R. L. Mattson*
J. E. Dammann

A Technique for Determining and Coding Subclasses in Pattern Recognition Problems

Abstract: The problem of organizing and partitioning large amounts of data into classes such that all data in one class will have similar properties is well known in pattern recognition research. The first step in the process, a cluster finding technique, involves grouping a large amount of data into clusters which must be detected and encoded so that automatic pattern recognition can take place. This paper describes a method for detecting and coding clusters. The principal advantages of this technique are that clusters need not be known *a priori* and no matrix inversion is required.

Introduction

The purpose of this paper is to provide a technique for pattern classification problems by which a set of patterns is partitioned into a useful group of subsets, so that classification can be made with a network of linear threshold elements. The method assumes that clusters within the space are not known *a priori.*

A linear threshold element (LTE) is a device which has n inputs $(x_1, \cdots, x_n)$ and a single output, f (see Fig. 1). Each x_i is a real number and the value of f is either 0 or 1. The value of f is determined by forming a weighted sum of the input quantities and comparing this sum S to a threshold T. If the sum is greater than or equal to the threshold, $f = 1$; if the sum is less than the threshold, $f = 0$. The value of f is expressed mathematically in Eqs. (1):

$$S = \sum_{i=1}^{n} x_i w_i; \tag{1a}$$

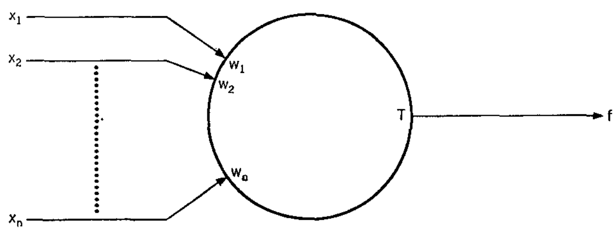
$$f = 1 \qquad \text{if } S \geq T; \tag{1b}$$

$$f = 0 \qquad \text{if } S < T. \tag{1c}$$

Many pattern classification systems use linear threshold elements to perform linear discriminations on the patterns of interest. In such an approach, n properties of patterns are measured and each property is assigned a real number, x_i. The n values $(x_1, \cdots, x_n)$ are used as a descriptor of the pattern. The pattern is classified into one of two classes by forming a weighted sum, S, of the values $(x_1, \cdots, x_n)$ and comparing this sum to a threshold value, T. If $S \geq T$ the pattern is classified into class 1; if $S < T$ the pattern is classified into class 0. Figure 2 shows an example for $n = 2$.

If patterns belong to more than two classes, two or more LTE's may be used in parallel to produce a binary code word for each class. For example, with four classes two threshold elements could be used for classification. If the outputs of both LTE's are 0, the code word 00 results and represents one of the four possible classes. The other combinations of outputs of the two LTE's, 01, 10, and 11 could be used to represent the other three classes of patterns. In Fig. 3 four patterns are shown as points in the Euclidean space with axes x_1 and x_2. If each point (pattern) belongs to a different class of patterns, the two threshold elements shown in Fig. 3 divide the patterns into four classes and assign the codes 00, 01, 10, and 11 to classes A, B, C, and D respectively. Linear threshold element

Figure 1 The symbolic representation of a linear threshold element (LTE).



* Assistant Professor, Electronics Laboratories, Stanford University.

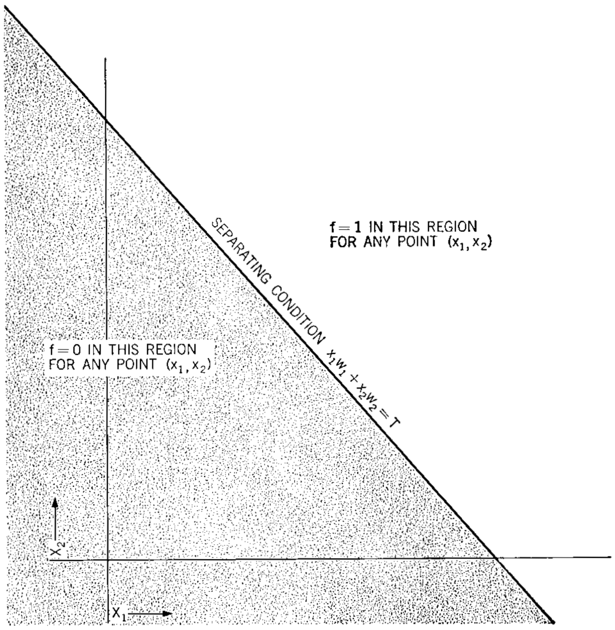


Figure 2 Euclidean space with axes x_1 and x_2.

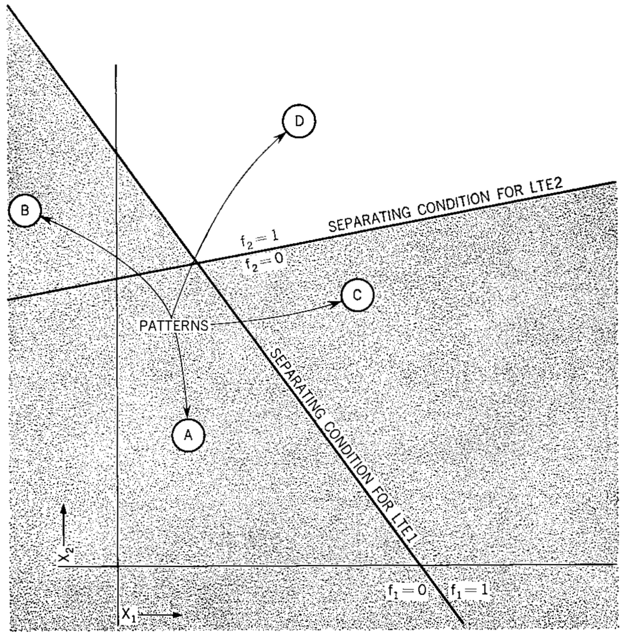


Figure 3 Four patterns classified into four classes with two linear threshold elements.

number 1 (LTE-1) classifies patterns A and B as 0, and patterns C and D as 1. LTE-2 classifies A and C as 0 and B and D as 1.

The code assigned to each of the classes is very important. If the codes 01, 10, 11, and 00 were assigned to the classes A, B, C, and D respectively, more than two LTE's would be required to classify the patterns. It is impossible for a single LTE to classify A and D as 0 and B and C as 1—two LTE's must be used (see Fig. 4).

In many pattern classification problems the existence of subclasses within a major class causes difficulty. For example, if the letters *B* and *b* are to be separated from the numbers 6 and 8, *B* and *b* would have to be put into one major class, "letters," and 6 and 8 into another

Figure 4 An impossible condition for a single separating line.

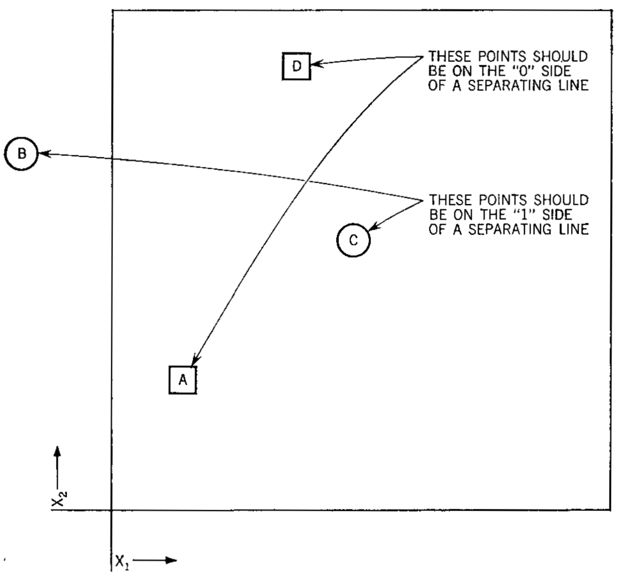
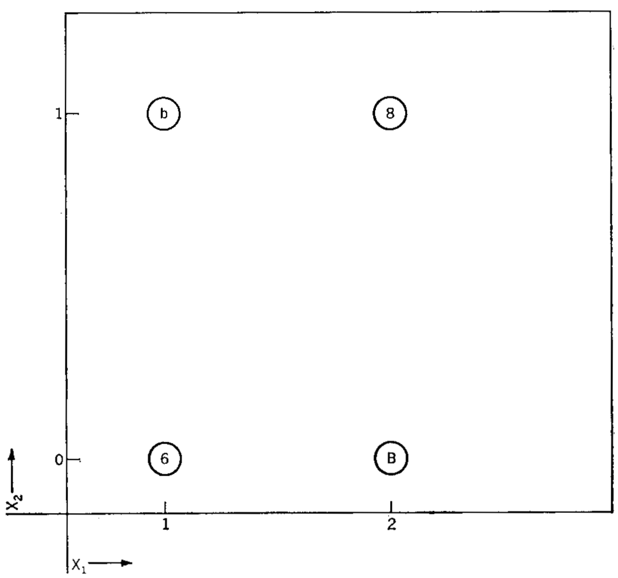


Figure 5 The Euclidean space for the "letter"-"not letter" problem.



major class, "not letters." Suppose that the properties to be used in this classification are x_1 = the number of loops in the character; $x_2 = 0$ if the lower loop is large, and $x_2 = 1$ if the lower loop is small. With these (x_1, x_2) quantities the four patterns belonging to the major classes "letters" and "not letters" would lie in Euclidean space shown in Fig. 5.

The existence of subclasses B and b within major class "letters" and subclasses 6 and 8 within major class "not letters" makes this problem unsolvable with a single LTE. However, if the subclasses are detected and given the proper binary code words, then this classification problem can be solved with two LTE's.

Early work in clustering techniques assumed that the clusters were known *a priori.* Fisher[1] in 1938 made contributions in this area. Other authors, particularly Bonner[2]; Hyvarinen[3]; Rogers and Tanimoto[4]; Firschein and Fischler[5]; Glazer[6]; Stark, Okajima, and Whipple[7]; Jakowitz, Shuey, and White[8]; and Ball and Hall[9] have considered the problem of unknown clusters. The method discussed in this paper differs from each of these approaches in that it does not require an *a priori* distance criterion to determine clusters of data points and does take into consideration the restrictions imposed by requiring a network of threshold elements to perform the desired classification. The present technique provides an approach to the decomposition of sample spaces and consequent coding of detected subsets using the transformation detailed in the next section. (Rao[10] discusses a technique for obtaining a transformation of this type based on a somewhat different rationale. Also, see Cooper and Cooper.[11])

A technique for determining clusters of patterns in an n-dimensional measurement space

• *Mathematical preliminaries*

The mathematical description of the threshold element is given in Eqs. (1). Each pattern $(x_{1k}, x_{2k}, \cdots, x_{nk})$ produces a sum S_k when substituted into Eq. (1a). Different patterns will typically produce different sums S. The distributions of S-values along an S-line can serve as an indication of how well the LTE will perform as a classifier. For example, if the distribution of S-values from major class "+" and major class "0" appears as shown in Fig. 6a, then the LTE which produced this distribution would probably do a good job of classification. If, however, the distribution of values on the S-line appeared as shown in Fig. 6b, the threshold element that produced this distribution would probably do a poor job of classification.

In many pattern recognition problems it is possible to describe mathematically the type of distribution that is desired along the S-line and then to calculate that set of weights which produces a distribution that is as close as possible to the desired distribution. For example, if two widely separated clusters of points, such as are shown in Fig. 7, are to be separated from each other by an LTE, a set of weights is desired which produces an S-line such as shown in Fig. 6a. What is desired, then, is to find a set of values w_i such that all points $(x_1, \cdots, x_n)$ in one cluster have values $S = \sum_{i=1}^{n} x_i w_i$ which are different from the values of $S = \sum_{i=1}^{n} x_i w_i$ produced by points $(x_1, \cdots, x_n)$ in the second cluster.[12] If such a set of values w_i can be found, the two separate clusters in $(x_1, \cdots, x_n)$-space can be detected by looking at the one-dimensional S-line and noting the two clusters of S-values.

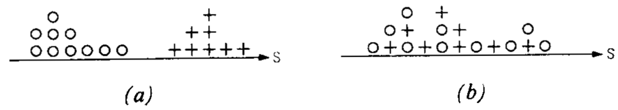


Figure 6 Two types of S-lines: (a) S-line for good separation; (b) S-line for poor separation.

Let there be p patterns in a set and denote the i^{th} component of the k^{th} pattern as x_{ik}. The value of S produced by the k^{th} pattern is

$$S_k = \sum_{i=1}^{n} x_{ik} w_i \ . \tag{2}$$

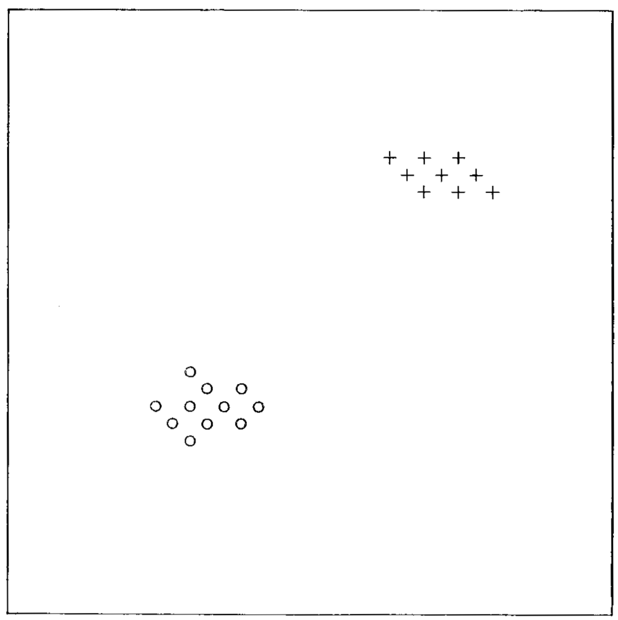
The average value of S produced by p patterns is

$$\bar{S} = \frac{1}{p} \sum_{k=1}^{p} S_k = \frac{1}{p} \sum_{k=1}^{p} \sum_{i=1}^{n} x_{ik} w_i$$

$$= \sum_{i=1}^{n} w_i \frac{1}{p} \sum_{k=1}^{p} x_{ik} = \sum_{i=1}^{n} \bar{x}_i w_i, \tag{3}$$

where $\bar{x}_i$ is the average value of the i^{th} component of the pattern vector $(x_1, \cdots, x_n)$.

Figure 7 Two clusters.



If two well separated, clustered subsets exist in the $(x_1, \cdots, x_n)$-space, the moment of inertia of the total set will be large around some axis due to the separation of the clusters. Similarly, the moment of inertia of the S-line distribution

$$M = \sum_{k=1}^{p} (S_k - \bar{S})^2 \tag{4}$$

will be large if two separated clusters of points exist in the space *and* the proper value of w_i's has been used to make the value of S different for each cluster.

Unfortunately the quantity M can become large without bound by allowing the absolute value of the weights (w_i) to become large without bound. If, however, the ratio

$$\lambda = \left[\sum_{k=1}^{p} (S_k - \bar{S})^2 \Big/ \sum_{i=1}^{n} w_i^2 \right] \tag{5}$$

is maximized, M will be large subject to the constraint that the sum of the squared values of w_i is small. Thus, what is desired is that set of weights which makes the value of λ as large as possible. This requirement can be simplified by noting that

$$M = \underline{\mathrm{w}}[A]\mathrm{w}], \tag{6}$$

$$\sum_{i=1}^{n} w_i^2 = \underline{\mathrm{w}}[I]\mathrm{w}], \tag{7}$$

where $\underline{\mathrm{w}}$ is a row vector $(w_1, \cdots, w_n)$, w] is a column vector $(w_1, \cdots, w_n)$, $[I]$ is the identity matrix, and $[A]$ is a matrix with elements

$$a_{ij} = \sum_{k=1}^{p} (x_{ik} - \bar{x}_i)(x_{jk} - \bar{x}_j), \tag{8}$$

with x_{ik} the i^{th} component of the k^{th} pattern, and $\bar{x}_i$ the average value of the i^{th} component. It should be noted that $[A]$ is proportional to the sample covariance matrix.

The quantity to be maximized[13] is then

$$\lambda = \frac{\underline{\mathrm{w}}[A]\mathrm{w}]}{\underline{\mathrm{w}}[I]\mathrm{w}]}. \tag{9}$$

Since $[A]$ is a symmetric matrix, differentiating λ with respect to each w_i yields the set of equations,

$$(\lambda[I] - [A])\mathrm{w}] = 0, \tag{10}$$

which can equal zero for non-zero w_i only if the determinant

$$|A - \lambda I| = 0. \tag{11}$$

The above determinant is a polynomial in λ equal to zero. Values of λ which are roots of this polynomial are eigenvalues of the matrix $[A]$. An eigenvector of the matrix $[A]$ is a vector such that

$$[A]\mathrm{w}] = \lambda \mathrm{w}]. \tag{12}$$

Thus, if w] is an eigenvector of $[A]$, then

$$\frac{\underline{\mathrm{w}}[A]\mathrm{w}]}{\underline{\mathrm{w}}[I]\mathrm{w}]} = \frac{\underline{\mathrm{w}}\,\lambda\,\mathrm{w}]}{\underline{\mathrm{w}}[I]\mathrm{w}]} = \frac{\lambda \sum w_i^2}{\sum w_i^2} = \lambda. \tag{13}$$

The eigenvector corresponding to the largest possible eigenvalue of the matrix $[A]$ is the set of weights which will maximize M while minimizing $\sum w_i^2$. (The direction of the eigenvector associated with the largest eigenvalue is in the direction of the major axis of the hyperellipsoid $\underline{\mathrm{w}}[A]\mathrm{w}] = C$. This is the direction of maximum dispersion.) This set of weights will yield a variable S such that two subclasses are displayed as two clusters of S values if it is possible to do so.

To determine the eigenvector corresponding to the largest eigenvalue several methods are known[14]. One possible method uses a result in matrix theory which states that any vector in the space spanned by the eigenvectors of $[A]$ can be written as a linear sum of the eigenvectors of the matrix $[A]$. Thus, if $\mathbf{E}_1, \mathbf{E}_2, \cdots, \mathbf{E}_n$ are the eigenvectors of a matrix $[A]$, a vector $\mathbf{V}]$[15] in the space spanned by these eigenvectors can be written as

$$\mathbf{V} = a_1\mathbf{E}_1 + a_2\mathbf{E}_2 + \cdots + a_n\mathbf{E}_n. \tag{14}$$

Multiplying $\mathbf{V}$ by $[A]^k$ results in

$$[A]^k\mathbf{V}] = a_1\lambda_1^k\mathbf{E}_1 + a_2\lambda_2^k\mathbf{E}_2 + \cdots + a_n\lambda_n^k\mathbf{E}_n. \tag{15}$$

If λ_1 is the largest eigenvalue of the matrix $[A]$, and k is large enough

$$[A]^k\mathbf{V}] \sim a_1\lambda_1^k\mathbf{E}_1. \tag{16}$$

This result indicates a method for approximating the eigenvector corresponding to the largest eigenvalue of a matrix $[A]$. The matrix $[A]$ is squared. This result is squared again. That result is squared again, and this process repeated k times. After k squaring operations, the matrix $[A]^{2^k}$ has been formed. Multiplying a vector $\mathbf{V}]$ (in the space spanned by the eigenvectors of $[A]$) by this matrix yields an approximation of the desired set of weight values.

$$[\mathrm{w}] = [A]^{2^k}\mathbf{V}] \sim a_1\lambda_1^{2^k}\mathbf{E}_1]. \tag{17}$$

Example of using the moment of inertia to determine clusters of points in measurement space

The following example is given to illustrate the technique of determining two clusters in (x_1, x_2, x_3, x_4)-space and to illustrate the determination of the eigenvector corresponding to the largest eigenvalue. Assume that six patterns $X = (x_1, x_2, x_3, x_4)$ from a major class have the following pattern vectors:

$$(0,\ 0,\ 0,\ 0),\quad (0,\ 0,\ 0,\ 1),\quad (1,\ 0,\ 1,\ 1)$$

$$(0,\ 1,\ 0,\ 0),\quad (1,\ 1,\ 1,\ 1),\quad (1,\ 1,\ 0,\ 1).$$

For these patterns, $\bar{x}_1 = 3/6$, $\bar{x}_2 = 3/6$, $\bar{x}_3 = 2/6$, $\bar{x}_4 = 4/6$. Also,

$$a_{11} = \tfrac{6}{4},\quad a_{22} = \tfrac{6}{4},\quad a_{33} = \tfrac{12}{9},\quad a_{44} = \tfrac{12}{9}$$

$$a_{12} = \tfrac{1}{2},\quad a_{13} = 1,\quad a_{14} = 1$$

$$a_{23} = 0 \quad a_{24} = 0$$

$$a_{34} = +\tfrac{2}{3}$$

$$[A] = \tfrac{1}{6}\begin{bmatrix} 9 & 3 & 6 & 6 \\ 3 & 9 & 0 & 0 \\ 6 & 0 & 8 & 4 \\ 6 & 0 & 4 & 8 \end{bmatrix}$$

To determine the eigenvector that corresponds to the largest eigenvalue of the matrix $[A]$, the technique of raising the matrix $[A]$ to a high power will be used. For this example, the vector $\mathbf{V}]$ is selected to be the "all one" vector, $\mathbf{V}] = (1, 1, 1, 1)$. A sequence of $\mathbf{W}]$ vectors is obtained from Eq. (17) by selecting $k = 0, 1, 2$, and 3. The values of the $\mathbf{W}]$ vector and the values of S are normalized so that the smallest value of S is zero, and the largest value of S is four. The sequence of vectors $\mathbf{W}]$ is given below, and the sequence of S-lines is given in Fig. 8.

$$\mathbf{W}_{-1} = \qquad \mathbf{V} = (1.0,\ 1.0,\ 1.0,\ 1.0)$$

$$\mathbf{W}_0 = [A]^1\mathbf{V} = (1.3,\ 0.7,\ 1.0,\ 1.0)$$

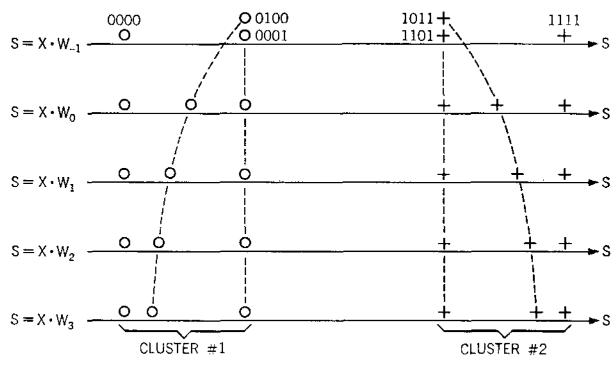
$$\mathbf{W}_1 = [A]^2\mathbf{V} = (1.4,\ 0.5,\ 1.0,\ 1.0)$$

$$\mathbf{W}_2 = [A]^4\mathbf{V} = (1.4,\ 0.4,\ 1.0,\ 1.0)$$

$$\mathbf{W}_3 = [A]^8\mathbf{V} = (1.5,\ 0.3,\ 1.1,\ 1.1).$$

In Fig. 8 each point represents a value of S determined by multiplying a pattern by a vector $\mathbf{W}$. Note the movement of the values of S as k is increased. See also the increase in separation of the two clusters of points.

Figure 8 S-line distributions for increasing values of k.



• *Example of using the moment of inertia to determine clusters of points and to code the points*

The following example is given to illustrate the technique of determining clusters in a two-dimensional measurement space and to illustrate how binary codes may be assigned to these clusters.

Consider the input space shown in Fig. 9. Here each axis represents an analog quantity x_1 or x_2 and o's indicate the location of points in two major classes, A and B. Class A has two subclasses, A_1 and A_2. If separation of the classes A and B by a straight line is all that is required, the line shown in Fig. 1 achieves the separation. However, it could be that the dotted line would be a better separating surface. The technique of maximizing the ratio of $M/\sum w_i^2 = \lambda$ will now be applied to the input space shown in Fig. 9.

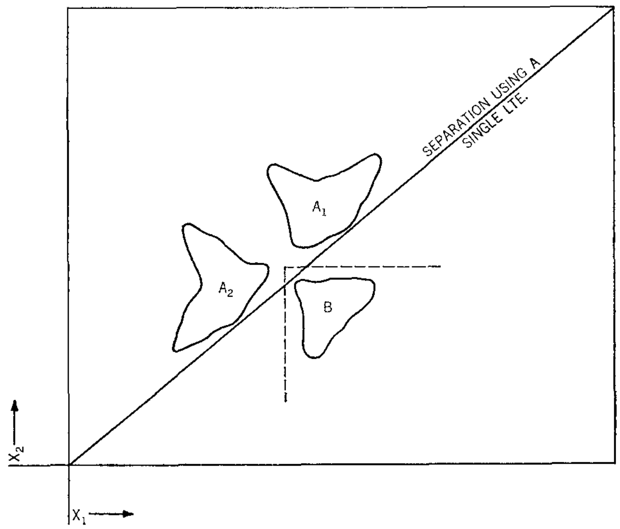
Using all data points in classes A and B, the eigenvector corresponding to the largest eigenvalue yields a distribution of points on an S-line as shown in Fig. 10a.

The best separation occurs between subclasses A_1 and A_2, and class B falls between these two subclasses. Thus, a single LTE, LTE-2, would adequately separate subclass A_1 from subclass A_2, but class B might be classified as either class A_1 or class A_2. To separate class B from subclasses A_1 and A_2, additional linear threshold elements are required.

To separate class B patterns from subclass A_1 patterns, patterns from these two groups are used (patterns from subclass A_2 are ignored) to maximize the λ ratio. The LTE (LTE-*b*) which maximizes this ratio produces the S-line shown in Fig. 10b and the separating line shown in Fig. 10c.

To separate class B patterns from subclass A_2 patterns, patterns from these two groups are used (patterns from

Figure 9 Measurement space for Example 2.



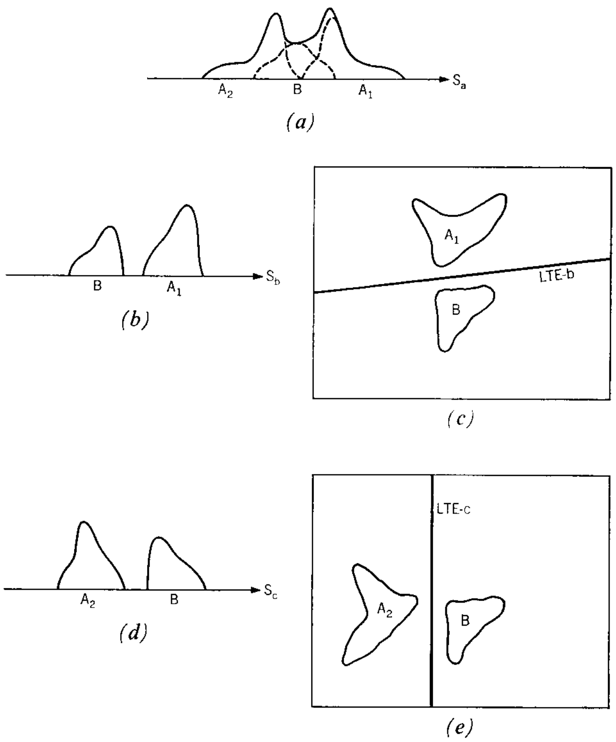


Figure 10 S-lines and separating lines for Example 2: (a) S-line determined by maximizing λ for all patterns; (b) S-line determined by maximizing λ for patterns in groups A_1 and B; (c) Separating line corresponding to the S-line of Fig. 10b; (d) S-line determined by maximizing for patterns in groups A_2 and B; (e) Separating line corresponding to the S-line of Fig. 10d.

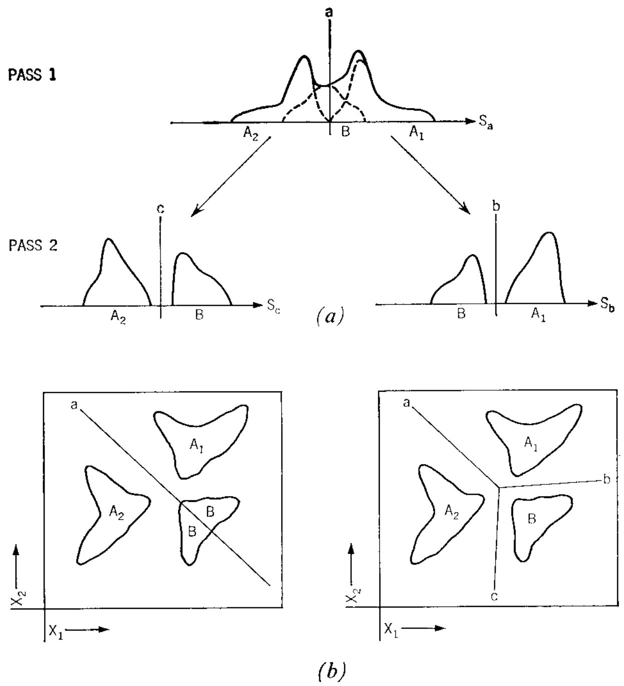


Figure 11 Sequence of S-line and separating lines for example 2: (a) Sequence of S-lines for Example 2; (b) Separating lines for Example 2.

subclass A_1 are ignored) to maximize the λ ratio. The LTE (LTE-c), which maximizes this ratio produces the S-line shown in Fig. 10d and the separating line shown in Fig. 10e. The sequence of S-lines and separating lines obtained in this process are reproduced in Figs. 11a and 11b.

The three linear threshold elements LTE-a, LTE-b, and LTE-c separate the three groups of patterns. LTE-a separates subclasses A_1 and A_2, but is of little value in classifying class B patterns. This fact is given in column a of Table 1. Likewise LTE-b and LTE-c are useful in separating class B from classes A_1 and A_2 respectively. Columns b and c of Table 1 show this fact.

Table 1 Coding of the classes

Group	Linear threshold element *a*	*b*	*c*
A_1	1	1	—
B	—	0	1
A_2	0	—	0

• *Example of analyzing and coding an unknown, multi-clustered space*

Consider the input space[16] shown in Fig. 12. This space has nine types of patterns, each being Gaussian-distributed with equal variances in the measurements x_1 and x_2.

In the analysis of this problem it will be assumed that the clusters are not known. If the space shown in Fig. 12 is considered to be analogous to a multi-dimensional digital space it will be understood that the clustering would not necessarily be obvious. The object of the analysis is to discover the clusters and to provide the weights and thresholds of an economical number of LTE's which will separate the clusters with good accuracy.

The first step in the analysis is to use all patterns available from the space to maximize the ratio λ. The S-line produced by this first pass is shown at the top of Fig. 13.

As a result of Pass 1, one set of points is seen to be separated from the rest of the points by an unpopulated segment of the S-line. This cluster may be one class or several bunched together by the particular view of the space afforded by this S-line. However, since it is markedly distinguished from the other points, a threshold is set for the middle value of the separation as indicated by the short bar shown in that part of the S-line in Fig. 13.

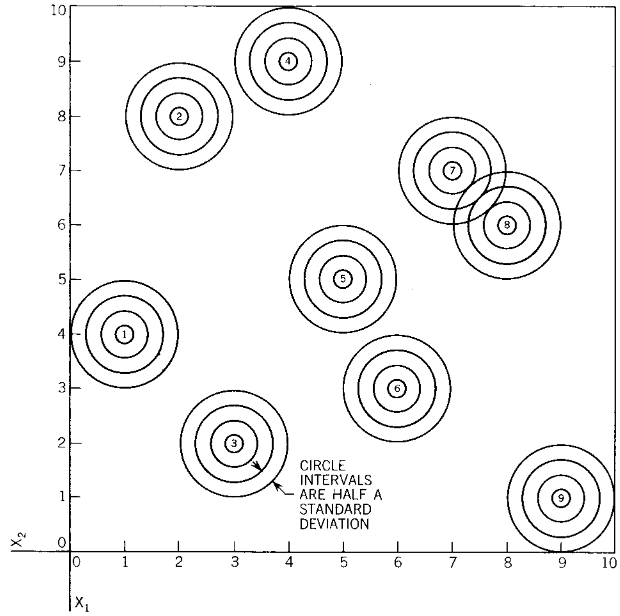


Figure 12 Nine two-dimensional Gaussian distributions with equal variances, Example 3.

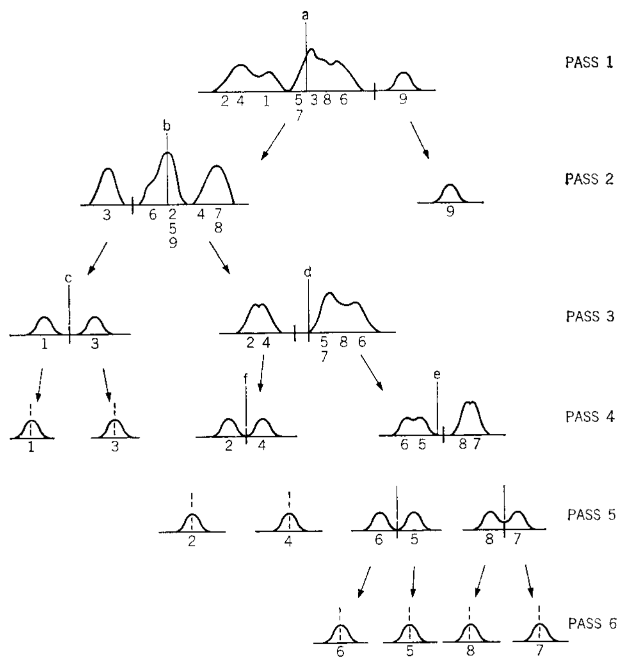


Figure 13 *S*-line distributions for Example 3.

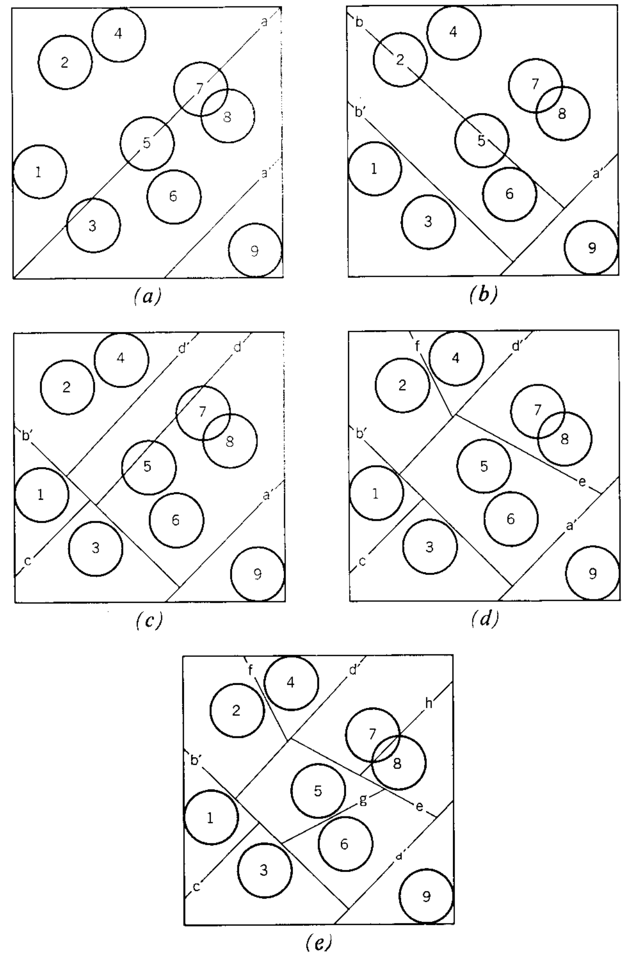


Figure 14 Separating lines for Example 3.

Table 2 Output code for Example 3

Classes	Linear threshold element *a*	*b*	*c*	*d*	*e*	*f*	*g*	*h*
1	0	1	1	x	x	x	x	x
2	0	0	x	1	x	0	x	x
3	0	1	0	x	x	x	x	x
4	0	0	x	1	x	1	x	x
5	0	0	x	0	0	x	0	x
6	0	0	x	0	0	x	1	x
7	0	0	x	0	1	x	x	0
8	0	0	x	0	1	x	x	1
9	1	x	x	x	x	x	x	x

There is another point on the *S*-line where a distinction could be made. This point is on the left side of the "a" *S*-line where the distribution drops to nearly zero. However, since this area is so narrow no distinction is made at this point. Distinctions made for zero threshold (marked "a") and the threshold to separate the first cluster (marked "a′") are shown in Fig. 14a.

Pass 2 is then run in two parts. All the patterns to the left of the threshold point are used together to maximize λ and determine a set of weights, and all the patterns to the right are similarly used. These runs result in the distribution shown on the second line in Fig. 13. The cuts through the space are shown in Fig. 14b. Since the cluster to the right of the threshold did not subdivide it is con-

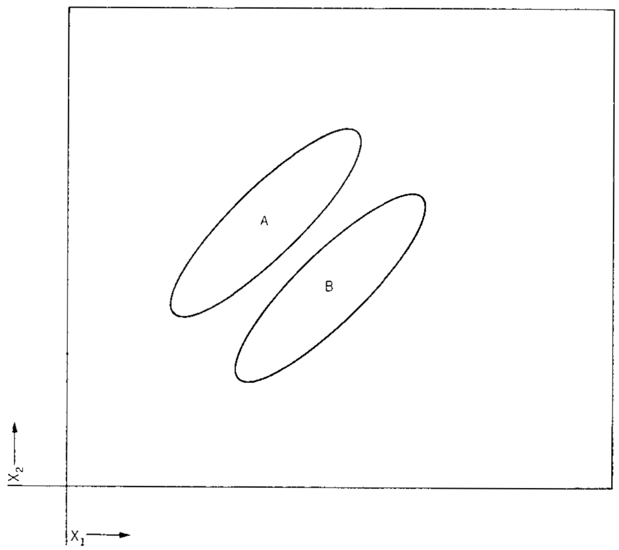


Figure 15 Input space.

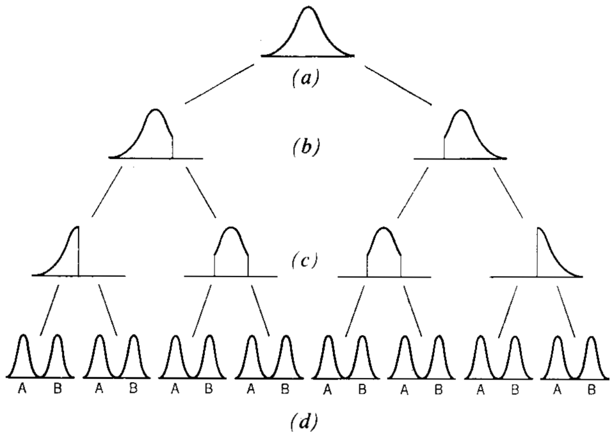


Figure 16 Analysis of the input space of Fig. 15 (a) pass 1; (b) pass 2; (c) pass 3; (d) pass 4.

sidered to be one cluster. For the left hand set of points a distinct separating area is again discovered and a threshold is set for this level. The two parts of this *S*-line are analyzed in pass three. The analysis continues until no cluster which subdivides is left. The results are shown in the remainder of Figs. 13 and 14.

The outcome of the analysis is that weights and thresholds are attained for a set of eight LTE's which will separate the clusters as shown in Table 2. This result would have been valuable even if it had been known that the space contained nine classes and if labeled sample points could have been obtained.

In cases where no definite separating area appears on the *S*-line a method of subdivision which involves selecting overlapping areas of the *S*-line is advantageous. In this way a class that is divided by separating the *S*-line at the point where the class falls may be retained intact in another branch of the analysis tree.

Conclusions

The proposed method of cluster analysis is obviously not applicable in every case. Configurations of clusters which do not yield readily to the method can be imagined. For instance, the input space shown in Fig. 15 gives unsatisfactory results as the analysis in Fig. 16 shows. In general, one would expect the method to achieve the best results when applied to input spaces where the inter-cluster dispersion is greater than the intra-cluster dispersion. The input space of Fig. 15 is not of this type.

A difficulty in applying the method is the determination of the point where the *S*-line should be separated when unpopulated stretches do not occur on it. This condition may mean that the data are not well clustered, which is a point of information in itself. Otherwise this choice must depend on the familiarity of the experimenter with the method and the input space of his problem. It is possible to devise a simple procedure which would search for unpopulated stretches of the *S*-line and continue the analysis automatically. Such a procedure might produce desirable results in cases where the data are well clustered. However, until further experience is acquired through the application of the procedure to practical pattern recognition problems, the method is viewed simply as a potential aid in finding and coding clusters for the designer of LTE systems.

The authors have achieved preliminary results applying the method to a 180-dimensional digital space with vectors generated by natural patterns. In this experiment, which will be reported, the eight clusters of the space were formed by measuring the spectra of eight spoken vowels. The eight clusters were successfully detected and coded with the premise that the space was initially unknown and multi-clustered.

References and footnotes

1. R. A. Fisher, "The Use of Multiple Measurements in Taxonomic Problems," *Annals of Eugenics* VII, 179–188 (1938).
2. R. E. Bonner, "A Logical Pattern Recognition Program," *IBM Journal* **6**, 353–360 (1962).
3. L. Hyvarinen, "Classification of Qualitative Data," *Nord. Tidskr. Info. Behandling* (*BIT*) **2**, No. 2, 83–89 (1962).
4. D. J. Rogers and T. T. Tanimoto, "A Computer Program for Classifying Plants," *Science* **132**, 115–118 (1960).

5. O. Firschein and M. A. Fischler, "Automatic Subclass Determination for Pattern Recognition Application," *IEEE-Trans. on Electronic Computers* EC-12, No. 2, 137–141 (1963).
6. E. M. Glazer, "Signal Detection by Adaptive Filters," *Tech. Report AF*-75, The Johns Hopkins University Radiation Laboratory, Baltimore, Md., April 1960.
7. O. Stark, M. Okajima, and G. H. Whipple, "Computer Pattern Recognition Techniques: Electrocardiographic Diagnosis," *Comm. ACM* 5, 527–531 (1962).
8. C. V. Jakowitz, R. L. Shuey, and G. M. White, "Adaptive Waveform Recognition," *Information Theory* (C. Cherry, Editor), Butterworths Publications, Washington, D. C.
9. G. H. Ball and D. J. Hall, "Some Fundamental Concepts and Synthesis Procedures for Pattern Recognition Preprocessors," presented at the *International Conference on Microwaves, Circuit Theory, and Information Theory*, Tokyo, Japan, Sept. 7–11, 1964.
10. C. Rao, *Advanced Statistical Methods in Biometric Research*, Wiley & Sons, New York (1952), p. 365.
11. D. B. Cooper and P. W. Cooper, "Nonsupervised Adaptive Signal Detection and Pattern Recognition," *Information and Control* 7, 416–444, (1964).
12. Fisher (Ref. 1) proposed a method for achieving this goal for problems where no subclasses are present.
13. This maximization is similar to the one proposed by Fisher, but the matrix inversion required by Fisher will not be required in this method.
14. "Calculation of Eigenvalues and Eigenvectors of *AZV-LAV* Where *Z* is Real and Symmetric, *V* is a Vector, *L* is a Scalar, and *A* is Real, Symmetric, and Positive Definite," *IBM SHARE Reference Library Program* 5.0.016.
15. In general one must select several different vectors V] to make certain that a vector is selected which has a component in the direction of the eigenvector with the largest eigenvalue.
16. This example is the same example that appears in Ref. 9.

Received January 15, 1965.