Reprinted from

Symposium on Machine Processing of Remotely Sensed Data

June 3 - 5, 1975

The Laboratory for Applications of Remote Sensing

Purdue University
West Lafayette
Indiana

IEEE Catalog No.
75CH1009-0 -C

ADAPTIVE BAYES CLASSIFIERS FOR REMOTELY SENSED DATA*

H. S. Raulston, M. O. Pace, and
R. C. Gonzalez

The University of Tennessee
Knoxville, Tennessee

I. ABSTRACT

A new technique for the adaptive estimation of statistics necessary for Bayesian classification is developed. The basic approach to the adaptive estimation procedure consists of two steps: (1) an optimal stochastic approximation of the parameters of interest and (2) a projection of the parameters in time or space. Comparative results of a practical application are shown.

II. INTRODUCTION

This paper reports development and testing of an algorithm for a learning, adaptive statistical pattern classifier for remotely sensed data. This algorithm incorporates adaptive estimation of the required statistics into a Bayesian classifier. The results reported here are for Gaussian data in which the mean vector of each class may vary with time or position after the classifier is trained.

The cases reported were chosen to test the effects of attempting to adapt the estimates to changing statistics. The Gaussian density has been found appropriate in practive (Tou and Gonzalez, 1974; Crane, Malila, and Richardson, 1972). From another treatment of estimating Gaussian class densities (Keehan, 1965) it can be shown that the additional problem of estimating the covariance matrices can be handled by estimating elements of the correlation matrix separately from the elements of the mean vector, and then combining these to form the covariance matrix. Therefore, if the covariance matrices were also variable, the same adaptive algorithms used here could be applied to their estimation.

Several notable contributions have been made to the problem of estimating the parameters for a classifier where the class statistics vary with time or space. One such adaptive estimator (Kriegler, Marshall, Horwitz, and Gordon, 1972) gave larger weight to more recent samples, as specified by an empirically determined weighting parameter; the consequent "limited memory" made the resultant average more up-to-date. A somewhat similar adaptive estimation algorithm (Chien and Fu, 1969) "projected" the current estimate to the next step by adding the amount of anticipated change, and then combining it with the next data sample in a weighted average with weights chosen to minimize the mean square error. The algorithm developed in this paper consists of "refine" and "project" steps. This algorithm differs from CF (Chien-Fu algorithm) in the sense that the former (1) makes projections suitable for more complex variations with time or position, and (2) is arranged to operate as part of a Bayes classifier. It will be seen that in both these algorithms the "refine" step of combining previous estimate and new data is in the form of a stochastic approximation formulation (Wilde, 1964)

III. ESTIMATION ALGORITHMS

An algorithm is next discussed for adaptively estimating the class density function parameters as inputs for a Bayes classifier. In this discussion the algorithm will be applied to the problem of maintaining optimum current estimates of changing mean vector elements. As indicated in the previous section, the same algorithm could also be used to adaptively estimate the correlation matrix elements for multidimensional data and, from that and the mean vector estimates, an updated covariance matrix estimate could be provided to the classifier.

The following notation is used to describe the algorithm.

*Work supported by NASA contract NAS8-30878

θ_n = true value of mean at time (position) n

Y_n = data sample classified into a certain class at time (position) n.

X_n = "refined" estimate of θ_n made after classification #n provides new data sample Y_n

X_n^* = "projected" estimate of θ_{n+1} made at preceding time (position) n

$\overline{e_n^2}$ = mean square error $\overline{(X_n^* - \theta_{n+1})^2}$

γ_{n-1} = weight used in "refine" step to minimize $\overline{e_n^2}$.

The operation of the algorithm is as follows. At each time or position (time will henceforth denote time or position unless specified otherwise) the order of operations is, "refine", then "project" because once data has been classified, the classifier will next require an estimate at the next time, not the present. The refine step makes an optimum compromise between the present mean estimate X^*_{n-1} made at previous step n-1, and the present data sample Y_n. The "project" operation then provides the classifier an estimate of the mean when needed by the classifier, viz. the next time; it also provides the next stochastic approximation an input which is still unbiased by variation. Therefore, the "project" operation should remove (in a statistical sense) the estimation bias due to time variation, while the "refine" operation reduces the estimation scatter due to zero-mean sampling noise.

A typical sequence of events for classifying and subsequently estimating class means at the next time, assuming current mean estimates have been made, is as follows:

Step 1. The current data sample Y_n is classified into a particular class using current mean estimates for all classes.

Step 2. A "refined" estimate of the mean of the class chosen in Step 1 is computed by stochastic approximation as $\theta_n \approx X_n = X^*_{n-1} + \gamma_{n-1}(Y_n - X^*_{n-1})$. This step is omitted for all other classes for lack of data Y_n.

Step 3. A "projected" estimate of θ_{n+1}, X_n^*, may be made by transforming X_n according to the way the algorithm assumes θ is changing with n. If the change is due to time, this step is made for all classes; if the change is due to position within the current class being scanned, this step is performed only for that class chosen in Step 1 above.

Step 4. Increment n by 1 and return to Step 1.

The algorithm used in the test examples is called a "polynomial fit" or PF algorithm. The particular algorithm presented was derived to make nonlinear estimates of degree two and can be specified as follows:

The refine step (#2, preceding list) is denoted

$$X_n = X^*_{n-1} + \gamma_{n-1}(Y_n - X_{n-1}) \approx \theta_n \quad (1)$$

and the project step (#3, preceding list)

$$X_n^* = X_n + \hat{S} \approx \theta_{n+1} \quad (2)$$

where

$$\theta_n \equiv \text{true value at step } n$$

and

$$\hat{S} = \{[i(i+1) - j(j+1)]Y_n - [i(i+1)]Y_{n-j} + [j(j+1)]Y_{n-i}\}/ij(i-j) \approx \theta_{n+1} - \theta_n \quad (3)$$

and

$$\gamma_{n-1} = \frac{\overline{e_n^2} - K_1\sigma^2}{\overline{e_n^2} + \sigma^2} \quad (4)$$

and the estimate of mean square error for use in the calculation of γ_n is

$$\overline{e_{n+1}^2} \equiv \overline{(X_n^* - \theta_{n+1})^2} = \frac{\overline{e_n^2}\,\sigma^2}{\overline{e_n^2} + \sigma^2}(K_1+1)^2 + (K_2^2 + K_3^2)\sigma^2 \quad (5)$$

the required terms for error calculation being

$$K_2 \equiv -\frac{j+1}{i(i-j)} \quad (6)$$

and

$$K_3 \equiv \frac{i+1}{j(i-j)} \quad (7)$$

with K_1 defined as the sum of these two or

$$K_1 \equiv K_2 + K_3. \qquad (8)$$

Here the variance of the density function from which samples Y_n are drawn is represented as σ^2.

The "project" operation of Step 3 and equation (2), takes a form suitable for the manner in which the mean is assumed to vary with time while in the CF algorithm, "projection" is accomplished as $X_n^* = (1+1/n)X_n$. $\hat{S}$ of equation (3) is an estimate of anticipated change over the next time increment based in this PF algorithm on the assumption that the true value varies as a second degree polynomial, which is in turn estimated by the values Y_n, Y_{n-i}, and Y_{n-j}. Equation (4) gives the optimum weight γ_{n-1} to minimize e_{n+1}^2. The classifier then uses X_n^* as the best available value for θ_{n+1} for the next classification, at step n+1.

The ability of the CF and PF algorithms to "track" the varying mean of a Gaussian density has been tested by computer simulation. The data $\{Y_n\}$ were drawn from a unit-variance, one-dimensional Gaussian density with mean $9(n-50)^2/2500+1$ for n=1 to 100, and the algorithms produced up-to-date estimates of this mean. Ten statistically independent runs were made for $1 \leq n \leq 100$; the CF algorithm performance is shown in Figure 1, while the PF algorithm performance is shown in Figure 2. For the sake of comparison the performance shown in Figure 3 is that resulting from a least mean square error fit of a second degree curve to the set $\{Y_k\}$, k=1,2,...,100.

IV. TEST OF ADAPTIVE BAYES CLASSIFIER

An adaptive Bayes classifier is realized by incorporating within the ordinary Bayes classifier an estimation operation which uses the PF algorithm to process samples classified into each class and to optimally estimate the class density mean vector at the next classification time(s). As a test, different data sets were generated (Bryan and Tebbe, 1970), each having two equally likely data classes. These data sets are composed of patterns synthetically produced to simulate a 128×128 pixel frame of two dimensional Gaussian spectral scan data. Both data classes were generated having covariance matrices

$$C = \begin{bmatrix} 1 & .5 \\ .5 & 1 \end{bmatrix}.$$

A photograph depicting a display of the true spatial boundary (not to be confused with the Bayes decision surface or boundary) between the two classes is shown in Figure 4. The area to the left of this wedge shaped boundary is referred to as class one; similarly, the area to the right of the boundary is class two. The shortest and longest rows of class one data are 32 patterns and 96 patterns respectively; likewise for class two. The data was generated a row at a time from left to right, with both mean components of class one varying according to the relation

$$\frac{5}{1024}(N-32)^2 + 5$$

to the boundary (N is simply the position index having an initial value of zero at the left edge of the frame and incremented by one at each new position to the right). Class two data was generated for the remainder of each row. Both components of the mean vector for class two data were constants, independent of position. A plot of the class one means versus position is shown in Figure 5. Two data sets were generated each possessing a different (constant) mean vector. The mean vector associated with class two of data set one is

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

while that of data set two, class two is

$$\begin{bmatrix} 1.5 \\ 1.5 \end{bmatrix}.$$

Classification of the two data frames involved treating each of the 128 individual horizontal lines of each as a separate, independent classifier test. Of the two data sets, the second is the more difficult to classify because, for this set, a greater degree of overlap exists.

An application suggested was that the adaptive classifier might be useful in locating or defining spatial boundaries between data classes, such as that shown in Figure 4. Figures 6 and 7 show boundaries specified utilizing an ordinary Bayes classifier for data sets one and two respectively. The ordinary classifier did not have an adaptive estimator. Instead, it was continuously supplied the initial mean vectors for both classes in each of the two cases.

Figures 8 and 9 show the boundary specified for data sets one and two by an adaptive Bayes classifier using the CF algorithm to estimate the mean vectors of both classes.

Figures 10 and 11 show the boundary specified for the same two data sets by an adaptive Bayes classifier using a modified CF algorithm. The modification was to subtract the initial value of each mean vector, apply the CF algorithm, and add the initial value back. This improvement was discovered in separate tests of the tracking ability of the CF algorithm.

Figures 12 and 13 show the boundary specified for data sets one and two by an adaptive Bayes classifier using the CF algorithm with an additional modification. The modification consisted of restarting (as if n were 1 again) the algorithm if the following "confidence interval" conditions were violated:

$$\left| \frac{1}{n} \sum_{i=1}^{n} (X_i - Y_i) \right| < \frac{3\sigma}{\sqrt{n}} .$$

The erroneous boundary points, in Figure 13 appeared mostly at points where the restart was made, due to poor initial tracking when the algorithm is first started with little prior training.

Figures 14 and 15 show the boundary specified for the two data sets by an adaptive Bayes classifier using the PF algorithm to estimate both class mean vectors.

V. CONCLUSION

It has been found that the class of estimation algorithms represented by the PF procedure can be used to make a Bayes classifier adapt to changing class statistics. Modifications of the CF algorithm have also been found suitable.

The PF is a class of algorithms that predict well; the second degree was used as an example but algorithms of this class can also be derived (with different S and γ formulas) for tracking parameters that vary with time as an nth degree polynomial, n=1,2,3,... . The PF type algorithm can also track variations not of the exact polynomial form assumed because of the limited memory characteristic of the "refine" step.

Although two classes and two dimensional data were chosen for these tests, the techniques described are equally applicable to more complex situations. These techniques can also be applied to the estimation of other statistics required by the Bayes classifier.

VI. REFERENCES

J.K. Bryan, D.L. Tebbe. "Generation of Multivariate Gaussian Data," Dept. of Electrical Engineering, University of Missouri at Columbia, April, 1970.

Y.T. Chien, K.S. Fu. "Stochastic Learning of Time-Variant Parameters in Random Environment," IEEE Trans. Systems Science and Cybernetics SSC-5, 237-246, July, 196

R.B. Crane, W.A. Malila, W. Richardson. "Suitability of the Normal Density Assumption for Processing Multispectral Scanner Data," IEEE Trans. Geoscience Electronics GE-10, 158-165, 1972.

D.G. Keehn. "A Note on Learning for Gaussian Properties," IEEE Trans. Information Theory IT-11, 126-132, January, 1965.

F.J. Kriegler, R.E. Marshall, H. Horwitz, M. Gordon. "Adaptive Multispectral Recognition of Agricultural Crops," Proc. 8th International Symposium on Remote Sensing of the Environment, 833-849, 1972.

J.T. Tou, R.C. Gonzalez. Pattern Recognition Principles, Addison-Wesley Pub. Co., 1974.

D.J. Wilde. Optimum Seeking Methods, Prentice-Hall, 1964.

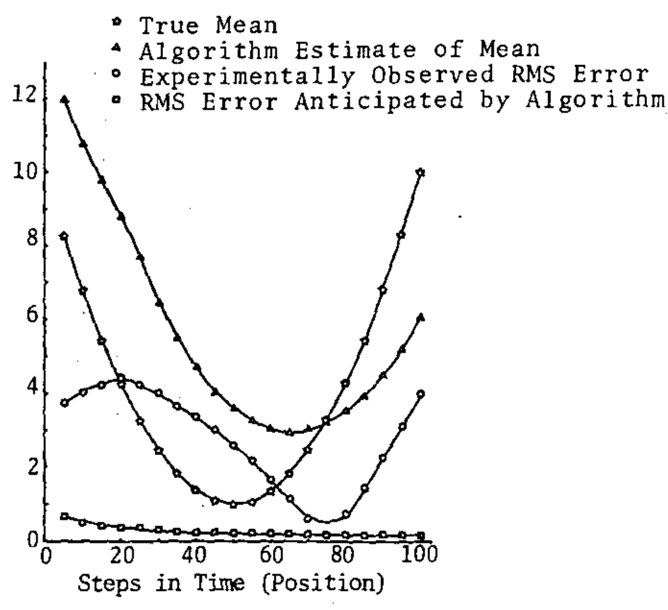

Figure 1. Performance of CF (Chien and Fu) algorithm.

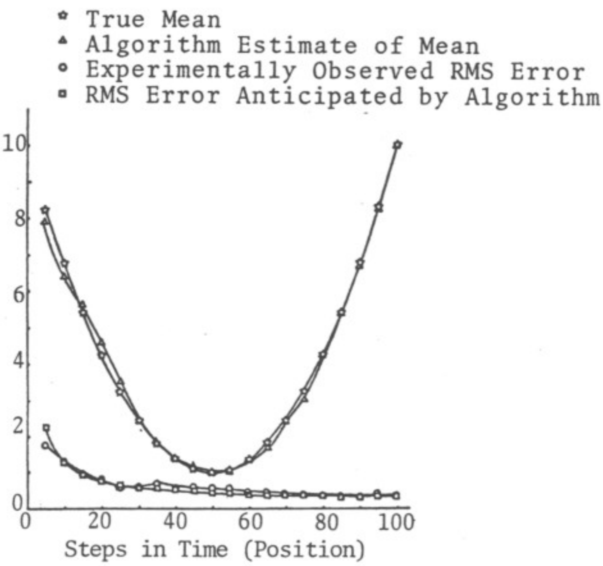

Figure 2. Performance of PF (polynomial fit) algorithm.

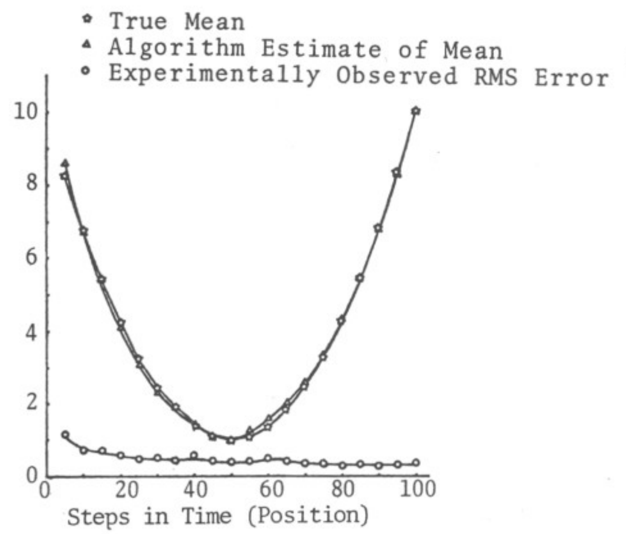

Figure 3. Performance of estimator operating as a least mean square error curve fit.

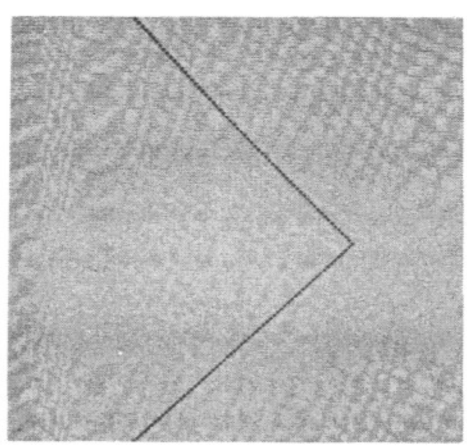

Figure 4. True spatial class boundary

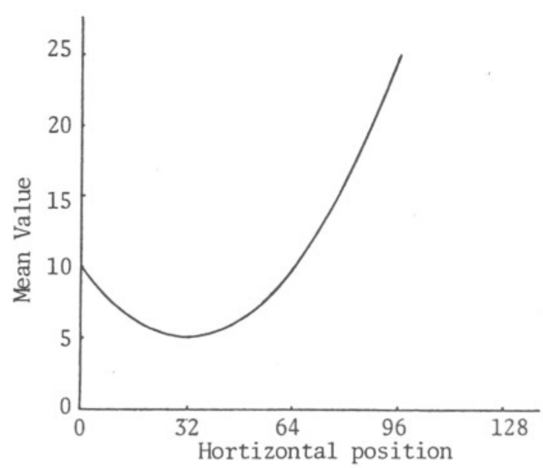

Figure 5. Both components of class two mean versus horizontal position.

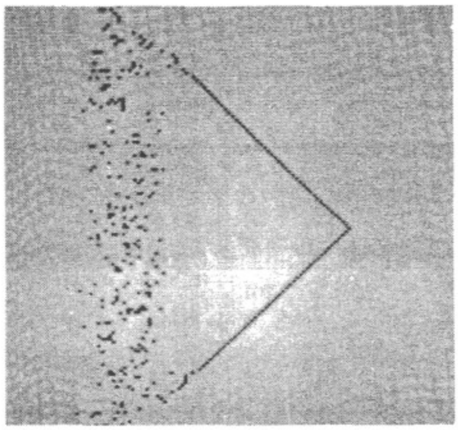

Figure 6. Spatial boundaries resulting from the application of an ordinary Bayes classifier to data set 1. Note the false boundaries.

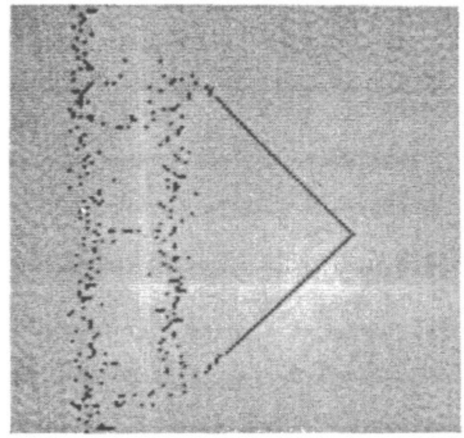

Figure 7. Spatial boundaries resulting from the application of an ordinary Bayes classifier to data set 2. Note the false boundaries.

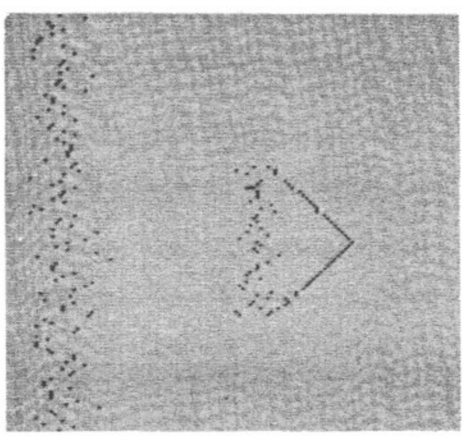

Figure 8. Spatial boundaries resulting from the application of an adaptive Bayes classifier using the CF algorithm to data set 1. Note the false boundaries.

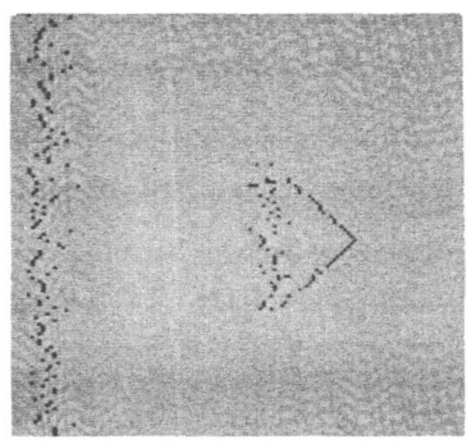

Figure 9. Spatial boundaries resulting from the application of an adaptive Bayes classifier using the CF algorithm to data set 2. Note the false boundaries.

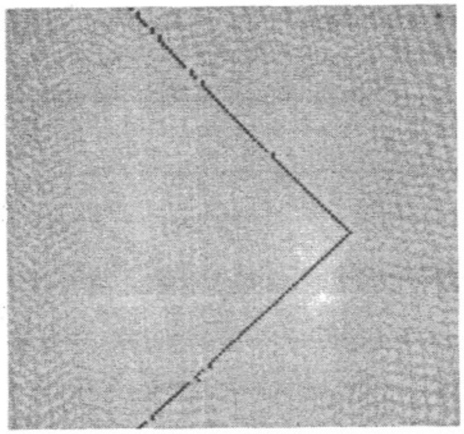

Figure 10. Spatial boundaries resulting from the application of an adaptive Bayes classifier using the modified CF algorithm to data set 1.

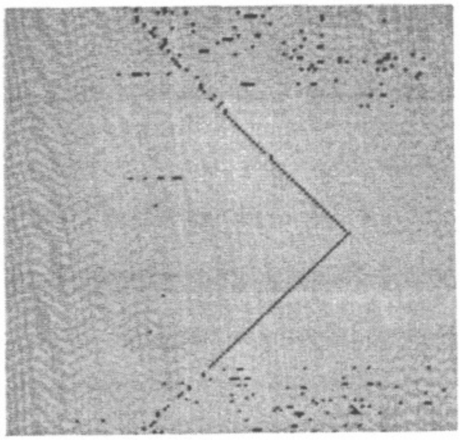

Figure 11. Spatial boundaries resulting from the application of an adaptive Bayes classifier using the modified CF algorithm to data set 2.

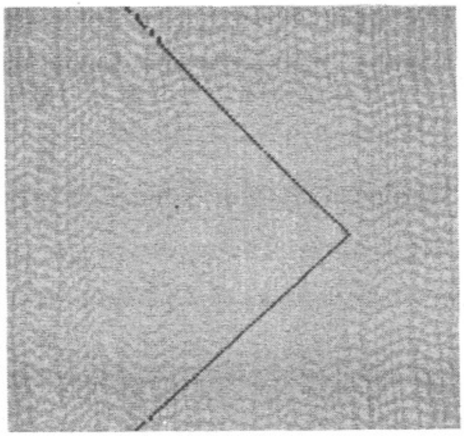

Figure 12. Spatial boundaries resulting from the application of an adaptive Bayes classifier using the modified CF algorithm and the divergence criterion to data set 1.

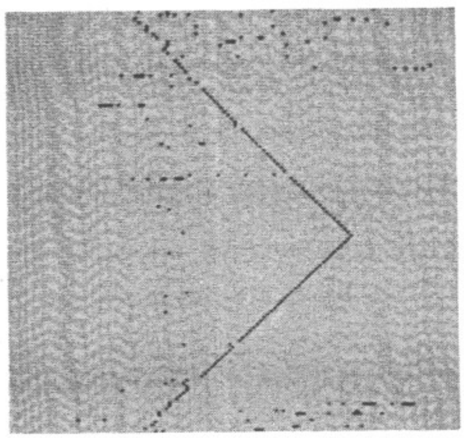

Figure 13. Spatial boundaries resulting from the application of an adaptive Bayes classifier using the modified CF algorithm and the divergence criterion to data set 2.

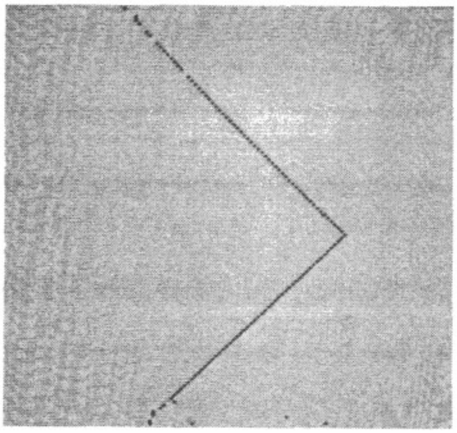

Figure 14. Spatial boundaries resulting from the application of an adaptive Bayes classifier using the PF algorithm to data set 1.

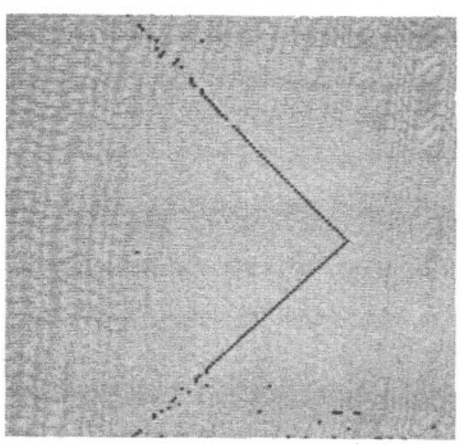

Figure 15. Spatial boundaries resulting from the application of an adaptive Bayes classifier using the PF algorithm to data set 2.