2

LARS Information Note 111572

Pattern Recognition: A Basis for Remote Sensing Data Analysis[1]

by

Philip H. Swain[2]

INTRODUCTION

Pattern recognition plays a central role in numerically oriented remote sensing systems. It provides an automatic procedure for deciding to which class any given ground resolution element should be assigned. The assignment is made in such a manner that on the average correct classification is achieved.

This information note describes briefly the theoretical basis for the pattern-recognition-oriented algorithms used in LARSYS, the multispectral data analysis software system developed by the Laboratory for Applications of Remote Sensing (LARS).

Figure 1 shows a model of a general pattern recognition system. In the LARS context the receptor or sensor is usually a multispectral scanner. For each ground resolution element the receptor produces n numbers or measurements corresponding to the n channels of the scanner. It is convenient to think of the n measurements as defining a point in n-dimensional Euclidean space which is referred to as the *measurement space*. Any particular measurement can be represented by the vector:

[1] Research reported here was supported by NASA Grant NGL 15-005-112.

[2] Program Leader for Data Processing and Analysis Research, LARS.

42

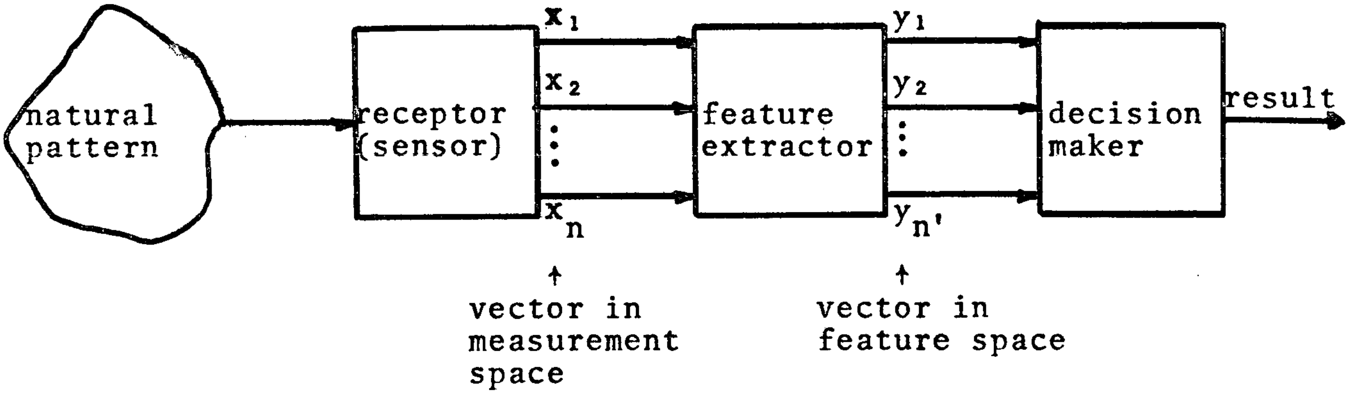


Figure 1. A Pattern Recognition System

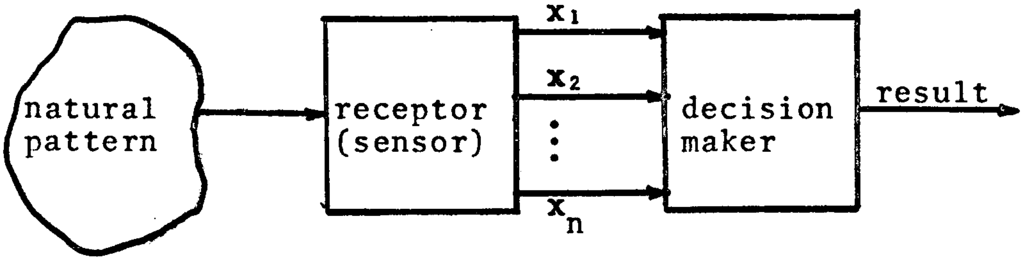


Figure 2. A Simplified Model of a Pattern Recognition System

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix} \qquad (1)$$

The feature extractor transforms the n-dimensional measurement vector into an n'-dimensional feature vector. In LARSYS, this consists simply of selecting a subset of the components of the measurement vector, but much more complex transformations are possible (see, for example, Ready et al, 1971).

The decision maker in Figure 1 performs calculations on the feature vectors presented to it and, based upon a decision rule, assigns the "unknown" data point to a particular class.

For the present, it will be sufficient to simplify the model to that shown in Figure 2. The vector X may subsequently be referred to as either a measurement vector ora feature vector.

DISCRIMINANT FUNCTIONS: QUANTIFYING THE DECISION PROCEDURE

Patterns arising in remote sensing problems exhibit some randomness due to the randomness of nature. As an example, one cannot in general expect the vector of measurements corresponding to a particular ground resolution element from one part of a wheat field to correspond exactly to the vector corresponding to a ground resolution element from another part of the field. Rather, vectors from the same class tend to form a "cloud" of points as shown in Figure 3. The job of the pattern classifier

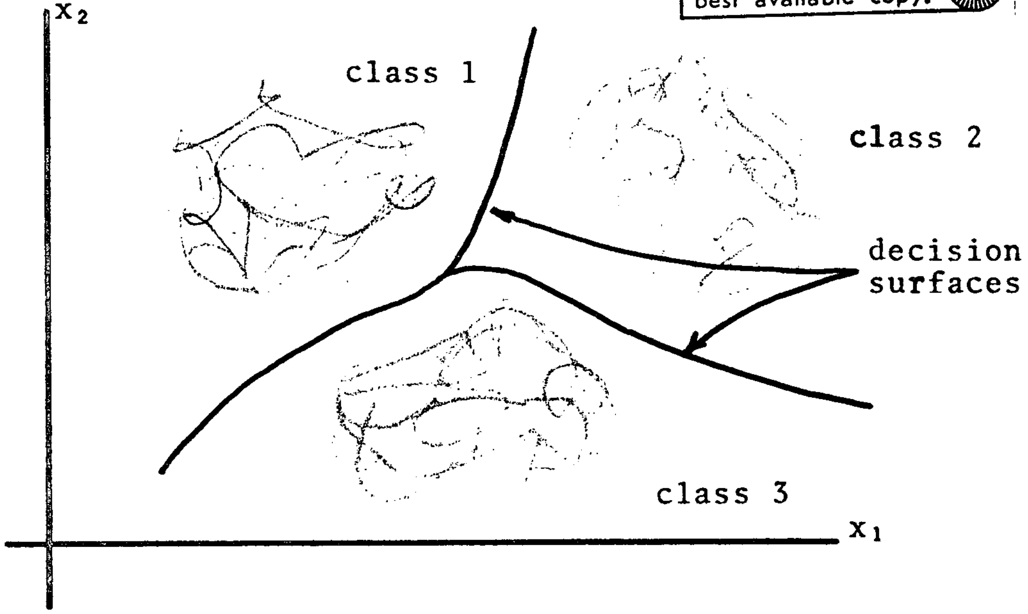


Figure 3. Decision Regions and Surfaces

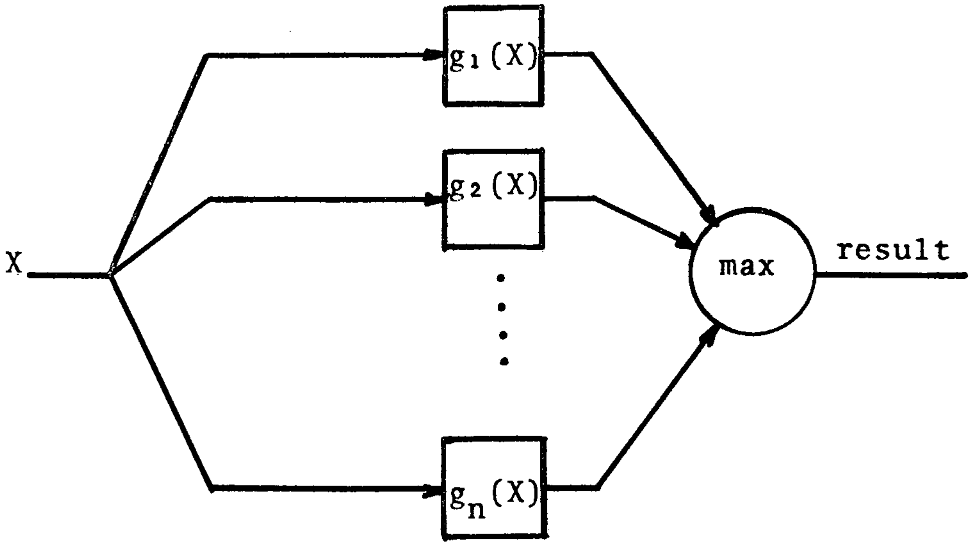


Figure 4. A Pattern Classifier Defined in Terms of Discriminant Functions

is to divide the feature space into *decision regions*, each region corresponding to a specific class. Any data point falling in a particular region is assigned to the class associated with that region. The surfaces separating the decision regions are known as *decision surfaces* Designing a pattern recognizer really boils down to devising a procedure for determining the decision surfaces so as to optimize some performance criterion, such as maximizing the frequency of correct classification.

These concepts can be put on a quantitative basis by introducing *discriminant functions*. Assume there are m pattern classes. Let $g_1(X)$, $g_2(X),\ldots,g_m(X)$ be scalar single-valued functions of X such that $g_i(X) > g_j(X)$ for all X in the region corresponding to the i^{th} class $(j \neq i)$. If the discriminant functions are continuous across the decision boundaries, the decision surfaces are given by equations of the form

$$g_i(X) - g_j(X) = 0. \tag{2}$$

A pattern classifier can then be represented by the block diagram of Figure 4.

By taking this approach the pattern classifier design problem is reduced to the problem of how to select the discriminant functions in an optimal fashion.

"TRAINING" THE CLASSIFIER

In some cases it is possible to select discriminant functions on the basis of theoretical considerations, experience, or perhaps even intuition. More commonly the discriminant functions are based upon a set of *training patterns.* Training patterns which are typical of those to be classified are "shown" to the classifier together with the identity of each pattern, and based on this information the classifier establishes its discriminant functions $g_i(X)$, i=1, 2, ..., m.

Example: Consider a two=dimensional, two-class problem in which the discriminant functions are assumed to have the form

$$\begin{aligned} g_1(X) &= a_{11}\, x_1 + a_{12}\, x_2 + b_1 \\ g_2(X) &= a_{21}\, x_1 + a_{22}\, x_2 + b_2 \end{aligned} \tag{3}$$

Then $g_1(X) - g_2(X) = 0$ is the equation of a straight line dividing the x_1, x_2 plane. Given a set of training patterns, how should the constants a_{11}, a_{12}, b_1, etc. be chosen? It can be proven that if the training patterns are indeed separable by a straight line, then the following procedure will converge (Nilsson, 1965):

> Initially select a's and b's arbitrarily. For example let
>
> $$\begin{aligned} a_{11} &= a_{12} = b_1 = 1 \\ a_{21} &= a_{22} = b_2 = -1 \end{aligned} \tag{4}$$
>
> Then take the first training pattern (say it is from ω_1, i.e., from class 1) and calculate $g_1(X)$ and $g_2(X)$. If $g_1(X) > g_2(X)$ the decision is correct; go on to the next training sample. If $g_1(X) < g_2(X)$ a wrong decision would

be made. In this case alter the coefficients so as to increase the discriminant function associated with the correct class and decrease the discriminant function associated with the incorrect class. If X is from ω_1 but ω_2 was decided, let

$$\begin{aligned} a'_{11} &= a_{11} + \alpha x_1 & \qquad a'_{21} &= a_{21} - \alpha x_1 \\ a'_{12} &= a_{12} + \alpha x_2 & \qquad a'_{22} &= a_{22} - \alpha x_2 \\ b'_1 &= b_1 + \alpha & \qquad b'_2 &= b_2 - \alpha \end{aligned} \tag{5}$$

where α is a convenient positive constant. If X is from ω_2 but ω_1 was decided, change the signs in Eq. (5) so as to increase g_2 and decrease g_1. Then go on to the next training pattern. Cycle through the training patterns until all are correctly classified.

Suggestion: Design and work out a numerical example to illustrate the training process described above. Assume two classes, two dimensions, and two training patterns per class.

THE STATISTICAL APPROACH

Remote sensing is typical of many practical applications of pattern recognition for which statistical methods are appropriate in the following respects:

- The data exhibit many incidental variations (noise) which tend to obsure differences between the pattern classes.
- There is often uncertainty, however small, concerning the true identity of the training patterns.
- The pattern classes of interest may actually overlap in the measurement space (may not always be discriminable),

suggesting the use of an approach which leads to decisions which are "most likely" correct.

Statistical pattern recognition techniques often make use of the probability density functions associated with the pattern classes (including the approach to be described here). However, the density functions are usually unknown and must be estimated from a set of training patterns. In some cases, the form of the density functions is assumed and only certain parameters associated with the functions are estimated. Such methods are called "parametric." Methods for which not even the form of the density functions is assumed are called "nonparametric." The parametric case requires more *a priori* knowledge or some basic assumptions regarding the nature of the patterns. The nonparametric case requires less initial knowledge and fewer assumptions but is generally more difficult to implement.

Let there be m classes characterized by the conditional probability density functions

$$p(X|\omega_i) \qquad\qquad i = 1, 2, \ldots, m. \qquad\qquad (6)$$

The function $p(X|\omega_i)$ gives the probability of occurrence of pattern X, given that X is in fact from class i.

An important assumption in the LARSYS algorithms is that the $p(X|\omega_i)$ are each multivariate gaussian (or normal) distributions. This is a parametric assumption which leads to a form of classifier which is relatively simple to implement. Under this assumption, a mean vector and covariance matrix are sufficient to characterize the probability distribution of any pattern class.

Returning to the problem of how to specify the discriminant functions, an approach based on statistical decision theory is taken. A set of loss functions is defined

$$\lambda(i|j) \quad i = 1, 2, \ldots, m; \; j = 1, 2, \ldots, m \qquad (7)$$

where $\lambda(i|j)$ is the loss (or cost) incurred when a pattern is classified into class i when it is actually from class j.

If the pattern classifier is designed so as to *minimize the average (expected) loss*, then the classifier is said to be *Bayes optimal*. This is the criterion to be used in specifying the classification algorithm.

For a given pattern X, the expected loss resulting from the decision $X\epsilon\omega_i$ is given by

$$L_X(i) = \sum_{j=1}^{m} \lambda(i|j)p(\omega_j|X) \qquad (8)$$

where $p(\omega_j|X)$ is the probability that a pattern X is from class j. Applying Bayes' rule, i.e.,

$$p(X,\omega_j) = p(X|\omega_j)p(\omega_j) = p(\omega_j|X)\; p(X) \qquad (9)$$

the expected loss can be written as

$$L_X(i) = \sum_{j=1}^{m} \lambda(i|j)p(X|\omega_j)p(\omega_j)/_{p(X)} \qquad (10)$$

where $p(\omega_j)$ is the *a priori* probability of ω_j.

Note that minimizing $L_X(i)$ with respect to i is the same as maximizing $-L_X(i)$. Thus a suitable set of discriminant functions is

$$g_i(X) = -L_X(i) \qquad i = 1, 2, \ldots, m. \tag{11}$$

A simple (and reasonable) loss function is

$$\begin{aligned} \lambda(i|j) &= 0 \qquad i = j \\ \lambda(i|j) &= 1 \qquad i \neq j \end{aligned} \tag{12}$$

(zero loss for correct classification, unit loss for any error). Then

$$g_i(X) = -\sum_{\substack{j=1 \\ j\neq i}}^{m} p(X|\omega_j)p(\omega_j)/p(X) \tag{13}$$

Here and at several points later in this paper it will be convenient to make use of the following fact: from any set of discriminant functions, another set of discriminant functions can be formed by taking the same monotonic function of each of the original discriminant functions. For example, if

$$g_i(X), \quad i = 1, 2, \ldots, m$$

is a set of discriminant functions, then so are the sets

$$g_i'(X) = g_i(X) + \text{constant} \qquad i = 1, 2, \ldots, m$$

and

$$g_i''(X) = \log[g_i(X)] \qquad i = 1, 2, \ldots, m.$$

Examining (13) note that p(X) is not a function of i so it is just as well to maximize

$$g_i'(X) = -\sum_{\substack{j=1 \\ j\neq 1}}^{m} p(X|\omega_j)p(\omega_j) = -\left[p(X)-p(X|\omega_i)p(\omega_i)\right]. \qquad (14)$$

But this is maximum if

$$g_i''(X) = p(X|\omega_i)p(\omega_i) \qquad (15)$$

is maximum. Thus, the decision rule is:

Decide

$X\varepsilon\omega_i$ if and only if

$$p(X|\omega_i)p(\omega_i) \geq p(X|\omega_j)p(\omega_j) \text{ for all } j* \qquad (16)$$

This is commonly referred to as the <u>maximum likelihood decision rule</u>.

Example: Consider two pairs of dice, one a standard pair and a second pair with two additional spots on each face. The probability functions associated with rolling a particular number with these dice are shown in Figure 5. Note how application of the decision rule (16) coincides with what you would do intuitively if the question were asked, "Given that a γ was rolled, decide which pair of dice was used." Let γ = 4, 7, 13. Note that p(standard dice) = p(augmented dice) = 0.5.

Consider the maximum likelihood discriminant function as it applies to remote sensing. The $p(\omega_i)$ represents the *a priori*

*Ties (the case of equality in (16)) may be arbitrarily decided by, say, always deciding $X\varepsilon\omega_i$ if $g_i(X) = g_j(X)$ and i>j.

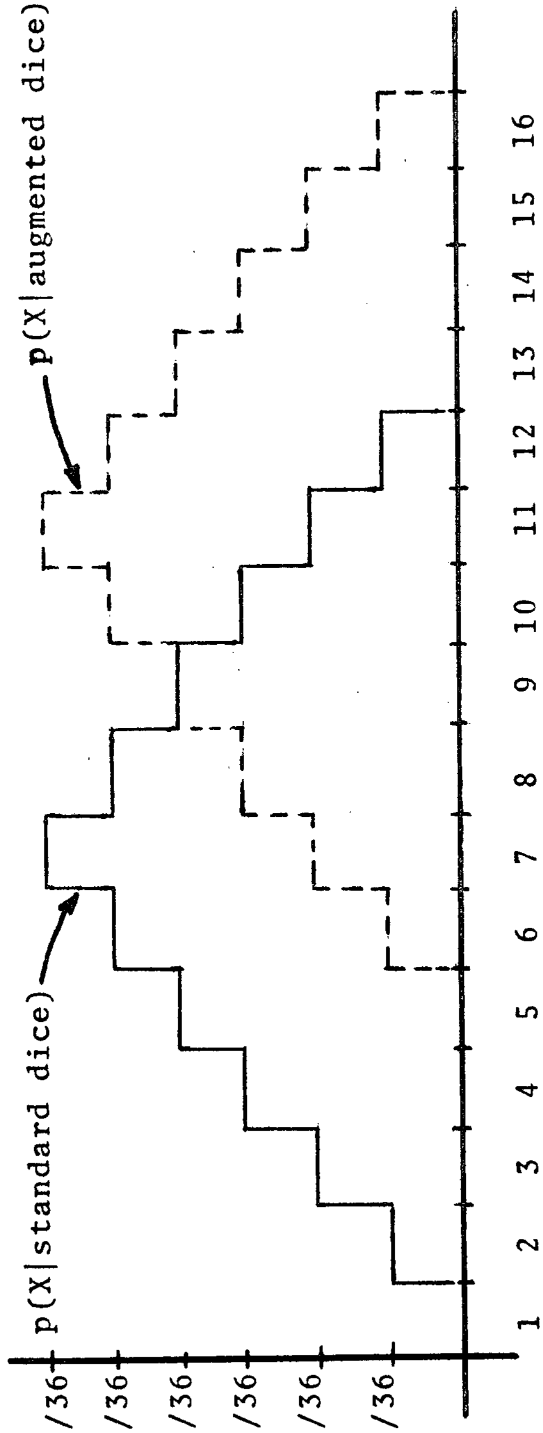


Figure 5. Probability Functions for Two Pairs of Dice

probability of the ith class. This can often be estimated. Taking agricultural crop types as an example, the $p(\omega_i)$ may be estimated from previous year yields, seed sales records or statistical reporting service information. The densities $p(X|\omega_i)$, on the otherhand, generally have to be estimated from training samples.

The assumption upon which the classification algorithms are based is that $p(X|\omega_i)$ is a multivariate gaussian probability density function. This basic assumption is supported by the following observations:

a) It is a reasonable model of the natural situation.

b) It results in a computationally simple (therefore inexpensive) discriminant function.

c) It works (or try it - you'll like it!).

Examining the maximum likelihood decision criterion in the one-dimensional gaussian case will serve both as a review of gaussian density functions and as a means of illustrating the principles of pattern classification. In this case (eg., one spectral channel)

$$p(x|\omega_i) = \frac{1}{(2\pi)^{1/2}\sigma_i} \exp\left[-1/2 \frac{(x-\mu_i)^2}{\sigma_i^2}\right] \qquad (18)$$

where $\mu_i = E[x]$ and $\sigma_i^2 = E[(x - \mu_i)^2]$ are the mean and variance for class i. In practice μ_i and $\sigma_i^2$are unknown and must be estimated from training samples. From statistical theory,

$$\hat{u}_i = m_i = \frac{1}{n_t} \sum_{j=1}^{n_t} x_j \tag{19}$$

$$\hat{\sigma}_i^2 = s_i^2 = \frac{1}{n_t - 1} \sum_{j=1}^{n_t} (x_j - m_i)^2 \tag{20}$$

(n_t = number of training patterns in class i)

are unbiased estimators of the mean and variance. Thus the estimated density function is

$$\hat{p}\,(x\,|\omega_i) = \frac{1}{(2\pi)^{1/2}\, s_i} \quad \exp\left[-1/2 \, \frac{(x-m_i)^2}{s_i^2} \right]. \tag{21}$$

Following the decision theory approach the discriminant function is

$$g_i(x) = \frac{p(\omega_i)}{(2\pi)^{1/2} s_i} \quad \exp\left[-1/2 \, \frac{(x-m_i)^2}{s_i^2} \right] \tag{22}$$

and since a monotonic function of a discriminant function may also be used as a discriminant function, we shall take the logarithm of the previous function to obtain

$$g_i'(x) = \log p(\omega_i) - 1/2 \log 2\pi - \log s_i - 1/2 \, \frac{(x-m_i)^2}{s_i^2} . \tag{23}$$

Since the constant term - 1/2 log 2π appears in all of the $g_{i,}$ (X) it may be dropped to yield

$$g_i''\,(x) = \log p\,(\omega_i) - \log s_i - 1/2 \, \frac{(x-m_i)^2}{s_i^2} . \tag{24}$$

Thus the decision rule becomes:

Decide X $\epsilon \omega_i$ if and only if

$$\log p(\omega_i) - \log s_i - 1/2 \; \frac{(x-m_i)^2}{s_i^{\,2}} \geq \log p(\omega_j) - \log s_j - 1/2 \; \frac{(x-m_j)^2}{s_j^{\,2}} \qquad (25)$$

The one dimensional case just described serves to illustrate the Bayes decision rule for gaussian statistics.

In the two dimensional case

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \qquad (26)$$

and

$$p(X|\omega_i) = \frac{1}{2\pi(\sigma_{i11}\ \sigma_{i22} - \sigma^2_{i12})^{1/2}} \cdot \qquad (27)$$

$$\exp\left\{ -\ 1/2 \; \frac{\dfrac{(x_1-\mu_{i1})^2}{\sigma_{i11}} - \dfrac{2\sigma_{i12}\ (x_1-\mu_{i1})(x_2-\mu_{i2})}{(\sigma_{i11} \cdot \sigma_{i22})^{1/2}} + \dfrac{(x_2-\mu_{i2})^2}{\sigma_{i22}}}{1 - \dfrac{\sigma^2_{i12}}{\sigma_{i11}\ \sigma_{i22}}} \right]$$

where

$$\mu_{i1} = E\ [x_1|\omega_i], \quad \mu_{i2} = E\ [x_2|\ \omega_i]$$

$$\sigma_{ijk} = E[(x_j-\mu_{ij})(x_k-\mu_{ik})|\omega_i] \qquad \begin{matrix} j,k = 1,2 \\ i = 1,2,\ldots R \end{matrix} \qquad (28)$$

This is a formidible expression, but by defining a mean vector and covariance matrix

$$U_i = \begin{bmatrix} \mu_{i1} \\ \mu_{i2} \end{bmatrix} \qquad (29)$$

$$\Sigma_i = \begin{bmatrix} \sigma_{i11} & \sigma_{i12} \\ \sigma_{i21} & \sigma_{i22} \end{bmatrix} \qquad (30)$$

the density can be rewritten in the simple form:

$$p(X|\omega_i) = \frac{1}{2\Pi|\Sigma_i|^{1/2}} \exp\left[-1/2\ (X-U_i)^T\ \Sigma_i^{-1}\ (X-U_i)\right] \qquad (31)$$

where $|\Sigma_i|$ is the determinant of Σ_i and $(X - U_i)^T$ is the transpose of $(X - U_i)$. The beauty of the matrix formulation is that it holds for n dimensions as well as for 2 dimensions. For the multivariate gaussian case, the maximum likelihood discriminant function is given by

$$g_i'(X) = p(X|\omega_i)p(\omega_i)$$

$$= p(\omega_i)\frac{1}{(2\pi)^{n/2}|\Sigma_i|^{1/2}} \exp\left[-\ 1/2(X-U_i)^T\ \Sigma_i^{-1}(X-U_i)\right] \qquad (32)$$

Taking the log and eliminating the constant term

$$g_i''(X) = \log p(\omega_i) - 1/2\ \log|\Sigma_i| - 1/2(X-U_i)^T\ \Sigma_i^{-1}(X-U_i) \qquad (33)$$

The corresponding decision rule is:

Decide $X\ \epsilon\omega_i$ if and only if

$$g_i''(X) \geq g_j''(X) \quad \text{all } i,j \qquad (34)$$

When U_i and Σ_i are not known, they must be estimated from training patterns. Denoting the estimates as $\hat{U}_i$ and $\hat{\Sigma}_i$ and dropping the subscripts indicating class to simplify the notation:

$$\hat{U} = M = \begin{bmatrix} m_1 \\ m_2 \\ \cdot \\ \cdot \\ \cdot \\ m_n \end{bmatrix} \quad \text{and } \hat{\Sigma} = S = \begin{bmatrix} s_{11} & s_{12} & \cdots & s_{1n} \\ s_{21} & s_{22} & \cdots & s_{2n} \\ \cdot & \cdot & & \\ \cdot & \cdot & & \\ \cdot & \cdot & & \\ s_{n1} & s_{n2} & \cdots & s_{nn} \end{bmatrix}. \quad (35)$$

Unbiased estimators are

$$m_j = \frac{1}{n_t} \sum_{\ell=1}^{n_t} x_{j\ell} \qquad j=1,2,\ldots n \qquad (36)$$

$$s_{jk} = \frac{1}{n_t - 1} \sum_{\ell=1}^{n_t} (x_{j\ell} - m_j)(x_{k\ell} - m_k) \qquad (37)$$

$$j=1,2,\ldots,n;\ k=1,2,\ldots,n$$

where n_t is the number of training patterns.

VECTOR CLASSIFICATION IN LARSYS

The classification algorithm currently in LARSYS is essentially the decision rule defined by Eq. (33) and (34), except that all class probabilities are assumed equal; i.e.,

$$p(\omega_1) = p(\omega_2) = \ldots = p(\omega_m) = \frac{1}{m}.$$

The required mean vectors and covariance matrices are computed from training patterns by the statistics processor. The classification processor computes the $g_i(X)$, $i=1,2,...m$ for every data vector in the area to be classified. For each vector the class decided and the value of the discriminant function computed for that class are written on magnetic tape for later use by the results display processor.

Inevitably there are points in the area classified which do not belong to *any* of the classes defined by the training samples. In agricultural settings such points might be from roads, fence lines, farmsteads, and the like. The classification procedure necessarily assigns these points to one of the training classes, but typically they may be expected to yield very small discriminant values. The later fact can be utilized to detect them, as will now be described.

Consider Figure 6. In this one-dimensional, two-class example, the points to be detected are those "not very much like" any of the training classes and therefore having a low probability of belonging to any of the training classes. Thus by "rejecting" or "thresholding" a very small percentage of the points *actually belonging* to the training classes, it is possible to reject a relatively large number of points not belonging to the training classes. This can be done simply by computing the probability density value associated with the data vector and "rejecting" the point if the value is below a user-specified threshold.

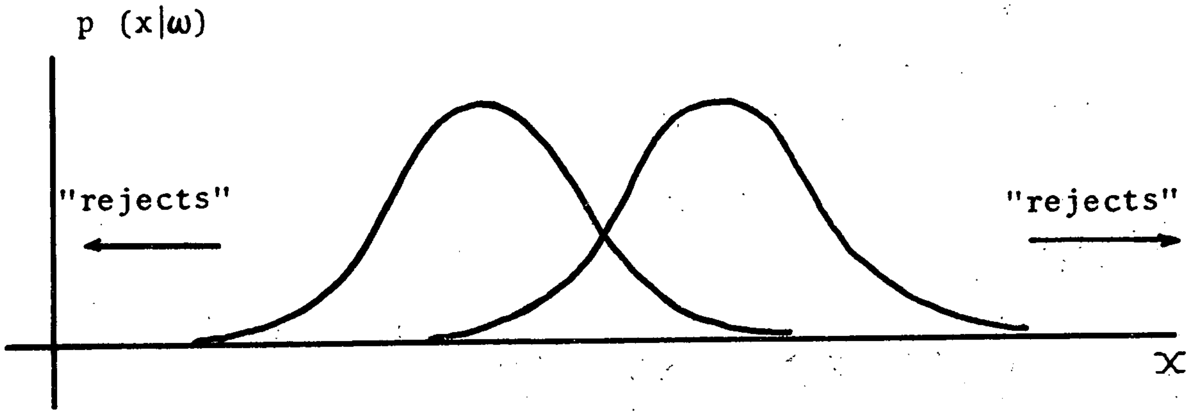
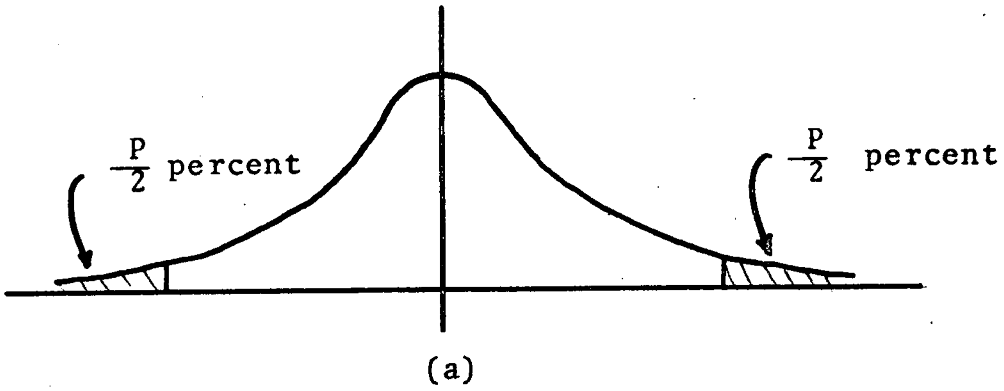


Figure 6. Points May Be "Rejected" as Unlike Any of the Defined Classes



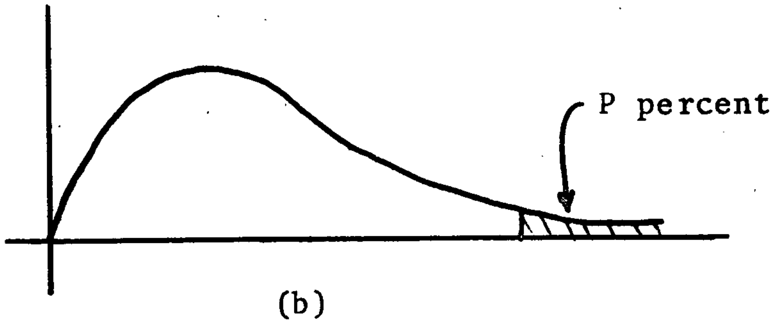


Figure 7. (a) P Percent of the Normal Distribution
(b) P Percent of the Chi-Square Distribution

But this can be accomplished just as well using the discriminant values stored as part of the classification result. If X is n-dimensional and normally distributed then the quadratic form

$$(X-U_i)^T \Sigma_i^{-1} (X-U_i) \tag{38}$$

has a chi-square distribution with n degrees of freedom ($C_n(\chi^2)$). Therefore to threshold, say, P percent of the normal distribution shown in Figure 7a, it is just as well to threshold P percent of the chi-square distribution of $(X - U_i)^T \Sigma_i^{-1}(X - U_i)$. This quadratic form is related to $g_i(X)$ in the following manner:

$$(X-U_i)^T\Sigma_i^{-1}(X-U_i) = -2g_i(X) + 2b_i \tag{39}$$

where

$$b_i = \log p(\omega_i) - 1/2 \log |\Sigma_i| \tag{40}$$

Thus, every point for which

$$-2g_i(X) + 2b_i > (\chi^2 \text{ for which } C_n(\chi^2) = P/100) \tag{41}$$

is rejected or thresholded. Note that a different threshold value may be applied to each class.

FEATURE SELECTION

Problem: Given a set of N features (eg., multispectral scanner channels), find a subset consisting of n channels which provides an optimal trade-off between classification costs (complexity and time for computation) and classification accuracy.

Ideally, one would like to solve this problem by computing the probability of misclassification associated with each n-feature subset and then selecting the one giving best performance. However, it is generally not feasible to perform the required computations. Even under the simplifying assumption of normal statistics, numerical integration is required which, in the multidimensional case, is impractical to carry out. To see this, consider that

$$\binom{N}{n} \triangleq \frac{N!}{n!(N-n)!} \qquad (42)$$

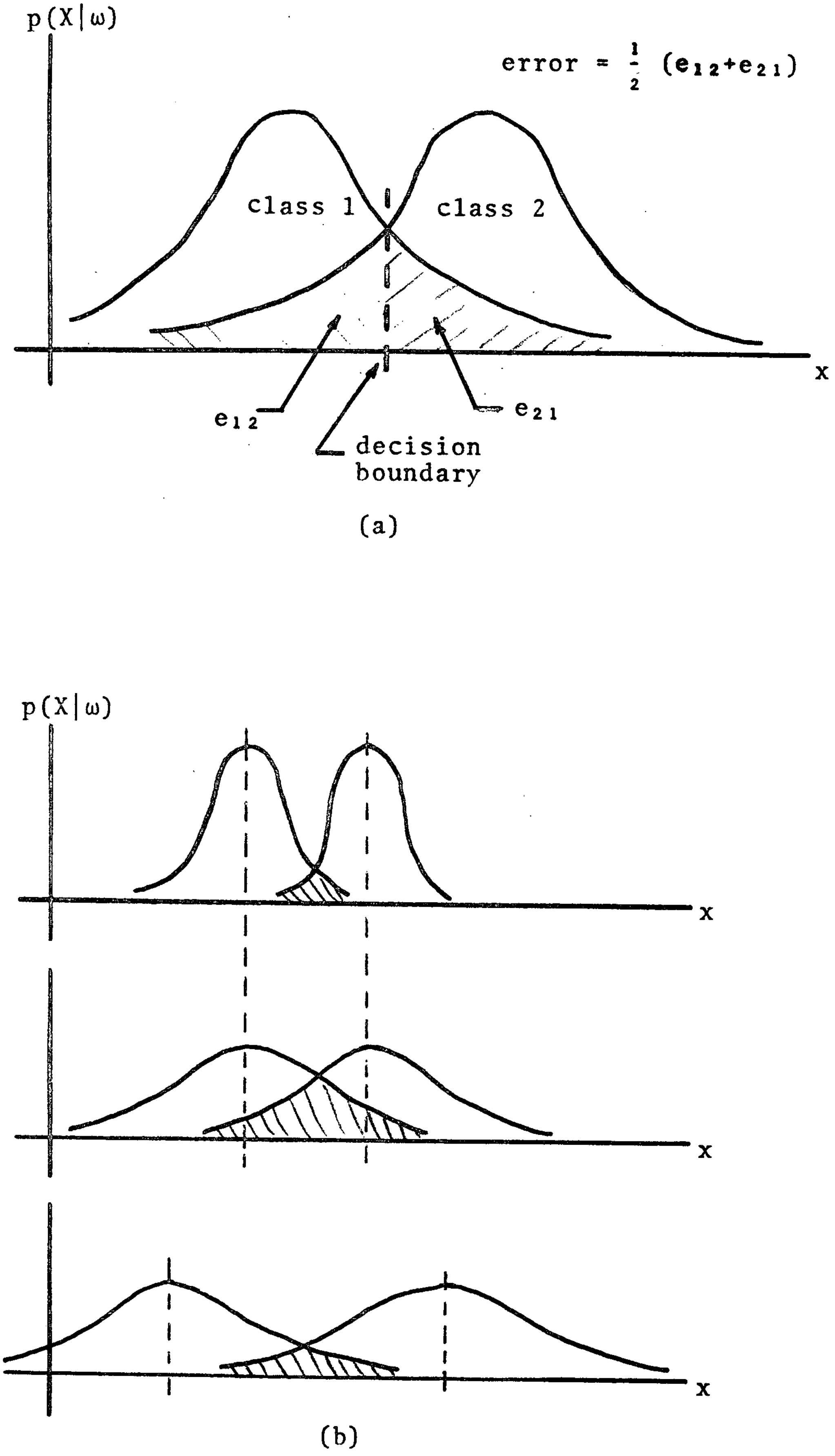
subsets of features must be evaluated. Thus, for example, to select the best 4 out of 12 available features requires

$$\binom{12}{4} = \frac{12!}{4!\ 8!} = 495 \qquad (43)$$

integrations in 4-dimensional space. Even on the fastest computers, such computations would be prohibitive. Alternative methods must be found for feature selection.

From Figure 8, the probability of error (proportional to the shaded area) can be seen to be a function of the "normalized distance" between the classes. That is, the error depends upon both the distance between the means as well as the variance of each class. The greater the "distance" the smaller the probability of error.

One measure of the distance between classes is known as divergence. Divergence is defined in terms of the *likelihood ratio*



(a)

(b)

Figure 8. Classification Error Depends on Distance Between Means and on Variance.

$$L_{ij}(X) = \frac{p(X|\omega_i)}{p(X|\omega_j)} \qquad (44)$$

which is a measure or indication of the separability of the densities at X. The logarithm of the likelihood ratio provides an equivalent indication of the separability of the densities:

$$L'_{ij}(X) = \log L_{ij}(X) = \log p(X|\omega_i) - \log p(X|\omega_j) \qquad (45)$$

Divergence is defined* as

$$D(i,j|c_1, c_2, .. c_n) \triangleq E[L^1_{ij}(X)|\omega_i] - E[L^1_{ij}(X)|\omega_j] \qquad (46)$$

for channels $c_1, c_2, \ldots, c_n$ where

$$E[L^1_{ij}(X)|\omega_i)] \triangleq \int_X L^1_{ij}(X) \quad p(X|\omega_i) \, dX \qquad (47)$$

Divergence has the following properties:

1) $D(i,j|c_1,\ldots,c_n) > 0$ for non-identical distributions
2) $D(i,i|c_1,\ldots,c_n) = 0$
3) $D(i,j|c_1,\ldots,c_n) = D(j,i|c_1, c_2, \ldots c_n)$ (48)
4) Divergence is additive for independent features
   $D(i,j|c_1, c_2,\ldots,c_n) = \sum_{k=1}^{n} D(i,j|c_k)$
5) Adding new features never decreases the divergence, i.e.,
   $D(i,j|c_1,\ldots,c_n) \leq D(i,j|c \ ,\ldots,c_n,c_{n+1})$

Divergence is defined for any two density functions. In the

[*] See for instance Kullback, 1959.

case of normal variables with unequal covariance matrices, it can be shown that

$$D(i,j|c_i,\dots,c_n) = 1/2\ \mathrm{tr}[(\Sigma_i-\Sigma_j)(\Sigma_j^{-1}-\Sigma_i^{-1})] + 1/2\ \mathrm{tr}[(\Sigma_i^{-1}+\Sigma_j^{-1})(U_i-U_j)(U_i-U_j)^T] \tag{49}$$

where tr[A] (trace A) is the sum of the diagonal elements of A.

Divergence is a measure of the dissimilarity of two distributions and thus provides an indirect measure of the ability of the classifier to discriminate successfully between them. Computation of this measure for n-tuples of the available features provides a basis for selecting an optimal set of n features.

Divergence is defined for *two* distributions. Remote sensing problems usually involve m > 2 classes. Several strategies have been suggested and used for feature selection in the multi-class case.

One strategy is to compute the *average divergence* over all pairs of classes and select the subset of features for which the average divergence is maximum. That is, maximize with respect to all n-tuples

$$D_{AVE}(c_1,c_2,\dots,c_n) = \frac{2}{m(m-1)} \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} D(i,j|c_1,c_2,\dots,c_n) \tag{50}$$

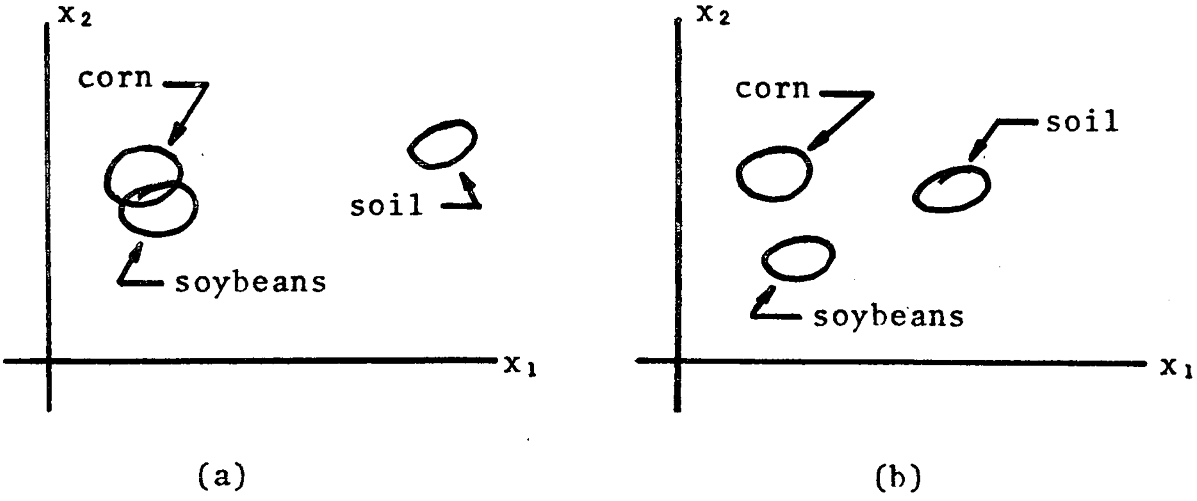
While this strategy is certainly reasonable there is no guarantee that it is optimal. It must be used with care. For instance, a single pairwise divergence, i.e., a single term in (50), if it were large enough, could make the average very large. This is illustrated in Figure 9. So in the process of ranking feature combinations by D_{AVE}, it is a good idea to examine each of the pairwise divergences as well.

Another strategy is to maximize the minimum pairwise divergence, i.e., to select the feature combination which does the best job of separating the hardest-to-separate pair of classes. This is not a Bayesian (minimum risk) strategy, but it is certainly a reasonable strategy for many remote sensing problems.

The problem illustrated in Figure 9 is amplified by the following fact: As the separability of a pair of classes increases, the pairwise divergence also increases without limit--but the probability of correct classification saturates at 100 percent (see Figure 10). A modified form of the divergence, referred to as the "transformed divergence," D_T, has a behavior more like probability of correct classification:

$$D_T = 2[1-\exp(-D/8)] \tag{51}$$

where D is the divergence discussed above. The saturating behavior of this function (see Figure 10) reduces the effects of widely separated classes when taking the average over all pairwise separations. D_{AVE} based on transformed divergence has been found a much more reliable criterion for feature selection than the D_{AVE} based on "ordinary" divergence.



Although D_{AVE} would be larger in (a), overall classification accuracy may be better for the situation in (b).

Figure 9. A Disadvantage of D_{AVE}.

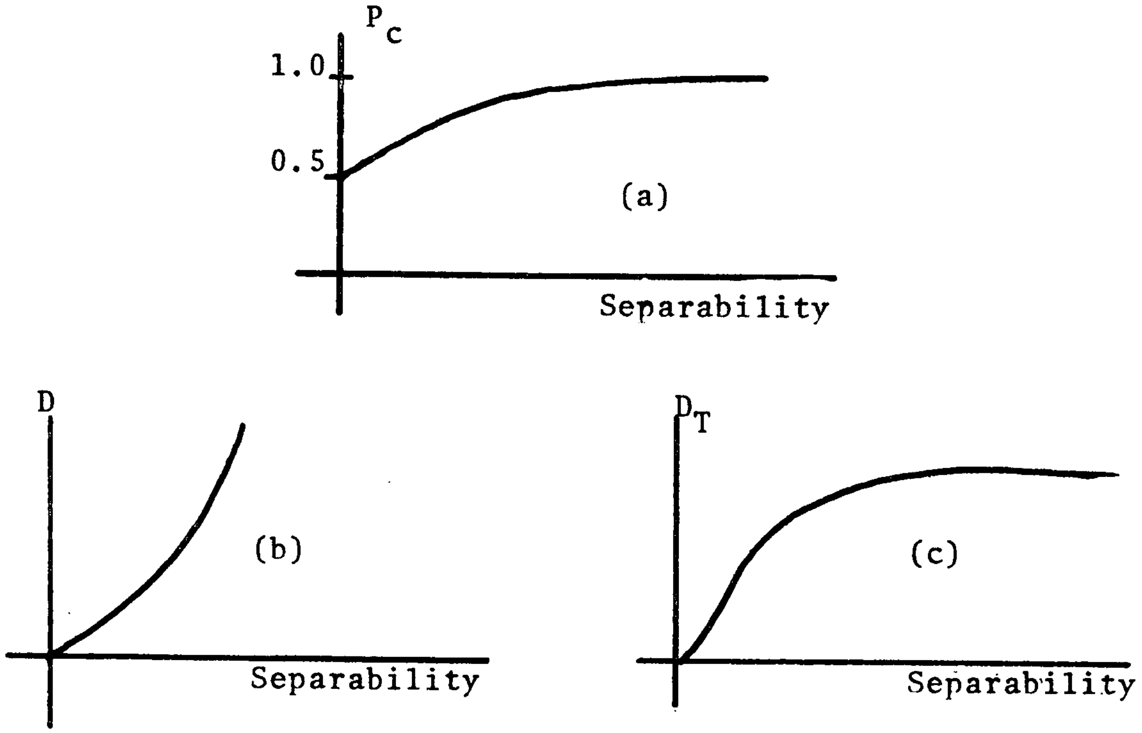


Figure 10. Relationship of Separability and (a) Probability of Correct Classification, (b) Divergence, (c) Transformed Divergence

CLUSTERING

Clustering is a data analysis technique by which one attempts to determine the "natural" or "inherent" relationships in a set of observations or data points. It is sometimes referred to as *unsupervised classification* because the end product is generally a classification of each observation into a "class" which has been established by the analysis procedure, based on the data, rather than by the person interested in the analysis.

To get an intuitive idea of what is meant by *natural* or *inherent relationships* in a set of data, consider the examples shown in Figure 11. If one were to plot height versus weight for a random sampling of students, without regard to sex, on a college campus, it is likely that two relatively distinct clusters of observations would result, one corresponding to the men in the sample (heavier and taller) and another corresponding to the women (lighter and shorter). Similarly, if the spectral reflectance of vegetation in a visible wave band were plotted against reflectance in an infrared wave band, dry vegetation and green vegetation could be expected to form discernible clusters.

If the data of interest never involved more than two attributes (measurements or dimensions), cluster analysis might always be performed by visual evaluation of two-dimensional plots such as those in Figure 11. But beyond two or possibly three dimensions, visual analysis is impossible. For such cases,

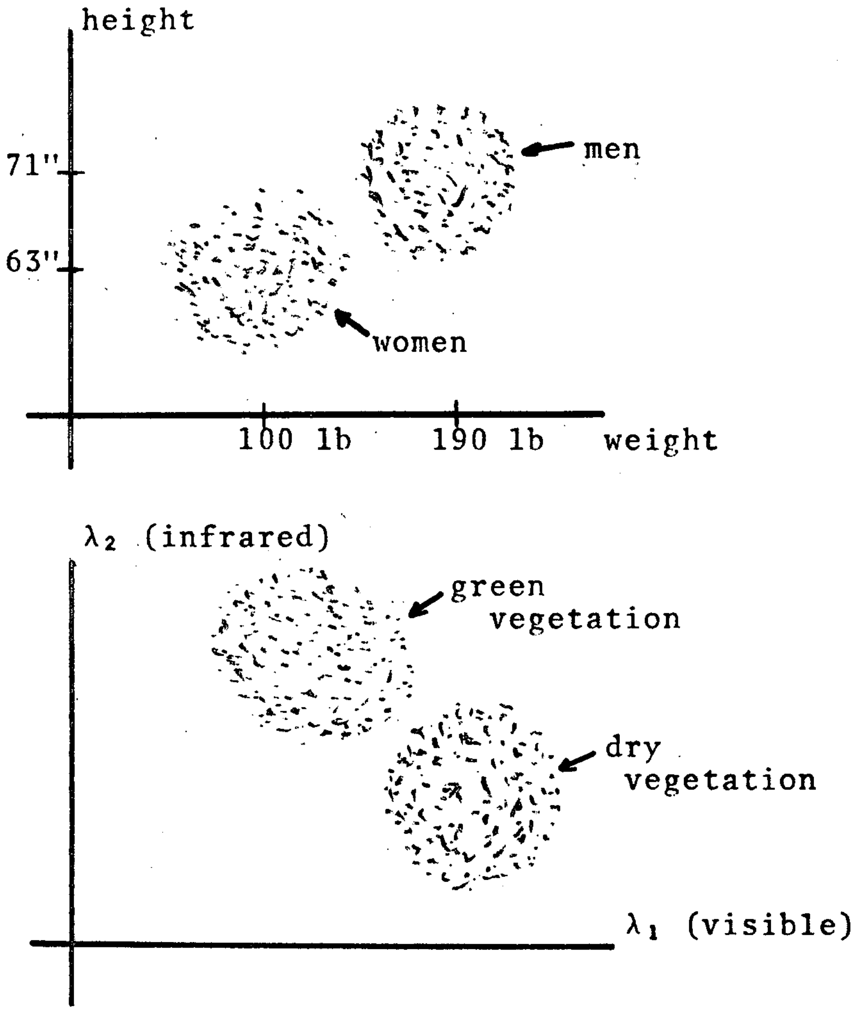


Figure 11. Examples of Data Clusters

it is desirable to have a computer perform the cluster analysis and report the results in a useful fashion.

Why is clustering a useful analysis tool? Clustering has been applied as a means of data compression (eg., for transmission or storage) and for the purpose of determining differentiating characteristics in complex data sets (eg., in numerical taxonomy). An increasingly important application is unsupervised classification, in which the clustering algorithm determines the classes based on the clustering tendencies in the data. The results of such a classification are useful if the "cluster classes" can be interpreted as classes of interest to the data analyst.

With respect to LARSYS, the greatest use of cluster analysis has been for the purpose of assuring that the data used to characterize the pattern classes do not seriously violate the assumption of gaussian statistics. In general it may be expected that each distinct cluster center will correspond to a mode in the distribution of the data. Therefore, by defining a pattern subclass for each cluster center, the possibility of multimodal (and hence definitely non-gaussian) class distributions is essentially eliminated.

The reader interested in the many possible ways of defining clustering in quantitative terms may consult the references (Wacker and Landgrebe, 1971; Hall, 1965). Essentially, the definition of a clustering algorithm depends on the specification of two distance measures: a measure of distance between data points or *individual* observations; and a measure of distance between *groups* of observations. Figure 12 is a block diagram for a typical clustering algorithm (including the LARSYS algorithm). The point-to-point distance measure is used in the step labelled "Assign each vector to nearest cluster center." The distance between groups of points (clusters, in this case) is calculated in the step "Compute separability information."

Euclidean distance, the most familiar point-to-point distance measure, is defined for two n-dimensional points or vectors X and Y as follows:

$$\text{Euclidean distance:} \quad D = \left[\sum_{i=1}^{n} (x_i - y_i)^2\right]^{1/2} \tag{52}$$

Several alternatives are available as candidate measures of distance between clusters, each having its peculiar advantages and disadvantages. One possibility is the divergence or transformed divergence used for feature selection. In LARSYS, a measure called "Swain-Fu distance" has been implemented, which compares the separation of cluster centers to the dispersion of the data in the clusters. The dispersion of the data in a cluster is measured in terms of the "ellipsoid of concentration" associated with the cluster.

Ellipsoid of concentration: Let the random vector X have a distribution with mean vector U and covariance matrix $\Sigma=\left[\sigma_{ij}\right]$. If Z is another random vector uniformly distributed over the volume of the ellipsoid given by

$$Q(Z)=\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{|\Sigma_{ij}|}{|\Sigma|}(z_i-u_i)(z_j-u_j) = n+2 \tag{53}$$

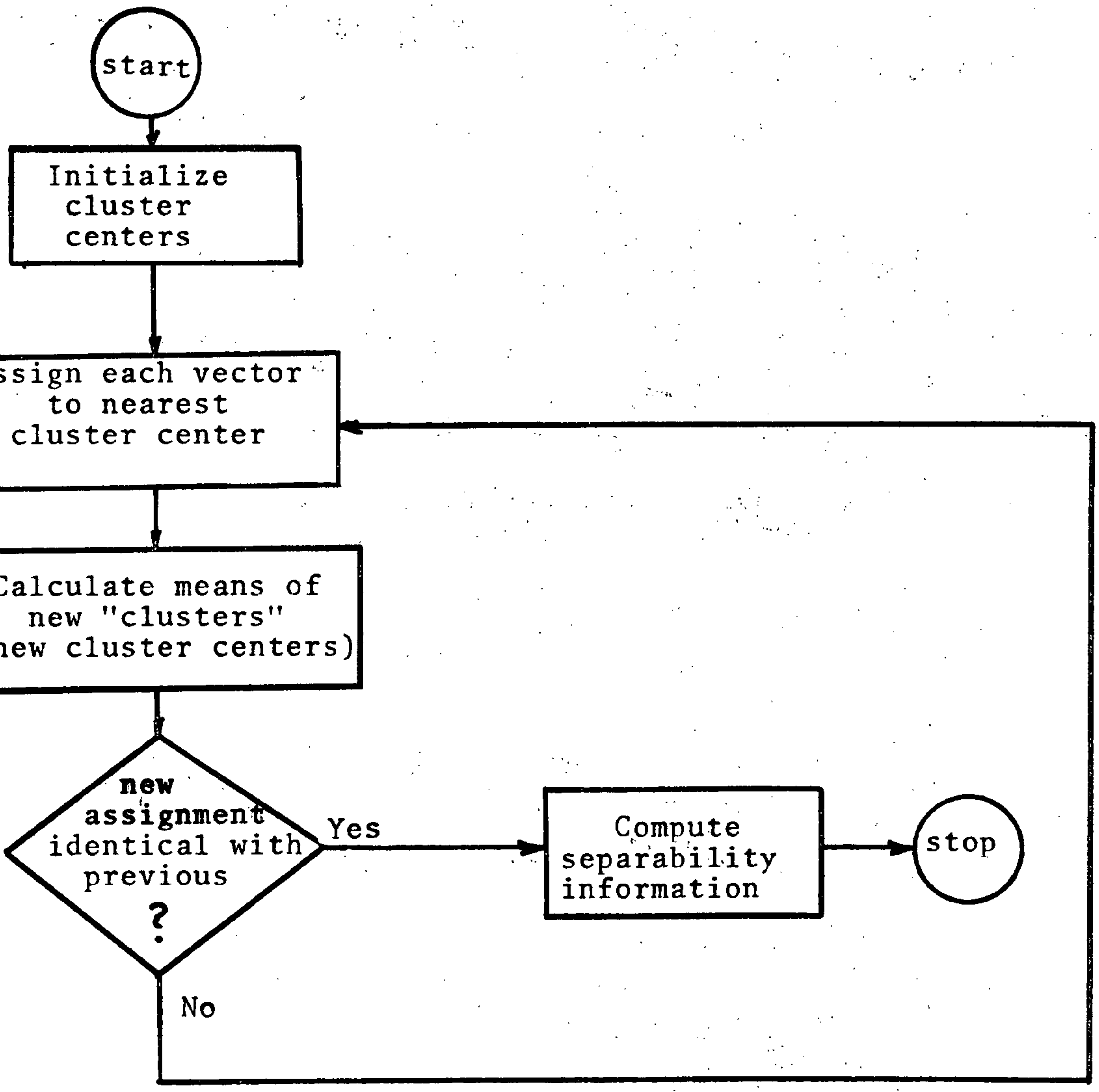

start
Initialize cluster centers
Assign each vector to nearest cluster center
Calculate means of new "clusters" (new cluster centers)
new assignment identical with previous ?
Yes
No
Compute separability information
stop

Figure 12. Clustering Algorithm

where n is the number of components in Z, and $|\Sigma_{ij}|$ is the cofactor of σ_{ij}, then Z also has zero mean and covariance matrix Σ. The ellipsoid Q is called the *ellipsoid of concentration* of the distribution of X.

Q as given by equation (53) is the ellipsoid of concentration of *any* distribution with mean U and covariance Σ and in particular serves as a geometrical characterization of the concentration (or equivalently, the dispersion) of these distributions.

Consider two clusters and their respective ellipsoids of concentration as shown in Figure 13. D_{12} is the distance between the cluster centers. D_1 is the distance from the center of cluster 1 to the surface of its ellipsoid of concentration along the line connecting the cluster centers. Similarly D_2 is the distance from the center of cluster 2 to the surface of its ellipsoid of concentration along the line connecting the cluster centers. In terms of these distances, D_1, D_2, D_{12}, the Swain-Fu distance is given by

$$\Delta = \frac{D_{12}}{D_1+D_2} \qquad (54)$$

In terms of the cluster centers (cluster means) and the covariance matrices associated with the clusters, the Swain-Fu distance can be expressed as

$$\Delta = \frac{\sqrt{c_1 \ c_2}}{\sqrt{c_1} + \sqrt{c_2}} \qquad (55)$$

where

$$c_k = \mathrm{tr}\{\Sigma_k{}^{-1} (U_1-U_2)(U_1-U_2)^T\}$$

tr{A} = trace of matrix A
Σ_k = covariance matrix for cluster k
U_k = mean vector for cluster k.

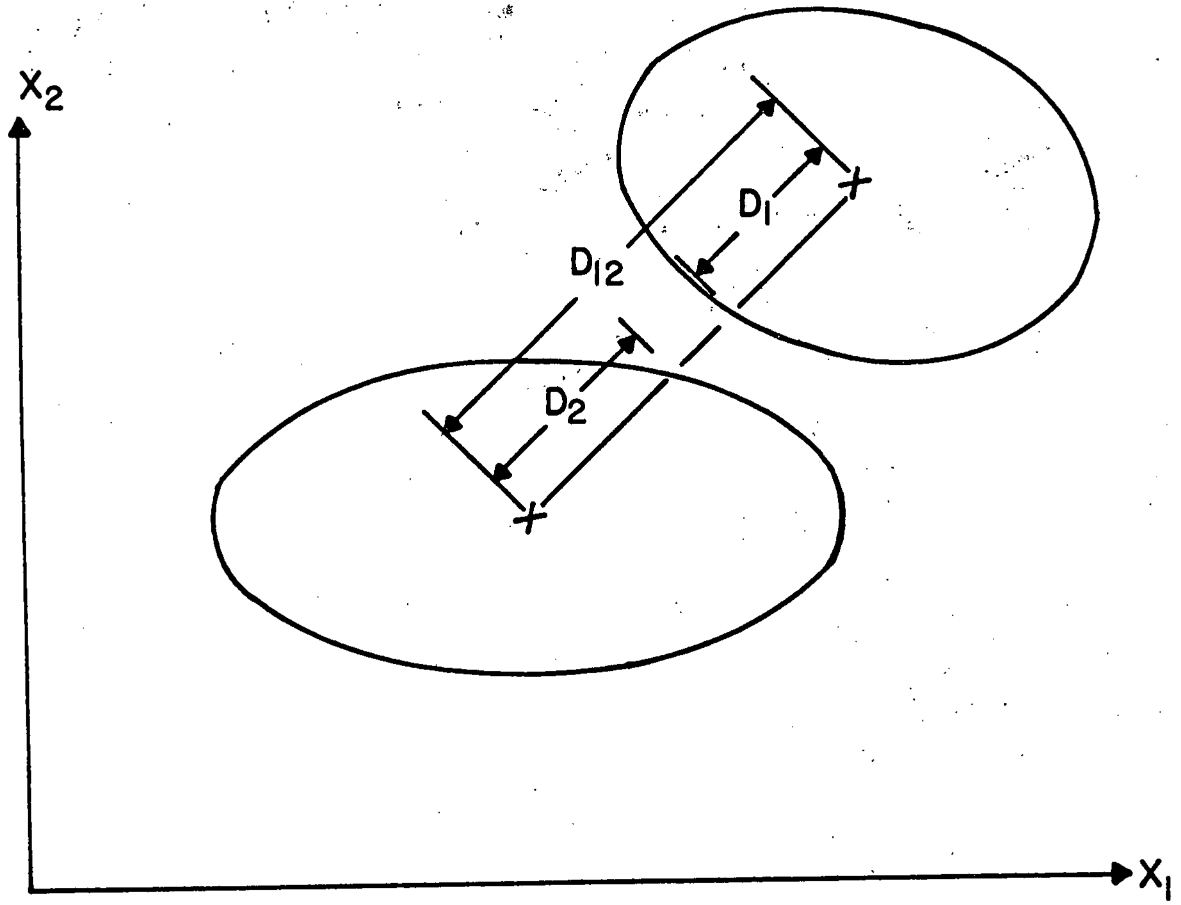


Figure 13. Separability of Clusters

Rule (distinctness): Clusters 1 and 2 as given above are considered distinct provide $\Delta > T$ where T is a suitable threshold.

Empirically, it is observed that two clusters for which Δ is greater than 0.75 will generally exhibit a multimodal distribution if pooled as a single class.

An illustration will provide some insight as to how the algorithm implemented in LARSYS produces clusters from a mass of data (refer to Figures 12 and 14). The first step is to select initial cluster centers. The analyst must specify how many clusters are to be isolated; the algorithm determines (arbitrarily) where the initial centers are to be located (the final results are relatively insensitive to the initial selection). Each data point is then labelled as "belonging" to the nearest cluster center (using Euclidean distance), effectively creating a cluster of data points associated with each center. The boundaries between clusters are formed by the lines (planes in n-dimensional space) which are the perpendicular bisectors of the lines connecting the centers. Next, new cluster centers are calculated. The new center for each cluster is the mean (in general, mean vector) of all points just assigned to that cluster. A check is made to see whether the algorithm has achieved the final result, which is the case when the new cluster centers are identical with the previous centers (or, equivalently, if no data points have changed their cluster "allegiance"). If necessary, the data points are assigned to the nearest new cluster center, and the process is cycled repeatedly. When no further change is detected, the *pairwise* distances (Swain-Fu distance) between the resulting clusters are

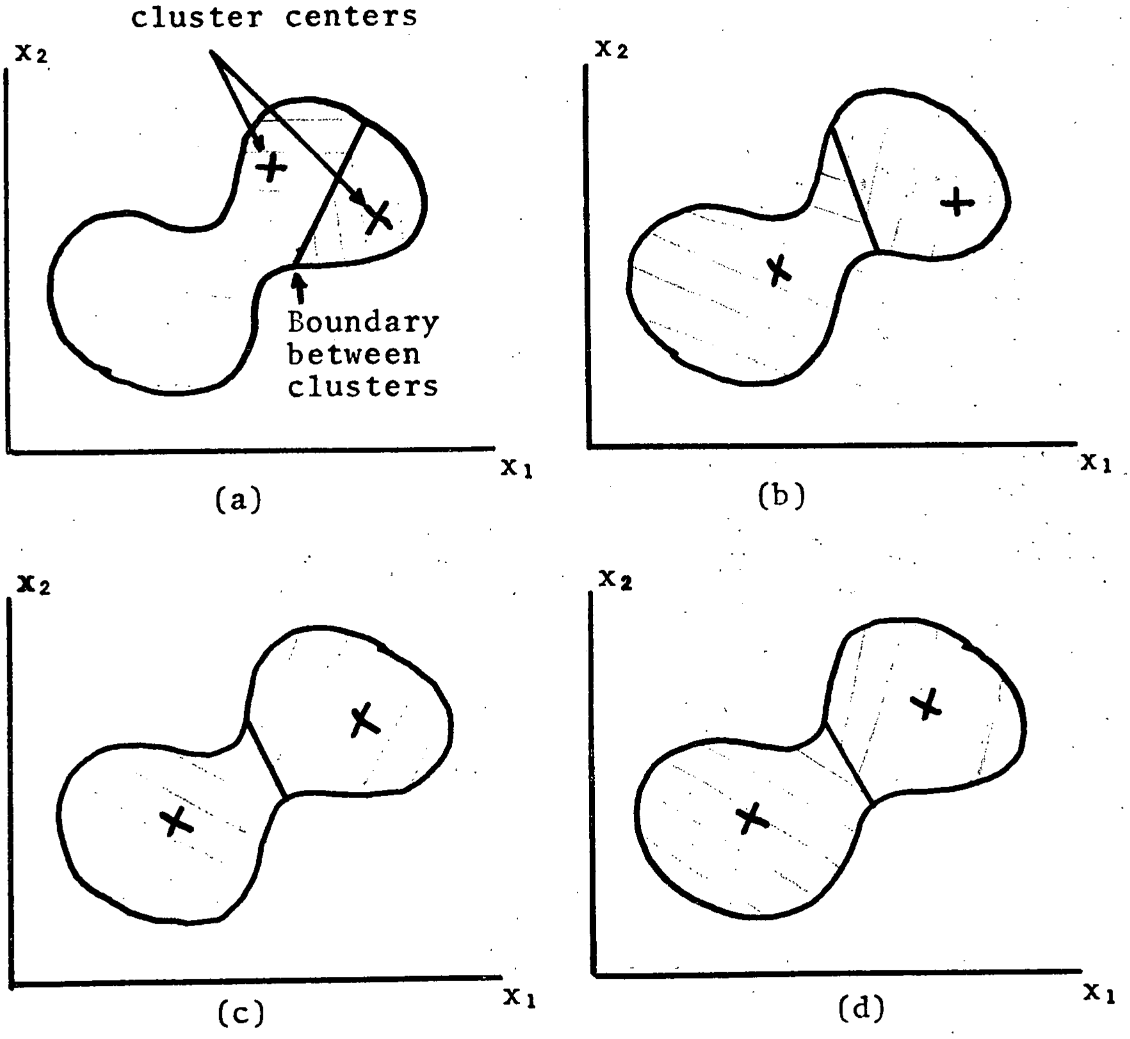


Figure 14. A Sequence of Clustering Iterations (a) Initial Cluster Centers (b) (c) Intermediate Steps (d) Final Center Configuration.

computed and all results are printed for evaluation by the analyst. These results include maps showing the final cluster assignments of all points in the area(s) analyzed, and all pairwise distances between clusters. The analyst must decide which of the resulting clusters are distinct and which should be pooled to define the classes for the maximum likelihood pattern recognition analysis.

SAMPLE CLASSIFICATION

Sample classification is a slight generalization of a concept which has been referred to in agricultural contexts as "per-field classification." In per-field classification, a statistical characterization of the data points in a field (actually, any rectangular area on the ground) is calculated and compared against the statistical characterizations of the pattern classes. Then the field (i.e., the aggregate of points in the field) is classified as a single unit. This is in contrast to the point-by-point classification method discussed previously in which each observation is given a classification which is assigned independently of all other observations. In sample classification an aggregate of data points is characterized and classified as in per-field classification except that the data points need not necessarily be taken from a spatially contiguous area (i.e., need not comprise a field). The only requirement is that the data points must all be assumed to be from the same class -- thus comprising a *sample* from a single population, in statistical terms.

The sample classification approach has some significant potential advantages over the more conventional point classification. Essentially, the decision process has at hand more information on which to base each classification decision, since it utilizes more than a single observation. The sample classification algorithm in LARSYS computes the sample mean and the sample covariance matrix for the data to be classified. The averaging process tends to eliminate the effects of system noise and other irrelevant variability in the data. The sample covariance matrix together with the class covariance matrices serve on one hand to provide appropriate factors for weighting the difference between the sample mean and each class mean; on the other hand, they may contain information which is important in itself for characterizing the pattern classes of interest and associating the sample with the appropriate class. An example of the latter phenomenon has been observed in analyzing flightlines containing both corn fields and forested areas. The average reflectance of the forest may be very much like the average reflectance of corn -- in fact, single observations from each may be very nearly identical. However, the spectral variability of forest cover is typically much greater than that of corn and this is reflected in the covariance matrices. As a result, the sample classifier can perform much more accurately than the point classifier in discriminating between corn and forest.

It should be clear to the reader from the preceding example that the sample classification approach is more powerful than an approach which would classify all points on an individual basis and then classify "fields" according to "majority rules."

Formally, the sample classification procedure may be defined as follows:

Let $d(\cdot,\cdot)$ be a measure defining the *distance* between two probability density functions and let $\{p(X|\omega_i), i = 1, 2, \ldots, m\}$ be a set of probability density functions corresponding to the classes $\omega_1, \omega_2, \ldots, \omega_m$. If $\{\mathbf{X}\}$ is a sample (a set of observations) with estimated probability density $p(X|\omega_x)$ then:

Decide $\{X\} \varepsilon\ \omega_i$ if and only if

$d[p(X|\omega_x), p(X|\omega_i)] \leq d[p(X|\omega_x), p(X|\omega_j)]$

for all i, j, = 1, 2, ..., m.

The concept of distance between probability density functions is the same as that discussed earlier with respect to feature selection. In fact, the same distance measure could be used, although a different distance measure, called *Jeffries-Matusita distance* (see Wacker and Landgrebe, 1971) has been implemented in LARSYS.

For writing the definition of Jeffries-Matusita distance (JM distance), it is convenient to use an abbreviated notation for the density functions. Let

$$p_i(X) = p\ (X|\omega_i).$$

Then the JM distance between density functions $p_1(X)$ and $p_2(X)$ is given by

$$d[p_1(X),p_2(X)] \triangleq [\int_X (\sqrt{p_1(X)} - \sqrt{p_2(X)})^2\ dX]^{1/2} \qquad (56)$$

where the integral is over the entire multi-dimensional space of X. By defining

$$\rho(p_1, p_2) = \int_X \sqrt{p_1(X)} \cdot \sqrt{p_2(X)} \quad dX \tag{57}$$

the JM distance can be expressed as

$$d[p_1(X), p_2(X)] = [2(1-\rho(p_1, p_2))]^{1/2}. \tag{58}$$

In the case of gaussian distributions with class mean vectors U_i, covariance matrices Σ_i, and a sample with mean U_x and covariance matrix Σ_x, Eq. (58) can be written in the form

$$\rho(p_x,p_i) = \frac{|\Sigma_x^{-1}\Sigma_i^{-1}|^{1/4}}{|\frac{1}{2}(\Sigma_x^{-1} + \Sigma_i^{-1})|^{1/2}} \cdot \tag{59}$$

$$\exp\left[-\tfrac{1}{4}\ \{-[(\Sigma_x^{-1} + \Sigma_i^{-1})(\Sigma_x^{-1}\ U_x + \Sigma_i^{-1}U_i)]^T[\Sigma_x^{-1}\ U_x + \Sigma_i^{-1}\ U_i]\right.$$

$$\left. + U_x^T\ \Sigma_x^{-1}U_x + U_i^T\Sigma_i^{-1}U_i\}\right]$$

It is significant that this expression can be evaluated without performing explicit integration.

In practice the U's and Σ's are usually not known, and estimates are used which are obtained from training patterns and from the sample to be classified.

CONCLUDING REMARKS

The foregoing is a description of the theoretical foundations of LARSYS, an approach to multispectral data analysis through pattern recognition and related computer-oriented techniques. The state-of-the-art of machine-assisted remote sensing data analysis is changing rapidly as more powerful methods are sought to meet ever-more-challenging remote sensing problems. It may be expected, however, that unless some radically different approach is developed which proves more effective, the techniques treated herein will continue to be extensively applied. The reader who can take time to develop a working understanding of this material will be well equipped to apply pattern recognition techniques to remote sensing data and to interpret with insight the analysis results he obtains.

References

Hall, G. H., "Data Analysis in the Social Sciences: What About the Details," Proc. Fall Joint Computer Conference, December, 1965.

Kullback, S., Information Theory and Statistics, Wiley, New York, 1959.

Nilsson, N. J., Learning Machines, McGraw-Hill, 1965.

Ready, P. J., P. A. Wintz, and D. A. Landgrebe, "A Linear Transformation for Data Compression and Feature Selection in Multispectral Imagery," LARS Information Note 072071, Laboratory for Applications of Remote Sensing, Purdue University, W. Lafayette, Indiana 47907, February 1971.

Swain, P. H., T. V. Robertson, and A. G. Wacker, "Comparison of the Divergence and B-Distance in Feature Selection," LARS Information Note 020871, Laboratory for Applications of Remote Sensing, Purdue University, W. Lafayette, Indiana 47907, February, 1971.

Wacker, A. G. and D. A. Landgrebe, "Minimum Distance Classification in Remote Sensing," LARS Information Note 030772, Laboratory for Applications of Remote Sensing, Purdue University, W. Lafayette, Indiana 47907, February, 1972.