Seabed classification using artificial neural networks and other nonparametric methods

Tom Kavli[1] Mats Carlin[1] Rune Madsen[2]

January 26, 1993

Abstract

Four nonparametric statistical classification methods are evaluated for the problem of segmentation and classification of the seabed based upon data recorded from a multibeam echo sounder. Two of the methods are the multilayered perceptron (MLP) and radial basis function (RBF) neural networks. The other two methods are polynomial regression models (PRM) and a B-spline modelling technique (ASMOD). The classification is based upon a set of features extracted from the strength of the echo recorded within a rectangular region (pixel) of the seabed. Feature extraction is not covered in this article, but we refer to another article on this subject in these proceedings.

Test results for two different geographical regions are presented, and the performance of the different classifiers is discussed.

1 Introduction

This work is part of a project that started in December 1990 with the aim of developing methods and algorithms for seabed classification based on sonar images of the seabed. Participating companies are Simrad Subsea A/S, Norwegian Computing Center (NR), and SINTEF SI (former Center for Industrial Research (SI)). The sonar images are all recorded with the Simrad EM 1000 multibeam echo sounder at 95 kHz. The extraction of features from the images used for the classification has been done by NR and is described elsewhere in these proceedings [Milvang *et al.*, 1993].

One of the main application areas for artificial neural networks (ANN) is pattern recognition and statistical classification, and ANN have been used for classification of sonar signals [Lippmann, 1987, Gorman and Sejnowski, 1988]. In the work described here two of the most common ANN paradigms are tested, namely the multilayered perceptron and the radial basis function neural networks. Two other nonparametric techniques are also evaluated. These are a B-spline based method named ASMOD and polynomial regression models.

All these methods are nonparametric in the sense that they do not assume any particular distributions of the data as e.g. normal distributions, but rather try to identify a general nonlinear function for the relative probabilities of the different classes. These models thus are more general in the sense that less assumptions are made on the statistical properties of the data. But this is gained at the cost of more complex model structures with a greater number of parameters. Hence, the danger of overfitting and loss of generalization when the classifiers are applied to new data may be increased. A thorough evaluation of the classifiers on new and independent data is thus of special importance for such models.

This article is organized as follows: Section 2 gives an overview of the data material used in this study. In section 3 a brief introduction to the different modelling paradigms is given, and the classification results obtained with the different methods are presented and discussed in section 4.

2 Description of the data

The data used in this project are logged from a Simrad EM 1000 sonar. Three series of data

[1] Tom Kavli and Mats Carlin are with the Instrumentation Division of SINTEF SI, Box 124 Blindern, 0314 Oslo, Norway

[2] Rune Madsen is with the Norwegian Computing Center, Box 114 Blindern, 0314 Oslo, Norway

were available.

The first set was logged in October 1991 at different locations in the Oslo fjord. Because the October 1991 data set seems to cover more classes than the other data sets, the training data was taken from this set.

Two of the locations covered are assumed to be homogeneous regions of rock and mud respectively. A third and larger region is assumed to contain different mixtures of clay, silt and sand. Based on manual inspection of the sonar plots three homogeneous regions are identified from this third region which seems to represent three different seabed types.

Data from the resulting five homogeneous regions were used to define five classes: *Seabed type 1* to *Seabed type 5*. The classes may correspond to *rock*, *sand*, *mixture of sand and clay*, *clay* and *mud* respectively, but we stress that this is not confirmed. The training data consisted of approximately 200 pixels from each class.

The second data set was logged in March 1992, and covers a cruise from Horten, around Bastø and up to Mølen in the Oslo fjord. This cruise was supplied with ground examination at 10 locations along the route. The seabed consists mainly of a mixture of clay, silt and sand, but there are also some areas of mud and some spots of hard bottom.

The third data set is logged in an area located around Nidingen outside the west coast of Sweden. The area covers different seabed types and will in the near future be supplied with ground examination. This area is expected to contain all seabed types.

2.1 Feature extraction

A ping from the sonar is divided into 60 beams numbered from 1 to 60, where beam 1 and 60 are the leftmost and rightmost beams respectively. Each ping is divided into four sections consisting of beam numbers 2 - 4, 5 - 24, 37 - 56, and 57-59 respectively. The different sections of 20 consecutive pings are put together to form four rectangular pixels on the seabed [Milvang *et al.*, 1993].

For each pixel a number of features are computed based on the variations of the backscatter strength within the pixel. All the classifiers described here, except the ASMOD, use nine of the available features. These can in general terms be described as three distribution parameters: *mean value*, *standard deviation*, and *3rd order moment*, two quantiles: *quantile 0.5* and *quantile 0.8*, three features extracted from the power spectra: *PaceM(5)* D_{f1}, *PaceM(5)* D_{f2}, *PaceM(5)* D_{f3}, and a gray level co-occurrence contrast measure (*GLCM contrast*). For the ASMOD method three features were used, namely *quantile 0.8*, *PaceM(5)* D_{f2} and *GLCM contrast*. The features are described in more detail in [Milvang *et al.*, 1993].

3 Description of methods

The tested methods all aim at identifying a model which maps the feature vector $\mathbf{x} \in \Re^{N_i}$ into a probability vector $\mathbf{y} \in \Re^{N_j}$, where N_i and N_j are integers representing the number of features and number of classes respectively.

The models are trained by minimizing the sum of squared errors for the N_t patterns in a training set of known classes. I.e. the cost function

$$E = \sum_{n=1}^{N_t} \sum_{j=1}^{N_j} (t_{nj} - y_{nj})^2 \qquad (1)$$

is minimized, where t_n is the the target vector for pattern n containing one in element j if the true class for pattern n is j, and zeros in all other elements.

It can be shown that when the cost function (1) is minimized, then the elements of the output vector $\mathbf{y}_n$ for a given feature vector $\mathbf{x}_n$ represent an estimate for the relative probabilities for the feature vector to be measured from the corresponding classes [Richard and Lippmann, 1991]. The class with the largest corresponding element in $\mathbf{y}_n$ is thus the class of highest probability. The relative sizes of the elements may be used to indicate the confidence in the classification.

The most common artificial neural networks used for classification, and all networks used in this article have a multilayered feedforward architecture as shown in figure 1. The nodes (processing elements) are arranged in layers with an input layer, a hidden layer and an output layer. The input layer does no processing of the data, but is used to distribute the input variables to all nodes of the hidden layer. The nodes of the hidden layer generally represent nonlinear functions, $\Phi_q(\mathbf{x})$, mapping the

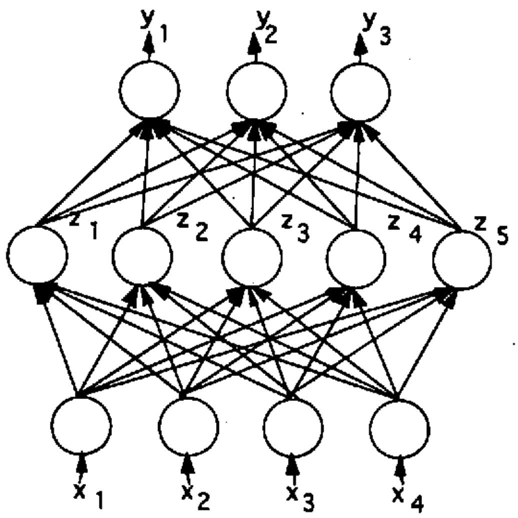

Figure 1: Structure of a multilayered neural network.

input vector into a vector of hidden variables $\mathbf{z} \in \Re^{N_z}$. The dimensionality N_z of the hidden vector corresponds to the number of hidden nodes, and may be smaller or larger than the number of input or output nodes. The output layer generally computes the output vector $\mathbf{y}$ as a nonlinear function of a weighted sum of the hidden variables.

The neural networks are trained by steepest descent methods, iteratively adjusting the model parameter vector θ in the direction giving the largest reduction of the cost function (1) for each new training pattern $(\mathbf{x}_n, \mathbf{t}_n)$. A momentum term is added to the steepest descent direction to avoid large fluctuations in the search path, resulting in a parameter update rule of the form

$$\Delta\theta^n = -\varepsilon\nabla_\theta E(\mathbf{x}_n, \mathbf{t}_n) + \eta\Delta\theta^{n-1}. \quad (2)$$

$\nabla_\theta E(\mathbf{x}_n, \mathbf{t}_n)$ is the gradient vector for the cost function (1) with respect to the parameter vector θ, and $\Delta\theta^n$ and $\Delta\theta^{n-1}$ are the incremental parameter update vectors for this and the previous training patterns respectively. ε and η are gain constants for the gradient and momentum terms respectively. The convergence may, for many paradigms, be quite slow. The training may also for some paradigms be trapped in local minima, so that repeated training may be required to find a global minimum.

3.1 Multi Layered Perceptron Neural Network (MLP)

One of the most commonly used neural networks is the multilayered perceptron networks (MLP) [Hecht-Nielsen, 1990, Beale and Jackson, 1990]. With one hidden layer the MLP has an architecture as shown in figure 1. The nodes of the hidden and output layers have all identical transfer functions given by

$$\psi = g(\sum_k w_k\xi_k + w_0)$$

where w_k are adjustable weighing coefficients for the input variables ξ_k received from the input or hidden nodes, w_0 is a bias term, and ψ is the output of the node transmitted to the output nodes or the output vector $\mathbf{y}$.

$g(\cdot)$ is a sigmoid function defined as $g(x) = (1 - e^{x/T})^{-1}$, where T specifies the steepness of the function.

The number of nodes in the input and output layers are determined by the dimension of the input and output vectors, whereas the number of hidden nodes must be manually chosen based on experience and by experimenting. Too few nodes gives too little flexibility in the model, while too many causes problems with overfitting. Practical experience has shown that problems of the type described her typically requires from 3 to 10 hidden nodes. The results reported here were obtained with a network of nine input nodes, 15 hidden nodes and 5 output nodes. The training gains were set to $\varepsilon = 0.3$ and $\eta = 0.4$. The training was run 2000 times through the training set of 996 training patterns. Training of one model took approximately 20 minutes on a large DEC 5000 workstation. We used the public domain program package 'Aspirin' [1] for the tests. All nine features listed in section 2 was used in the input vector.

3.2 Radial Basis Function Networks (RBF)

Radial Basis Function Networks (RBF) have been proposed by a number of researchers for multivariate statistical classification

[1]©1992 by Russel Leighton and the MITRE Corporation.

[Moody and Darken, 1988]. The RBF networks use radial transfer functions in the hidden layer. Different radial functions can be used. In the work described here Gaussian functions with variable and independent standard deviations σ_i^q for each input dimension i and function q are used:

$$\Phi_i(\mathbf{x}) = e^{-\frac{1}{2}\sum_i \frac{(x_i-\mu_i^q)^2}{\sigma_i^{q^2}}} . \quad (3)$$

The center positions μ_q and standard deviations σ_q of each radial basis function as well as the weights of the output layer w_q are all trained with the gradient descent algorithm (2). For the output nodes a linear transfer function was used, with clipping of the output to the range [0,1]. A detailed description of the parameter update rules can be found in [Carlin, 1991].

The Hierarchical Self-Organized Learning learning scheme (HSOL) proposed by [Lee and Kil, 1991] was used. This algorithm does incremental recruitement of new hidden nodes (RBFs) until a satisfactory result is reached, or some stop criterion is met. The stop criterion can typically be that the generalization on an independent test data set starts to degrade.The HSOL algorithm was slightly modified to yield better initialization of new hidden nodes[Carlin, 1991].

An important feature of this scheme is that it starts learning global features of the mapping based on a small initial number of hidden nodes, and proceeds with learning more detailed information by increasing the number of nodes. The final solution will typically use several RBFs for modelling the probability functions of each class, allowing arbitrary and not necessarily normal distributions to be modelled.

The main drawback of this learning scheme is that the performance of the classifier may depend on the choice of several parameters values that need to be specified. Also, as for the MLP networks, the training is slow and may be trapped in local minima.

An RBF network was trained using all 9 features listed in section 2. The number of hidden nodes was limited upwards to 15, and all 15 nodes were recruited in the final model. The training gains were set to $\epsilon = 0.000004$ and $\eta = 0.25$.The training set of 996 patterns were run through 2000 times. Training of one model took approximately 10 hours on a 486 PC.

3.3 Spline modelling (ASMOD)

The ASMOD (Adaptive Spline Modelling of Observation Data) scheme uses B-splines for its internal representation [Kavli, 1992].

Spline models use a grid partitioning of the input domain, making a set of hypercubes that fill up the input domain. A polynomial function is defined for every hypercube in such a way that the intersection between the different hypercubes are smooth. The resulting global function is a piecewise polynomial function with special merits with respect to modelling general multidimensional and nonlinear functions.

The ASMOD models can also be mapped into the architecture in figure 1 by using B-spline basis functions in the hidden layer. The set of basis functions are completely defined by the selected partitioning of the input space. The basis functions hence contain no parameters that need to be trained, leaving only the weight coefficients of the output layer to be determined during training. Since linear output nodes are used, the weights can be analytically determined by standard least square fitting methods. The results obtained here however, are obtained with an iterative training algorithm equivalent to (2). The momentum term was not used ($\eta = 0$). Since the optimization problem is linear and there is no local minima, the training converges fast to the global minimum (typically 10 - 20 seconds on a Macintosh).

One of the important properties of the ASMOD scheme is that it incorporates a method for successive refinement of the model. Starting with a coarse partitioning of the input domain, the partitioning is incrementally refined and adapted tu the structure of the learned functions. The resulting model thus has an automatically determined structure which is determined by the problem, allowing arbitrary simple or complex functions to be modelled.

The feature used for the ASMOD models were the quantile 0.8, PaceM (5) D_{f2}, and GLCM Contrast.

3.4 Polynomial Regression Model (PRM)

Polynomial models were identified for each of the five classes. The models for each class had the form given in equation (4), with variable polynomial degree (linear, quadratic, and cubic, including the corresponding cross products)

$$y_j = u_j + \sum_i v_{ji} x_i + \sum_{i \le k} w_{jik} x_i x_k + \cdots \quad (4)$$

Standard linear regression techniques were used to find the weight coefficients u, v, w, etc. These are given with a linear system that may be solved exactly. This method has the advantage over neural networks that it is very fast to train and we are always sure to find the global minimum of the error function (1).

Several models with different polynomial degrees and variable number of feature variables were identified. With too many features or too high polynomial degree a more "noisy" (inhomogeneous) classification of the seabed was observed. This was most likely due to overfitting to the training data. The results presented here use linear, quadratic and cross product dependencies of all nine feature variables.

4 Results

Method	Features	Train data	Test data
MLP-15	9	94 %	70 %
MLP-5	9	90 %	68 %
RBF	9	86 %	68 %
ASMOD	3	82 %	64 %
PRM	9	83 %	64 %
Gauss	3	84 %	- %
KNN	3	86 %	- %

Table 1: Correct classification rates for the training data and a test data set with the different methods.

Table 1 summarize the rate of correct classification of the training data and a test data set using the different methods. We used a training set consisting of 996 patterns and a test set consisting of 1331 patterns. For comparison results with a Gaussian classifier and a k-nearest neighbour classifier based on the same training data are included. The test data are taken from the same data files as the training data (October 1991), and from relatively homogeneous regions similar to but not overlapping the regions used for training. The training and test data are therefore quite similar, and an evaluation of the methods based upon the test data may be susceptible to unrecognized overfitting of the models. MLP models with 15 and 5 hidden nodes were trained, both giving good results on the training and test data. But, due to the above mentioned problems a ranking of the different methods is difficult based on this information.

A more reliable evaluation can be done using completely independent data such as the Mars 1992 and November 1992 data sets. Unfortunately there exists at the present time only a very limited knowledge of the true seabed types of these regions, and an evaluation and comparison of the methods must be done by visual inspection of how consistent and homogeneous the seabed is classified when data from multiple passes over a geographical region is processed.

Figures 2 and 3 show such geographical plots for two regions generated by the four methods. Figure 2 shows the classification of data taken from three passes over the same region, with a different heading of the ship for each pass. In figure 3 the data are taken from 5 different passes over the plotted region, here with four different headings of the ship.

As can be seen, the ASMOD and the RBF network give very similar results and consistent classification for the different passes. The region plotted in figure 2 contains four of the five bottom types (all but mud). The only sample within this area (indicated by a dark cross) consisted of clay and silt. This area is classified as seabed type 3 which is assumed to correspond to a medium hard seabed type. Also the other bottom samples agrees well with the classification of these two methods.

Classification with the multilayered perceptron model and the polynomial regression model give a somewhat more "noisy" result. As can be seen these methods both include seabed type 5, and pixels from different passes over the same location may be classified as differently as type 1 and type 5 (rock and mud). Such extreme local variations seem very unlikely, and it is reasonable to interprete this as misclassifications.

The area plotted in figure 3 is by ASMOD and RBF consistently classified either as type 1 or type 2. Also in this area the MLP model mix up seabed type 1 and 5 within small areas. This assumed misclassification may either be due to overfitting of the models, or a problem with finding a good representation of the probability functions (improper interpolation or extrapolation).

5 Conclusions

Four methods for nonparametric statistical modelling has been tested for seabed classification. The methods are nonparametric in the sense that no particular distribution of the data is assumed. This is in contrast to e.g. Gaussian classifiers which assumes normal distributions. This gives a potential for the models to find more accurate models for the data distributions, and hence more accurate classification borders. But the increased flexibility may also cause problems due to overfitting to the training data.

One of the methods, namely multilayered perceptron neural networks (MLP) gave higher classification rates on the training data and a test set, but when evaluated on new and independent data this method seemed to give less consistent and less accurate classification. This was probably due to overfitting or extrapolation/interpolation problems. Similar obervations were done for polynomial regression moldes. The other two methods, namely radial basis function networks (RBF) and a spline based method (ASMOD) gave similar and reasonable results on the independent data sets. They also gave results which were similar to results obtained with normal distribution (Gaussian) classifiers. An internal ranking of the RBF and the ASMOD classifiers is difficult to do based on results reported here. Ground truthing of one of the classified areas is expected to be available in the near future, enabling a more reliable evaluation of the methods.

The Gaussian, PRM and ASMOD models all have the advantage to the neural network paradigms that analytical solutions can be found for the model parameters, thus avoiding the well known problems of neural networks with long training times, sensitivity to a number of parameters, and the possibility of getting trapped in local minima during training.

References

[Beale and Jackson, 1990] R. Beale and T. Jackson. *Neural Computing - An introduction*. Adam Hilger, IOP, 1990.

[Carlin, 1991] Mats Carlin. Neural nets for empirical modelling. Master's thesis, Norwegian Institute of Technology (NTH), 1991.

[Gorman and Sejnowski, 1988] R. Paul Gorman and Terrence J. Sejnowski. Analysis of hidden units in a layered network trained to classifiy sonar targets. *Neural Networks*, 1:75 - 89, 1988.

[Hecht-Nielsen, 1990] Robert Hecht-Nielsen. *Neurocomputing*. Addison-Wesley Publishing Company, New York, 1990.

[Kavli, 1992] Tom Kavli. Asmod - An algorithm for Adaptive Spline Modelling of Observation Data. Tech. report, SINTEF SI, ISBN-No. 82-411-0410-3, 1992.

[Lee and Kil, 1991] Sukhan Lee and Rhee M. Kil. A gaussian potential function network with hierarchically sel-organizing learning. *Neural Networks*, 4:207 - 224, 1991.

[Lippmann, 1987] Richard P. Lippmann. An introduction to computing with neural nets. *IEEE ASSP Magazine*, pages 4 - 22, Apr 1987.

[Milvang *et al.*, 1993] Otto Milvang, Ragnar Bang Huseby, Katrine Weisten, and Anne Solberg. Feature extraction from backscatter sonar data. In *Int. conference on Acoustic Classification and Mapping of the Seabed*, Bath University, England, 1993.

[Moody and Darken, 1988] John Moody and Cristian J. Darken. Learning with localized receptive fields. In Touretzky et.al., editor, *Proceedings of the 1988 Connectionist Models Summer School*, pages 133 -143. Morgan-Kaufman, 1988.

[Richard and Lippmann, 1991] Michael D. Richard and Richard P. Lippmann. Neural network classifiers estimate bayesian a posteriori probabilities. *Neural Computation*, 3:461 - 483, 1991.

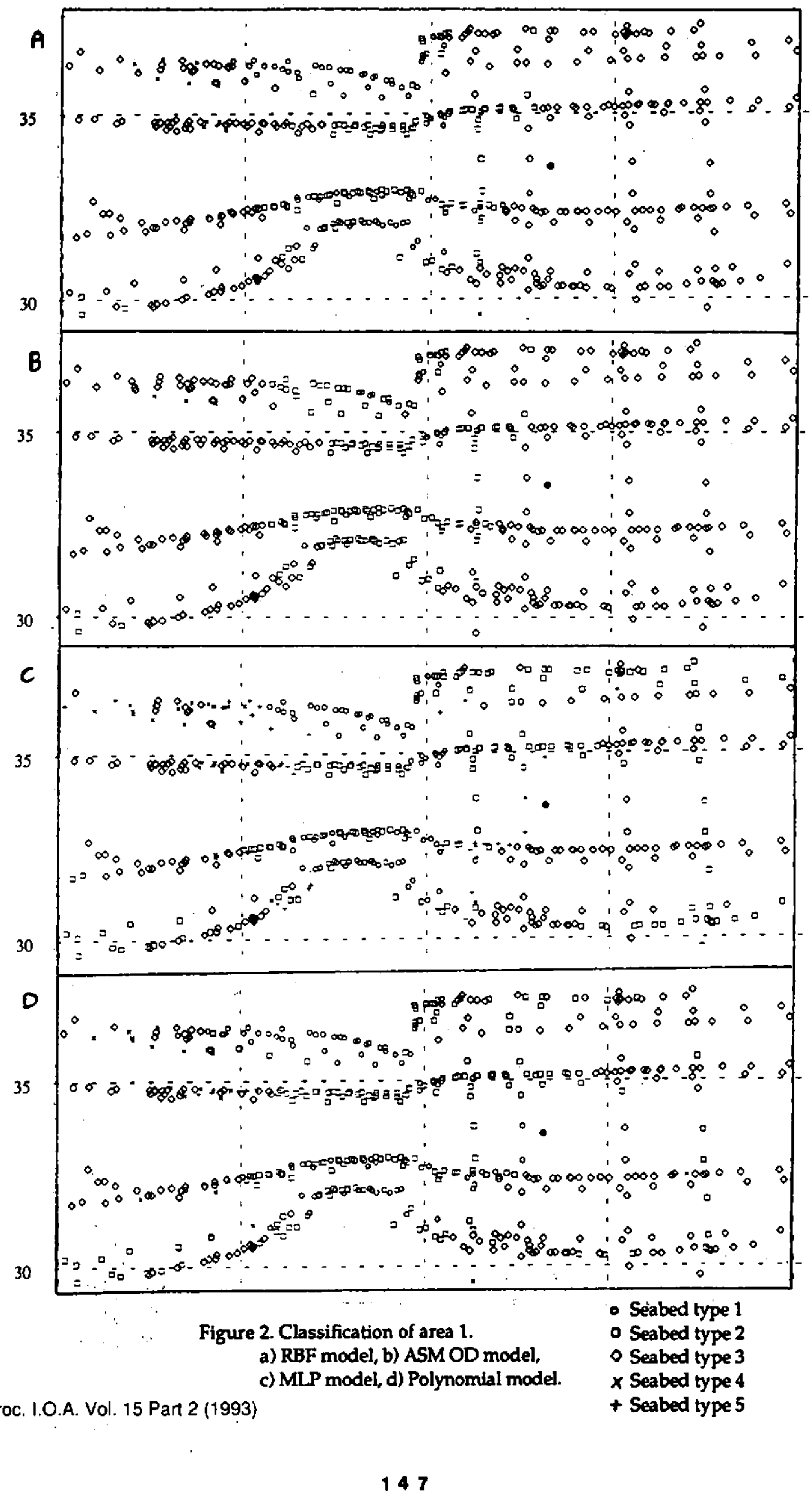

Figure 2. Classification of area 1.
a) RBF model, b) ASM OD model,
c) MLP model, d) Polynomial model.

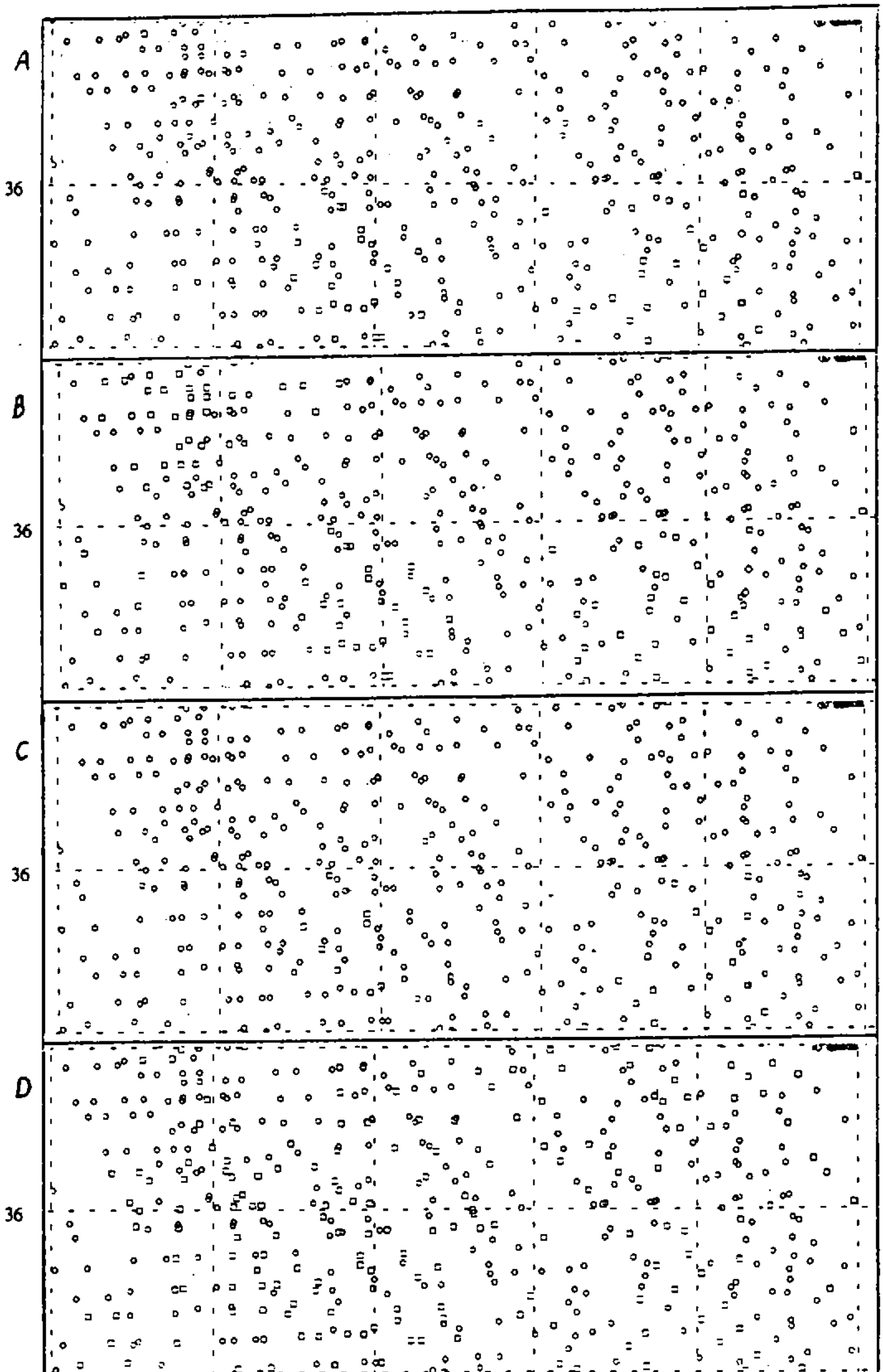

Figure 3. Classification of area 2.
a) RBF model, b) ASM OD model,
c) MLP model, d) Polynomial model.