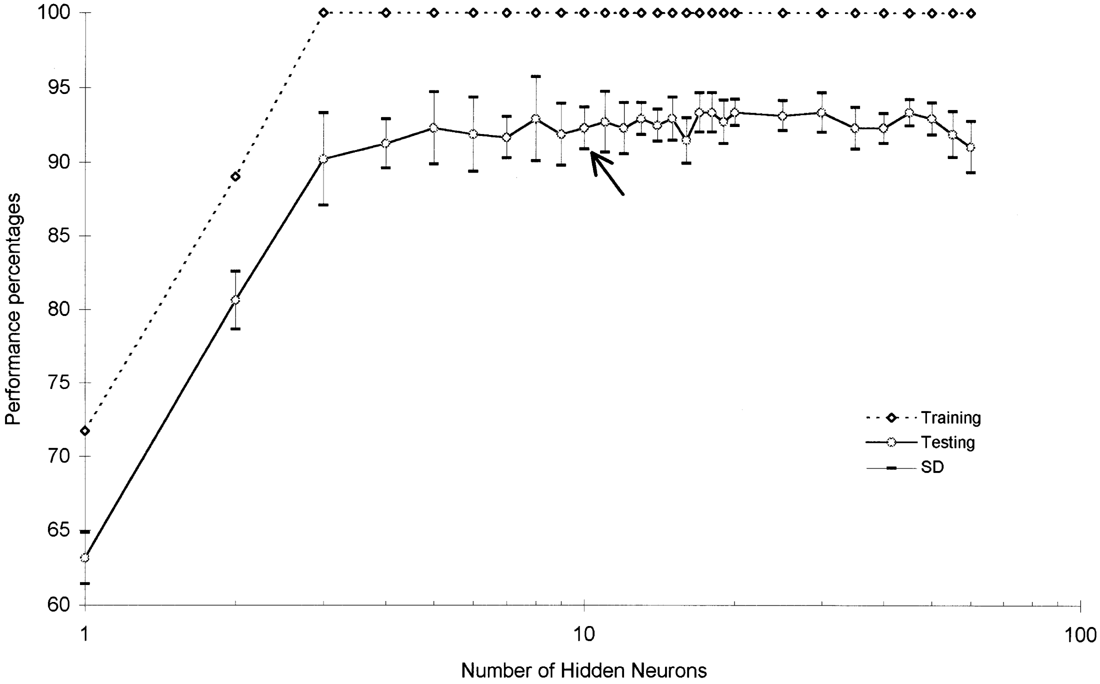

Fig. 3. Mean performance and standard deviation as a function of the hidden unit numbers (the arrow indicates the result for the 10 hidden-neuron layer). S.D.: standard deviation).

where: Y_{pj} is the expected output value, $\hat{Y}_{pj}$ is the estimated value by the network, $j = 1 \ldots N$ is the number of records and $p = 1 \ldots P$ is the number of neurons in the output layer.

The structure of the network, the number of records in the data set and the number of iterations that determine the training duration. In our study, for a hundred records characterised by 32 input variables and four output variables, and only one hidden layer with 10 neurons, 300 iterations last about 3 minutes with an Intel 486 DX2_66 processor.

2.3. Testing the network

After training, the performance of the network has to be tested. As in discriminant analysis, a first indication is given by the percentage of correct classifications of the training set records. Nevertheless, the performance of the network with a test set (set of similar data unused during training) is more relevant.

In the test step, the input data are fed into the network and the desired values are compared to the network's output values. The agreement or disagreement of the results thus give an indication of the performance of the trained network.

3. Material and methods

3.1. Sound recording and processing

Groans of four fallow deer bucks were tape-recorded during the rut, in October 1993. Each individual's set of vocalisations is a mixture of recordings made at different mating stands, and throughout the period of the rut, in order to prevent the recognition of recording conditions rather than individuality. Each vocalization was digitized using the Mc recorder program (8 bits amplitude resolution, 11 kHz sampling rate, anti-aliasing filter). Fast Fourier transformations (FFT size = 64, overlap = 50%, Hanning window func-

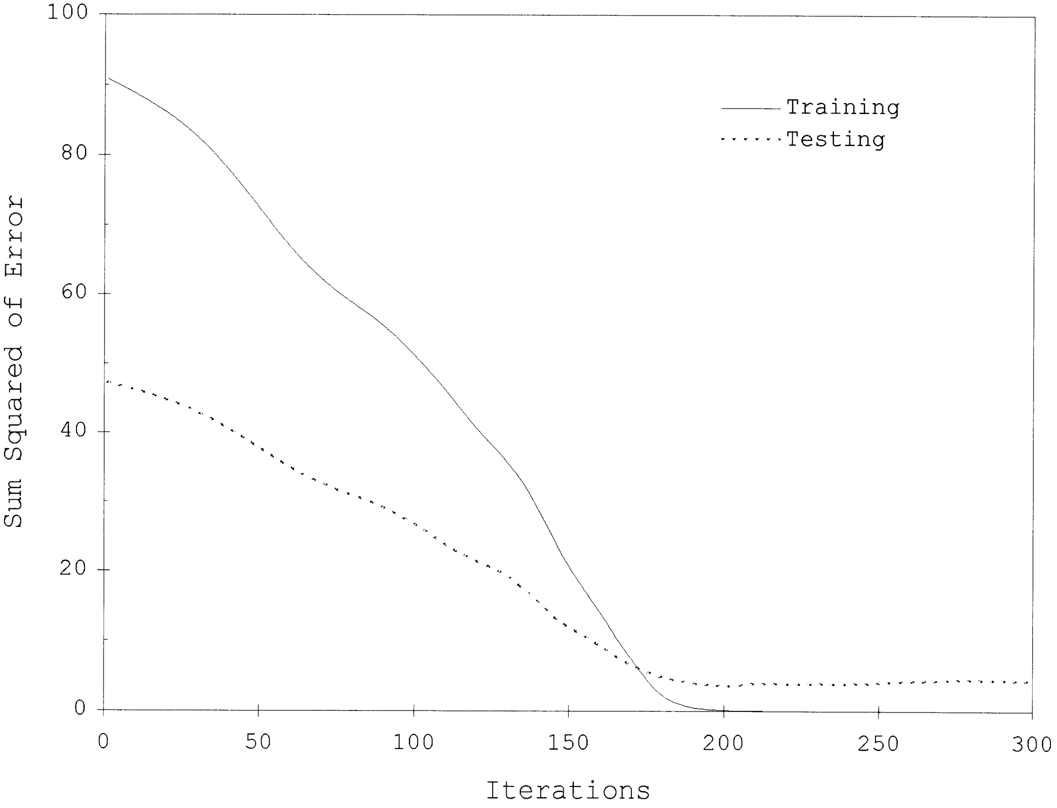

Fig. 4. Sum of squared error (SSE) as a function of the training iteration number.

tion) were performed on the numeric files, in order to depict the frequency distribution of sound pressure with a 32 value power spectrum.

3.2. Preparing data

The NN comprised 32 input neurons (32 harmonics of the mean spectrum) and four output neurons (four deer). Input variables, the 32 values of the power spectrum, differed in order of magnitude. So as a standardization, they were converted into reduced centred variables by:

$$Z_s = \frac{X_c - \bar{X}}{\sigma_X}$$

where Z_s are the standardized values, X_c are the original values, $\bar{X}$ and σ_X are the average and standard deviation of the variable under consideration.

3.3. Experimental protocol

The final activity of four output neurons (dependent variables: $f(a_j)$) was digitally coded by 0 if $f(a_j) < 0.5$ or 1 if $f(a_j) > 0.5$. So, in our example 1 0 0 0 designates deer 1, 0 1 0 0 designates deer 2, etc.).

The number of neurons in the hidden layer was empirically determined. First, we increased the number of neurons, one by one from 1 to 20, then five by five to 60. This gave 28 different configurations. For each configuration, 10 training sessions were performed with randomly different initial weight factors. The results were examined after 300 iterations.

Then, we studied the evolution of the network performance up to 300 iterations, with a 10 neuron hidden layer, by examining the SSE and the percentage of correct classifications in the training

Table 1
Confusion matrix of the five validation sets (Test 1…5).

Actual membership	Predicted group membership				Total	Proportion correctly classified
	1	2	3	4		
Test 1						
1	17	3	0	0	20	0.850
2	0	8	0	1	9	0.889
3	0	0	5	0	5	1.000
4	0	0	0	14	14	1.000
Test 2						
1	17	2	0	1	20	0.850
2	1	8	0	0	9	0.889
3	0	0	3	2	5	0.600
4	3	0	0	11	14	0.786
Test 3						
	1	2	3	4		
1	18	1	0	1	20	0.900
2	1	8	0	0	9	0.889
3	0	1	3	0	5	0.800
4	1	0	1	12	14	0.857
Test 4						
1	18	1	0	1	20	0.900
2	1	8	0	0	9	0.889
3	0	1	4	0	5	0.800
4	1	0	1	12	14	0.857
Test 5						
1	20	0	0	0	20	1.000
2	1	8	0	0	9	0.889
3	0	0	5	0	5	1.000
4	1	0	0	13	14	0.929

and test sets (in order to validate the 300 iteration training).

Lastly, we tested the repeatability of the neural network results. The point here was to verify that the discrimination quality was independent of the composition of the training set where there were few observations. From all the available records (141 recordings for the four males) we randomly sampled one training set of 93 recordings and one test set of 48 recordings, by respecting the proportion of individuals of each class in the total data set. This sampling method was performed 5 times in order to obtain five training sets and five test sets. Ten neuron hidden layer networks were trained with the five random training sets (300 iterations). After training, the weights of the networks were used in the testing phase with the five test sets.

Table 2
Mean and S.D. of the correctly classified proportion for the five validation sets.

Group	1	2	3	4
Mean	0.900	0.889	0.840	0.886
S.D.	0.061	0	0.167	0.082

4. Results

4.1. Number of neurons in the hidden layer

The average recognition (training set) and prediction (test set) percentages for each configura-

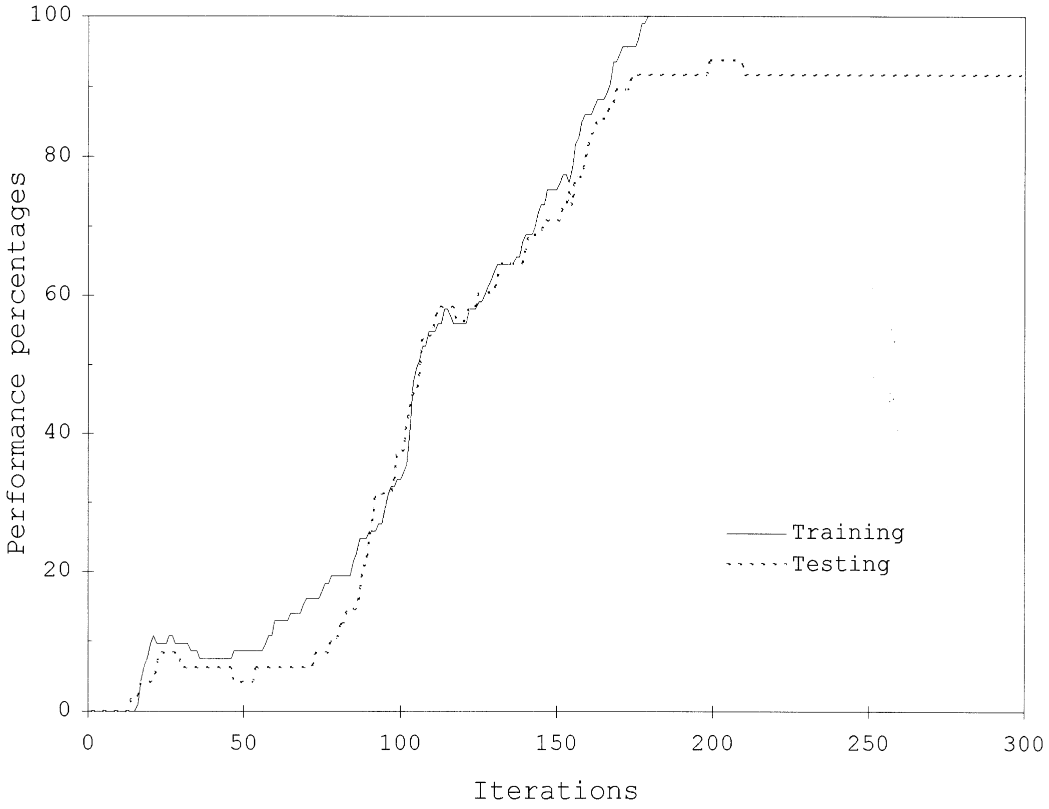

Fig. 5. Performance (percentage of correctly classified records) as a function of the training iteration number.

tion are represented in Fig. 3. We can see that performance increases very quickly with the number of neurons in the hidden layer: 72% recognition and 63% prediction with one neuron, 82% and 90% recognition, 89% and 100% prediction respectively for two and three hidden neurons. The performance became stable thereafter and, with more than 10 neurons, changes were negligible. Furthermore, performance declined slightly when the number of hidden neurons exceeded 50 units. In accordance with these results, we chose to work with a 10 hidden neuron layer which had a relatively small S.D.

4.2. *Number of iterations*

The SSE between observed values and values estimated by the network declined very rapidly from a high starting value to become 0 after 200 iterations in the training set (Figs. 4 and 5). In the test set, a similar variation was observed, with minimal values close to 0, but without actually reaching 0 as in the training set. Values of SSE stabilized after 200 iterations; then rose after 300 iterations (over fitting). The percentages of correct classifications (0% up to 20 iterations), first increased very slowly up to about 100 iterations (Fig. 5) and then climbed very quickly to reach 100% for 180 iterations with the training set and 92% with the test set. After this number of iterations the percentage of prediction fluctuated slightly between 92 and 96%. In most of our studies, therefore, we stopped training of the network at 300 iterations.

4.3. Discrimination performance

Results of prediction of the 5 NN trained with random sampled sets are presented in Table 1. Noticeable variations appeared between the different networks. Prediction percentages varied from 60 to 100% with most values around 90%. The strongest fluctuations were observed for deer 3, which had the smallest number of records ($n = 14$). For the other deer with more records ($n = 26$, 41, 60), homogeneity was observed between the tests; this homogeneity is confirmed by small standard deviations (especially for deer 2, Table 2). The overall average performance of prediction of the network was 88 ± 3%.

5. Conclusion

The first results presented here show that performance of prediction was improved with an increasing number of neurons in the hidden layer until an asymptote was reached. Increasing this number further enhanced the risk of overfitting (Gallant, 1993). The progressive improvement of the prediction performance with the number of iterations is obvious. A limit is also quickly reached and it is not recommended to extend the training time. Indeed, time is saved by avoiding overfitting. Finally, we observed very good performance of the networks with 100% recognition and nearly 90% prediction (note that these results were obtained with a fairly small set of data).

A successful example of individual identification by analysing vocalisations of one mammal species has been presented here. NN can be used in any behavioural study whenever it is necessary to: (i) demonstrate the relevance of pre-judged categories (recognition capacity), and (ii) classify additional records (individuals, behaviours, etc.) in defined categories (prediction capacity).

NN are an alternative method for classification. They can work with large numbers of qualitative variables such as behaviours, provided that they can be coded, and they are able to use non-linear linked variables (Lek et al., 1996). Moreover, once the weight factors are saved, NN can easily be reused to immediately classify additional records, as with discriminant functions.

References

Andreassen, H., Bohr, H., Bohr, J., Brunak, S., Bugge, T., Cotterill, M.J., Jacobsen, C., Kusk, P., Lautrup, B., Petersen, S.B., Saermark, T. and Ulrich, K., 1990. Analysis of the secondary structure of the human immunodeficiency virus (HIV) proteins p17, gp20, and gp41 by computer modelling based on neural network methods. J. Acq. Immun. Defic. Syndrome, 3: 615–622.

Belliustin, N.S., Kuznetsov, S.O., Nuidel, I.V. and Yakhno, V.G., 1991. Neural networks with close nonlocal coupling for analysing composite image. Neurocomputing, 3: 231–246.

Figueredo, A.J., Ross, D.M. and Petrinovich, L., 1992. The quantitative ethology of the Zebra finch: a study in comparative psychometrics. Multiv. Behav. Res., 27 (3): 435–458.

Fukushima, K. and Wake, N., 1990. Alphanumeric character recognition by the neocognitron. In: R. Eckmiller (Editor), Advanced Neural Computers, Elsevier Science, Amsterdam, pp. 263–270.

Gallant, S.I., 1993. Neural Network Learning and Expert Systems. M.I.T. Press, London, pp 365.

Gemello, R. and Mana, F., 1991. A neural approach to speaker independant isoled word recognition in an uncontrolled environment. In: Proc. Int. Neural Networks Conf., Paris July 9–13, 1990, Kluwer Academic Publishers, Dordrecht, Vol. 1, pp. 35–37.

Iwata, A., Tohma, T., Matsuo, H. and Suzumura, N., 1990. A large scale neural network 'CombNET' and its application to Chinese character recognition. In: Proc. Int. Neural Networks Conf., Paris July 9–13, 1990, Kluwer Academic Publishers, Dordrecht, Vol. 1, pp. 83–86.

Kosko, B., 1992. Neural Networks for Signal Processing. Prentice-Hall, London, pp. 399.

Lefebvre, T., Nicolas, J.M. and Dagoul, P., 1990. Numerical to symbolical conversion for acoustic signal classification using a two-stage neural architecture. In: Proc. Int. Neural Networks Conf., Paris July 9–13, 1990, Kluwer Academic Publishers, Dordrecht, Vol. 1, pp. 119–122.

Lek, S., Lauga, J., Giraudel, J.L., Baran, P. and Delacoste, M., 1994. Example of an application of formal neural networks in the environmental sciences. In: V. Brétagnolle and F. Beninel (Editors), Proc. Int. Meeting 'Ecology and statistical methods', C.N.R.S., Chizé, pp. 73–82.

Lek, S., Belaud, A., Lauga, J., Dimopoulos, I. and Moreau, J., 1995. An improved estimation of the food consumption of fish populations using neural networks. Aust. J. Mar. Freshwater Res., 46 (8): 1229–1236.

Lek, S., Delacoste, M., Baran, P., Lauga, J. and Aulagnier, S., 1996. Application of neural networks to modelling nonlinear relationships in ecology. Ecol. Model., 90: 39–52.

Le Pape, G. and Chevalet, P., 1992. Description des données d'observation continue du comportement par une technique d'analyse de texte. 2: comparaisons des conduites maternelles dans trois souches de souris. Behav. Proc., 26 (1): 23–30.

Maravall, D., Rfos, J., Pérez-Castellano, M., Carpintero, A. and Gïmez-Calcerrada, J., 1991. Comparison of neural networks and conventional techniques for automatic recognition of a multilingual speech database. In: A. Prieto (Editor), Artificial Neural Networks, Proc. Int. Workshop IWANN'91, Granada (SPAIN), Sep-91, pp. 377–384.

Martindale, S., 1980. On the multivariate analysis of avian vocalizations. J. Theor. Biol., 83: 107–110.

Nicolas, J.M., Lemer, A. and Legitimus, D., 1989. Identification automatique de bruit impulsifs en acoustique sous-marine par réseaux multicouches. In: Neuro-Nîmes'89, Int. Workshop Neural Networks and their Applications, Nîmes (FRANCE) November 13–16, 1989, EC2, Nanterre, pp. 269–278.

Omatu, S., Fukumi, M. and Teranisi, M., 1990. Neural network model for alphabetical letter recognition. In: Proc. Int. Neural Networks Conf., Paris July 9–13, 1990, Kluwer Academic Publishers, Dordrecht, Vol. 1, pp. 19–22.

Qian, N. and Sejnowski, T.J., 1988. Prediction of the secondary structure of globular proteins using neural networks models. J. Mol. Biol., 202: 865–884.

Rumelhart, D.E., Hinton, G.E. and Williams, R.J., 1986. Learning representations by back-propagating error. Nature, 323: 533–536.

Smits, J.R.M., Breedveld, L.W., Derksen, M.W.J., Kateman, G., Balfoort, H.W., Snoek, J. and Hofstraat, J.W., 1992. Pattern classification with artificial neural networks: classification of algae, based upon flow cytometer data. Anal. Chim. Acta, 258: 11–25.

Sparling, D.W. and Williams, J.D., 1978. Multivariate analysis of avian vocalizations. J. Theor. Biol., 74: 83–107.

Terhune, J.M., Burton, H. and Green, K., 1993. Classification of diverse call types using cluster analysis techniques. Bioacoustics, 4: 245–258.

Tirakis, A., Sukissian, L. and Kollias, S., 1990. An adaptive technique for segmentation and classification of textured images. In: Proc. Int. Neural Networks Conf., Paris, Vol. 1, pp. 31–34.

Van Allen, E., Menon, M.M. and Dicaprio, N., 1990. A modular architecture for object recongnition using neural networks. In: Proc. Int. Neural Networks Confer., Paris July 9–13, 1990, Kluwer Academic Publishers, Dordrecht, Vol. 1, pp. 35–37.

Waibel, A., Hanazawa, T. Hinton, G. Shikano, K. and Lang, K.J., 1989. Phoneme recognition using time-delay neural networks. IEEE Trans. Acoust. Speech Signal Proc., 37(3): 328–339.