ELSEVIER

Ecological Modelling 120 (1999) 109–118

ECOLOGICAL MODELLING

www.elsevier.com/locate/ecomodel

Radial basis function networks with partially classified data

Isabella Morlini *

Istituto di Statistica, Università degli Studi di Parma, Via J.F. Kennedy 6, I-43100 Parma, Italy

Abstract

The problem of estimating a classification rule with partially classified observations, which often occurs in biological and ecological modelling, and which is of major interest in pattern recognition, is discussed. Radial basis function networks for classification problems are presented and compared with the discriminant analysis with partially classified data, in situations where some observations in the training set are unclassified. An application on a set of morphometric data obtained from the skulls of 288 specimens of *Microtus subterraneus* and *Microtus multiplex* is performed. This example illustrates how the use of both classified and unclassified observations in the estimate of the hidden layer parameters has the potential to greatly improve the network performances.

Keywords: Classification; Discriminant analysis; Mixture analysis; Radial basis function networks

1. Introduction

One of the major problems related to practical applications in pattern recognition is the presence of partially classified data. In these situations the population from which the sample is taken consists itself of a number of several homogeneous sub-populations, but the group membership of the training data is known only for some input vectors. If the quantity of data available is sufficiently large, and the proportion of unclassified observations is small, then the simplest solution is to discard those patterns from the data set. This approach, however, is implicitly assuming that the cause of the omission of the group membership is independent of the data itself. If the reason of the omission of the group membership depends on the data, then this approach will modify the effective data distribution (Bishop, 1995). When there is too little data to discard the unclassified one, or when the proportion of unclassified observations is high, it becomes important to use all the information which is potentially available from the incomplete patterns. It is intuitively clear, in fact, that the unclassified observations, as well as the classified ones, contain some knowledge about the distribution of the measured variables in the different groups.

The purpose of this work is to show the benefits of using the information contained in a partially classified data set to the maximum extent. Radial basis function networks are introduced and demonstrated to be a suitable method in situations where some observations in the training data are unclassified. An application on an ecological

* Fax: + 39-0521-902375.

E-mail address: morlini@economia.econ.unipr.it (I. Morlini)

0304-3800/99/$ - see front matter
PII: S0304-3800(99)00095-2

problem, which illustrates how to include unclassified observations in the network training, and which compares the network performances with those reached by conventional discriminant analysis and by discriminant analysis with partially classified observations, is presented. The network performances are measured in terms of classification error rate and generalisation to unobserved patterns.

2. Radial basis function networks

Radial basis function (RBF) networks provide a powerful technique for generating multivariate, non-linear mappings (Broomhead and Lowe, 1988). Unlike the widely used multi-layer perceptron, that is based on units which compute a non-linear function of the scalar product of the input vector and a weight vector, the activation of a RBF hidden neuron is determined by the distance between the input vector and a prototype vector. The RBF network mapping from a d-dimensional input space x to a c-dimensional target space t is a linear combination of a set of M basis functions, which take the form:

$$y_k(x) = \sum_{j=1}^{M} w_{kj}\phi_j(\|x-\mu_j\|) + w_{k_0} \quad k = 1,\ldots, c \qquad (1)$$

where x is the d-dimensional input vector with elements x_i and μ_j is the vector determining the centre of basis function ϕ_j and has elements μ_{ij}. The basis functions can be normalised (Moody and Darken, 1989) through lateral connections between different hidden units in the network diagram, so that the output becomes:

$$y_k(x) = \sum_{j=1}^{M} w_{kj} \frac{\phi_j(\|x-\mu_j\|)}{\sum_{j=1}^{M} \phi_j(\|x-\mu_j\|)} \quad k = 1,\ldots, c \qquad (2)$$

Usually the distance $\|x-\mu_j\|$ is taken to be Euclidean and several form of basis functions can be considered, the most common being the Gaussian:

$$\phi_j(\|x-\mu_j\|) = \exp\left(-\frac{\|x-\mu_j\|^2}{2\sigma_j^2}\right) \qquad (3)$$

where the standard deviation σ_j, also called smoothing parameter, determines the width of the hidden unit. If the basis functions are Gaussians, then the hidden units assume a localised nature: the network forms a representation in the space of hidden units which is local with respect to the input space, because, for a given input vector, only few hidden units will have significant activations. The use of radial basis functions can be motivated from a number of different concepts as function approximation, noisy interpolation, density estimation and optimal classification theory (Bishop, 1995). In this work we are considering the use of such networks for a classification problem. A multilayer perceptron can separate classes by using hidden units, which form hyperplanes, or hypersurfaces in the input space, and for this reason can be related to discriminant analysis. A RBF network is able to model each class distribution by local kernel functions, and so can be rather compared with the kernel discriminant analysis. If, in a classification problem, the goal is to model the posterior probabilities $p(C_k|x)$ for each of the classes C_k, $(k = 1,\ldots, c)$, then these probabilities can be obtained through Bayes' theorem, using prior probabilities $p(C_k)$ as follows:

$$P(C_k|x) = \frac{p(x|C_k)P(C_k)}{p(x)} = \frac{p(x|C_k)P(C_k)}{\sum_{k=1}^{c} p(x|C_j)P(C_j)} \qquad (4)$$

where $P(\cdot)$ indicates a probability and $p(\cdot)$ a probability density function. If the class-conditional distributions are obtained by using not a single kernel function, but a mixture model constituted by a common pool of M basis functions, labelled by an index j and equal for every density, then the probabilities $p(x/C_k)$ and $p(x)$ can be written as

$$p(x|C_k) = \sum_{j=1}^{M} p(x|j)P(j|C_k) \qquad (5)$$

and

$$p(x) = \sum_{k=1}^{c} p(x|C_k)P(C_k) = \sum_{j=1}^{M} p(x|j)P(j) \qquad (6)$$

where priors $P(j)$ are given by

$$P(j) = \sum_{k=1}^{c} P(j|C_k)P(C_k) \qquad (7)$$

The posterior probabilities can be obtained by substituting Eqs. (5) and (6) into Bayes' theorem (4) and adding an extra factor of $1 = P(j)/P(j)$ to give:

$$P(C_k|x) = \frac{\sum_{j=1}^{M} P(j|C_k)p(x|j)P(C_k)P(j)}{\sum_{j'=1}^{M} p(x|j')P(j')P(j')} = \sum_{j=1}^{M} w_{kj}\phi_j(x) \quad (8)$$

This expression represents a radial basis function network (Bishop, 1995), in which the normalised basis functions are given by

$$\phi_j(x) = \frac{P(x|j)P(j)}{\sum_{J'=1}^{M} p(x|j')P(j')} = P(j|x) \quad (9)$$

and the second layer weights are given by

$$w_{kj} = \frac{P(j|C_k)P(C_k)}{P(j)} = P(C_k|j) \quad (10)$$

After training, for a particular partition of the data into c groups, the value of each k output neuron, ($k = 1, \ldots, c$) can be interpreted as the posterior probability of corresponding class membership. Thus, following the optimal classification rule (Anderson, 1984), in a two class problem an observation should be classified as belonging to group k if the value of the corresponding output unit is bigger than 0.5. In practice, when least squares are used to set the second layer parameters and the target values are coded with the 1-of-c coding scheme (so that they sum to unity), the output values are forced to sum to unity but they are not forced to lie in the range [0, 1]. If the output values do not lie in this range, they should be normalised.

The major problems related to a RBF network are the determination of the number of basis functions and the choice of the parameters. The faster and simplest procedure is to create a Probabilistic Neural Network (Specht, 1990) which has N localised hidden units centred on each input vector. In these networks the parameters σ_j are usually heuristically determined. One approach is to choose all σ_j to be equal and to be given by some multiple of the average distance between the basis function centres. This ensures that the basis functions overlap to some degree and hence give a relatively smooth representation of the distribution of the training data. In order to determine the number of basis functions by the complexity of the data, rather by the size of the data set, a subset of the input vectors can be chosen by forward selection or orthogonal least squares to serve as centres. A different approach is to choose the number of basis functions and determine the parameters by supervised or unsupervised methods. An exhaustive list of these methods, together with their theoretical issues, is in Bishop (1995). A k-means procedure is adopted in the example of section 4. This procedure proposed by Moody and Darken (1988), sets the centres of basis functions equal to the cluster centres found by the k-means clustering algorithm, and the standard deviations σ_j equal to the average distances to the z-nearest clusters. Moody and Darken (1988) report good empirical results for using this procedure. The main drawback of this method is that the number of basis functions must be defined a priori. This leads to similar problems as the 'number of hidden units' dilemma in the multi layer perceptron, since it is very difficult to estimate an appropriate number of basis functions. In Section 4 we determine the optimal number of clusters (and, therefore, the optimal number of basis functions in the RBF network) on the basis of the within-groups and between-groups deviances, for different number of groups. Once the parameters of the hidden layer are determined, the network has to be trained to produce the optimal values of the second layer weights. When the error function is a quadratic function of these weights, its minimum can be found in terms of the solution of a set of linear equations. In fact, if we indicate with N the number of training cases and with $t_k(x_n)$ the target value for output unit k when network is presented with input vector x_n ($n = 1, \ldots, N$; $k = 1, \ldots, c$), then the sum of squares error function is given by

$$E = \sum_{n=1}^{N} \sum_{k=1}^{c} \{y_k(x) - t_k(x)\}^2 \quad (11)$$

where y_k is defined in Eqs. (1) and (2). Training is then very fast and does not have the problem of local minima.

3. Estimating group membership with partially classified observations

In real applications, especially in biological and ecological modelling, it sometimes happens that group membership is known only for a subset of the original sample. This can arise, for example, when the exact determination of group membership requires high laboratory costs. In these situations, classical supervised methods, like the discriminant analysis or the multi-layer perceptron, are often applied. Classified observations are used to estimate the discrimination rule and this rule is then applied to unclassified observations, to determine the corresponding group membership. Evidently, this procedure does not use the information contained in the data to the maximum extent, since it is clear that the unclassified observations contain some information about the distribution of the measured variables in the groups, as well. There is also some theoretical literature on the benefits of using unclassified observations for estimation (O'Neill, 1978; McLachlan and Basford, 1988). On the other hand, using an unsupervised procedure (like mixture analysis, cluster analysis or the Kohonen network) over the entire data set means ignoring group membership of classified observations and, therefore, discarding important available information. Airoldi et al. (1995) found that mixture analysis, compared with discriminant analysis on a data set with partially classified observations, reveals highly unstable estimates. They conclude that ignoring group membership is a bad idea. In statistics, an iterative method that uses the information contained in both classified and unclassified observations in the parameter estimation is fairly well developed under the name of discriminant analysis with partially classified data (*discrimix*). This method (McLachlan and Basford, 1988; Airoldi et al., 1995) has the potential to greatly improve the estimation of the classification rule. However, it is a re-estimation procedure which may involve some technical problems in the solution of the equation system. These drawbacks are the computational time and costs, the eventual convergence to a singular estimate of the covariance matrix (that will cause the algorithm to fail), the absence of convergence or the convergence to a local maximum. Some of these problems can be overcome with a constrained maximum solution and the availability of good computer programs. Therefore, the main drawback of this method seems to be the assumption of multivariate normality of the density function in each group. This assumption is indispensable in discriminant analysis with partially classified data, since the density function appears explicitly in one equation of the system. This is also a crucial difference to discriminant analysis, where calculus can be justified without assuming normality or any other particular distribution.

RBF networks in which the basis functions parameters are estimated by unsupervised procedures are particularly advantageous for applications with partially classified observations, since the hidden layer parameters can be determined using both labelled and unlabelled data, leaving a relatively small number of parameters in the second layer to be determined using the classified data. It must be remarked that using unsupervised methods for determining the hidden units parameters, doesn't mean ignoring group memberships in the entire procedure, since the second layer parameters are determined by the solutions of a set of linear equations, which includes target values. One advantage of RBF networks, over discriminant analysis with partially classified data, is that they do not require iterative procedures in the estimate of the second layer parameters. Moreover, they do not need the assumption of multivariate normality or any other particular distribution of the density function of the input variables in each group.

Next section illustrates how the use of unsupervised procedures for the determination of the basis function parameters and, consequently, the use of unlabelled data in the estimate of the classification rule in a problem with partially classified observation, can improve the performances of a RBF network. RBF networks are

also compared with discriminant analysis and discriminant analysis with partially classified observations.

4. Real data set example

4.1. The microtus data

This example is based on the classification of two species of voles (Flury, 1997, pp. 333–339). The two species, *Microtus multiplex* and *Microtus subterraneus*, differ in the number of chromosomes, but are morphometrically difficult to distinguish. The geographic ranges of distribution of the two species overlap to some extent in the Alps of southern Switzerland and northern Italy (Krapp, 1982; Niethammer, 1982). *M. subterraneus* is smaller than *M. multiplex* in most measurements. It usually occurs at elevations from 1000 m to over 2000 m, but it is also found at lower elevations. *M. multiplex* is found at similar elevations, and also at latitudes from 200 to 300 m (South of the Alps). Much of the data available are in form of skull remains, either fossilised or from owl pellets. Till now, no reliable criteria based on cranial morphology have been found to distinguish the two species. The data set consists of eight variables measured on the skulls of 288 specimens found at various places in central Europe: X1 = width of upper left molar 1; X2 = width of upper left molar 2; X3 = width of upper left molar 3; X4 = length of incisive foramen; X5 = length of palatal bone; X6 = condylo incisive length or skull length; X7 = skull height above bullae; X8 = skull width across rostrum. Variables X1 to X5 are measured in mm/1000; variables X6 to X8 are in mm/100. These cranial measurements are relatively inexpensive to carry out, since they can be measured with a measurescope (accurancy 1/1000 mm) and dial calipers (accuracy i/100 mm). Nevertheless, the exact determination of the species requires a costly chromosomal investigation. For this reason, only 89 of the skulls were analysed to identify their species: 43 specimens were from *M. multiplex* and 46 from *M. subterraneus*. The chromosomes were not analysed and species was not determined for the remaining 199 observations.

Airoldi et al. (1995) report a discriminant analysis, a finite mixture analysis and a discriminant analysis with partially classified observations (which they call *Discrimix*) of this data set. Here, we seek to analyse the data with RBF networks and to compare the classification capabilities of different models. The analysis is first performed using both classified and unclassified observation in the optimisation of the basis function parameters. In order to reach better generalisation capabilities, a pre-processing stage is then applied to the network. Results are finally compared with those reached by a RBF network with parameters determined using the sole 89 classified specimens and with those reached by other statistical models.

In the RBF networks considered in the following the input variables are combined via the Euclidean distance function, so that the contribution of an input variables depends heavily on its variability relative to other inputs. In order to give the same importance to every input variable, variables are standardised to zero mean and unit variance before every process.

4.2. Computation of the error rates

Two types of error rates are used to assess the performance of classification procedure. The first, the simplest and most popular error, is the *plug-in error rate:* it is the proportion of observations misclassified when the classification rule is applied to the data in the training sample. The second, the *cross-validation error* (Stone, 1974), is obtained as follows. The sample is divided in k subsets of equal size. The network is trained k times, each time leaving out one of the subsets from training, and using the omitted subset to compute the error rate. If k equal the sample size, and only one observation is used each time to compute the proportion of observation misclassified, than cross validation reduces to the *leave-one-out* error rate. The plug-in error rate is very fast to compute and, since it uses the entire sample to train the network, it is very advantageous when only a little sample is available. The main drawback of the plug-in error rate is that it tends to be overly optimistic, that is, it tends to underestimate the

Table 1
ANOVA table for different number of clusters

Number of clusters	Deviance between	Degree of freedom	Deviance between	Degree of freedom	Deviance total	Degree of freedom	R^2
2	1153.229	1	1142.771	286	2296	1	0.5023
3	1447.640	2	848.360	285	2296	1	0.6305
4	1565.199	3	730.801	284	2296	1	0.6817
5	1636.804	4	659.196	283	2296	1	0.7129
6	1681.827	5	614.173	282	2296	1	0.7325
7	1715.849	6	580.151	281	2296	1	0.7473

probability of misclassifying future observations, since the error is calculated over the same data employed during training. Cross validation gives a better estimate of the generalisation error, namely, the average misclassification rate over the entire space of possible inputs. For this reason, cross validation is often preferred, but if k gets too small, the error estimate is pessimistically biased because of the difference in sample size between the full-sample analysis and the cross-validation analyses. For this reason, a value of $k = 10$ is chosen, since it is shown to offer good empirical results in literature.

4.3. Using both classified and unclassified observations in a RBF network

Eq. (9) points out that the basis functions depend solely on the input data and ignore any target information. In particularly, the basis function parameters should be chosen to form a representation of the probability density of the input data and the centre μ_j should be regarded as *prototype* of the input vectors. This justifies the use of unsupervised procedures to determine the basis function parameters, which are usually very fast and can be run a number of time, in order to test the robustness of the results, with low computational costs. Following Moody and Darken (1989), the *k-means* clustering algorithm is performed to optimise both the basis function centres and the widths. The optimal number of clusters is heuristically chosen comparing the within-groups and between-groups deviances, for different values of k. Due to an increase in the number of clusters, the deviance between groups (which indicates the share of total deviance 'explained' by the aggregation of the observations in clusters) increases, while the deviance within (which indicates the error minimised by the algorithm) decreases. As long as the increase in the deviance between groups is considerable, we think it justifies the increase in the complexity of the grouping structure (due to the addition of new groups). We stop adding clusters when this increase becomes poor, in order to reach a good compromise between the proportion of the total deviance 'explained' by the aggregation in groups and a parsimonious number of clusters (which means a clearer and simpler representation of the data set). The ANOVA table obtained running the k-means cluster analysis for the 288 observations, for different values of k (using the package SPSS for Windows, release 7.5), is reported in Table 1. The coefficient R^2 is the ratio between the deviance between groups and the total deviance. The increase in R^2 from 2 to 3 clusters is considerable. From 3 to 4 groups it is still fairly great, while from 4 to 5 clusters it becomes poor. From 5 to 6 and from 6 to 7 groups the increase in R^2 is nearly negligible. The 'optimal' grouping structure, the one which appears to lead to the best *trade off* between number of clusters and variance in each cluster, seems therefore to be associated with $k = 4$.

In a RBF network with eight input nodes (one for each variable), four hidden nodes with centres determined by the cluster means and widths determined by the minima distance between all the other clusters, and second layer weights determined by linear regression, the plug in error rate is 5.62%, while the cross validation error rate is 2.28%.

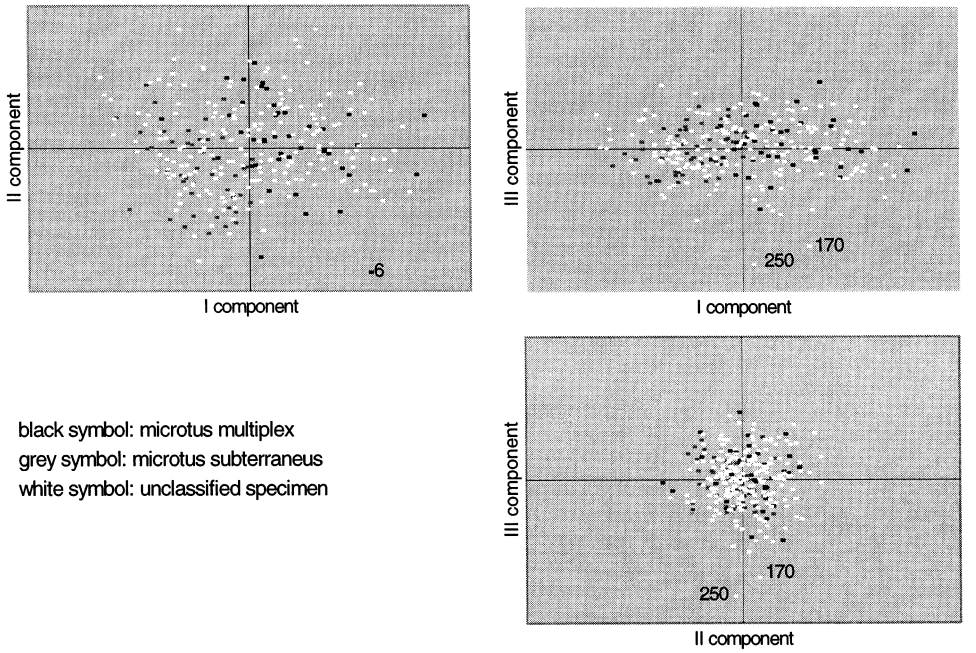

Fig. 1. Scatter plot matrix of the first three principal components.

Analysing the correlation matrix of the input data, it can be noted that the eight variables are highly correlated and the information related to many of these variables is therefore redundant. When input variables are highly correlated, a subset of these variables or a linear transformation of these into new, fewer variables may describe the data equally well and, in accordance with the principle of parsimony (or 'Occam's Razor') the simplest model, the one with fewer variable, should be preferred. Moreover, the network performances may improve with a reduction of the input vector dimensionality (and the related loss of information), since a network with fewer inputs has fewer adaptive parameters to be determined. These parameters are more likely to be properly constrained by a data set of limited size, leading to a network with better generalisation properties. As a pre-processing stage, a principal component analysis is performed in order to form linear combinations of the original variables and generate new (less) input variables for the network. The scatter plot diagram of the first three principal components is reported in Fig. 1. Using the scores of the first n principal components as input variables, the proportion of original information that is preserved can be measured. Since the first three principal components retain the 88% of the original variance, only the 12% of original information is lost using these scores as input variables. The scatter plot diagram of Fig. 1 also reveals the presence of possible multivariate outliers, since observations 6, 170 and 250 clearly stand aside from the cloud of points. In order to determine the basis function parameters, cluster analysis is then performed with $k = 4$ and without this three possible outliers. The second layer parameters are determined by least squares, with a training set of 88 observations (unit 6 is discarded also for linear regression). With this preprocessing step, the plug in error rate of the RBF network is 3.37%, while the cross validation one is 4.49%.

An alternative pre-processing concerning discard of six (redundant) input variables and elimination of the three possible outliers is also applied. Performing the analysis with the sole variables X1 and X4, in which the two groups are

Table 2
Error rates of a radial basis function networks with parameters determined using both classified and unclassified observations

Error rate	RBF network with eight input variables (%)	RBF network with three input variables	RBF network with two input variables (%)
Plug-in	5.62	3.37	3.37
Cross validation	2.28	4.49	3.37

well separated (see Airoldi et al., 1995), the plug in and the cross validation error rates are both 3.37%.

Table 2 summarises the results obtained in the different analysis. Particularly attention must be paid to the first analysis, since the cross validation error rate of the RBF network is less than the plug-in one.

This is a fairly unusual and unexpected result, even if it is not impossible in theory. The explanation of this phenomenon can be related to the normalisation of the basis functions. Normalisation is desirable for a classification problem, since at every point in the input space the sum of the basis function is forced to sum to unity so that, in mixture underlying model, the activation of each basis function can be interpreted as the posterior probability of the presence of corresponding feature in input space (see Eq. (9)) and the network outputs can be interpreted as Bayesian posterior probabilities of group memberships (Bishop, 1995). However, normalisation leads to a number of side effects which are described in Murray-Smith (1994). Some of these side effects should be considered here, in order to motivate the better performances of the network in the test set rather then in the training set. The first one is that when the basis functions are Gaussians, the normalisation results the whole of the input space being covered and not just the region of the input space defined by the training data. The second one is that basis functions with different widths (which are used in the application) can become multimodal, meaning that their activations increase as the distance function between the input vector and the centre decreases (this phenomenon is called 'reactivation' of the basis functions). A final side effect, which also concerns basis functions with different widths, is that the maxima may no longer be at their centres. These three normalisation effects, which are more pronounced as the input dimension increases, due to the increased number of neighbouring units in higher dimensions, justify results reported in the first column of Table 2. From a heuristic point of view, we have noted that, performing the analysis with an unnormalised RBF network, the plug is error rate is less than the cross-validation one.

4.4. Using only classified observations in a RBF network

In a classical set of a probabilistic neural network, the 89 specimens with known group membership should constitute the training sample and, in a subsequent stage, the trained neural network should be used to assign the remaining 199 specimens to either the *M. multiplex* or the *M. subterraneus* group. Using a probabilistic neural network with eight input nodes, one for each explanatory variable and 89 hidden nodes with equal width parameters and centres determined by the input vectors, the following numerical results are obtained. The plug in error rate is 1.12% and the cross validation error rate is 10.1%. Using the first three principal components as input variable, the plug in and the cross validation error rates are both 6.82. Performing the analysis with the two variables X1 and X4, the misclassified observations in the training set are 5 and the plug in error rate is therefore 5.62%. The cross validation error is 8.99%. The reduction of the input vector dimensionality improves the generalisation properties of the network, but these numerical results are still remarkably worse that those previously obtained. The advantage of using a RBF network with basis function parameters determined using both classified and unclassified observations is there-

fore apparent, since generalisation of a result obtained from a particular data set is one of the most important concerns in quit every real applications.

4.5. Comparisons with other concurrent methods

For the 89 classified observations and using the discriminant analysis the following numerical results are obtained for all eight variables (for theoretical and empirical comparisons between discriminant analysis and other classification tools see, for example, Hand, 1981; Ripley, 1994). With prior probabilities given by the relative frequencies of observations in each group, the plug in error rate is 5.62% and the cross validation one is 6.74%. With equal prior probabilities the error rates are, respectively, 4.49 and 6.74%. Using variables X1 and X4, only, the plug in error rate is 4.49% and the cross validation is 5.62%, both for equal and different prior probabilities. Numerical results and parameter estimations obtained from discriminant analysis with partially classified observations are reported in Airoldi et al. (1995). Here it should be noted that error rates obtained with two input variables are remarkably similar to those obtained by conventional discriminant analysis. The advantage of *discrimix* over discriminant analysis is apparent performing bootstrap analysis, since it reveals that the estimates from *discrimix* are typically much smaller. From a numerical point of view, RBF network with basis function parameters given by k-means cluster analysis outperforms procedure *discrimix*. However, comparison between *discrimix* and RBF network should be more detailed, since the purposes of these two methods are different. Discriminant analysis with partially classified observations (like conventional discriminant analysis and mixture analysis) attempts to estimate the parameters of a population which is known to be composed of a fixed number of homogeneous sub-populations. It directly models the class distributions by Gaussian mixtures in the sampling paradigm. The outputs of a RBF network represent, in an underlying mixture model, the posterior probabilities of class memberships. However, procedure k-means partition a data set deterministically into subgroups and the number of these sub-populations is heuristically determined. The hidden layer of a RBF network is used to learn bout the class distributions and to estimate the number of sub-clusters in the training data, when this number is unknown. Procedure k-means can be seen as a particular limit of the expectation-maximisation (EM) algorithm used in *discrimix*. It can be shown that in case of Gaussian basis functions with a common width parameter σ and in the limit $\sigma \to 0$, the EM update formula for a basis function centre reduces to the k-means update formula (Dempster et al., 1977). However, means and variances of the k-clusters are not in general considered as estimators of the parameters of the component densities. Similarly, the mixing coefficient w_{kj}, which are determined by the EM algorithm in *discrimix*, are given by least squares in the RBF network and should be motivated from a geometrical point of view rather than from the principle of maximum likelihood. A final observation relates to the assumption of multivariate normality of the density function in each group. In procedure *discrimix* this density function appears explicitly in the update formula. On the contrary, calculus performed by a RBF network can be justified without assuming normality or any other particular distribution.

If the classification rules found by *discrimix* and RBF network are applied to the observations with unknown group membership, results are remarkably similar. Of the 199 unclassified specimens, 100 are classified as *M. multiplex*, 75 as *M. subterraneus*, and 24 observations are near the classification boundary, giving rise to considerable uncertainty in allocating them in one of the two groups both with *discrimix* and RBF network.

The CPU time is not a real problem, for the *Microtus* data, in any case. Running *Discrimix* takes about 10 s of CPU time on a 486PC, using the Gauss software (Airoldi et al., 1995). Running the principal components for the pre processing stage in the neural network set-up takes about 3 s of CPU time on a pentium PC, using the SPSS for Windows release 7.5. It takes less then 3 s for each run of the k-means cluster analysis and for the solution of the linear equations, to determine the network parameters. However, for very large data

sets, the computational costs are usually higher in *discrimix*. A further technical problem of *discrimix* if that the re-estimation formula must not deterministically converge, while convergence is demonstrated for the k-means algorithm.

5. Discussion

The idea of using RBF networks to process inclomplete data is not new (see Bishop, 1995, p. 184). This work is an attempt to explain and illustrate the use of RBF networks in situations where partially classified data sets occur and to show the differences between this methodology and other competitive methods which are often used in these situations. The goal of this paper is to make RBF networks more popular, since they appear to be rather less well known than the classical multi-layer perceptron, in the neural networks field, and than discriminant analysis and discriminant analysis with partially classified observations, in statistics. The application on the *Microtus* data demonstrates that RBF networks are a suitable methodological tool for ecological modelling, since the example is a rather typical case. The benefits of using RBF networks with partially classified observations is that no information is wasted and if very few observations are labelled the only alternative to estimate a classification rule is procedure *discrimix*. On the other hand, procedure *discrimix* is not a suitable tool in situations where the normality of the density function in each group is not verified and, for very large data sets, can lead to some technical problems in the solution of the equation systems. These problems are overcome in a RBF network in which the basis functions are trained with the k-means algorithm and the second-layer weights are given by least squares.

References

Anderson, T.W., 1984. An Introduction to Multivariate Statistical Analysis. Wiley, NY, p. 374.

Airoldi, J.P., Flury, B., Salvioni, M., 1995. Discrimination between two species of Microtus using both classified and unclassified observations'. J. Theor. Biol. 177, 247–262.

Bishop, M.C., 1995. Neural Networks for Pattern Recognition. Clarendon Press, Oxford, UK, p. 482.

Broomhead, D.S., Lowe, D., 1988. Multi-variable functional interpolation and adaptive networks. Complex Syst. 2, 321–335.

Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. J. R. Stat. Soc., B 39 (1), 1–38.

Flury, B., 1997. A First Course in Multivariate Statistics. Springer-Verlag, NY, p. 713.

Hand, D.J., 1981. Discrimination and Classification. Wiley, NY, p. 218.

Krapp, F., 1982. *Microtus multiplex* (Fatio, 1905) Alpen-Kleinwühlmaus. In Niethammer, J. and Krapp, F., Handbuch der Säugetier Europas, Band 2/I, Nagetiere II, Akademische Verlagsgesellschaft, pp. 319–428.

McLachlan, G.J., Basford, K.E., 1988. Mixture Models: Inference and application to Clustering. Marcel Dekker, NY, p. 272.

Moody, J., Darken, C.J., 1988. Learning with localised receptive fields. In: Touretzky, D., Hinton, G., Sejnowsky, T. (Eds.), Proceedings of the 1988 Connectionist Models Summer School. Morgan and Kauffman, San Mateo, pp. 133–143.

Moody, J., Darken, C.J., 1989. Fast learning in networks of locally-tuned processing units. Neural Comput. 1 (2), 281–294.

Murray-Smith, R., 1994. A Local Model Network Approach to Nonlinear Modelling. Ph.D. Thesis, Department of Computer Science, University of Strathclyde, Glasgow, Scotland, Nov.1994, pp. 71–79.

Niethammer, J., 1982. *Microtus subterraneus* (de Sélys-Longchamps, 1836). In Niethammer, J. and Krapp, F., Handbuch der Säugetier Europas, Band 2/I, Nagetiere II, Akademische Verlagsgesellschaft, pp. 397–418.

O'Neill, T.J., 1978. Normal discrimination with unclassified observations. J. Am. Stat. Assoc. 73, 821–826.

Ripley, B.D., 1994. Neural networks and related methods for classification. J. R. Stat. Soc., B 56 (3), 409–456.

Specht, D.F., 1990. Probabilistic Neural Networks. Neural Networks 3, 110–118.

Stone, M., 1974. Cross-validatory choice and assessment of statistical predictor. J. R. Stat. Soc., B 36, 111–147.