1968

A method of supervised pattern recognition by an adaptive hypersphere decision threshold

Darroll Steven McCormack

115p.

A METHOD OF

SUPERVISED PATTERN RECOGNITION BY AN

ADAPTIVE HYPERSPHERE DECISION THRESHOLD

BY

DARROLL S^toven^. McCORMACK, 1940

A

THESIS

submitted to the faculty of

THE UNIVERSITY OF MISSOURI AT ROLLA

in partial fulfillment of the requirements for the

Degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

Rolla, Missouri

1968

Approved by

Frank J Kern (advisor) E C Bertnolli

C E Antle

Abstract

In this study the Bayes likelihood detector is combined with an adaptive decision threshold classifier to solve the multicategory pattern recognition problem. It is assumed that the pattern classes can be represented by an n-dimensional vector sample taken from a multivariate gaussian probability distribution.

This study presents (1) the derivation of the Adaptive Hypersphere Decision Threshold classifier (AHDT classifier) and shows (2) how the AHDT classifier minimizes the probability of error using the learning patterns. Finally the AHDT classifier is applied to the solution of a physical problem through computer simulation.

Acknowledgements

The author wishes to express his appreciation to Dr. Frank J. Kern for his many constructive suggestions and hours of help which he provided in the preparation of this thesis.

A special thanks is also extended to Messrs. P. O. Brown and J. Krebbers of McDonnell-Douglas Corporation.

TABLE OF CONTENTS

LIST OF SYMBOLS v

LIST OF FIGURES viii

LIST OF TABLES x

I. INTRODUCTION 1

1.1 Problem Statement 1
1.2 Statistical Model 1

II. REVIEW OF LITERATURE PERTAINING TO PATTERN RECOGNITION 3

2.1 General Review 3
2.2 Bayes Approach to Supervised Pattern Recognition 5
2.3 Adaptive Decision Thresholds 11

III. THE ADAPTIVE HYPERSPHERE DECISION THRESHOLD CLASSIFIER 15

3.1 Approach 15
3.2 Probability of Error 28

IV. IMPLEMENTATION AND COMPUTER SIMULATION OF THE AHDT CLASSIFIER 35

4.1 Quantizing and Coding the Parameter Space 35
4.2 Results 49

V. SUMMARY 81

VI. SUGGESTIONS FOR FURTHER RESEARCH 82

REFERENCES 83

BIBLIOGRAPHY 85

APPENDIX A 88

VITA 104

LIST OF SYMBOLS

(in order of occurrence)

η - Number of pattern classes.

ω_ι - The ι^{th} pattern class.

X_ι - Number of learning patterns for the ι^{th} class

X - A n-dimensional vector sample

X_ι - The ι^{th} vector sample.

Φ - A n-dimensional weight vector.

Φ_ι - The ι^{th} weight vector.

W - A threshold level.

M_ι - Mean n-dimensional vector of X for the ι^{th} class

R_H - Hypersphere radius magnitude.

$P(A/B)$ - Conditional probability of the event A given the event B has occured.

$P(B)$ - Probability of the event B.

$P(A, B)$ - Probability of the event A and B.

V_ι - The ι^{th} class covariance matrix.

$M_{\iota j}$ - The j^{th} mean vector of X for the ι^{th} class.

V_ι^{-1} - The inverse covariance matrix of X for the ι^{th} class.

X^t - The transpose of the vector X.

$\widetilde{M}_\iota$ - The estimate of the mean n-dimensional vector of X for the ι^{th} class.

$X_{\iota\gamma}$ - The γ^{th} learning pattern for the ι^{th} class.

$X_{\iota\gamma\tau}$ - The τ^{th} vector sample of $X_{\iota\gamma}$.

$\widetilde{V}_\iota$ - The estimate of the covariance matrix of X for the ι^{th} class.

$\mathcal{L}_{\kappa\iota}\{X\}$ - Likelihood ratio that the vector X originated in the ι^{th} class versus the κ^{th} class.

$\ln$ - Natural logarithm.

P_α - Probability of misclassifying a vector originating in class α.

$\Phi_{\kappa\iota}$ - The ι^{th} weight vector for the κ^{th} threshold level.

$h(\Phi)$ - Mean-square-error function.

S_d - Desired output.

ζ - constant

$\nabla h[\Phi(\lambda)]$ - The gradient of the function $h[\Phi(\lambda)]$.

f_ℓ - Mean-square-error function.

Y - A η-dimensional logarithm of the likelihood ratio vector.

Y_ι - The ι^{th} vector sample of the logarithm of the likelihood ratio vector.

$\widehat{\ln\{\mathcal{L}_{\alpha\iota}[X]\}/_{\omega_\kappa}}$ - The estimated value of $\ln\{\mathcal{L}_{\alpha\iota}(X)\}$ for class ω_ι

Ψ_ι - The mean η-dimensional vector of Y for the ι^{th} class.

$R_{H\iota}$ - First level hypersphere radius magnitude for the ι^{th} class.

R_A - Second level hypersphere radius magnitude.

Γ - A η-dimensional vector in the logarithm of the likelihood ratio space.

Z - A η-dimensional vector in the logarithm of the likelihood ratio space.

Δ - A η-dimensional vector in the logarithm of the likelihood ratio space.

H_ι - The ι^{th} first level hypersphere.

Ψ_δ - The vector to the origin of the second level hypersphere in the logarithm of the likelihood ratio space.

K_1 - Constant

β - Variable

A_i - The event of the i^{th} learning pattern selection.

R_i - The magnitude of Υ for the A_i learning pattern.

$V(\ )$ - The variance of a variable.

μ - Constant

σ - Constant

C - Constant

P_o - A probability value.

P - A probability value.

$f(i/j)$ - Misclassification rate of vectors from the j^{th} class into the i^{th} class.

$f_i(t)$ - The i^{th} pattern class function.

T_i - Period of the i^{th} pattern class.

P_{ti} - Power content of the i^{th} pattern class.

C_i - Constant for the i^{th} pattern class.

N - A normally distributed random number.

U - A uniformally distributed random number.

σ_{ij} - Covariance between the i^{th} and j^{th} vector samples.

K_2 - Constant

m - Slope of a line.

K_3 - Constant

LIST OF FIGURES

Figure		Page
2.1	2-Dimensional Decision Space	9
2.2	Piecewise Linear Separation	10
2.3	Nonlinear Separation	11
2.4	A Piecewise Linear Decision Function	12
2.5	Mean-Square-Error Function	14
3.1	AHDT Pattern Classifier	16
3.2	Hypersphere Decision Threshold	20
3.3	Second Level Hypersphere	20
3.4	Hypersphere Within a Hypersphere	25
3.5	Union of Three Adaptive Hyperspheres	27
3.6	Probability Bounds	34
4.1	Pattern Class No. 1	37
4.2	Pattern Class No. 2	37
4.3	Pattern Class No. 3	40
4.4	Pattern Class No. 4	40
4.5	Pattern Class No. 5	42
4.6	Pattern Class No. 6	42
4.7	Pattern Class No. 7	44
4.8	Pattern Class No. 8	44
4.9	Error Function	48
4.10	Maximum, Average and Minimum First Level Hypersphere Threshold Magnitude	50
4.11	Percent of Unknown Patterns Falling Within the Union of First Level Hypersphere Thresholds, S/N=10.	52

4.12 Percent of Unknown Patterns Falling Within the Union of First Level Hypersphere Thresholds,, S/N=2. 52

4.13 Percent of Unknown Patterns Falling Within the Union of First Level Hypersphere Thresholds, S/N=1. 53

4.14 Percent of Unknown Patterns Falling Within the Union of First Level Hypersphere Thresholds, S/N=.5 53

4.15 Percent of Unknown Patterns Falling Within the Union of First Level Hypersphere Thresholds, S/N=.2 54

4.16 Percent of Unknown Patterns Falling Within the Union of First Level Hypersphere Thresholds, S/N=.1 54

4.17 Average Error Rate Comparison 55

4.18 Average Correct Classification Comparison 57

4.19 AHDT Separation of Class NU 59

4.20 AHDT Separation of Class 7 60

4.21 AHDT Separation of Class 8 61

4.22 AHDT Separation of Class 9 62

4.23 Maximum Likelihood Ratio Classifier Separation of Signal and Noise 63

4.24 AHDT Classifier Separation of Signal and Noise, Total Patterns 64

4.25 AHDT Classifier Separation of Signal and Noise, Classified Patterns 65

LIST OF TABLES

Table		Page
I.	First Level Hypersphere Threshold Magnitude	66
II.	Number of Unknown Patterns Falling Within the Union of the First Level Hypersphere Thresholds	66
III.	Number of Training Patterns Falling Within the First Level Hypersphere Threshold, S/N=10.	67
IV.	Training Patterns Second Level Hypersphere Threshold Class to Class Separation Matrix, S/N=10.	67
V.	Number of Training Patterns Falling Within the First Level Hypersphere Threshold, S/N=2.	68
VI.	Training Patterns Second Level Hypersphere Threshold Class to Class Separation Matrix, S/N=2.	68
VII.	Number of Training Patterns Falling Within the First Level Hypersphere Threshold, S/N=1.	69
VIII.	Training Patterns Second Level Hypersphere Threshold Class to Class Separation Matrix, S/N=1.	69
IX.	Number of Training Patterns Falling Within the First Level Hypersphere Threshold, S/N=.5	70
X.	Training Patterns Second Level Hypersphere Threshold Class to Class Separation Matrix, S/N=.5	70
XI.	Number of Training Patterns Falling Within the First Level Hypersphere Threshold, S/N=.2	71
XII.	Training Patterns Second Level Hypersphere Threshold Class to Class Separation Matrix, S/N=.2	71

XIII. Number of Training Patterns Falling Within the First Level Hypersphere Threshold, S/N=.1 72

XIV. Training Patterns Second Level Hypersphere Threshold Class to Class Separation Matrix, S/N=.1 72

XV. AHDT Simulation Error Rates 73

XVI. Number of Unknown Patterns Falling Within the First Level Hypersphere Threshold, S/N=10. 74

XVII. AHDT Classification Matrix, S/N=10. 74

XVIII. Number of Unknown Patterns Falling Within the First Level Hypersphere Threshold, S/N=2. 75

XIX. AHDT Classification Matrix, S/N=2. 75

XX. Number of Unknown Patterns Falling Within the First Level Hypersphere Threshold, S/N=1. 76

XXI. AHDT Classification Matrix, S/N=1. 76

XXII. Number of Unknown Patterns Falling Within the First Level Hypersphere Threshold, S/N=.5 77

XXIII. AHDT Classification Matrix, S/N=.5 77

XXIV. Number of Unknown Patterns Falling Within the First Level Hypersphere Threshold, S/N=.2 78

XXV. AHDT Classification Matrix, S/N=.2 78

XXVI. Number of Unknown Patterns Falling Within the First Level Hypersphere Threshold, S/N=.1 79

XXVII. AHDT Classification Matrix, S/N=.1 79

XXVIII. Percent of Unknown Patterns in a Class Exceeding the First Level Hypersphere Threshold 80

CHAPTER I

INTRODUCTION

1.1 Problem Statement

The problem to be investigated is the categorization of a vector taken from an unclassified set of vector samples into one of the available pattern classes. The categorization must be preformed by the pattern classifier using the information derived from the set of learning vector samples for each pattern class. The solution to this problem is based on the the vector sample features which are common to all pattern classes and through which the classes can be distinguished. For example, the power spectral densities of η groups of voltage signals (η classes) could be such that a vector sample could be created by a discrete equally spaced sampling of the power spectral density. The pattern classifier could operate upon this vector sample to classify an unknown voltage signal into one of the η categories. Therefore, the pattern classifier must have the capabilities of detecting the vector sample features and categorizing the vector sample belonging to an unknown class with some predictable error of misclassification.

1.2 Statistical Model

The basic statistical model for pattern recognition with learning observations is as follows:

1) There exist η pattern categories (classes) denoted by $\omega_\iota = \{\omega_1, \omega_2, \omega_3, \ldots, \omega_\eta\}$.

2) For each category ω_ι the observer is given a set of X_ι learning patterns and is told to which class each

observation belongs.

3) Each sample pattern consists of a n-dimensional vector sample, $X = \{X_1, X_2, X_3, \ldots, X_n\}$.

4) Upon receiving the n^{th} sample of an unknown vector sample a decision is made as to the pattern class membership of the vector sample.

This model is similar to the models various authors have used to approach the pattern recognition problem.

CHAPTER II

REVIEW OF LITERATURE PERTAINING TO PATTERN RECOGNITION

2.1 General Review

The pattern recognition problem has been divided into specialized but related areas. Keehn [4], Abramson, Braverman and Sebestyen [6], Koford and Groner [1], Scudder [7], Cooper and Cooper [6], Patrick and Hancock [3,20], and Spragins [16] have considered pattern recognition in terms of supervised and nonsupervised learning. For each learning method, learning patterns existed for all pattern classes; however, the difference was the information given the pattern classifier. The statistical model in Chapter I is similar to the model used by Koford and Groner [1] and Keehn [4] in their study of supervised pattern recognition with learning observations. This model has been identified as "Learning with a Teacher" [14]. If the observer is not told to which class each learning observation belongs, the statistical model would represent a "Learning without a Teacher" pattern recognition problem [14]. This is the nonsupervised pattern recognition model that Scudder [7] and Spragins [16] have investigated.

Problems which are common to supervised and nonsupervised learning are the subsidiary problems:

1) The selection of a set of measurements or features to classify the patterns (feature detection).
2) The determination of a method to partition the measurements or features.

These subsidiary problems are pointed out by Abramson, Braverman and Sebestyen [14], Keehn [4] and Patrick and Hancock [3].

An example of feature selection is the time domain pattern recognition problem. Petersen and Middleton [21] submitted that discrete periodic sampling has become a standard technique for monitoring of continuous data sources in the time domain. The pattern features would consist of the n-dimensional vector sample mean and covariance matrix obtained by operating on several data sets of length n using either supervised or nonsupervised learning. The basic problem of feature detection is to maximize the difference between the pattern classes.

The partition of the measurement space can be accomplished with either a linear or nonlinear separation function or both. Probably the most investigated function is the hyperplane [1,2,11,12]. The hyperplane has been used to obtain linear and piecewise linear separation. Akers [2] applied the the piecewise linear concept to a 2-dimensional pattern recognition problem containing several pattern classes. The procedure is described by Akers as a chain of linear threshold gates with each gate driving the gates ahead of it. Akers [2] and Yau and Chuang [11] defined that the pattern classes are linearly separable if a weight vector Φ exists such that the linear decision rule does not result in any misclassification, Eq. (2.1).

$$\sum_{i=1}^{n} \Phi_i X_i \geq W \quad \text{, decide class } \omega_\alpha \tag{2.1a}$$

$$\text{otherwise decide not class } \omega_\alpha \tag{2.1b}$$

A piecewise linear separation for a Bayes likelihood classifier is shown in Figure (2.2), page 10. This piecewise linear separation occurs if and only if the covariance matrices are equal for all classes.

Nonlinear partition methods which have investigated in some detail are hyperspheres and hyperquadratics [14]. Cooper's investigations [18,19] of the hypersphere presented the spherical decision rule as

$$\left(X-M_\alpha\right)^t \left(X-M_\alpha\right) \le R_\mu \quad \text{, decide class } \omega_\alpha \qquad (2.2a)$$

$$\text{otherwise decide not class } \omega_\alpha \qquad (2.2b)$$

Cooper has also investigated the hyperquadratic rule [10].

An important measure of a pattern classifier is its probability of error (misclassification). Albrecht and Werner [5] and Scudder [7] have investigated this characteristic of a supervised pattern classifier. In order to minimize the probability of error the pattern classifier must optimally partition the measurement space given the a priori knowledge derived from the learning samples for each class. This optimum partition assumes that the mean and covariance estimates obtained from the learning samples give a good approximation of the actual statistical parameters, which according to the weak law of large numbers [17] becomes a better approximation as the number of learning samples increase.

2.2 Bayes Approach to Supervised Pattern Recognition

The Bayes classifier is referred to as the optimum classifier, which computes the conditional probability of one event given that another event has occurred. The Bayes' law is given by Eq. (2.3). Where in this instance, A is the unclassified vector sample and B is the pattern class, conditioned upon the learning samples given for each pattern class.

$$P(A/B) = \frac{P(B/A)\ P(A)}{\sum_{\text{ALL } A} P(B/A)\ P(A)} \tag{2.3}$$

The Bayes classifier makes its decision based upon the likelihood ratio of the joint probabilities.

$$\frac{P(A,B)}{P(A,C)} > 1 \quad \text{, decide class B} \tag{2.4a}$$

$$\frac{P(A,B)}{P(A,C)} = 1 \quad \text{, decide class B or C} \tag{2.4b}$$

$$\frac{P(A,B)}{P(A,C)} < 1 \quad \text{, decide class C} \tag{2.4c}$$

One can write this as

$$\frac{P(A,B)}{P(A,C)} = \frac{P(A/B)\ P(B)}{P(A/C)\ P(C)} \tag{2.5}$$

By assuming that all classes are equally likely Eq. (2.5) reduces to

$$\frac{P(A,B)}{P(A,C)} = \frac{P(A/B)}{P(A/C)} \tag{2.6}$$

Keehn [4] also shows that the term the Bayes classifier needs to consider in the classification of A is the conditional probabilities in Eq. (2.6).

If it is assumed that each pattern may be represented by an n-dimensional column vector taken from a multivariate gaussian distribution, one can write [1,4,13]

$$P\left(X/\omega_\iota\right)=\frac{1}{\left(2\pi\right)^{n/2}\left(\left|V_\iota\right|\right)^{1/2}}\exp\left[-\frac{1}{2}\left(X-M_\iota\right)^t V_\iota^{-1}\left(X-M_\iota\right)\right] \quad (2.7)$$

Where M_ι is the vector sample mean, $M_\iota=\left\{M_{\iota 1},M_{\iota 2},\cdots,M_{\iota n}\right\}$, and V_ι is the vector sample covariance matrix of the ι^{th} pattern class.

The sample mean vector is obtained from the estimate

$$\widetilde{M}_\iota=\frac{1}{X_\iota}\sum_{\gamma=1}^{X_\iota} X_{\iota\gamma} \quad (2.8)$$

and the unbiased covariance matrix estimate from

$$\widetilde{V}_\iota=\frac{1}{X_\iota-1}\sum_{\gamma=1}^{X_\iota}\sum_{\tau=1}^{n}\sum_{J=1}^{n}\left(X_{\iota\gamma\tau}-\widetilde{M}_{\iota\tau}\right)\left(X_{\iota\gamma J}-\widetilde{M}_{\iota J}\right) \quad (2.9)$$

The parameters τ and J denote vector samples of the ι^{th} pattern class and γ represents the γ^{th} pattern sample from the ι^{th} class.

The Bayes classifier operates upon the likelihood that the unclassified vector sample X originated in class ω_ι versus class ω_K. The likelihood ratio is the ratio of the conditional probabilities.

$$\pounds_{K\iota}\{X\}=\frac{P\left(X/\omega_\iota\right)}{P\left(X/\omega_K\right)} \quad (2.10)$$

The substitution of Eq. (2.7) into Eq.(2.10) and elimination of the exponential terms by taking the logarithm yields

$$\ln\left\{\pounds_{K\iota}\{X\}\right\}=\frac{1}{2}\ln\left\{\frac{\left|\widetilde{V}_K\right|}{\left|\widetilde{V}_\iota\right|}\right\}-\frac{1}{2}\left\{\left(X-\widetilde{M}_\iota\right)^t\widetilde{V}_\iota^{-1}\left(X-\widetilde{M}_\iota\right)-\left(X-\widetilde{M}_K\right)^t\widetilde{V}_K^{-1}\left(X-\widetilde{M}_K\right)\right\} \quad (2.11)$$

This will shift the decision threshold of Eq. (2.4) such that

$$\ln\left\{\pounds_{\kappa\iota}\{X\}\right\} > 0 \qquad \text{, decide class } \omega_\iota \qquad (2.12a)$$

$$\ln\left\{\pounds_{\kappa\iota}\{X\}\right\} = 0 \qquad \text{, decide class } \omega_\kappa \text{ or } \omega_\iota \qquad (2.12b)$$

$$\ln\left\{\pounds_{\kappa\iota}\{X\}\right\} < 0 \qquad \text{, decide class } \omega_\kappa \qquad (2.12c)$$

In a multicategory pattern recognition problem the Bayes classifier will place the vector sample in the ω_ι class for which the logarithm of the likelihood ratio is a maximum. The category κ is a fixed class in calculating Eq. (2.11) for all $\iota = 1, 2, 3, \cdots, \eta$. If it is assumed that $\tilde{V}_\iota = \tilde{V}_\kappa = \tilde{V}$, Eq. (2.11) reduces to

$$\ln\left\{\pounds_{\kappa\iota}\{X\}\right\} = X^t \tilde{V}^{-1}\left(\tilde{M}_\iota - \tilde{M}_\kappa\right) - \frac{1}{2}\left(\tilde{M}_\iota + \tilde{M}_\kappa\right)^t \tilde{V}^{-1}\left(\tilde{M}_\iota - \tilde{M}_\kappa\right) \qquad (2.13)$$

In order to demonstrate the probability-of-error optimality of the Bayes classifier the problem will be restricted to two categories, since diagrams in other than a 2-dimensional space are difficult to draw. Figure (2.1) illustrates the two category probability distribution, where $V_1 = V_2$, for which the following conditions hold

$$\ln\left\{\pounds_{12}\{X\}\right\} < 0 \qquad \text{, if } X < \frac{M_2}{2} \qquad (2.14a)$$

$$\ln\left\{\pounds_{12}\{X\}\right\} \geq 0 \qquad \text{, if } X \geq \frac{M_2}{2} \qquad (2.14b)$$

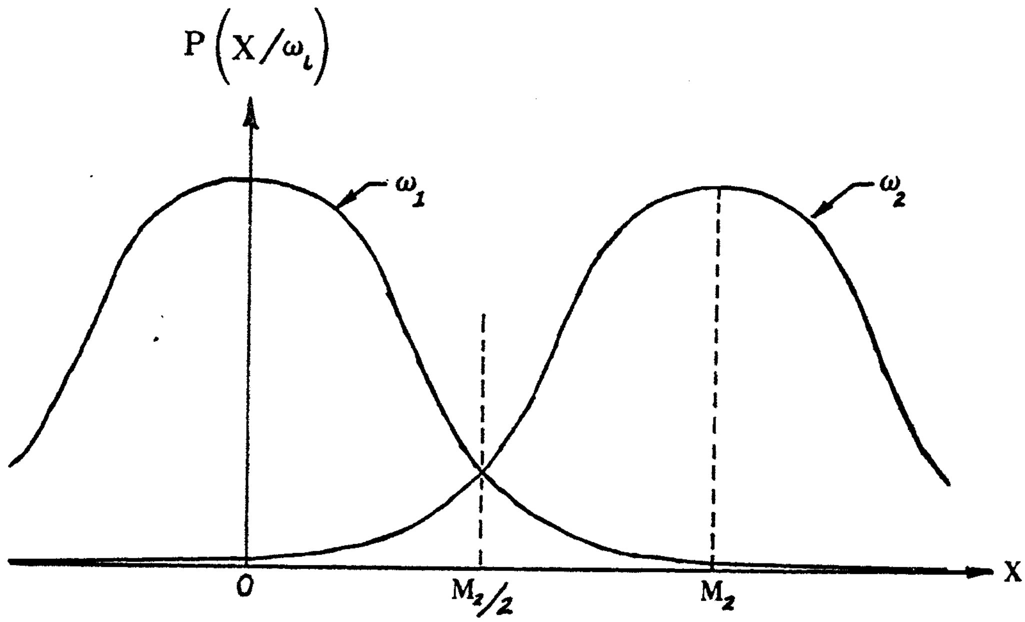


Figure (2.1). 2-Dimensional Decision Space

Let P_1 be the probability of misclassification in class ω_2 given the vector originated in class ω_1 and P_2 the probability of misclassification in class ω_1 given the vector originated in class ω_2 for some threshold W. Then one can write

$$P_1 = \frac{1}{\sqrt{2\pi V_2}} \int_W^{\infty} exp\left[-\frac{1}{2}\,\frac{X^2}{V_2}\right] dx \tag{2.15}$$

$$P_2 = \frac{1}{\sqrt{2\pi V_2}} \int_{-\infty}^{W} exp\left[-\frac{1}{2}\frac{\{X-M_2\}^2}{V_2}\right] dx \tag{2.16}$$

Hence, from Eq. (2.15) and (2.16), the total probability of error, P_1 plus P_2, is a monotonically decreasing function with a minimum at $\ln\left\{\pounds_{12}\{X\}\right\} = 0$. Thus the optimum value for W is $M_2/2$.

For a pattern recognition problem in which three or more pattern classes exist the separation of the classes would be linear if $V_i = V_K$

for a 3-dimensional pattern recognition problem. When $V_i \neq V_k$ for all i,k the optimum separation between classes is nonlinear as shown in Figure (2.3). The likelihood ratio decision thresholds in Figure (2.2) and (2.3) would be somewhat distorted if the actual values of V_1, V_2, and V_3 were replaced with their estimates $\tilde{V}_1, \tilde{V}_2$ and $\tilde{V}_3$. It is this area that is treated in succeeding sections. The idea of operating on the likelihood ratios by additional decision levels, such that the distortion induced by the covariance estimates is smoothed, will be investigated.

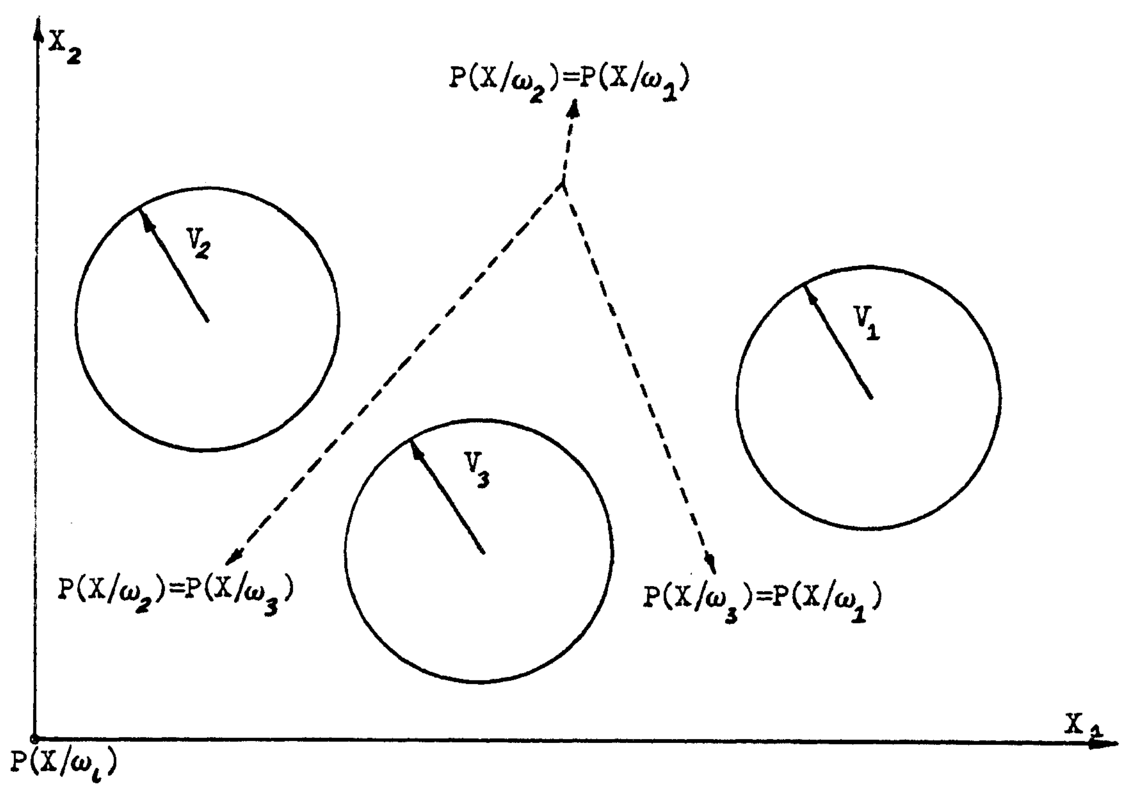


Figure (2.2). Piecewise Linear Separation

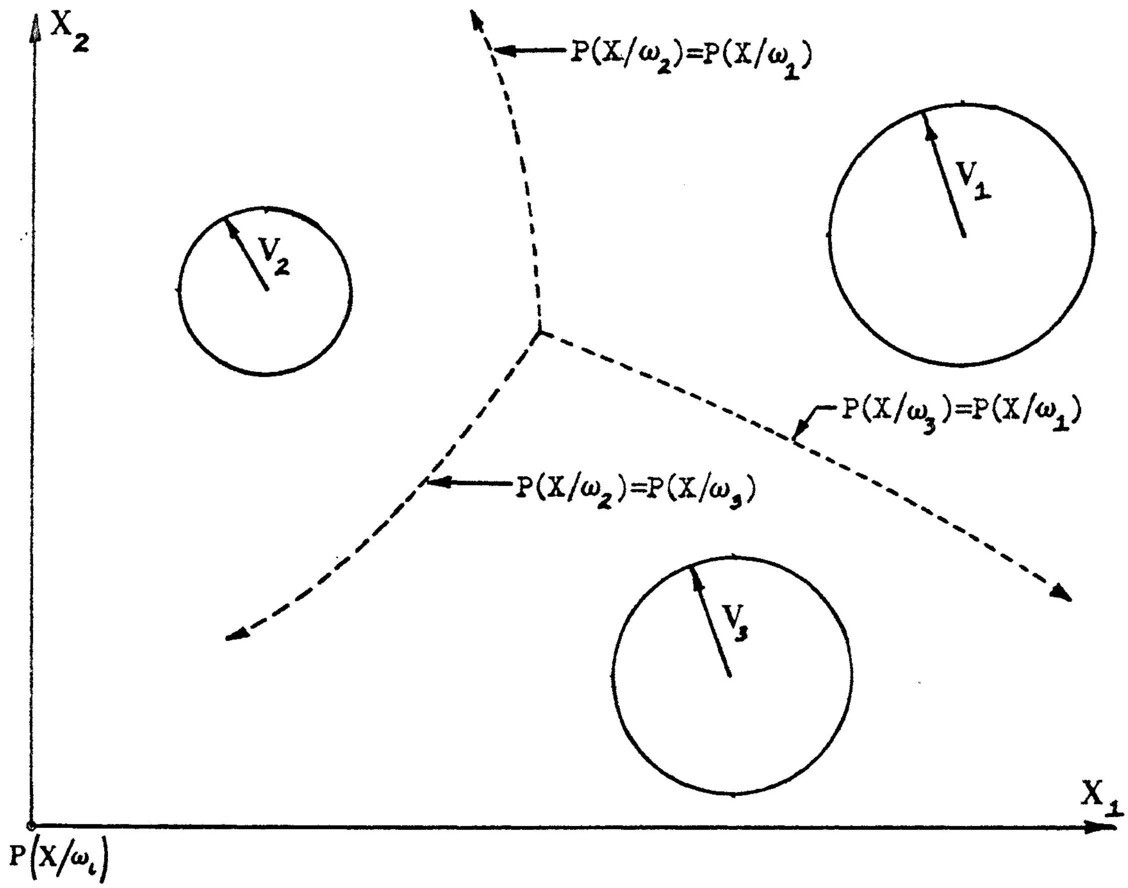


Figure (2.3). Nonlinear Separation

2.3 Adaptive Decision Thresholds

The adaptive pattern classification system concept requires that the classifier have a variable internal structure. The system is adaptive in the sense that the internal structure (decision procedure) is automatically adjusted based upon the learning patterns. The adjustment is made according to some criterion of the system preformance (minimum mean-square-error, no misclassification, etc.). Several authors [1.2.6.10.11.19] have investigated the adaptive decision threshold using various schemes.

Akers [2] presented linear and piecewise linear adaptive thresholds. Where the linear scheme consisted of finding a set of weights to form a hyperplane decision threshold, Eq. (2.1). The piecewise linear decision threshold consisted of a cascading approach in which each threshold gate was driving all gates ahead of it. Figure (2.4) shows a two level piecewise linear decision threshold for which the threshold function is

$$W_2 = \sum_{\tau=1}^{n} \Phi_{2\tau} X_\tau + \mu W_1 \geq 0 \quad \text{, decide class } \omega_\alpha \tag{2.17a}$$

$$\text{otherwise decide not class } \omega_\alpha \tag{2.17b}$$

where

$$W_1 = \sum_{\tau=1}^{n} \Phi_{1\tau} X_\tau \tag{2.18}$$

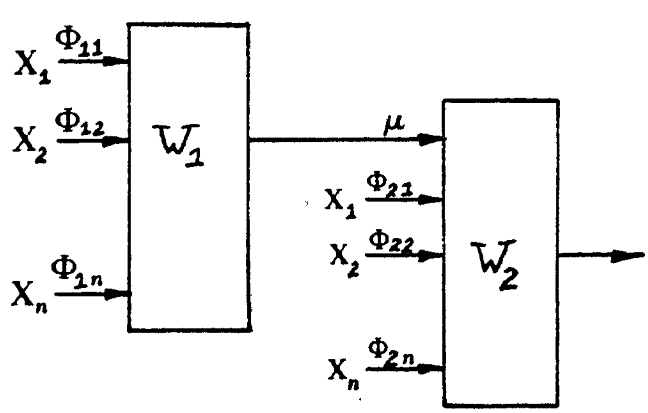


Figure (2.4) A Piecewise Linear Decision Function

The adaptive decision threshold formulated by Koford and Groner [1] was intended to minimize the mean-square-error between the desired and the actual outputs. The classification was obtained using the linear decision rule, Eq. (2.1), in the form

$$X^t\Phi + W \geq 0 \qquad \text{, decide class } \omega_i \tag{2.19a}$$

$$\text{otherwise decide not class } \omega_i \tag{2.19b}$$

The study defines a mean-square-error function $h(\Phi)$, according to Figure (2.5), for a 2-class pattern recognition problem

$$h(\Phi) = \frac{1}{X_1 + X_2} \sum_{\gamma=1}^{2} \sum_{\tau=1}^{X_\gamma} \left[X^t_{\gamma\tau}\Phi - s_d \right]^2 \tag{2.20}$$

and proceeds to formulate an equation for the weight vector in terms of the mean-square-error function.

$$\Phi(\lambda+1) = \Phi(\lambda) - \zeta \; \nabla h(\Phi[\lambda]) \tag{2.21}$$

The constant ζ determines the rate of convergence and stability of the iterative process in obtaining the desired mean-square-error minimization. If ζ is small enough $\nabla h(\Phi[\lambda])$ approaches zero and Eq. (2.21) approaches a minimum. The authors point out that this algorithm always converges to a unique set of weights (determined by the learning patterns and their desired output). The disadvantage is that this unique set of weights may allow some misclassification even

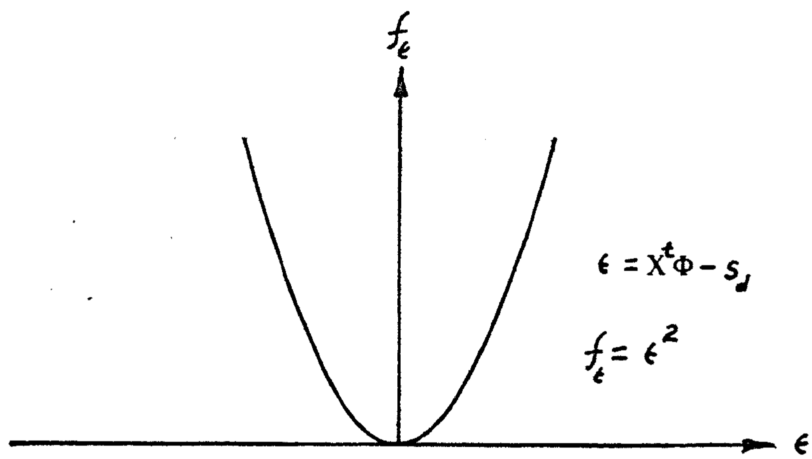


Figure (2.5). Mean-Square-Error Function

Cooper [18,19] considered the pattern recognition problem in terms of a hypersphere decision rule, Eq. (2.2). The decision threshold was adaptive in the sense that the origin, M_i, and/or the magnitude, R_H, of the hypersphere was modified to correctly classify the learning patterns. Cooper's investigations only considered the 2-class pattern recognition problem. This study is an attempt to extend the hypersphere decision rule to a multiple pattern class problem.

CHAPTER III

THE ADAPTIVE HYPERSPHERE DECISION THRESHOLD CLASSIFIER

3.1 Approach

As stated in the previous chapter the purpose of this study is to extend the previous work with the adaptive hypersphere decision threshold to a multiple class pattern recognition problem. A schematic of the proposed Adaptive Hypersphere Decision Threshold classifier, AHDT classifier, is shown in Figure (3.1). The objective of this classifier is to determine, with the help of a teacher, the mean and covariance matrix of the pattern classes. The internal structure of the AHDT classifier is adapted using the logarithm of the likelihood ratio vector of the training patterns. The classifier is expected to classify an unknown vector sample using the η-dimensional logarithm of the likelihood ratio vector. Toward that end, the classifier threshold levels consist of (1) a first level hypersphere, which includes all learning patterns for the ω_i class, and (2) a second level hypersphere, which minimizes the error of misclassification between classes, due to the union of two first level hyperspheres.

With the assumption that each pattern can be represented by an n-dimensional column vector taken from a multivariate gaussian distribution one can write that

$$P\left(X/\omega_i\right)=\frac{1}{\left(2\pi\right)^{n/2}\left(\left|V_i\right|\right)^{1/2}}\exp\left[-\frac{1}{2}\left(X-\widetilde{M}_i\right)^t\widetilde{V}_i^{-1}\left(X-\widetilde{M}_i\right)\right] \qquad (3.1)$$

The logarithm of the likelihood ratio based upon the ω_α class would

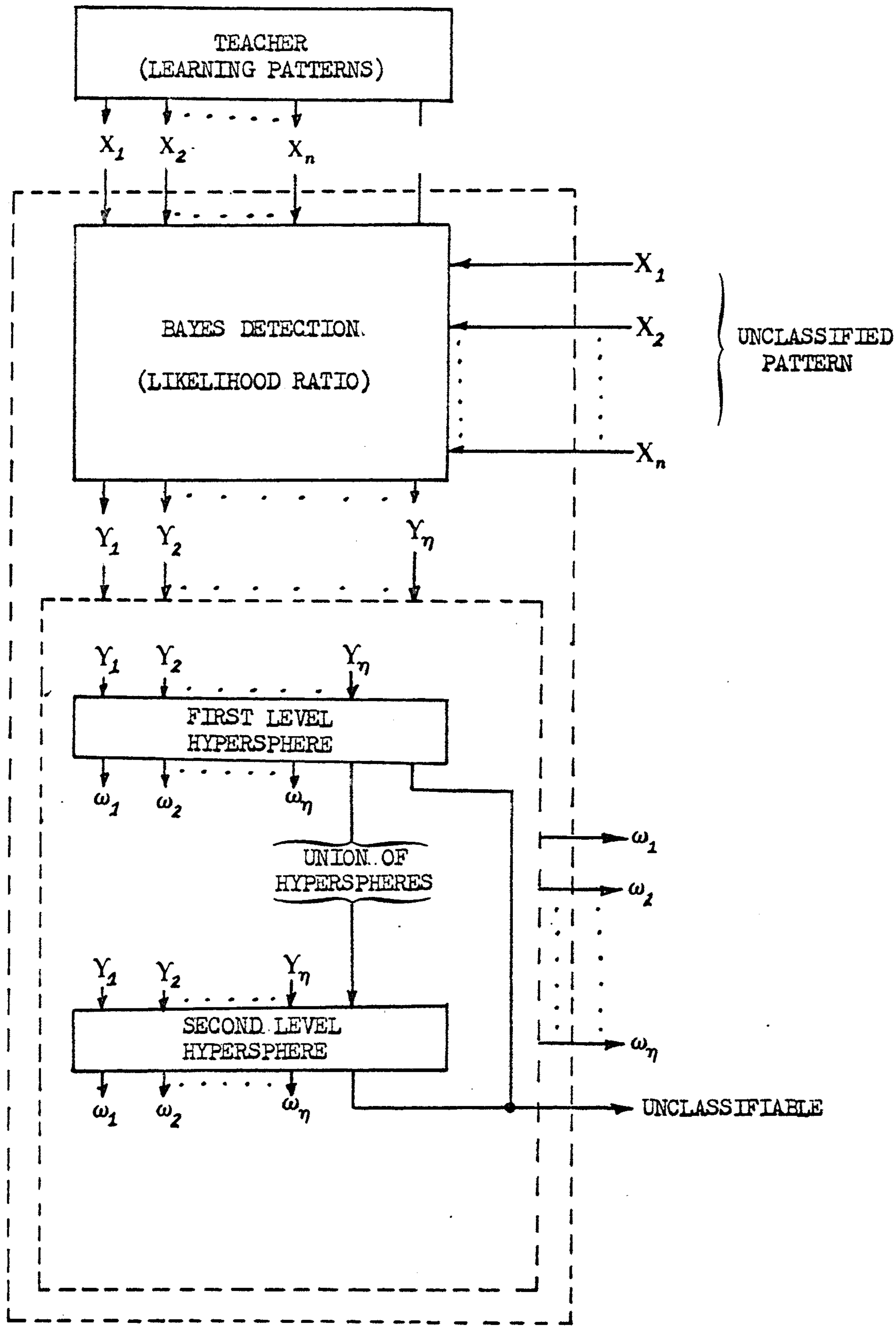


Figure (3.1). AHDT Pattern Classifier

$$\ln\left\{\pounds_{\alpha\iota}\{X\}\right\} = \frac{1}{2}\ln\left[\frac{|\tilde{V}_\alpha|}{|\tilde{V}_\iota|}\right] - \frac{1}{2}\left[\left(X-\tilde{M}_\iota\right)^t \tilde{V}_\iota^{-1}\left(X-\tilde{M}_\iota\right) - \left(X-\tilde{M}_\alpha\right)^t \tilde{V}_\alpha^{-1}\left(X-\tilde{M}_\alpha\right)\right] \quad (3.2)$$

The question remains as to how class ω_α should be specified. To minimize the affect of large numbers, it is proposed that class ω_α be the pattern class centroid. The criterion for the pattern class centroid selection can be obtained by letting the received vector sample be equivalent to the mean vector for class ω_α. The substitution of this equivalancy into Eq. (3.2) will yield

$$\ln\left\{\pounds_{\alpha\iota}\{\tilde{M}_\alpha\}\right\} = \frac{1}{2}\ln\left[\frac{|\tilde{V}_\alpha|}{|\tilde{V}_\iota|}\right] - \frac{1}{2}\left(\tilde{M}_\alpha-\tilde{M}_\iota\right)^t \tilde{V}_\iota^{-1}\left(\tilde{M}_\alpha-\tilde{M}_\iota\right) \quad (3.3)$$

This can be rewritten as

$$\ln\left\{\pounds_{\alpha\iota}\{\tilde{M}_\alpha\}\right\} = \frac{1}{2}\ln\left[\frac{|\tilde{V}_\alpha|}{|\tilde{V}_\iota|}\right] - \frac{1}{2}\left(\tilde{M}_\iota-\tilde{M}_\alpha\right)^t \tilde{V}_\iota^{-1}\left(\tilde{M}_\iota-\tilde{M}_\alpha\right) \quad (3.4)$$

Summing up Eq. (3.4) for all ι to obtain

$$\sum_{\iota=1}^{\eta}\ln\left\{\pounds_{\alpha\iota}\{\tilde{M}_\alpha\}\right\} = \frac{1}{2}\sum_{\iota=1}^{\eta}\ln\left[\frac{|\tilde{V}_\alpha|}{|\tilde{V}_\iota|}\right] - \frac{1}{2}\sum_{\iota=1}^{\eta}\left(\tilde{M}_\iota-\tilde{M}_\alpha\right)^t \tilde{V}_\iota^{-1}\left(\tilde{M}_\iota-\tilde{M}_\alpha\right) \quad (3.5)$$

one sees that under the conditions $V_\iota=V_\alpha=V$, where V is an identity matrix, Eq. (3.5) will reduce to

$$\sum_{\iota=1}^{\eta}\ln\left\{\pounds_{\alpha\iota}\{\tilde{M}_\alpha\}\right\} = -\frac{1}{2}\sum_{\iota=1}^{\eta}\left(\tilde{M}_\iota-\tilde{M}_\alpha\right)^t \left(\tilde{M}_\iota-\tilde{M}_\alpha\right) \quad (3.6)$$

Since the magnitude of $(\tilde{M}_\iota-\tilde{M}_\alpha)^t(\tilde{M}_\iota-\tilde{M}_\alpha)$ is a positive number for all ι, then one can write

$$\left| \ln\left\{ \pounds_{\alpha\iota}\{\widetilde{M}\} \right\} \right| \le \left| \sum_{\iota=1}^{\eta} \ln\left\{ \pounds_{\alpha\iota}\{\widetilde{M}\} \right\} \right| \qquad (3.7)$$

This shows that to minimize the affect of large numbers Eq. (3.6) should be minimized. Thus the pattern class centroid would be selected utilizing the criterion that

$$(\widetilde{M}_\iota - \widetilde{M}_\alpha)^t (\widetilde{M}_\iota - \widetilde{M}_\alpha) = \text{minimum} \qquad (3.8)$$

The Bayes classifier would place an unclassified vector sample in that class having the maximum likelihood ratio with an optimum misclassification. Generally there is some error associated with the calculated covariance matrix and mean vector sample for each class resulting in a nonoptimum misclassification. An attempt to reduce the misclassification through additional signal processing (smoothing) by letting the logarithm of the likelihood ratios be a η-dimensional vector input for an adaptive threshold classifier is proposed here.

The adaptive decision threshold can be formulated using the expected value of the input function described by Eq. (3.2). The expected value of the function given that the sample pattern came from the ω_K class can be written as

$$\widetilde{\left[\ln\left\{\pounds_{\alpha\iota}\{X\}\right\}/\omega_K\right]} = \frac{1}{2}\ln\left[\frac{|\widetilde{V}_\alpha|}{|\widetilde{V}_\iota|}\right] - \frac{1}{2}\left[\left(\widetilde{M}_K - \widetilde{M}_\iota\right)^t \widetilde{V}_\iota^{-1}\left(\widetilde{M}_K - \widetilde{M}_\iota\right) - \left(\widetilde{M}_K - \widetilde{M}_\alpha\right)^t \widetilde{V}_\alpha^{-1}\left(\widetilde{M}_K - \widetilde{M}_\alpha\right)\right] \qquad (3.9)$$

This is the concept that Marill and Green [9] used to formulate the expected value of the logarithm of the likelihood ratio. Let the class ω_K mean vector in the logarithm of the likelihood ratio space be represented by

$$\Psi_K = \left\{ \overline{\left[\ln\left\{\pounds_{\alpha_1}\{X\}\right\}/\omega_K\right]},\ \overline{\left[\ln\left\{\pounds_{\alpha_2}\{X\}\right\}/\omega_K\right]}, \cdots\cdots, \overline{\left[\ln\left\{\pounds_{\alpha\eta}\{X\}\right\}/\omega_K\right]} \right\} \quad (3.10)$$

and a vector sample by

$$Y = \left\{ \ln\left\{\pounds_{\alpha_1}\{X\}\right\}, \ln\left\{\pounds_{\alpha_2}\{X\}\right\}, \cdots\cdots, \ln\left\{\pounds_{\alpha\eta}\{X\}\right\} \right\} \quad (3.11)$$

The first level hypersphere threshold in the logarithm of the likelihood ratio space is, Figure (3.2),

$$\left(Y - \Psi_l\right)^t \left(Y - \Psi_l\right) \le \left(R_{Hl}\right)^2 \quad \text{, decide class } \omega_l \quad (3.12a)$$

$$\text{otherwise decide not class } \omega_l \quad (3.12b)$$

The vector sample to be classified will generate the η-dimensional vector Y in the logarithm of the likelihood ratio space. The learning patterns for each class will provide the estimate of Ψ_l and the magnitude of R_{Hl}, which is increased to include all learning patterns of class ω_l. It should be noted that the value of R_{Hl} has not been restricted to a constant value for all classes.

The union of two or more first level hyperspheres in a η-class pattern recognition problem can result in a number of unclassifiable vector samples; however, this number can be reduced by using multiple

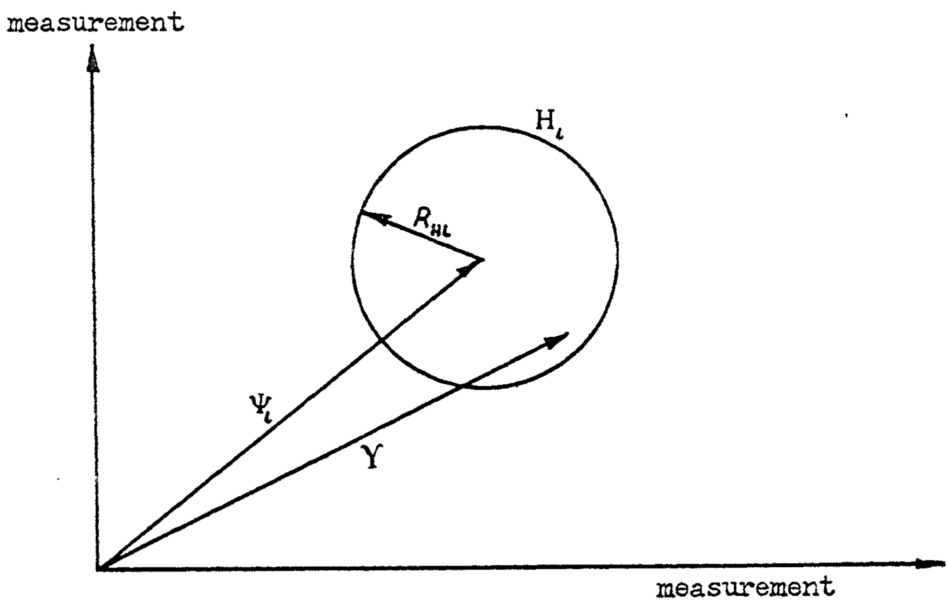


Figure (3.2). Hypersphere Decision Threshold

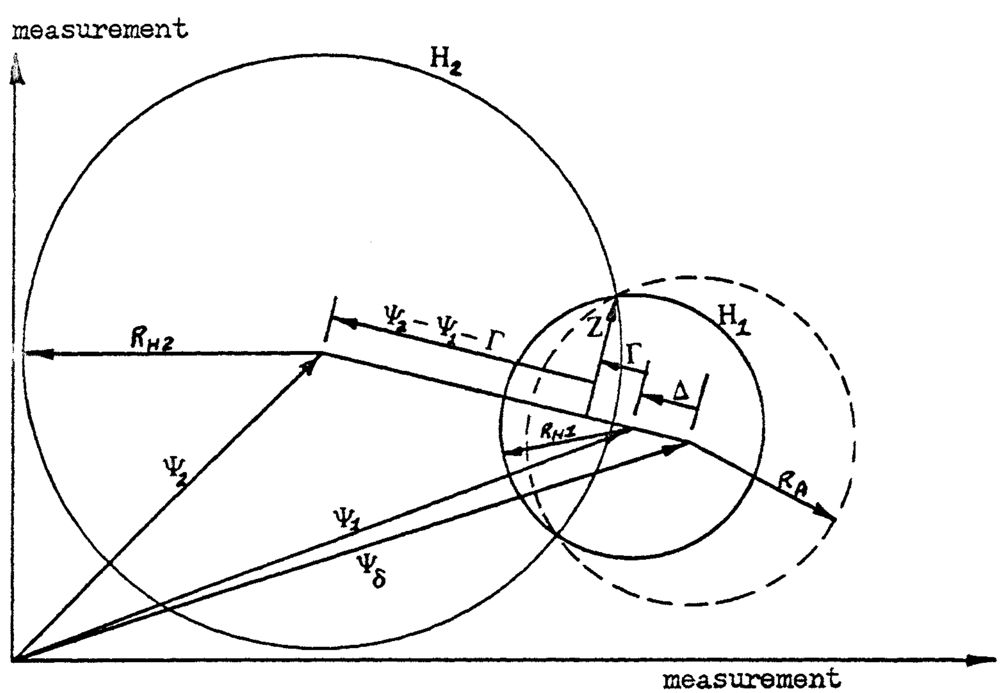


Figure (3.3). Second Level Hypersphere

adaptive hypersphere decision threshold to minimize the error of misclassification. Figure (3.3) illustrates this problem and shows the resultant second level adaptive hypersphere required to separate two classes. The threshold for the second level hypersphere in Figure (3.3) is

$$\left(Y - \Psi_\delta\right)^t \left(Y - \Psi_\delta\right) \le \left(R_A\right)^2 \text{, decide class } \omega_1 \qquad (3.13a)$$

$$\text{otherwise decide class } \omega_2 \qquad (3.13b)$$

The problem that remains is how the value of Ψ_δ should be assigned. Since the objective is to separate the union of the hyperspheres, let the adaptive hypersphere intersect the intersection of the two hyperspheres, as shown in Figure (3.3). Under this condition one can write the relations

$$\left(Z + \Gamma\right)^t \left(Z + \Gamma\right) = \left(R_{H1}\right)^2 \qquad (3.14)$$

$$\left(Z + \Psi_2 - \Psi_1 - \Gamma\right)^t \left(Z + \Psi_2 - \Psi_1 - \Gamma\right) = \left(R_{H2}\right)^2 \qquad (3.15)$$

and

$$\left(\Gamma + \Delta + Z\right)^t \left(\Gamma + \Delta + Z\right) = \left(R_A \quad\right)^2 \qquad (3.16)$$

where

$$\Gamma = K_1\left(\Psi_2 - \Psi_1\right) \tag{3.17}$$

$$\Delta = \beta\left(\Psi_2 - \Psi_1\right) \tag{3.18}$$

Now, it is possible to rewrite Eq. (3.15) and expand it into

$$\left(R_{H2}\right)^2 = \left(Z + \Gamma\right)^t\left(Z + \Gamma\right) + \left(Z + \Gamma\right)^t\left(\Psi_2 - \Psi_1 - 2\Gamma\right) + \left(\Psi_2 - \Psi_1 - 2\Gamma\right)^t\left(Z + \Gamma\right) + \left(\Psi_2 - \Psi_1 - 2\Gamma\right)^t\left(\Psi_2 - \Psi_1 - 2\Gamma\right) \tag{3.19}$$

The substitution of Eq. (3.14) and (3.17) into Eq. (3.19) will yield

$$\left(R_{H2}\right)^2 = \left(R_{H1}\right)^2 + \left[1 - 2K_1\right]\left\{Z^t\left(\Psi_2 - \Psi_1\right) + \left(\Psi_2 - \Psi_1\right)^t Z + \left(\Psi_2 - \Psi_1\right)^t\left(\Psi_2 - \Psi_1\right)\right\} \tag{3.20}$$

Now by expanding Eq. (3.16)

$$\left(R_A\right)^2 = \Delta^t\Delta + \Delta^t\left(Z + \Gamma\right) + \left(Z + \Gamma\right)^t\Delta + \left(Z + \Gamma\right)^t\left(Z + \Gamma\right) \tag{3.21}$$

and substituting Eq. (3.14), (3.17) and (3.18) into Eq. (3.21) to obtain

$$\left(R_A\right)^2 = \left[\beta^2 + 2K_1\beta\right]\left(\Psi_2 - \Psi_1\right)^t\left(\Psi_2 - \Psi_1\right) + \beta\left\{Z^t\left(\Psi_2 - \Psi_1\right) + \left(\Psi_2 - \Psi_1\right)^t Z\right\} + \left(R_{H1}\right)^2 \tag{3.22}$$

This equation can be rearranged as

$$Z^t\left(\Psi_2-\Psi_1\right)+\left(\Psi_2-\Psi_1\right)^t Z=\frac{1}{\beta}\left\{\left(R_A\right)^2-\left(R_{H1}\right)^2-\left[\beta^2+2\beta K_1\right]\left(\Psi_2-\Psi_1\right)^t\left(\Psi_2-\Psi_1\right)\right\} \tag{3.23}$$

and substituted into Eq. (3.20) to eliminate the variable Z.

$$\left(R_{H2}\right)^2=\left(R_{H1}\right)^2+\left[1-2K_1\right]\left\{\frac{1}{\beta}\left[\left(R_A\right)^2-\left(R_{H1}\right)^2-\left\{\beta^2+2\beta K_1\right\}\left(\Psi_2-\Psi_1\right)^t\left(\Psi_2-\Psi_1\right)\right]\right.$$
$$\left.+\left(\Psi_2-\Psi_1\right)^t\left(\Psi_2-\Psi_1\right)\right\} \tag{3.24}$$

Solving for $\left(R_A\right)^2$ in Eq.(3.24) yields

$$\left(R_A\right)^2=\left(R_{H1}\right)^2\left[1-\frac{\beta}{1-2K_1}\right]+\left(R_{H2}\right)^2\left[\frac{\beta}{1-2K_1}\right]$$
$$+\beta\left[\beta+2K_1-1\right]\left(\Psi_2-\Psi_1\right)^t\left(\Psi_2-\Psi_1\right) \tag{3.25}$$

The constant K_1 may be found from the initial condition for β. Under the condition $\beta=0$, the second level hypersphere would initially coincide with the H_1 hypersphere such that

$$\left(R_A\right)^2=\left(R_{H1}\right)^2 \tag{3.26}$$

Thus, Eq. (3.24) in the initial condition becomes

$$\left(R_{H2}\right)^2=\left(R_{H1}\right)^2+\left[1-2K_1\right]^2\left(\Psi_2-\Psi_1\right)^t\left(\Psi_2-\Psi_1\right) \tag{3.27}$$

and by rearranging Eq. (3.27) and solving for K_1

$$K_1 = \frac{1}{2} \pm \frac{1}{2}\sqrt{\frac{(R_{H2})^2 - (R_{H1})^2}{(\Psi_2 - \Psi_1)^t(\Psi_2 - \Psi_1)}} \tag{3.28}$$

Eq. (3.28) imples that a restriction is placed on the adaptive hypersphere threshold. This restriction is

$$\left(R_{H2}\right)^2 \geq \left(R_{H1}\right)^2 \tag{3.29}$$

since K_1 is defined to be a real number. This restriction would limit the maximum positive value for K_1 to 1/2.

Consideration must also be given to the problem of a hypersphere within a hypersphere as shown in Figure (3.4). In this case the magnitude of the vector Z to the H_1 and H_2 hypersphere intersection is zero, since an intersection does not exist. Thus, one obtains the relations

$$\left(R_{H2}\right)^2 \geq \left(\Psi_2 - \Psi_1 - \Gamma\right)^t\left(\Psi_2 - \Psi_1 - \Gamma\right) \tag{3.30}$$

$$\left(R_{H1}\right)^2 = \Gamma^t\Gamma \tag{3.31}$$

and

$$\left(R_A\right)^2 = \left(\Gamma + \Delta\right)^t\left(\Gamma + \Delta\right) \tag{3.32}$$

Substitution of Eq. (3.17) and (3.31) into Eq. (3.30) will yield

$$\left(R_{H2}\right)^2 \geq \left[1-2K_1\right]\left(\Psi_2-\Psi_1\right)^t\left(\Psi_2-\Psi_1\right)+\left(R_{H1}\right)^2 \tag{3.33}$$

From the substitution of Eq. (3.17) into Eq. (3.31) it follows that

$$K_1 = \pm\sqrt{\frac{\{R_{H1}\}^2}{\left(\Psi_2-\Psi_1\right)^t\left(\Psi_2-\Psi_1\right)}} \tag{3.34}$$

From Eq. (3.17) and Figure (3.4) it is observed that K_1 would have a negative magnitude. From Eq. (3.17), (3.18) and (3.32) one obtains

$$\left(R_A\right)^2 = \left[\beta+K_1\right]^2\left(\Psi_2-\Psi_1\right)^t\left(\Psi_2-\Psi_1\right) \tag{3.35}$$

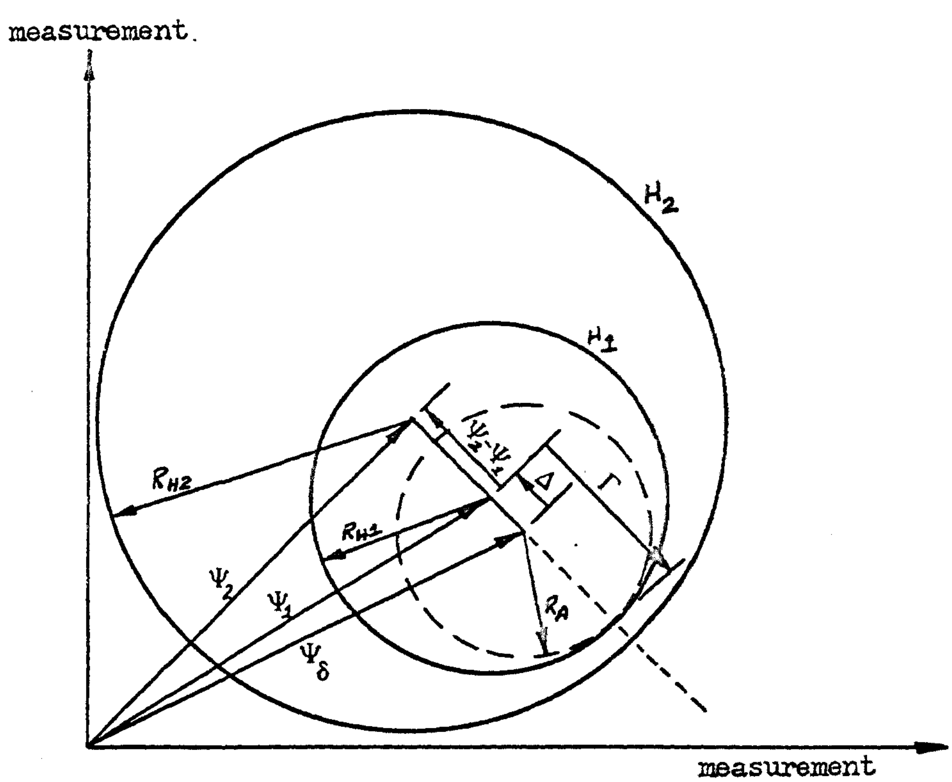


Figure (3.4). Hypersphere Within a Hypersphere

In the computer simulation problem Eq. (3.34) must be calculated and the inequality of Eq. (3.30) proved or disproved. From this either Eq. (3.25) or Eq. (3.35) would be used to calculate the adaptive hypersphere threshold. Based upon the restriction of Eq. (3.29) and Figure (3.4), the range of K_1 is found to be

$$-\sqrt{\frac{\{R_{H1}\}^2}{\left(\Psi_2-\Psi_1\right)\left(\Psi_2-\Psi_1\right)}} \leq K_1 \leq \frac{1}{2} \tag{3.36}$$

After solving for the magnitude of K_1, some perturbation magnitude for β must be assigned. Thus, substitution of the relation

$$\Psi_\delta = \Psi_1 - \Delta \tag{3.37}$$

and Eq. (3.18) into Eq. (3.13) gives the second level adaptive hypersphere threshold.

$$\left(\Upsilon+\beta\Psi_2-\beta\Psi_1-\Psi_1\right)^t\left(\Upsilon+\beta\Psi_2-\beta\Psi_1-\Psi_1\right)\leq\left(R_A\right)^2, \text{ decide class } \omega_1 \tag{3.38a}$$

otherwise decide class ω_2 (3.38b)

A problem that still exist is the union of three or more adaptive hyperspheres. This could be overcome by additional levels of adaptive thresholds with increased complexity. Figure (3.5) illustrates how such a region could exist. For purposes of this study all vector samples falling within this region are considered as unclassifiable. The evaluation of this problem will be suggested for further research.

All vector samples falling in the convex hull region D would form the following logic from the second level hypersphere threshold gates: $W_1=\bar{B}$, $W_2=\bar{C}$, and $W_3=\bar{A}$.

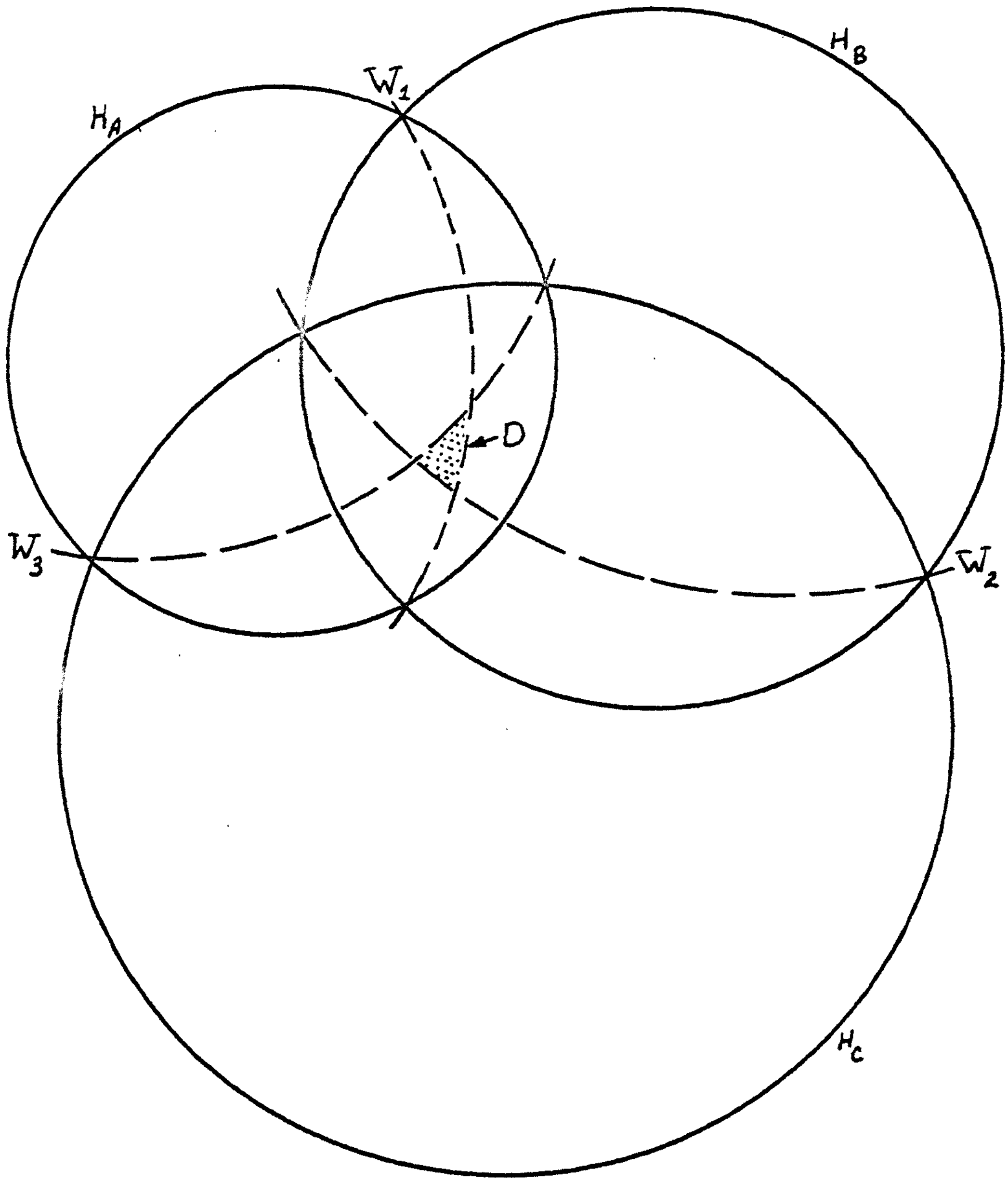


Figure (3.5). Union of Three Adaptive Hyperspheres

3.2 Probability of Error

Two sources of error resulting from the limited number of learning patterns are (1) unknown vectors samples falling outside the first level hypersphere and (2) unknown vector samples misclassified by the second level hypersphere.

The first error can be determined by the probability that an unknown vector is greater than the learning patterns. Since there are an infinite number of possible patterns per class, consider that a learning pattern is selected at random and is independent of any previous learning pattern selection. Letting the event A_i be a pattern selection with some $\{R_i\}^2$, one can write

$$P\left(A_1, A_2, A_3, \ldots, A_\chi\right) = P\left(A_1\right) P\left(A_2\right) \cdots\cdots P\left(A_\chi\right) \quad (3.39)$$

where

$$\left(R_i\right)^2 = \left(Y - \Psi_\alpha\right)^t \left(Y - \Psi_\alpha\right) \quad (3.40)$$

Chebyshev's inequality [17] can be used to evaluate the probability that $\{R_i\}^2$ of pattern A_i exceeds some value. Let $\{R_i\}^2$ be a random variable with $E\{(R_i)^2\} = \mu$ and $V\{(R_i)^2\} = \sigma^2$. Then for any positive number C one can consider

$$P\left(\left|\{R_i\}^2 - \mu\right| \geq C\sigma\right) \leq \frac{1}{C^2} \quad (3.41)$$

or

$$P\left(\left|\{R_i\}^2-\mu\right| < C\sigma\right) \geq 1-\frac{1}{C^2} \tag{3.42}$$

If χ learning patterns are generated at random then from Eq. (3.39)

$$P\left(\left|\{R_i\}^2-\mu\right|_{ALL\ i} \geq C\sigma\right) = P\left(\left|\{R_1\}^2-\mu\right| \geq C\sigma\right) \cdots\cdots$$

$$\cdots\cdots P\left(\left|\{R_\chi\}^2-\mu\right| \geq C\sigma\right) \tag{3.43}$$

and

$$P\left(\left|\{R_i\}^2-\mu\right|_{ALL\ i} < C\sigma\right) = P\left(\left|\{R_1\}^2-\mu\right| < C\sigma\right) \cdots\cdots$$

$$\cdots\cdots P\left(\left|\{R_\chi\}^2-\mu\right| < C\sigma\right) \tag{3.44}$$

which simplifies to

$$P\left(\left|\{R_i\}^2-\mu\right|_{ALL\ i} \geq C\sigma\right) \leq \left(\frac{1}{C^2}\right)^{\chi} \tag{3.45}$$

and

$$P\left(\left|\{R_i\}^2-\mu\right|_{ALL\ i} < C\sigma\right) \geq \left(1-\frac{1}{C^2}\right)^{\chi} \tag{3.46}$$

Now consider the problem in terms of the probability that $\chi-1$ learning patterns are less than any learning pattern selected at random, A_K. By setting

$$\left| \{R_K\}^2 - \mu \right| = C\sigma \tag{3.47}$$

Eq. (3.46) can be rewritten in the form

$$P\left(\left| \{R_\iota\}^2 - \mu \right|_{\text{ALL } \iota \neq K} < C\sigma \right) \geq \left(1 - \frac{1}{C^2}\right)^{X-1} \tag{3.48}$$

The probability that the $(R)^2$ of an unclassified pattern

$$\{R\}^2 = \left(Y - \Psi_\alpha\right)^t \left(Y - \Psi_\alpha\right) \tag{3.49}$$

is bounded by the value of $(R_K)^2$ for all ι is

$$P\left(\{R\}^2 < \{R_K\}^2 \right) = P_o \tag{3.50}$$

If the expected value is subtracted from both sides of the inequality then

$$P\left(\{R\}^2 < \{R_K\}^2 \right) = P\left(\{R\}^2 - \mu < \{R_K\}^2 - \mu \right) \tag{3.51}$$

However, since

$$P\left(\{R\}^2 - \mu < \{R_K\}^2 - \mu \right) = P\left(\left| \{R\}^2 - \mu \right| < \left| \{R_K\}^2 - \mu \right| \right) + P\left(\{R\}^2 - \mu < \{R_K\}^2 - 2\mu \right) \tag{3.52}$$

it follows that

$$P\left(\{R\}^2-\mu<\{R_K\}^2-\mu\right)\geq P\left(\left|\{R\}^2-\mu\right|<\left|\{R_K\}^2-\mu\right|\right) \quad (3.53)$$

Then from Eq. (3.47) and (3.53) one obtains

$$P\left(\{R\}^2-\mu<\{R_K\}^2-\mu\right)\geq P\left(\left|\{R\}^2-\mu\right|<C\sigma\right) \quad (3.53)$$

The substitution of Eq. (3.42) into Eq. (3.53) will yield

$$P\left(\{R\}^2-\mu<\{R_K\}^2-\mu\right)\geq 1-\frac{1}{C^2} \quad (3.54)$$

and finally

$$P\left(\{R\}^2<\{R_K\}^2\right)\geq 1-\frac{1}{C^2} \quad (3.55)$$

With the development of Eq. (3.46), (3.48) and (3.55) some idea as to the probability an unclassified vector sample will lie within the first level hypersphere can be obtained. From Eq. (3.46) it is possible to calculate the value for which

$$P\left(\left|\{R_L\}^2-\mu\right|_{\text{All } L}<C\sigma\right)=\rho \quad (3.56)$$

The substitution of Eq. (3.46) into (3.56) will yield

$$\left(1-\frac{1}{C^2}\right)^{X}=\rho \quad (3.57)$$

We can solve for C^2 and obtain

$$C^2 = \frac{1}{1-\rho^{1/\chi}} \tag{3.58}$$

and finally the substitution of Eq. (3.58) into (3.48) will yield

$$P\left(\left|\{R_i\}^2-\mu\right|_{ALL\ i\neq K} < C\sigma\right) \geq \rho^{\frac{\chi-1}{\chi}} \tag{3.59}$$

which is the probability that $\chi-1$ learning patterns are bounded by the first level hypersphere threshold. The substitution of Eq. (3.58) into (3.55) will yield the probability that any pattern selected at random will fall within the first level hypersphere threshold.

$$P\left(\{R\}^2 < \{R_K\}^2\right) \geq \rho^{\frac{1}{\chi}} \tag{3.60}$$

Eq. (3.59) and (3.60) are plotted in Figure (3.6) for selected values of ρ. This can be used to obtain the probability that an unknown vector sample is within the first level hypersphere threshold. As an example, selection of some $C\sigma$ such that $\rho=.5$ (50 percent) in Eq. (3.56) with $\chi=20$ then from Figure (3.6) the probability an unknown vector sample is bounded by the first level hypersphere is 96.6 %.

The purpose of the second level hypersphere, as previously stated, is to separate the union of two first level hyperspheres. Using Eq. (3.38), the misclassification function may be defined as

$$f_r \triangleq f(\omega_2/\omega_r) - f(\omega_1/\omega_r) \tag{3.61}$$

The second level hypersphere magnitude as a function of β, Eq. (3.25) or (3.35), would be calculated to determine the magnitude of β which minimizes Eq. (3.61) for the learning patterns. Thus one would obtain the best estimate in minimizing the second error source.

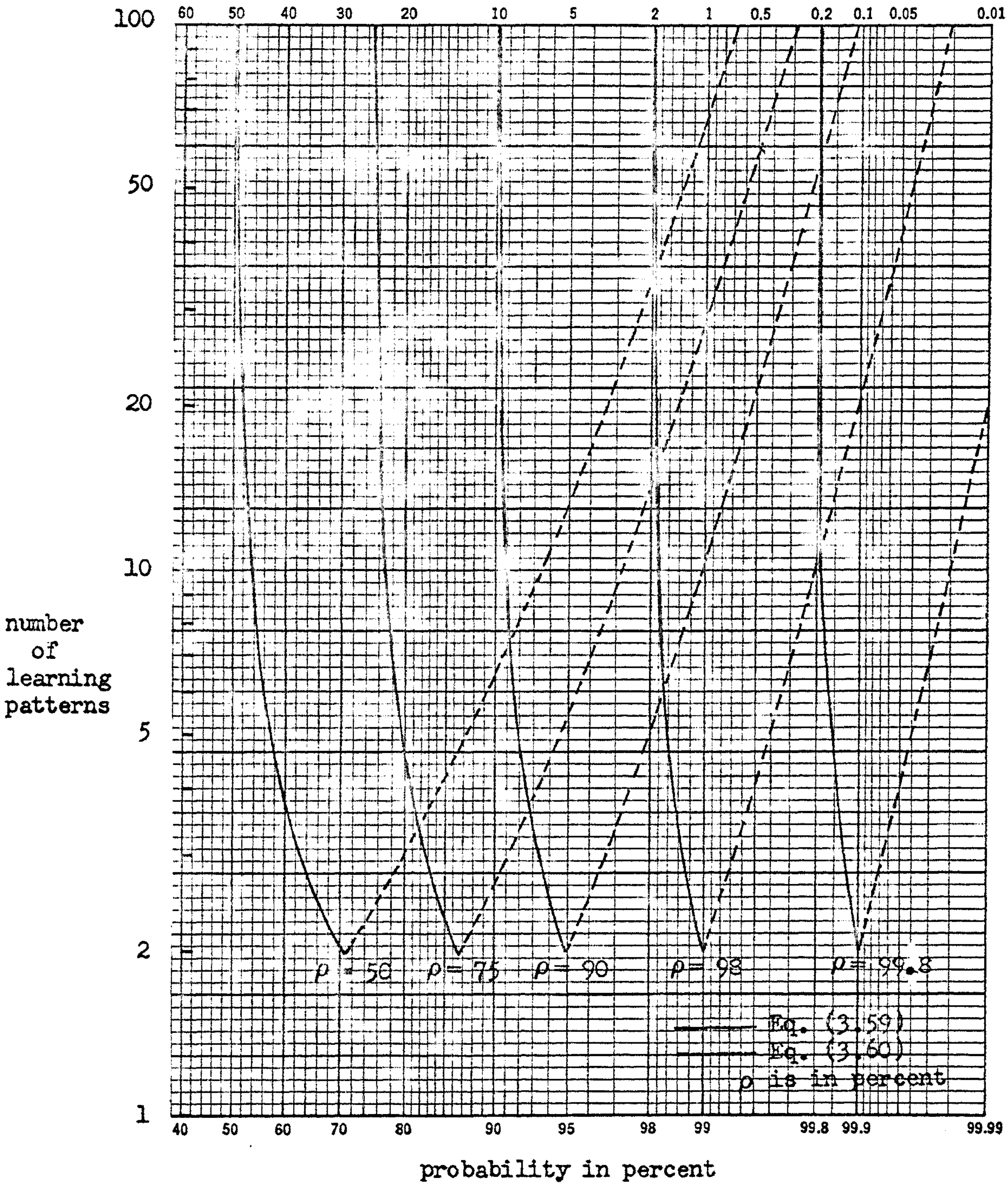


Figure (3.6). Probability Bounds

CHAPTER IV

IMPLEMENTATION AND COMPUTER SIMULATION OF THE AHDT CLASSIFIER

4.1 Quantizing and Coding the Parameter Space

An infinite number of pattern classes could be generated for the computer simulation of the AHDT classifier; however, for practical purposes the number of pattern classes will be limited to some finite number. Let the following characteristics be common to all pattern classes.

1) All patterns are real and symmetrical about $T_i/2$.

$$f_i[t] = f_i[T_i - t] \qquad (4.1)$$

2) All patterns are of equal period.

$$T_i = T = 100 \qquad (4.2)$$

3) All patterns are periodic and have equal power content.

$$P_{t\,i} = \frac{1}{T}\int_0^T \left\{ f_i[t] \right\}^2 dt \qquad (4.3)$$

Based upon these characteristics several pattern classes will be constructed.

The amplitude of pattern class No. 1 is defined by Eq. (4.4). A plot of this pattern is shown in Figure (4.1).

$$f_1[t] = t \qquad , 0 \leq t \leq T/2 \tag{4.4a}$$

$$f_1[t] = T - t \qquad , T/2 < t \leq T \tag{4.4b}$$

The power content P_{t1} is determined by

$$P_{t1} = \frac{1}{T}\int_0^T \left\{ f_1[t] \right\}^2 dt \tag{4.5}$$

and becomes in this case

$$P_{t1} = \frac{1}{T}\int_0^{T/2} t^2 dt + \frac{1}{T}\int_{T/2}^{T} (T-t)^2 dt \tag{4.6}$$

The evaluation of the integrals yield

$$P_{t1} = \frac{T^2}{12} \tag{4.7}$$

Pattern class No. 2, Figure (4.2), is described by the equation

$$f_2[t] = C_2 \, \text{Sin}\left[\frac{\pi t}{T}\right] \qquad , 0 \leq t \leq T \tag{4.8}$$

The constant C_2 can be evaluated using the equal power content requirement. Where

$$P_{t2} = \frac{1}{T}\int_0^T \left\{ C_2 \text{Sin}\left[\frac{\pi t}{T}\right] \right\}^2 dt \tag{4.9}$$

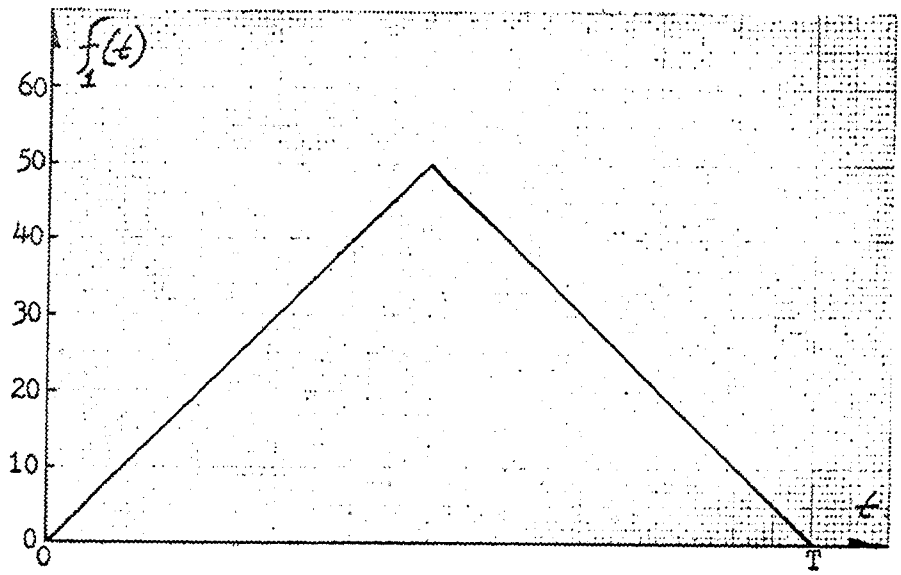


Figure (4.1). Pattern Class No. 1

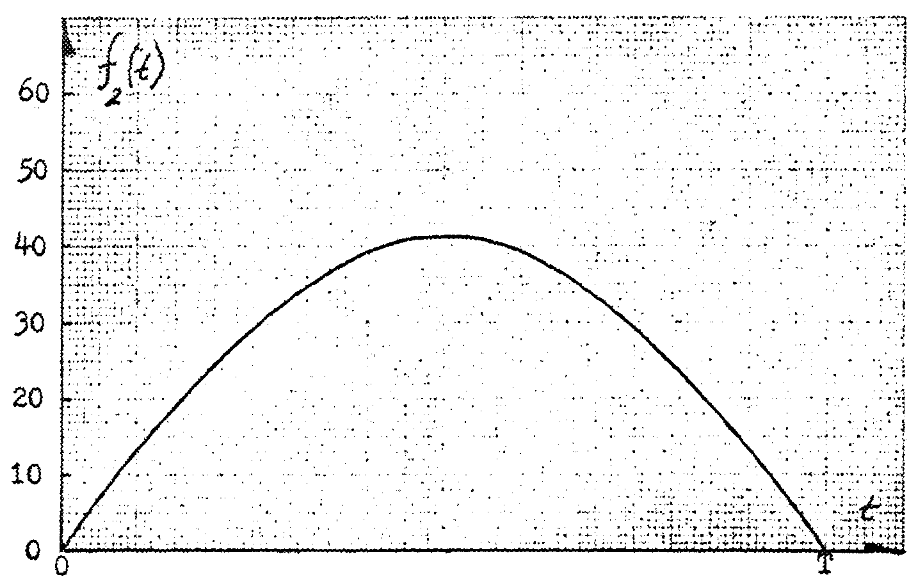


Figure (4.2). Pattern Class No. 2

The evaluation of P_{t2} will yield

$$P_{t2} = \frac{1}{2}\left(C_2\right)^2 \tag{4.10}$$

Thus

$$C_2 = T\sqrt{\frac{1}{6}} \tag{4.11}$$

A plot of pattern class No. 3, which is described by

$$f_3[t] = C_3 t^2 \quad , \quad 0 \le t \le T/2 \tag{4.12a}$$

$$f_3[t] = C_3\left(T-t\right)^2 \quad , \quad T/2 < t \le T \tag{4.12b}$$

is shown by Figure (4.3). Again the constant is evaluated from the power content requirement.

$$P_{t3} = \frac{1}{T}\int_0^{T/2}\left\{C_3 t^2\right\}^2 dt + \frac{1}{T}\int_{T/2}^{T}\left\{C_3\left(T-t\right)^2\right\}^2 dt \tag{4.13}$$

Solving for

$$P_{t3} = \left(C_3\right)^2 T^4 \left(\frac{1}{80}\right) \tag{4.14}$$

The value of C_3 is then

$$C_3 = \frac{1}{T}\sqrt{\frac{20}{3}} \tag{4.15}$$

The amplitude of pattern class No. 4

$$f_4[t] = C_4\left(1 - \cos\left\{\frac{\pi t}{T}\right\}\right) \quad , \quad 0 \le t \le T/2 \quad (4.16a)$$

$$f_4[t] = C\left(1 + \cos\left\{\frac{\pi t}{T}\right\}\right) \quad , \quad T/2 < t \le T \quad (4.16b)$$

is shown by Figure (4.4). The power content is

$$P_{t4} = \frac{1}{T}\int_0^{T/2}\left\{C_4\left(1 - \cos\left\{\frac{\pi t}{T}\right\}\right)\right\}^2 dt + \frac{1}{T}\int_{T/2}^{T}\left\{C_4\left(1 + \cos\left\{\frac{\pi t}{T}\right\}\right)\right\}^2 dt \quad (4.17)$$

The solution of this equation will yield

$$P_{t4} = \left(C_4\right)^2\left(\frac{3\pi - 8}{2\pi}\right) \quad (4.18)$$

The constant C_4 is evaluated using the requirement that $P_{t4} = P_{t1}$. Thus

$$C_4 = T\sqrt{\frac{\pi}{18\pi - 48}} \quad (4.19)$$

Pattern Class No. 5 is shown by Figure (4.5). The amplitude is defined by

$$f_5[t] = C_5\left(1 - \exp\left\{-\frac{t}{T/8}\right\}\right) \quad , \quad 0 \le t \le T/2 \quad (4.20a)$$

$$f_5[t] = C_5\left(1 - \exp\left\{-\frac{T - t}{T/8}\right\}\right) \quad , \quad T/2 < t \le T \quad (4.20b)$$

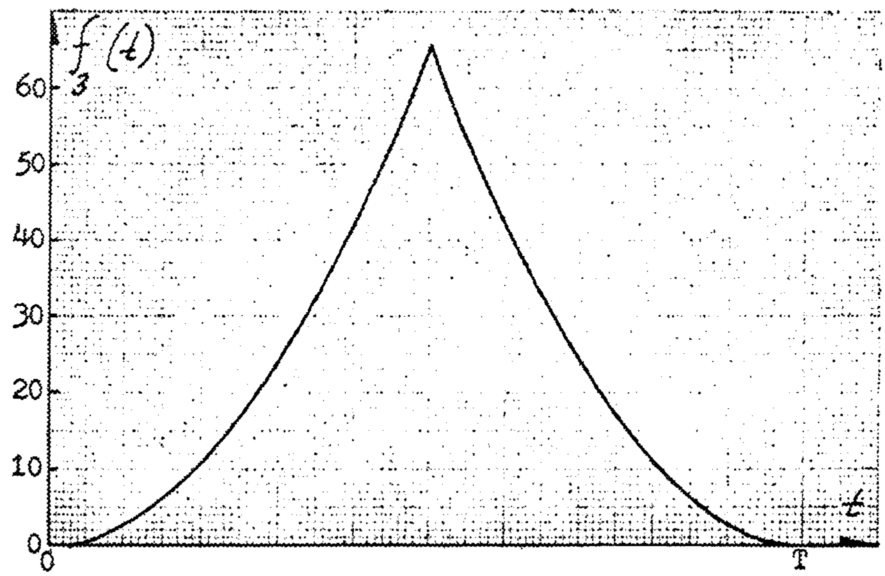


Figure (4.3). Pattern Class No. 3

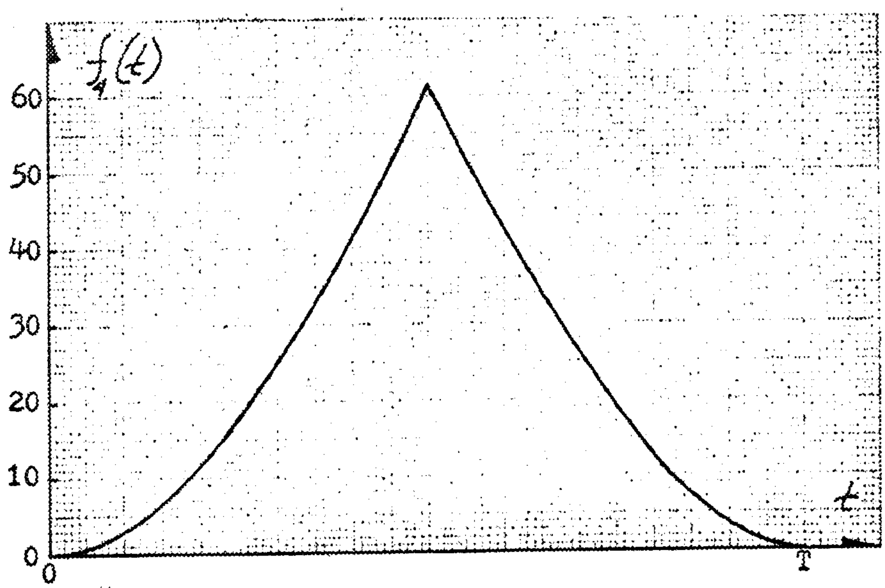


Figure (4.4). Pattern Class No. 4

$$P_{t5} = \frac{1}{T}\int_{0}^{T/2}\left\{C_5\left(1-\exp\left\{-\frac{t}{T/8}\right\}\right)\right\}^2 dt + \frac{1}{T}\int_{T/2}^{T}\left\{C_5\left(1-\exp\left\{-\frac{T-t}{T/8}\right\}\right)\right\}^2 dt \quad (4.21)$$

will yield

$$P_{t5} = \left(C_5\right)^2\left(\frac{5}{8} + \frac{e^{-4}}{2} - \frac{e^{-8}}{8}\right) \quad (4.22)$$

From the equality $P_{t5} = P_{t1}$, the value of C_5 is

$$C_5 = T\sqrt{\frac{1}{7.5 + 6e^{-4} - 1.5e^{-8}}} \quad (4.23)$$

Figure (4.6) is a plot of pattern Class No. 6, where

$$f_6\left[t\right] = C_6\left(1 - \mathrm{Cos}\left\{\frac{2\pi t}{T}\right\}\right) \quad , \quad 0 \le t \le T \quad (4.24)$$

The power content is

$$P = \frac{1}{T}\int_{0}^{T}\left\{C_6\left(1 - \mathrm{Cos}\left\{\frac{2\pi t}{T}\right\}\right)\right\}^2 dt \quad (4.25)$$

This will yield a power content of

$$P_{t6} = \left(C_6\right)^2\left(\frac{3}{2}\right) \quad (4.26)$$

solving for C_6, where $P_{t6} = P_{t1}$,

$$C_6 = T\sqrt{\frac{1}{18}} \quad (4.27)$$

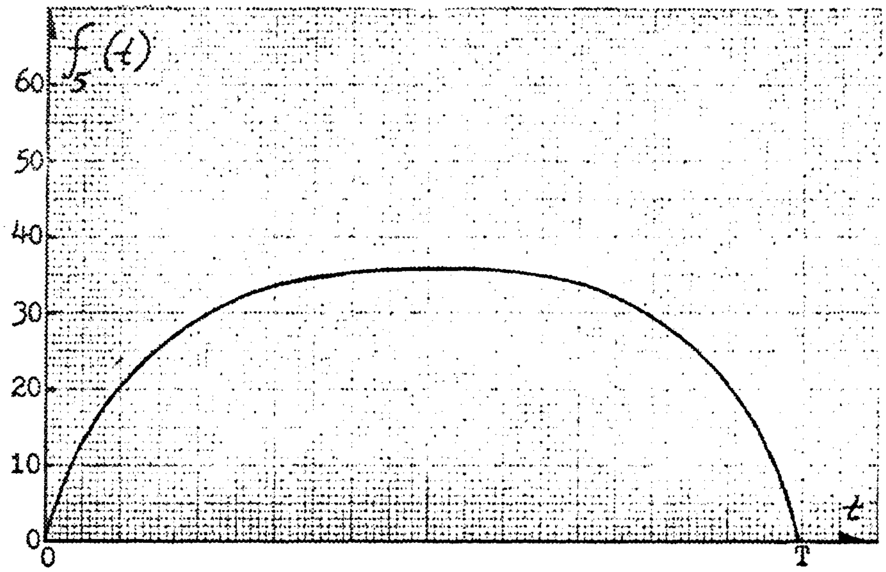


Figure (4.5). Pattern Class No. 5

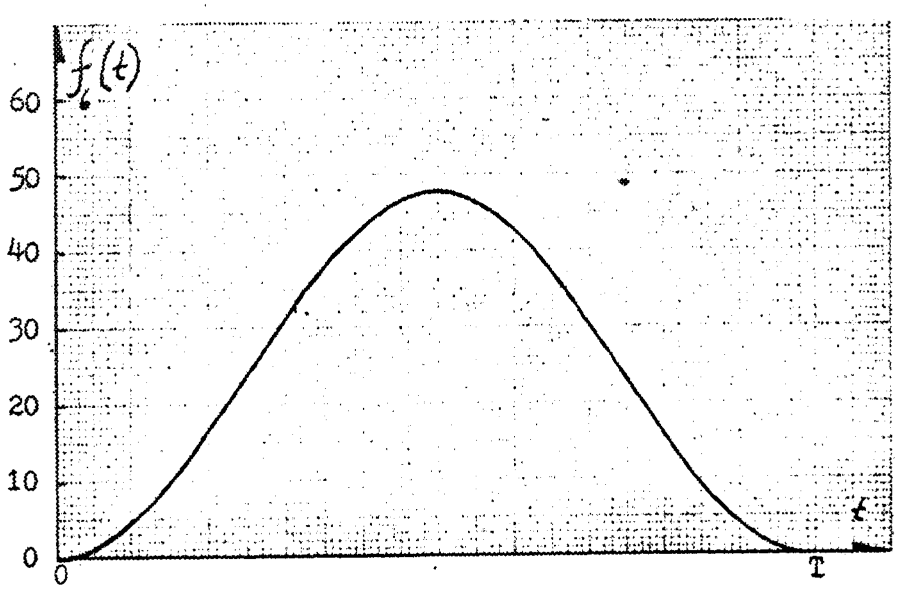


Figure (4.6). Pattern Class No. 6

Pattern class No. 7, Figure (4.7), is a square wave pulse whose amplitude is

$$f_7[t] = 0 \quad , \quad 0 \leq t < T/4 \qquad (4.28a)$$

$$f_7[t] = C_7 \quad , \quad T/4 \leq t \leq 3T/4 \qquad (4.28b)$$

$$f_7[t] = 0 \quad , \quad 3T/4 < t \leq T \qquad (4.28c)$$

The power content

$$P_{t7} = \frac{1}{T}\int_{T/4}^{3T/4} \left(C_7\right)^2 dt \qquad (4.29)$$

is

$$P_{t7} = \left(C_7\right)^2 \frac{1}{2} \qquad (4.30)$$

From the equality $P_{t7} = P_{t1}$.

$$C_7 = T\sqrt{\frac{1}{6}} \qquad (4.31)$$

Figure (4.8) shows pattern class No. 8, where the amplitude is

$$f_8[t] = 0 \quad , \quad 0 \leq t < 2T/5 \qquad (4.32a)$$

$$f_8[t] = C_8 \quad , \quad 2T/5 \leq t \leq 3T/5 \qquad (4.32b)$$

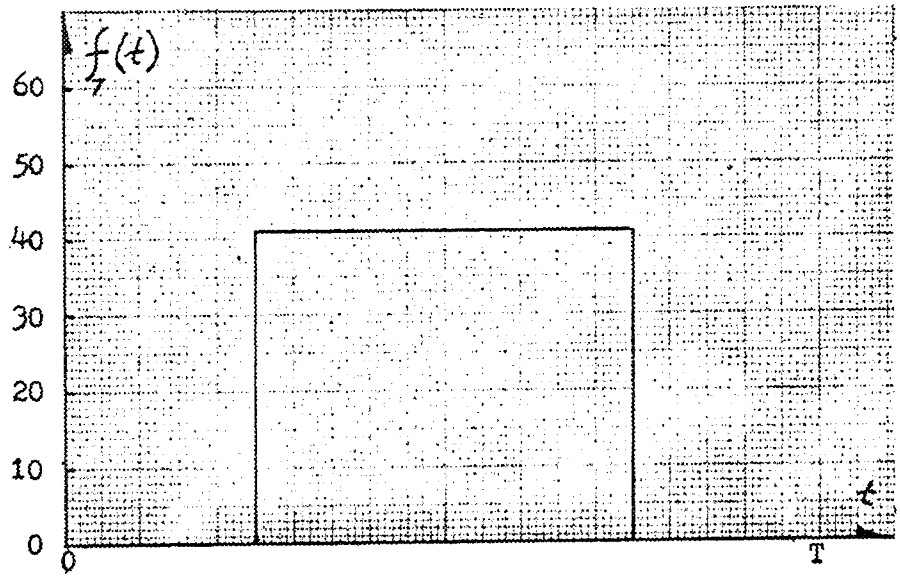


Figure (4.7). Pattern Class No. 7

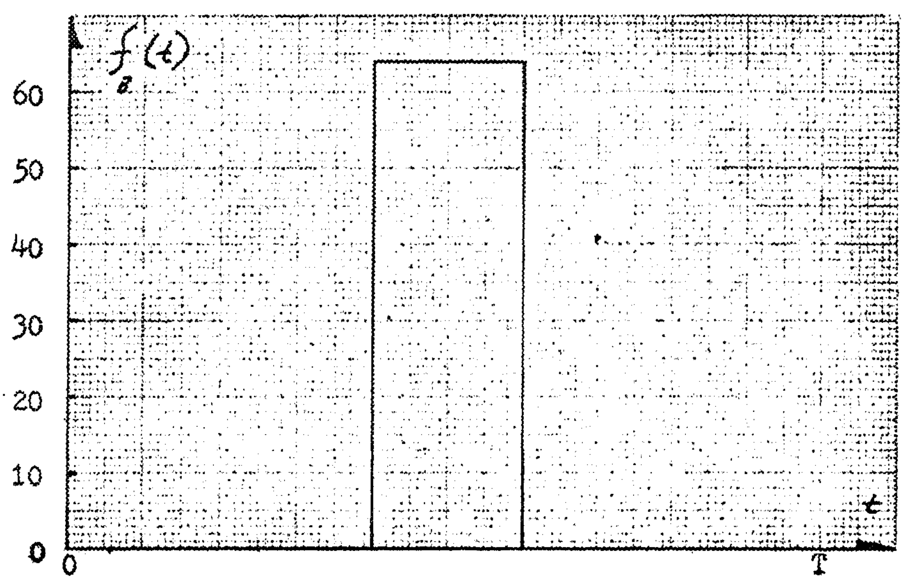


Figure (4.8). Pattern Class No. 8

$$f_8\left[t\right]=0 \qquad , \; 3T/5 < t \leq T \qquad (4.32c)$$

The pattern class power content

$$P_{t8}=\frac{1}{T}\int_{2T/5}^{3T/5}\left(C_8\right)^2 dt \qquad (4.33)$$

is

$$P_{t8}=\frac{1}{5}\left(C_8\right)^2 \qquad (4.34)$$

where C_8 is solved using the equality $P_{t8} = P_{t1}$.

$$C_8 = T\sqrt{\frac{5}{12}} \qquad (4.35)$$

These patterns were chosen with the thought of minimizing the difference between the pattern classes. This would supply information on the ability to separate similar patterns. In a real world sense the pattern shape may be known or obtained by data sampling. The computer simulation uses the fact that the actual patterns are known, as shown in Figures (4.1) through (4.8), to generate the pattern mean vector sample at twenty-five (25) discrete points. This would eliminate the error associated with a mean vector sample estimate obtained from data sampling.

The additive gaussion noise is approximated, using the central limit theorem, as

$$N = \frac{\sum_{T=1}^{K} \left(U_T - E\{U\} \right)}{\sqrt{K\ V\{U\}}} \tag{4.36}$$

where U_T is a uniformally distributed random number between 0 and 1, inclusive. The expect value and variance of U are $E\{U\}=1/2$ and $V\{U\}=1/12$. If we sum up twenty random values of U, $K=20$, then Eq. (4.38) can be rewritten as

$$N = \frac{\sum_{T=1}^{20} U - 10}{\sqrt{\frac{20}{12}}} \tag{4.37}$$

Eq. (4.37) will yield an approximate normally distributed random number truncated at ±10. with a zero mean value and an approximate variance of one.

A covariance matrix estimate is generated for each pattern class. The covariance matrix is constructed by generating a sequence of twenty-five (25) random numbers, Eq. (4.37). A total of five-hundred (500) sequences are used to calculate the points in each pattern class covariance matrix using the equation

$$\sigma_{ij} = \frac{1}{499} \sum_{\gamma=1}^{500} N_{\gamma i}\, N_{\gamma j} \tag{4.38}$$

where

$$V_K = \begin{bmatrix} \sigma_{11} & \sigma_{21} & \cdots & \sigma_{i1} \\ \sigma_{12} & \sigma_{22} & & \vdots \\ \vdots & & \ddots & \vdots \\ \sigma_{1j} & \cdots & \cdots & \sigma_{ij} \end{bmatrix} \tag{4.39}$$

The inverse covariance matrix and determinant are computed and these inverse covariance matrices are used for all signal to noise power ratios by using the relationship.

$$V^{-1}_{\iota, S/N=K_2} = \frac{V^{-1}_{\iota}}{K_2 P_{t\iota}} \tag{4.40}$$

The learning patterns are arbitrarily set at twenty (20) per pattern class for each signal to noise power ratio. These learning patterns are used to generate the adaptive hypersphere thresholds derived in Chapter 3. The amount of computation time for the second level hypersphere threshold is held to a minimum by continuously predicting the solution giving a minimum learning pattern misclassification. For example, given the misclassification function f_t , Eq. (3.61), $f_t[\beta]$ at $\beta = a$ can be calculated and some perturbation introduced such that $\beta = b$. The value of $\beta_{\bullet}$ in Figure (4.9) can be predicted using the linear equation

$$y = m\,\beta + K_3 \tag{4.41}$$

with the slope of the line connecting S_1 and S_2 being

$$m = \frac{f_t[b] - f_t[a]}{b - a} \tag{4.42}$$

The constant K_3 is evaluated at the point $y = f_t(b)$ and $\beta = b$.

$$K_3 = f_t(b) - m\,b \tag{4.43}$$

Substitution of Eq. (4.42) and (4.43) into (4.41) and solving for β at $y=0$ will give the predicted value of β_o yielding the minimum misclassification.

$$\beta_o = \frac{a f_\epsilon(b) - b f_\epsilon(a)}{f_\epsilon(b) - f_\epsilon(a)} \tag{4.44}$$

This linear prediction method is continued until a sign change in f_ϵ occurs. The program logic then switches out of the linear prediction and converges to the point $f_\epsilon = 0$, which is bounded by the values of β in the last linear prediction.

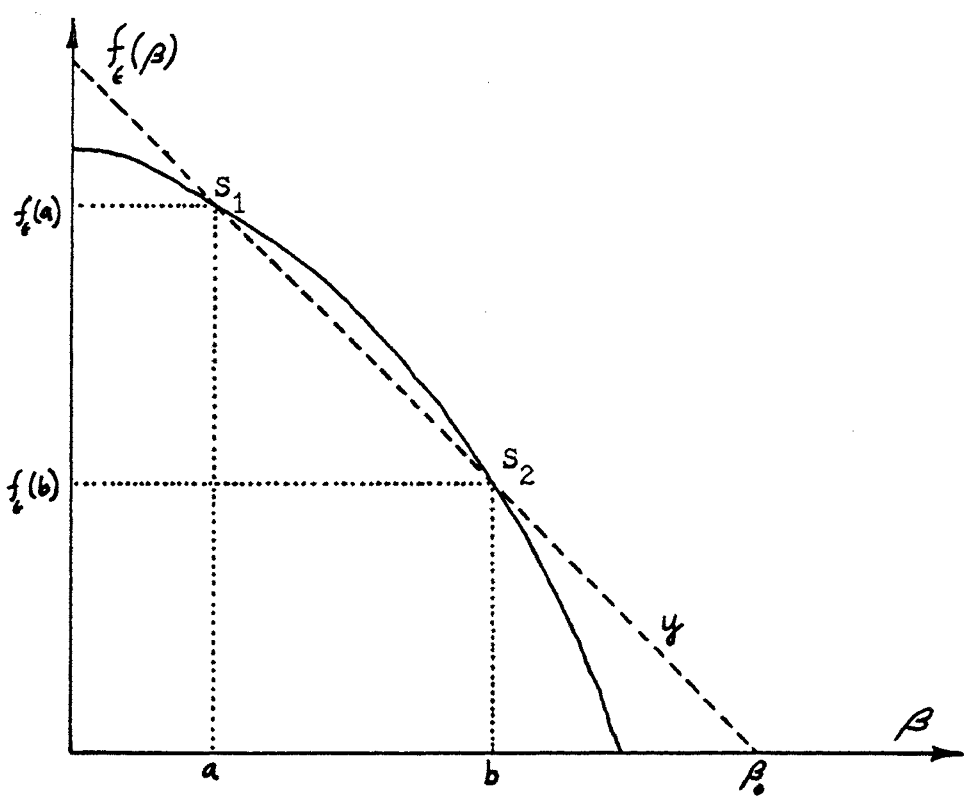


Figure (4.9). Error Function

4.2 Results

The derivations in Chapter III generated several questions about the AHDT classifier. These include:

1) What magnitudes are associated with the first level hypersphere threshold?

2) What is the frequency of the union of the first level hypersphere threshold?

3) Will a hypersphere within a hypersphere exist in an actual case?

4) How does the AHDT classifier compare with the maximum likelihood ratio classifier?

5) Can the second level hypersphere threshold separate two classes in an actual case?

6) How well does an actual case compare with the probability bounds in Figure (3.6)?

7) Will the adaptive hypersphere threshold optimumally separate the pattern classes?

The answers to these questions are supplied by the computer simulation. A listing of the AHDT classifier simulation program is presented in Appendix A. Approximately thirty-one (31) minutes of IBM 360-75 system time is needed for the computations in the main program.

The resultant first level adaptive hypersphere threshold magnitudes are tabulated in Table I. These results are based on a training set of twenty (20) patterns per class. The maximum, average and minimum values are plotted in Figure (4.10). A review of Table I indicates that for a fixed S/N ratio the pattern class order giving a

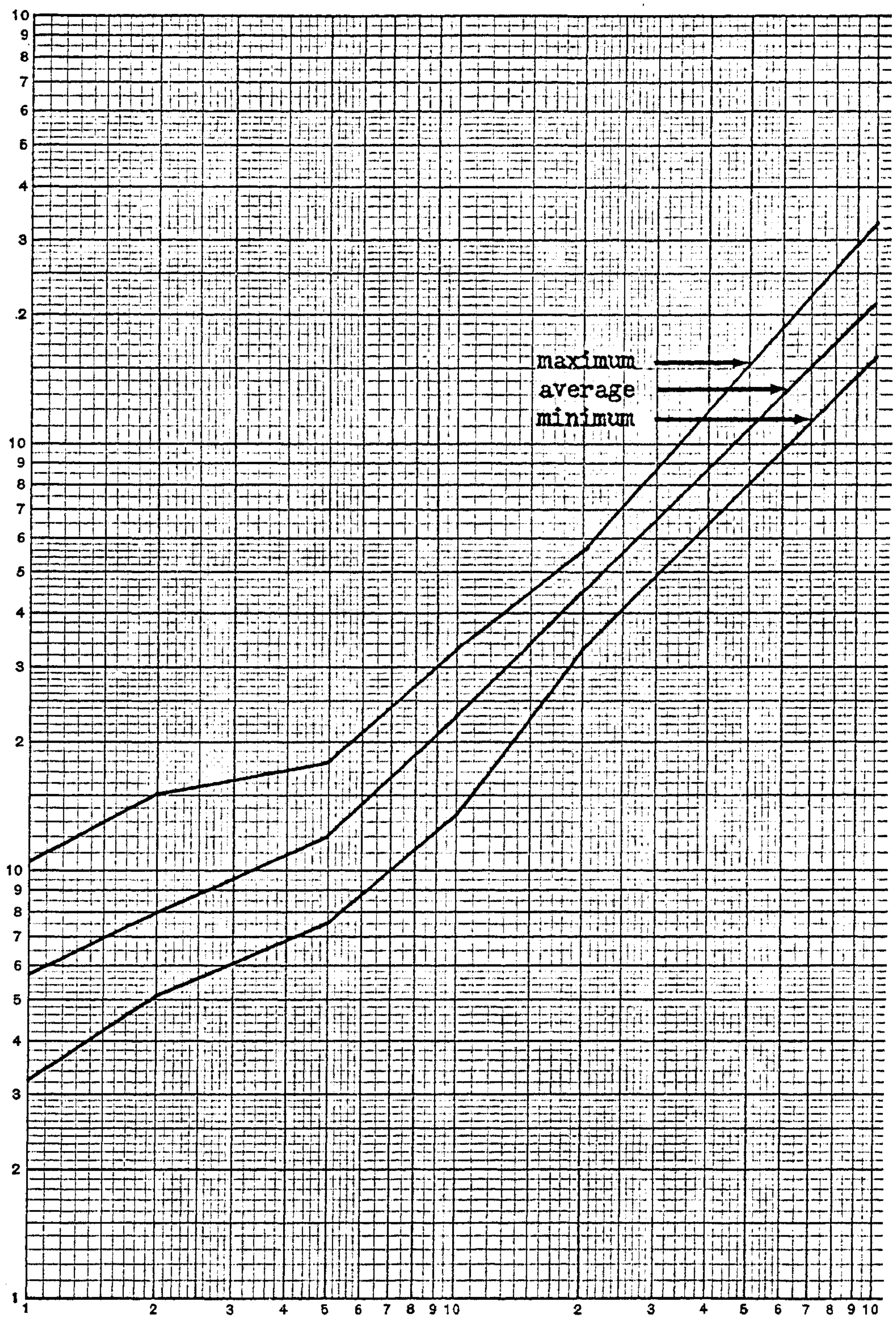
10000

1000

first level hypersphere threshold magnitude

100

10



.1 1.0 10.

signal to noise power ratio

Figure (4.10). Maximum, Average and Minimum First Level Hypersphere Threshold Magnitude which could be explained as resulting from the random training pattern selection. It will be pointed out later that additional research is needed in this area.

Figure (3.5), page 27, illustrates a type of event which occurs with the hypersphere decision threshold. This union of the hyperspheres did occur in the AHDT simulation. In addition, the hypersphere within a hypersphere occurred. The results of the one-hundred (100) unknown patterns per class are presented in Table II, page 66. This data has been converted to a percent of patterns falling within the union and plotted in Figures (4.11), (4.12), (4.13), (4.14), (4.15) and (4.16) for the six (6) S/N ratios. It can be observed in these figures that the data is shifting to a larger number of first level hypersphere thresholds in union. As the S/N is decreased this is to be expected, since the cluster of hyperspheres becomes more compact as the S/N ratio decreases.

The AHDT simulation supplies four (4) error rates. These include: the maximum likelihood ratio classifier misclassification, the AHDT classifier misclassification, the unclassifiable patterns exceeding the first level hypersphere threshold and the unclassifiable patterns not separated by the second level hypersphere. The error rate data, presented in Table XV and plotted in Figure (4.17) as an average misclassification, indicates the usefullness of the AHDT classifier averaged over all classes is suboptimum to the maximum likelihood ratio classifier when the average correct classification of a fixed total is considered. If one ignores unclassifiable patterns, then for S/N ratios less than 2, the AHDT correct classification as a percent

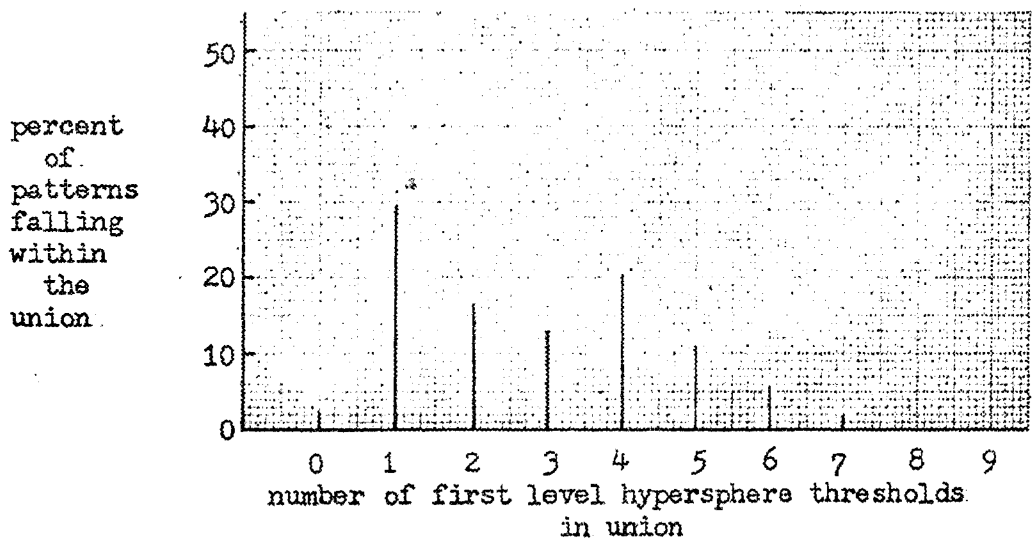


Figure (4.11). Percent of Unknown Patterns Falling Within the Union of First Level Hypersphere Thresholds, S/N=10.

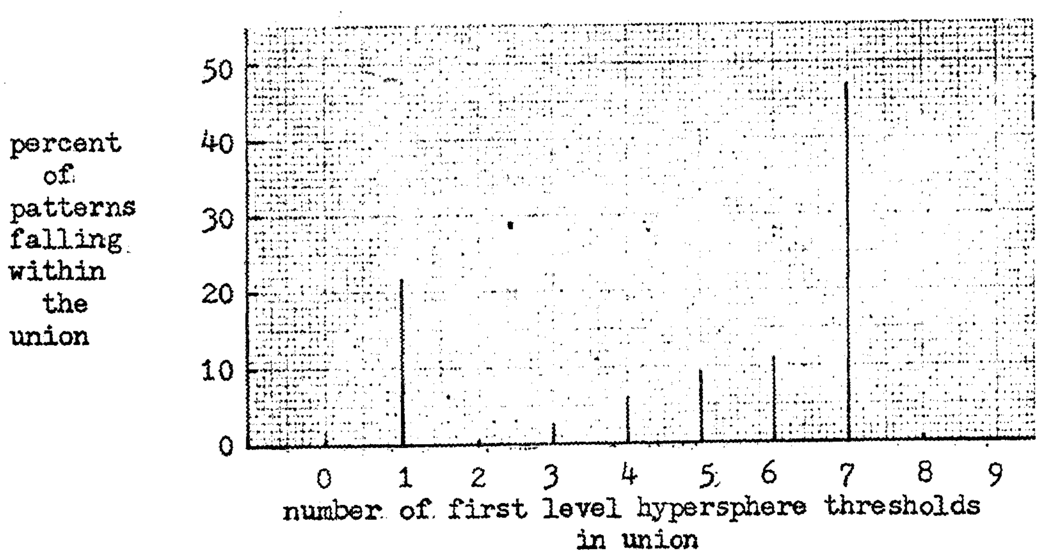


Figure (4.12). Percent of Unknown Patterns Falling Within the Union of First Level Hypersphere Thresholds, S/N=2.

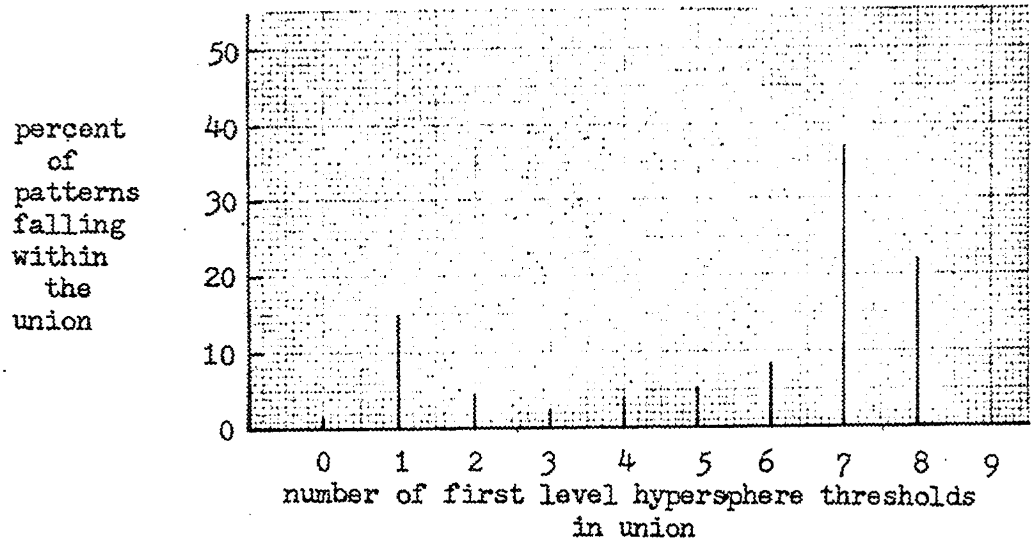


Figure (4.13). Percent of Unknown Patterns Falling Within the Union of First Level Hypersphere Thresholds, S/N=1.

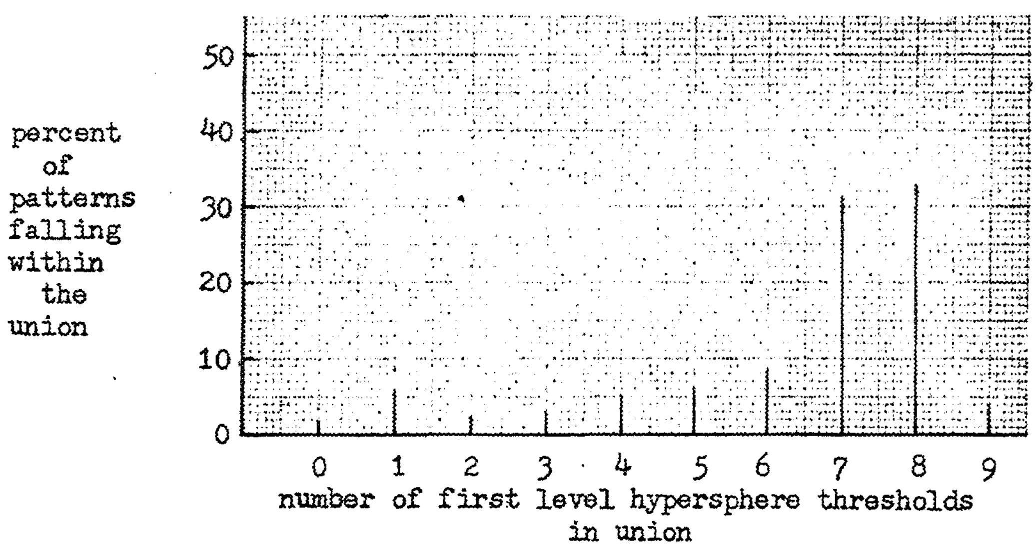


Figure (4.14). Percent of Unknown Patterns Falling Within the Union of First Level Hypersphere Thresholds, S/N=.5

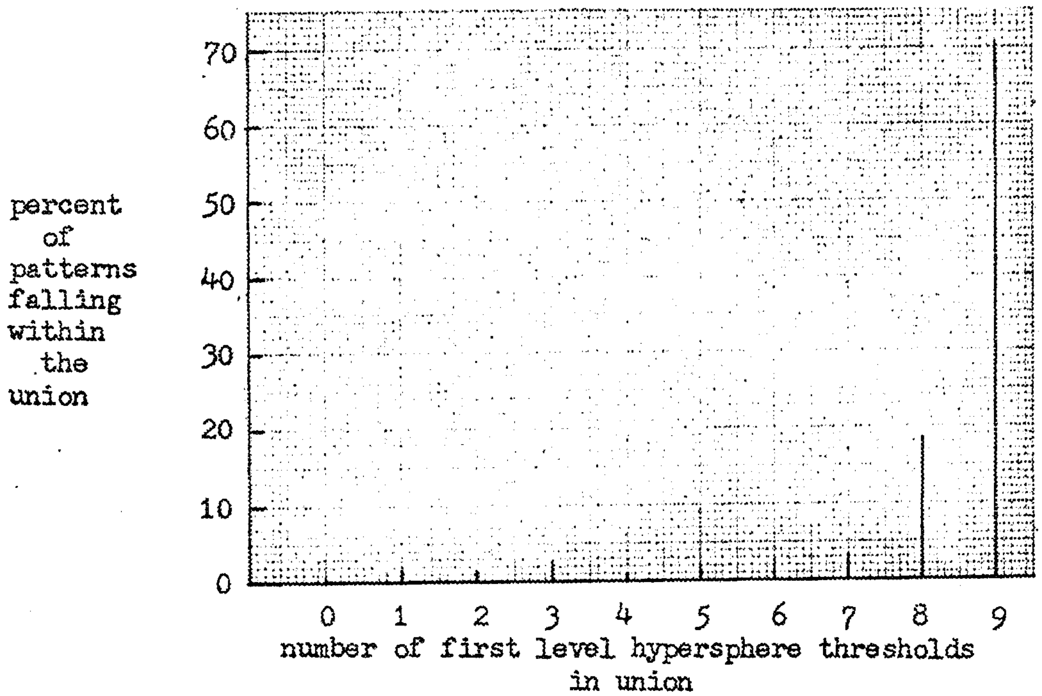


Figure (4.15). Percent of Unknown Patterns Falling Within the Union of First Level Hypersphere Thresholds, S/N=.2

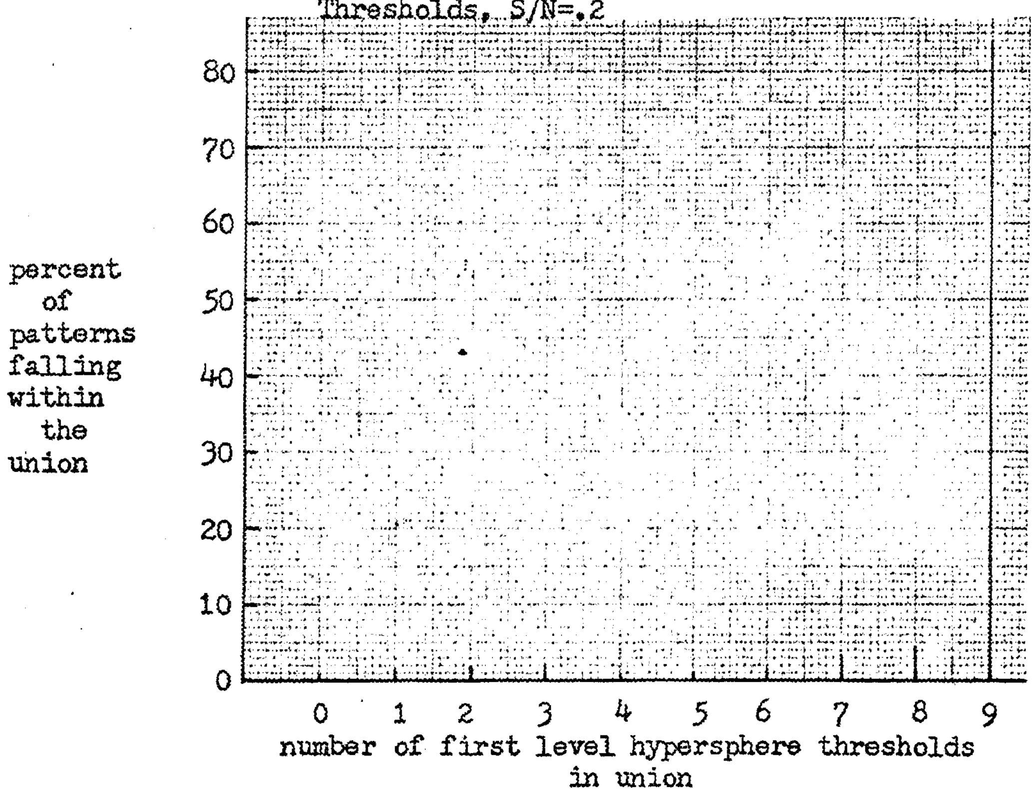


Figure (4.16). Percent of Unknown Patterns Falling Within the Union of First Level Hypersphere Thresholds, S/N=.1

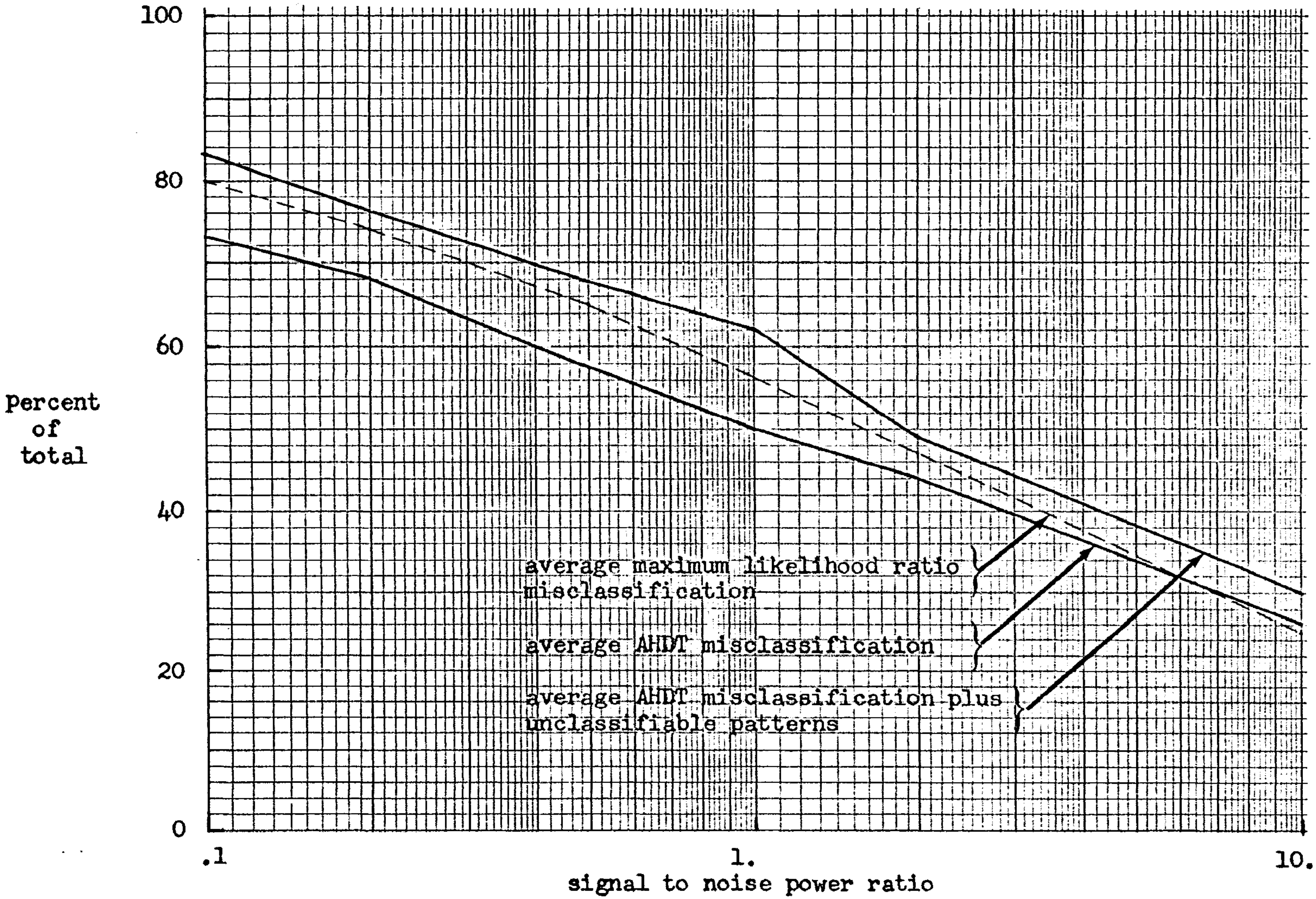


Figure (4.17). Average Error Rate Comparison likelihood correct classification, Figure (4.18). A review of the training set data contained in Table III through XIV shows the objective to minimize the misclassification between two classes was accomplished for each S/N ratio. A comparison between the number of training patterns, Tables III, V, VII, IX, XI and XIII, and the unknown patterns, Tables XVI, XVIII, XX, XXII, XXIV and XXVI, falling within the first level hypersphere threshold indicates a maximum difference of sixty-nine (69) percent. The variation in the first level hypersphere threshold magnitude indicates it as the primary problem source.

If one compares Table XXVIII with Figure (3.6), page 33, there are several cases in which the percent of unknown patterns exceeding the adaptive first level hypersphere threshold falls below the $\rho = 50\%$ curve. In Chapter III, page 32, a sample case was presented for which it was found with $\rho = 50\%$ the probability that an unknown vector sample is bounded by the first level hypersphere is 96.6%. This does not compare with the values listed in Table XXVIII. Thus, additional research is required and recommended to find a method which would optimize the first level hypersphere threshold magnitude selection. Having obtained this optimization, it could be substituted into the AHDT simulation. The results should be compared with Tables XVII, XIX, XXI, XXIII, XXV and XXVII to see if the classification bias has been reduced or eliminated. A review of these tables would show that the computer simulation is biased toward Class 5 with a S/N =10, S/N =2 and S/N =1, toward Class 1 with a S/N =.5 and toward Class 4 with a S/N =.2 and S/N =.1.

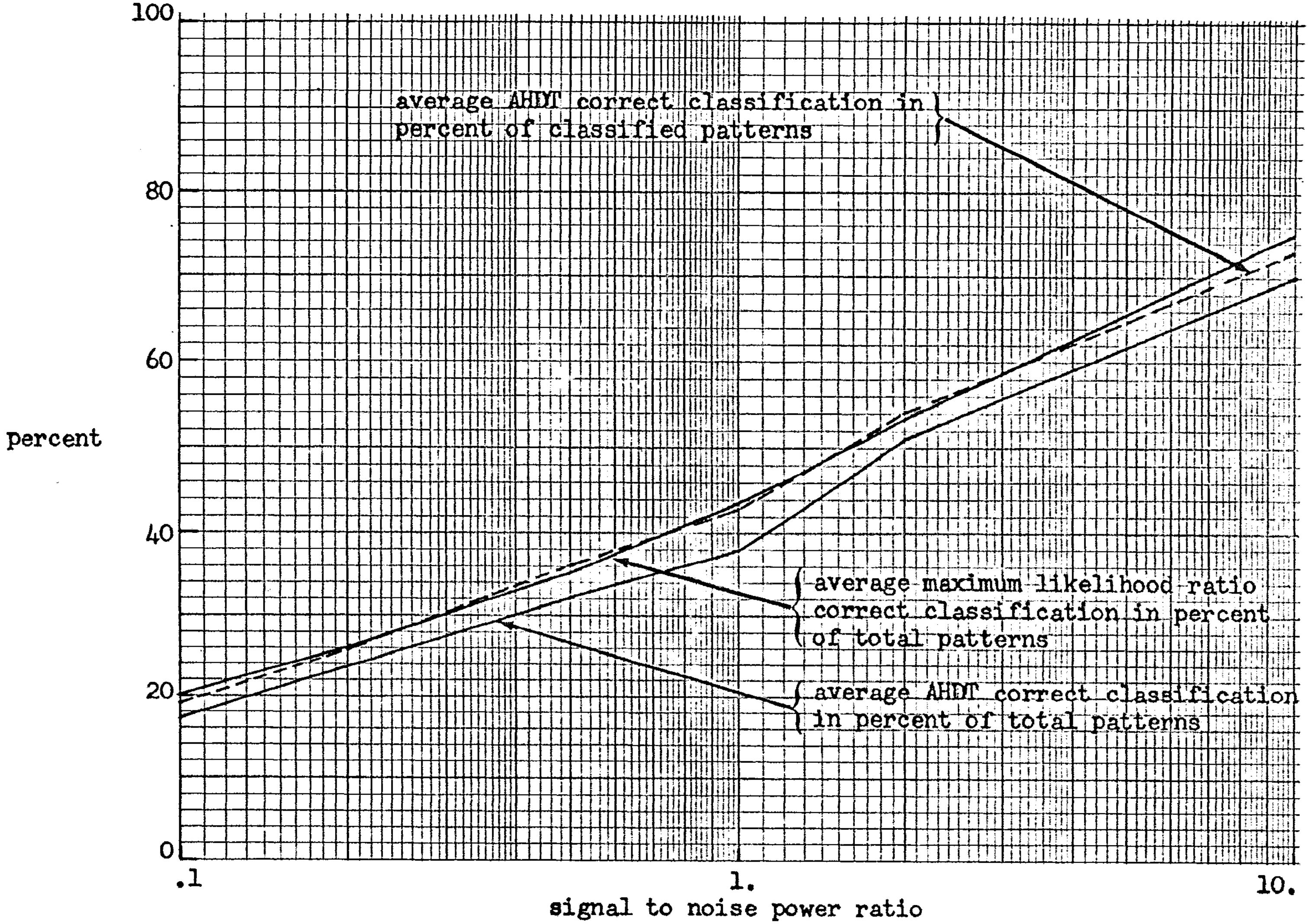


Figure (4.18). Average Correct Classification Comparison

A review of Figures (4.1) through (4.8) indicates the similarity of Class 1 through 6 is such that these classes could be considered as subset classes of a class NU. With this idea Figure (4.19) is presented. It is observed that class NU has a larger percentage of correct classifications than the average correct classification of classes 1 through 6. This increase is due to the difficulty in separating these similar subset classes. It was hoped that the pattern separation using the AHDT would offer an improvement. This is not obvious in the Figures (4.19), (4.20), (4.21), and (4.22).

A comparison between the maximum likelihood ratio classifier and the AHDT classifier is obtained from Figure (4.23) and (4.24). The comparison is based on the separation of signal and noise. It is obvious that the AHDT adds a bias to the maximum likelihood ratio classifier threshold. This bias reduced the false alarm rate by 14% and the correct signal classification by 15% at S/N = .1, based on the total patterns. If unclassifiable patterns are neglected then from Figure (4.25) the bias reduced the false alarm rate and the correct signal classification by 10% and 6.5%, respectively. This indicates an improvement in performance can be obtained with the additional signal processing supplied by the AHDT classifier.

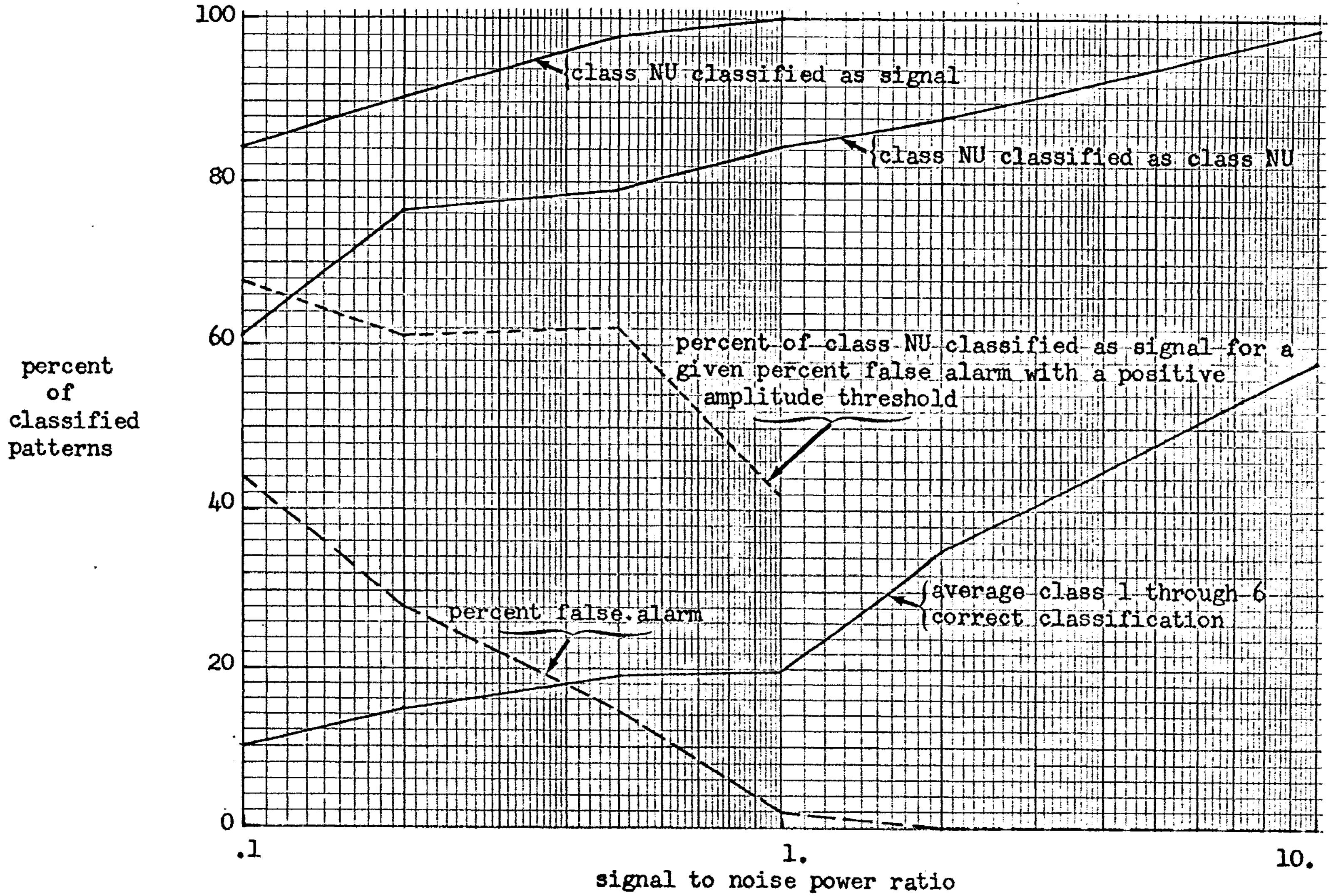


Figure (4.19). AHDT Separation of Class NU

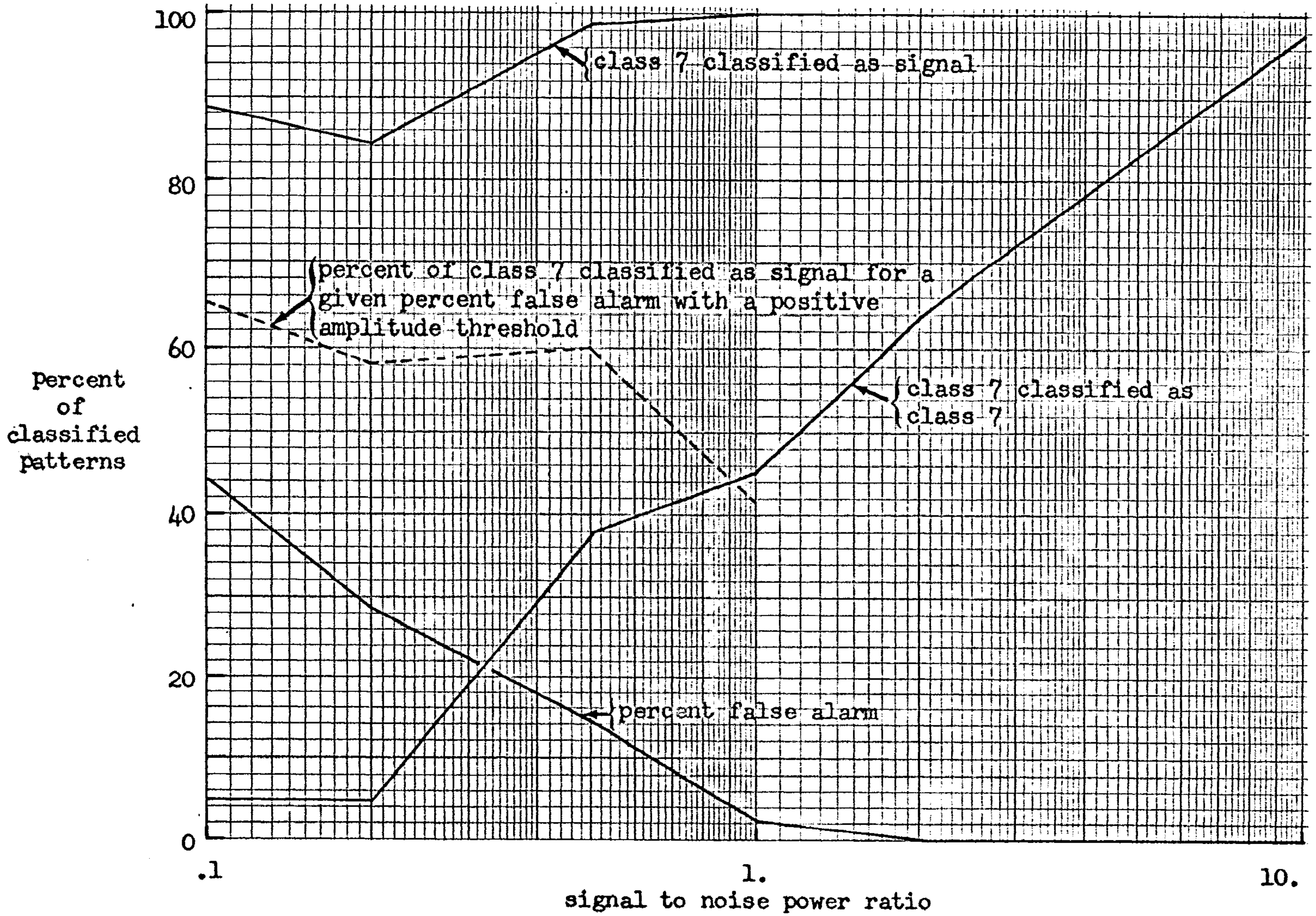


Figure (4.20). AHDT Separation of Class 7

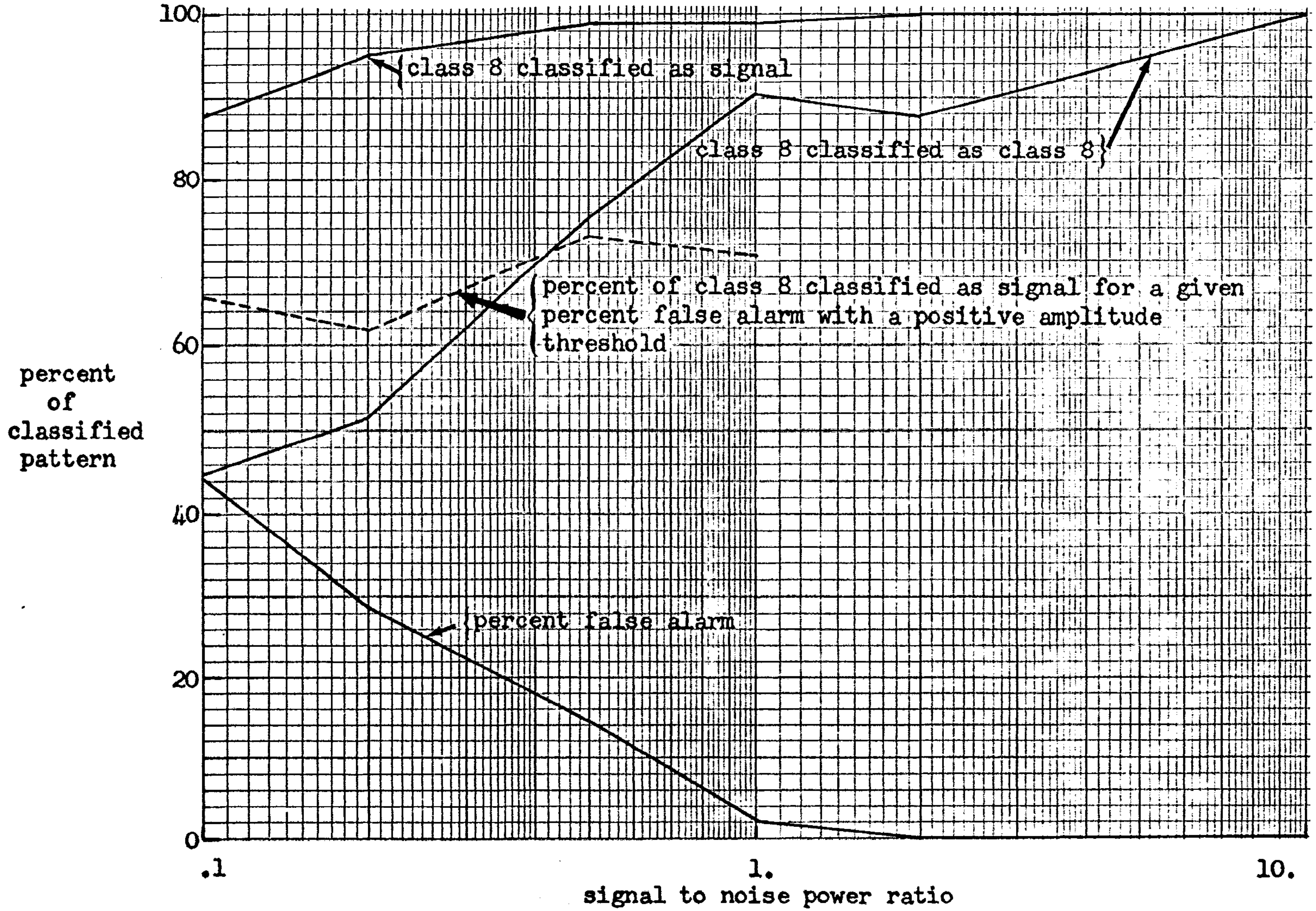


Figure (4.21). AHDT Separation of Class 8

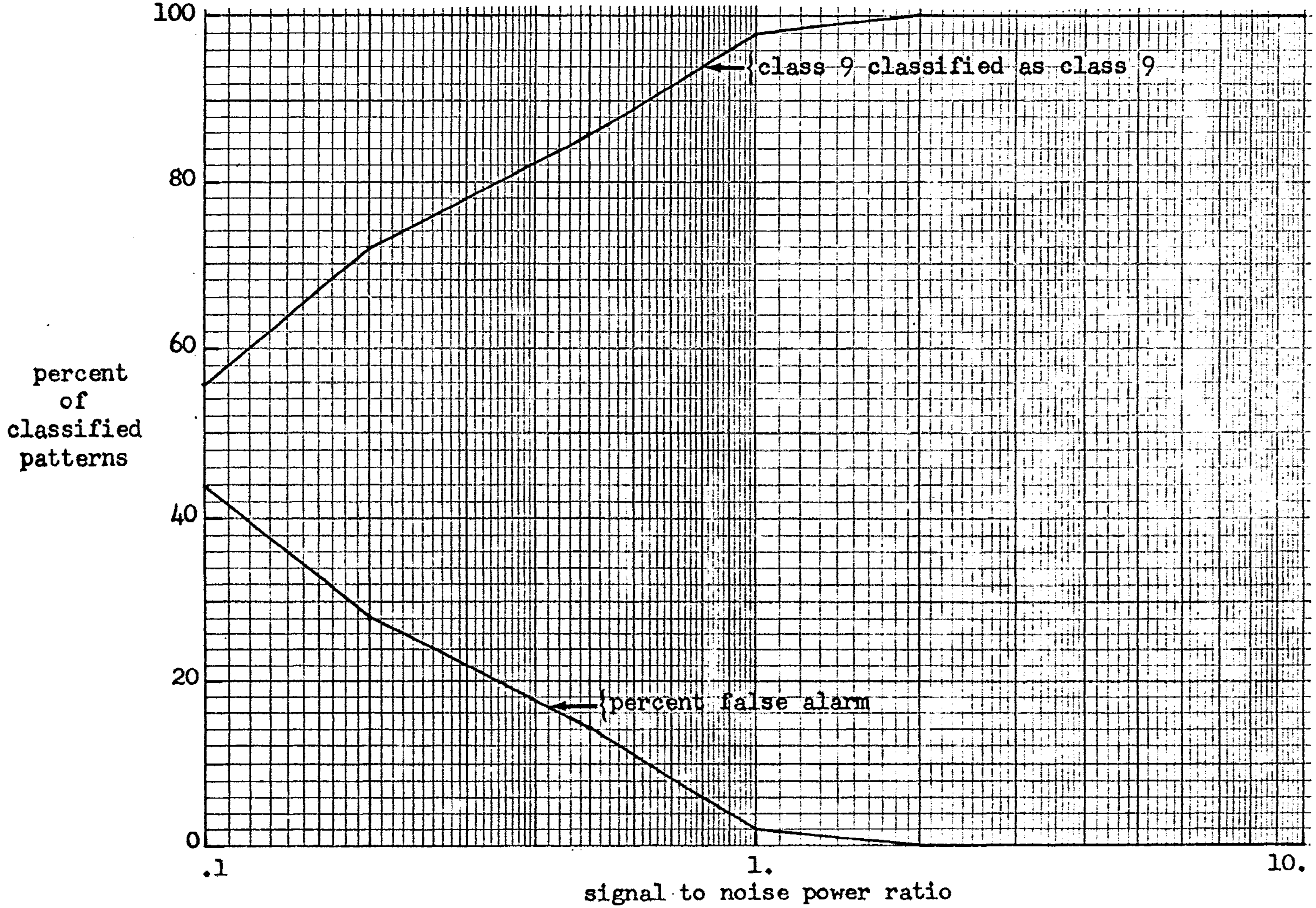


Figure (4.22). AHDT Separation of Class 9

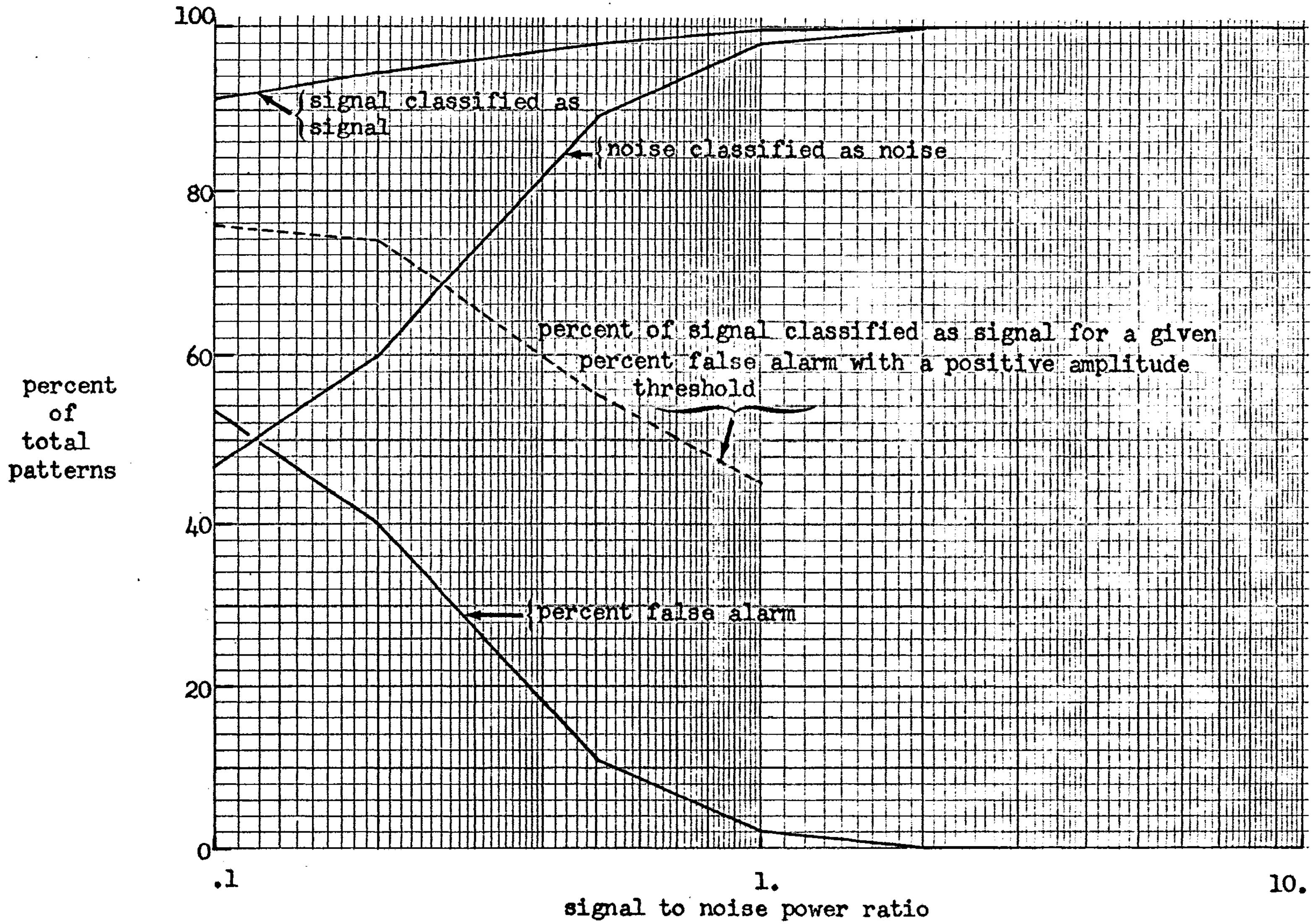


Figure (4.23). Maximum Likelihood Ratio Classifier
Separation of Signal and Noise

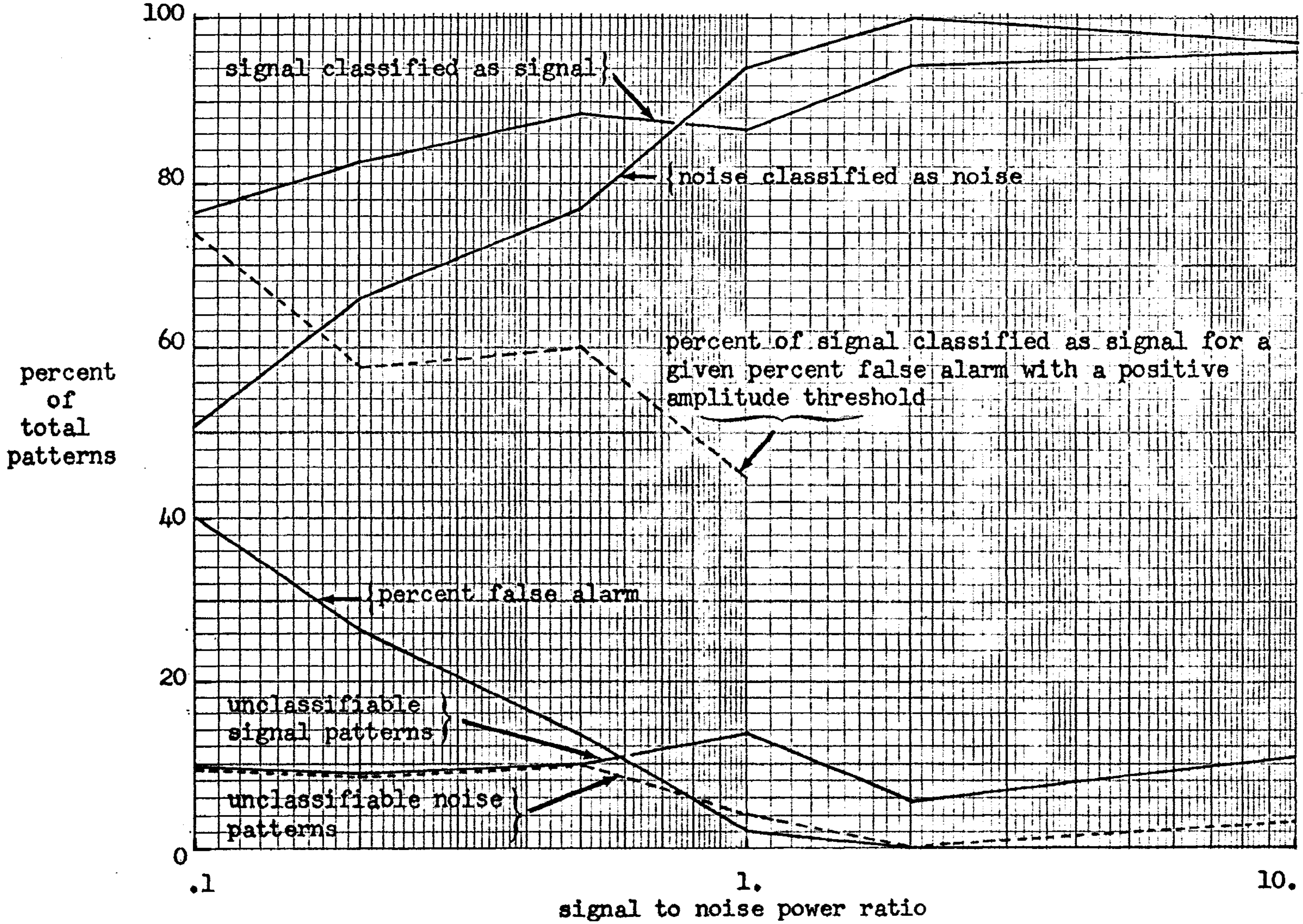


Figure (4.24). AHDT Classifier Separation
of Signal and Noise, Total Patterns

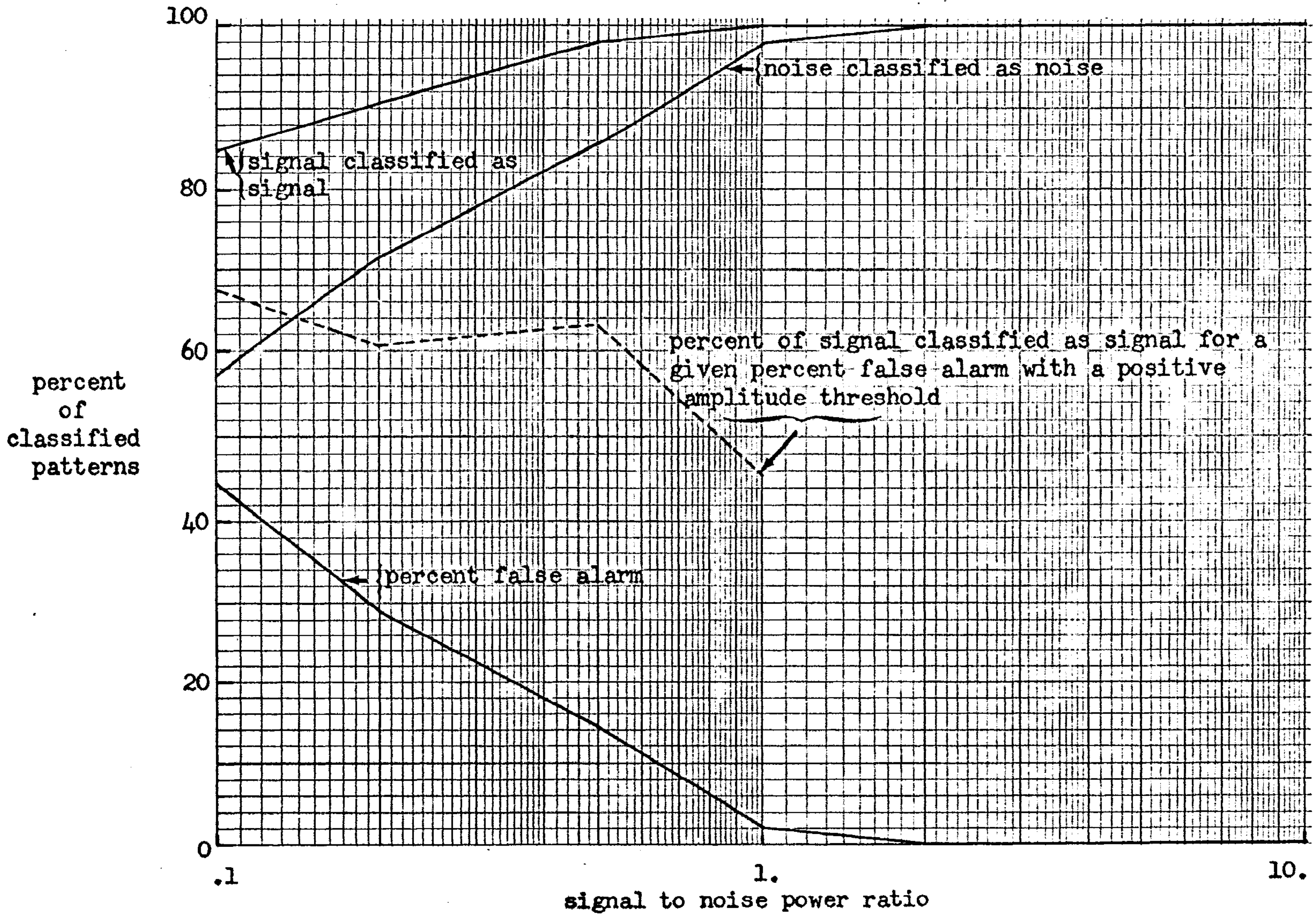


Figure (4.25). AHDT Classifier Separation of Signal and Noise, Classified Patterns

Table I. First Level Hypersphere Threshold Magnitude

Class	Signal to Noise Power Ratio 10.	2.	1.	.5	.2	.1
1	2192.	450.3	255.8	78.42	65.69	38.33
2	2136.	328.6	303.8	120.2	50.82	53.63
3	2327.	406.9	235.4	171.0	152.2	48.21
4	1859.	488.9	134.3	119.8	50.96	32.34
5	3322.	475.1	138.9	170.0	62.52	47.38
6	1692.	464.4	215.6	179.0	75.38	65.27
7	2202.	382.1	208.9	75.57	68.70	51.17
8	1580.	506.6	329.1	83.63	117.4	105.4
9	1827.	557.1	249.1	87.40	75.35	73.37
Average	2126.	451.1	230.1	120.6	79.89	57.23

Table II. Number of Unknown Patterns Falling Within the Union of the First Level Hypersphere Thresholds

Number of Thresholds in Union	Signal to Noise Power Ratio 10.	2.	1.	.5	.2	.1
0	23	7	14	15	3	8
1	266	196	133	52	17	14
2	147	7	42	20	17	15
3	115	24	26	26	25	17
4	182	55	34	47	8	3
5	100	87	46	52	20	7
6	50	97	77	78	24	11
7	17	422	331	281	30	30
8	0	5	197	295	122	38
9	0	0	0	34	634	757

Table III. Number of Training Patterns Falling Within the First Level Hypersphere Threshold, S/N=10.

Training Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	20	17	5	12	5	20	5	0	0
2	18	20	1	2	15	13	0	0	0
3	3	1	20	20	0	17	9	0	0
4	6	1	18	20	0	18	8	0	0
5	12	18	0	0	20	4	0	0	0
6	18	12	7	12	0	20	7	0	0
7	0	1	6	11	0	8	20	0	0
8	0	0	0	0	0	0	0	20	0
9	0	0	0	0	0	0	0	0	20

Table IV. Training Patterns Second Level Hypersphere Threshold Class to Class Separation Matrix, S/N=10.

Training Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	20	7	1	1	2	3	1	0	0
2	7	20	0	1	4	1	0	0	0
3	1	0	20	7	0	2	1	0	0
4	1	1	7	20	0	3	2	0	0
5	2	4	0	0	20	0	0	0	0
6	3	1	2	3	0	20	1	0	0
7	1	0	1	2	0	1	20	0	0
8	0	0	0	0	0	0	0	20	0
9	0	0	0	0	0	0	0	0	20

Table V. Number of Training Patterns Falling Within the First Level Hypersphere Threshold, S/N=2.

Training Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	20	20	18	18	18	20	19	0	0
2	19	20	9	13	18	18	14	0	0
3	18	15	20	19	8	17	19	6	0
4	19	18	20	20	14	19	19	6	0
5	20	20	10	10	20	17	13	0	0
6	19	20	18	19	17	20	19	0	0
7	15	15	15	18	14	16	20	0	0
8	0	0	6	4	0	0	0	20	0
9	0	0	0	0	0	0	0	0	20

Table VI. Training Patterns Second Level Hypersphere Threshold Class to Class Separation Matrix, S/N=2.

Training Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	20	7	5	5	4	5	4	0	0
2	7	20	4	4	6	6	2	0	0
3	5	4	20	10	1	5	3	2	0
4	5	4	10	20	2	6	3	1	0
5	4	6	1	2	20	2	1	0	0
6	5	6	5	6	2	20	4	0	0
7	4	2	3	3	1	4	20	0	0
8	0	0	2	1	0	0	0	20	0
9	0	0	0	0	0	0	0	0	20

Table VII. Number of Training Patterns Falling Within the First Level Hypersphere Threshold, S/N=1.

Training Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	20	19	19	20	19	20	19	6	0
2	20	20	19	20	20	20	19	6	0
3	18	16	20	20	16	19	19	14	0
4	14	13	16	20	10	18	17	6	0
5	15	13	9	17	20	14	7	0	0
6	19	18	19	20	17	20	19	6	0
7	16	15	18	20	15	19	20	7	0
8	4	6	16	17	4	13	18	20	3
9	0	0	0	0	0	0	1	0	20

Table VIII. Training Patterns Second Level Hypersphere Threshold Class to Class Separation Matrix, S/N=1.

Training Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	20	10	3	4	7	5	2	1	0
2	10	20	4	6	8	7	3	2	0
3	3	4	20	6	3	7	7	2	0
4	4	6	6	20	4	8	6	2	0
5	7	8	3	4	20	4	1	0	0
6	5	7	7	8	4	20	4	1	0
7	2	3	7	6	1	4	20	1	0
8	1	2	2	2	0	1	1	20	0
9	0	0	0	0	0	0	0	0	20

Table IX. Number of Training Patterns Falling Within the First Level Hypersphere Threshold, S/N=.5

Training Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	20	16	19	14	18	19	17	9	0
2	20	20	19	20	19	19	19	10	1
3	20	20	20	20	19	19	20	20	4
4	20	18	19	20	18	19	20	15	1
5	20	20	19	20	20	19	20	13	5
6	20	20	20	20	19	20	20	18	5
7	17	14	18	17	17	16	20	9	0
8	7	5	11	11	8	6	12	20	0
9	4	1	1	4	4	1	5	3	20

Table X. Training Patterns Second Level Hypersphere Threshold Class to Class Separation Matrix, S/N=.5

Training Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	20	9	8	9	11	9	7	3	0
2	9	20	6	7	11	9	5	2	0
3	8	6	20	10	7	7	7	3	0
4	9	7	10	20	9	6	10	4	1
5	11	11	7	9	20	10	7	2	0
6	9	9	7	6	10	20	7	2	0
7	7	5	7	10	7	7	20	4	0
8	3	2	3	4	2	2	4	20	0
9	0	0	0	1	0	0	0	0	20

Table XI. Number of Training Patterns Falling Within the First Level Hypersphere Threshold, S/N=.2

Training Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	20	20	17	20	18	19	20	16	16
2	18	20	17	19	17	18	18	14	12
3	20	20	20	20	20	20	20	20	20
4	18	17	17	20	15	19	17	16	13
5	20	20	17	20	20	18	19	15	15
6	20	20	17	20	20	20	20	19	17
7	20	20	17	20	18	19	20	17	17
8	20	20	18	20	18	20	20	20	19
9	17	18	14	16	12	16	16	16	20

Table XII. Training Patterns Second Level Hypersphere Threshold Class to Class Separation Matrix, S/N=.2

Training Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	20	10	6	11	7	11	11	5	4
2	10	20	5	8	7	10	7	4	4
3	6	5	20	7	8	6	8	8	1
4	11	8	7	20	6	9	10	6	4
5	7	7	8	6	20	8	5	4	3
6	11	10	6	9	8	20	8	5	5
7	11	7	8	10	5	8	20	6	4
8	5	4	8	6	4	5	6	20	3
9	4	4	1	4	3	5	4	3	20

Table XIII. Number of Training Patterns Falling Within the First Level Hypersphere Threshold, S/N=.1

Training Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	20	19	19	20	19	17	19	18	16
2	20	20	20	20	20	18	20	19	18
3	20	19	20	20	20	17	19	19	18
4	19	19	18	20	18	15	18	15	16
5	20	19	20	20	20	18	20	18	18
6	20	20	20	20	20	20	20	19	18
7	20	19	20	20	20	18	20	19	18
8	20	20	20	20	20	20	20	20	20
9	20	19	19	20	20	20	20	19	20

Table XIV. Training Patterns Second Level Hypersphere Threshold Class to Class Separation Matrix, S/N=.1

Training Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	20	10	8	10	11	7	8	5	6
2	10	20	9	9	10	6	9	5	5
3	8	9	20	10	10	10	8	7	4
4	10	9	10	20	12	10	10	8	7
5	11	10	10	12	20	6	7	5	5
6	7	6	10	10	6	20	9	9	6
7	8	9	8	10	7	9	20	6	4
8	5	5	7	8	5	9	6	20	5
9	6	5	4	7	5	6	4	5	20

Table XV. AHDT Simulation Error Rates

Signal to Noise Power Ratio		Class 1	2	3	4	5	6	7	8	9
10.	EMLR	35	37	45	56	17	34	0	0	0
	ERR	51	31	51	41	18	42	2	0	0
	UNC	4	1	1	0	0	1	2	11	3
	UNR	2	0	3	1	1	2	3	0	0
2.	EMLR	76	72	55	81	42	71	18	8	0
	ERR	68	54	51	82	24	72	33	12	0
	UNC	1	0	0	1	1	1	1	2	0
	UNR	9	10	4	6	1	3	3	0	0
1.	EMLR	85	74	71	81	60	90	38	8	2
	ERR	80	61	64	73	38	74	48	9	2
	UNC	0	0	2	1	4	1	0	3	3
	UNR	15	18	12	11	11	17	11	1	1
.5	EMLR	86	77	76	79	64	90	68	33	11
	ERR	49	84	77	83	54	77	58	24	13
	UNC	0	0	2	0	1	0	2	5	5
	UNR	13	12	8	9	6	10	5	6	5
.2	EMLR	80	81	89	87	80	90	72	47	40
	ERR	93	67	76	58	71	87	87	47	26
	UNC	0	1	0	1	0	0	1	0	0
	UNR	8	6	10	10	11	9	9	3	8
.1	EMLR	88	80	88	86	83	94	87	61	53
	ERR	94	82	85	64	83	78	83	50	40
	UNC	0	2	1	0	2	1	1	1	0
	UNR	6	7	12	9	3	11	12	9	9

EMLR= Maximum likelihood ratio misclassification
ERR= AHDT misclassification not including the unclassifiable patterns
UNC= Unclassifiable patterns falling outside the first level hypersphere thresholds
UNR= Unclassifiable patterns not separated by the second level hypersphere thresholds

Table XVI. Number of Unknown Patterns Falling Within the First Level Hypersphere Threshold, S/N=10.

Unknown Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	93	85	34	40	62	90	23	0	0
2	92	98	4	4	97	63	4	0	0
3	35	3	98	99	2	46	37	0	0
4	53	3	99	96	0	63	49	0	0
5	28	76	0	0	99	7	0	0	0
6	94	73	63	67	33	93	54	0	0
7	13	0	44	30	0	22	98	0	0
8	0	0	0	0	0	0	0	89	0
9	0	0	0	0	0	0	0	0	97

Table XVII. AHDT Classification Matrix, S/N=10.

Unknown Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	43	26	1	5	2	17	0	0	0
2	13	68	0	0	18	0	0	0	0
3	1	0	45	48	0	1	1	0	0
4	0	0	34	58	0	5	2	0	0
5	2	16	0	0	81	0	0	0	0
6	19	6	3	12	0	55	2	0	0
7	0	0	0	2	0	0	93	0	0
8	0	0	0	0	0	0	0	89	0
9	0	0	0	0	0	0	0	0	97

Table XVIII. Number of Unknown Patterns Falling Within the First Level Hypersphere Threshold, S/N=2.

Unknown Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	96	92	83	94	95	97	88	1	0
2	96	94	66	87	99	95	72	0	0
3	91	61	99	100	56	97	93	29	0
4	95	63	96	99	57	97	86	24	0
5	86	87	38	57	99	79	39	0	0
6	8	89	92	96	89	98	91	3	0
7	93	68	83	92	74	97	96	5	0
8	7	0	20	22	0	9	10	98	0
9	0	0	0	0	0	0	0	0	100

Table XIX. AHDT Classification Matrix, S/N=2.

Unknown Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	22	17	7	5	17	10	12	0	0
2	8	36	1	0	36	5	4	0	0
3	3	2	45	13	1	10	21	1	0
4	8	8	51	11	0	5	10	0	0
5	3	20	0	1	74	0	0	0	0
6	6	26	9	8	3	24	20	0	0
7	0	8	7	3	5	11	61	1	0
8	0	0	11	1	0	0	0	86	0
9	0	0	0	0	0	0	0	0	100

Table XX. Number of Unknown Patterns Falling Within the First Level Hypersphere Threshold, S/N=1.

Unknown Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	97	100	86	73	70	97	85	45	1
2	98	100	87	65	79	95	87	41	1
3	96	96	97	79	49	93	89	71	3
4	99	99	98	90	51	98	94	78	3
5	90	96	68	45	78	78	67	18	1
6	99	99	95	78	73	96	93	53	1
7	96	99	95	79	47	95	97	60	5
8	36	31	65	25	0	30	31	97	1
9	2	3	1	0	0	1	2	4	96

Table XXI. AHDT Classification Matrix, S/N=1.

Unknown Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	5	15	11	13	23	7	9	2	0
2	1	21	9	9	29	8	5	0	0
3	4	3	22	15	5	10	19	8	0
4	3	2	28	15	7	11	18	4	0
5	5	24	1	4	47	2	2	0	0
6	6	13	18	11	16	7	9	2	0
7	1	16	12	5	3	10	40	2	0
8	0	0	0	0	0	0	0	87	1
9	0	1	0	0	0	0	1	0	94

Table XXII. Number of Unknown Patterns Falling Within the First Level Hypersphere Threshold, S/N=.5

Unknown Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	84	97	96	93	99	98	77	35	15
2	90	98	98	95	100	100	78	27	16
3	84	90	98	96	95	98	79	58	10
4	85	96	100	96	98	100	85	52	9
5	82	95	95	89	99	99	68	19	11
6	80	91	97	91	97	99	73	47	6
7	88	94	99	95	98	98	89	48	12
8	38	55	94	84	66	93	43	89	7
9	10	19	35	20	39	39	10	6	88

Table XXIII. AHDT Classification Matrix, S/N=.5

Unknown Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	38	0	8	2	22	9	6	2	0
2	34	4	9	1	25	4	7	2	2
3	21	0	13	7	9	5	17	17	1
4	27	2	18	8	3	6	14	10	3
5	31	5	2	1	40	4	4	2	4
6	26	1	8	5	14	13	15	7	1
7	21	2	7	1	15	3	35	8	1
8	3	0	10	4	1	0	3	67	1
9	0	0	2	0	5	1	2	3	77

Table XXIV. Number of Unknown Patterns Falling Within the First Level Hypersphere Threshold, S/N=.2

Unknown Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	93	86	100	86	91	96	93	99	71
2	94	90	99	90	94	99	96	99	80
3	96	92	100	93	95	97	96	99	86
4	90	84	99	85	89	93	92	95	75
5	94	88	100	86	94	94	92	93	77
6	95	89	100	89	95	96	96	97	78
7	95	88	99	87	91	97	95	97	78
8	89	73	100	87	85	92	91	100	75
9	71	58	100	55	70	82	80	97	96

Table XXV. AHDT Classification Matrix, S/N=.2

Unknown Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	0	17	16	25	7	7	2	11	7
2	0	25	12	16	14	9	5	5	7
3	0	8	14	33	7	3	4	12	9
4	1	14	14	31	3	2	4	18	2
5	2	17	12	17	18	3	4	3	13
6	1	17	16	26	5	3	2	8	13
7	0	9	12	22	6	8	4	15	14
8	1	3	11	25	1	0	1	50	5
9	0	1	2	10	4	0	2	7	66

Table XXVI. Number of Unknown Patterns Falling Within the First Level Hypersphere Threshold, S/N=.1

Unknown Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	92	96	95	85	95	98	95	100	97
2	91	96	93	89	96	97	94	98	96
3	89	95	94	85	94	98	95	99	95
4	89	96	95	87	94	97	95	100	95
5	84	91	87	77	87	94	88	98	90
6	90	96	92	90	93	97	95	99	96
7	92	96	95	89	96	99	96	99	96
8	87	95	94	81	91	97	94	99	93
9	81	90	88	74	89	96	90	100	99

Table XXVII. AHDT Classification Matrix, S/N=.1

Unknown Pattern Class Origin	Class 1	2	3	4	5	6	7	8	9
1	0	5	4	21	17	8	4	20	15
2	1	8	4	21	18	7	3	15	14
3	0	4	2	19	14	12	3	19	14
4	0	2	7	27	12	10	1	20	12
5	0	10	4	26	12	6	1	19	17
6	1	3	5	20	12	10	1	17	19
7	1	1	3	29	13	11	4	15	10
8	1	2	8	16	3	6	3	40	11
9	0	1	2	10	6	6	4	11	51

Table XXVIII. Percent of Unknown Patterns in a Class Exceeding the First Level Hypersphere Threshold

Signal to Noise Power Ratio	Class 1	2	3	4	5	6	7	8	9
10.	7	2	2	6	1	7	2	11	3
2.	4	6	1	1	1	2	4	2	0
1.	3	0	3	10	22	4	3	3	4
.5	16	2	2	4	1	1	11	11	12
.2	7	10	0	15	6	4	5	0	4
.1	8	4	6	13	13	3	4	1	1

CHAPTER V

SUMMARY

The computer simulation pointed out various areas of the adaptive hypersphere decision threshold concept which requires additional research. The AHDT, as implemented, was found to have a bias classification for each S/N ratio level. This bias in the pattern class separation appears to be a function of the first level hypersphere threshold magnitude. In the case where unclassifiable patterns can be neglected the AHDT classifier offers an improvement over the maximum likelihood ratio classifier in the separation of signal and noise. This improvement emphasizes the need for additional research into the adaptive hypersphere decision threshold.

CHAPTER VI

SUGGESTION FOR FURTHER RESEARCH

This investigation has indicated the need for additional research. Areas which are evident include:

1) A method for optimal selection of the first level hypersphere threshold magnitude.

2) Generation of third and higher order levels of adaptive hypersphere decision thresholds to minimize unclassifiable patterns.

REFERENCES

1. J. S. Koford and G. F. Groner, "The Use of an Adaptive Threshold Element to Design a Linear Optimal Pattern Classifier," IEEE Transactions on Information Theory, Vol. IT-12, pp. 42-50, January 1966.

2. S. B. Akers, "Techniques of Adaptive Decision Making," General Electric Report No. G-137, September 1966.

3. E. A. Patrick and G. C. Hancock, "Nonsupervised Sequential Classification and Recognition of Patterns," IEEE Transactions on Information Theory, Vol. IT-12, pp. 362-372, July 1966.

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

5. R. Albrecht and W. Werner, "Error Analysis of a Statistical Decision Method," IEEE Transactions on Information Theory, Vol. IT-10, pp. 34-38, January 1964.

6. D. B. Cooper and P. W. Cooper, "Nonsupervised Adaptive Signal Detection and Pattern Recognition," Information and Control 7, pp. 416-444, 1964.

7. H. J. Scudder, "Probability of Error of Some Adaptive Pattern-Recognition Machines," IEEE Transactions on Information Theory, Vol. IT-11, pp. 363-371, July 1965.

8. G. Sebestyen and J. Edie, "An Algorithm for Non-Parametric Pattern Recognition," IEEE Transactions on Electronic Computers, Vol. EC-15, pp. 908-915, December 1966.

9. T. Marill and D. M. Green, "On The Effectiveness of Receptors in Recognition Systems, " IEEE Transactions on Information Theory, Vol. IT-9, pp. 11-17, January 1963.

10. P. W. Cooper, "Quadratic Discriminant Functions in Pattern Recognition," IEEE Transactions on Information Theory, Vol. IT-12, pp. 313-315, April 1965.

11. S. S. Yau and P. C. Chuang, "Feasibility of Using Linear Pattern Classifiers for Probabilistic Pattern Classes," Proceedings of the IEEE, pp. 1957-1959, December 1966.

12. R. O. Duda and H. Fossum, "Pattern Classification by Iteratively Determined Linear and Piecewise Linear Discriminant Function," IEEE Transactions on Electronic Computers, Vol. EC-15, pp. 220-232, April 1966.

13. D. Middleton, <u>An Inroduction to Statistical Communication Theory</u>, New York: McGraw-Hill, 1960.

14. N. Abramson, D. Braverman and G. Sebestyen, "Pattern Recognition and Machine Learning," IEEE Transactions on Information Theory, Vol. IT-9, pp. 257-261, October 1963.

15. R. W. Sears, "Adaptive Representations for Pattern Recognition," IEEE Transactions on Systems Science and Cybernetics, Vol. SSC-1, pp. 59-66, November 1965.

16. J. Spragins, "Learning Without A Teacher," IEEE Transactions on Information Theory, Vol. It-12, pp. 223-230, April 1966.

17. P. L. Meyer, Introductory Probability and Statistical Applications, Massachusetts: Addison-Wesley, 1965.

18. P. W. Cooper, "The Hypersphere in Pattern Recognition," Information and Control 5, pp. 324-346, 1962.

19. P. W. Cooper, "A Note on an Adaptive Hypersphere Decision Boundary," IEEE Transactions on Electronic Computers, Vol. EC-15, pp. 948-949, December 1966.

20. E. A. Patrick and J. C. Hancock, "The Nonsupervised Learning of Probability Spaces and Recognition of Patterns," 1965 IEEE Internat'l Conn. Rec., Pt. II.

21. D. P. Petersen and D. Middleton, "Sampling and Reconstruction of Wave-Number Limited Functions in N-Dimensional Euclidean Spaces," Information and Control 5, pp. 279-323, 1962.

BIBLIOGRAPHY

1. P. W. Cooper, "The Hypersphere in Pattern Recognition," Information and Control 5, pp. 324-346, 1962.

2. D. P. Petersen and D. Middleton, "Sampling and Reconstruction of Wave-Number Limited Functions in N-Dimensional Euclidean Space," Information and Control 5, pp. 279-323, 1962.

3. T. Marill, "A Note on Pattern Recognition Techniques and Game-Playing Programs," Information and Control 6, pp. 213-217, 1963.

4. M. Sakaguchi, "Information Pattern, Learning Structure, and Optimal Decision Rule," Information and Control 6, pp. 218-229, 1963.

5. A. M. Hormann, "Programs for Machine Learning, Part II," Information and Control 7, pp. 55-77, 1964.

6. P. Mermelstein and M. Eden, "Experiments on Computer Recognition of Connected Handwritten Words," Information and Control 7, pp. 255-270, 1964.

7. D. B. Cooper and P. W. Cooper, "Nonsupervised Adaptive Signal Detection and Pattern Recognition," Information and Control 7, pp. 416-444, 1964.

8. I. Selin, "The Sequential Estimation and Detection of Signals in Normal Noise, I," Information and Control 7, pp. 512-534, 1964.

9. I. Selin, "The Sequential Estimation and Detection of Signals in Normal Noise, II," Information and Control 8, pp. 1-35, 1965.

10. R. J. Spinrad, "Machine Recognition of Hand Printing," Information and Control 8, pp. 124-142, 1965.

11. T. Kailath, "Some Results on Singular Detection," Information and Control 9, pp. 130-152, 1966.

12. D. O. Clayden, M. B. Clowes and J. R. Parks, "Letter Recognition and the Segmentation of Running Text," Information and Control 9, pp. 246-264, 1966.

13. C. Chen, "A Note on Sequential Decision Approach to Pattern Recognition and Machine Learning," Information and Control 9, pp. 549-562, 1966.

14. R. W. Sears, "Adaptive Representations for Pattern Recognition," IEEE Transactions on Systems Science and Cybernetics, Vol. SSC-1, pp. 59-66, November 1965.

15. J. D. Patternson and B. F. Womack, "An Adaptive Pattern· Classification System," IEEE Transactions on Systems Science and Cybernetics, Vol. SSC-2, pp. 62-67, August 1966.

16. C. K. Chow and C. N. Liu, "An Approach to Structure Adaptation in Pattern Recognition," IEEE Transactions on Systems Science and Cybernetics, Vol. SSC-2, pp. 73-80, December 1966.

17. C. K. Chow, "A Class of Nonlinear Recognition Procedures," IEEE Transactions on Systems Science and Cybernetics, Vol. SSC-2, pp. 101-108, December 1966.

18. R. O. Duda and H. Fossum, "Pattern Classification by Iteratively Determined Linear and Piecewise Linear Discriminant Function," IEEE Transactions on Electronic Computers, Vol. EC-15, pp. 220-232, April 1966.

19. C. A. Rosen and D. J. Hall, "A Pattern Recognition Experiment with Near-Optimum Results," IEEE Transactions on Electronic Computers, Vol. EC-15, August 1966.

20. R. E. Bonner, "Pattern Recognition with Three Added Requirements," IEEE Transactions on Electronic Computers, Vol. EC-15, 770-781, October 1966.

21. G. Sebestyen and J. Edie, "An Algorithm for Non-Parametric Pattern Recognition," IEEE Transactions on Electronic Computers, Vol. EC-15, pp. 908-915, December 1966.

22. P. W. Cooper, "A Note on an Adaptive Hypersphere Decision Boundary," IEEE Transactions on Electronic Computers, Vol. EC-15, pp. 948-949, December 1966.

23. T. Marill and D. M. Green, "On The Effectiveness of Receptors in Recognition Systems," IEEE Transactions on Information Theory, Vol. IT-9, pp. 11-17, January 1963.

24. N. Abramson, D. Braverman and G. Sebestyen, "Pattern Recognition and Machine Learning," IEEE Transactions on Information Theory, Vol. IT-9, pp. 257-261, October 1963.

25. R. Albrecht and N. Werner, "Error Analysis of a Statistical Decision Method," IEEE Transactions on Information Theory, Vol. IT-10, pp. 34-38, January 1964.

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

27. P. W. Cooper, "Quadratic Discriminant Functions in Pattern Recognition," IEEE Transactions on Information Theory, Vol. IT-11, pp. 313-315, April 1965.

28. H. J. Scudder, "Probability of Error of Some Adaptive Pattern-Recognition Machines," IEEE Transactions on Information Theory, Vol. IT-11, pp. 363-371, July 1965.

29. J. S. Koford and G. F. Groner, "The Use of an Adaptive Threshold Element to Design a Linear Optimal Pattern Classifier," IEEE Transactions on Information Theory, Vol. IT-12, pp. 42-50, January 1966.

30. G. Nagy and G. L. Shelton, "Self-Corrective-Character Recognition System," IEEE Transactions on Information Theory, Vol. IT-12, pp. 215-222, April 1966.

31. J. Spragins, "Learning Without a Teacher," IEEE Transactions on Information Theory, Vol. IT-12, pp. 223-230, April 1966.

32. E. A. Patrick and J. C. Hancock, "Nonsupervised Sequential Classification and Recognition of Patterns," IEEE Transactions on Information Theory, Vol. IT-12, pp. 362-372, July 1966.

33. S. G. Fralick, "Learning to Recognize Patterns Without a Teacher," IEEE Transactions on Information Theory, Vol. IT-13, pp. 57-64, January 1967.

34. S. B. Akers, "Techniques of Adaptive Decision Making," General Electric Report No. G-137, September 1966.

35. S. S. Yau and P. C. Chuang, "Feasibility of Using Linear Pattern Classifiers for Probabilistic Pattern Classes," Proceedings of the IEEE, pp. 1957-1959, December 1966.

36. J. M. Pitt and B. F. Womack, "A Sequentialization of the Pattern Classifier," Proceedings of the IEEE, pp. 1987-1988, December 1966.

37. D. Middleton, An Introduction to Statistical Communication Theory, New York: McGraw-Hill, 1960.

38. E. A. Patrick and J. C. Hancock, "The Unsupervised Learning of Probability Spaces and Recognition of Patterns," 1965 IEEE Internat'l Conn. Rec., Pt. II.

39. W. W. Bledsoe, "Some Results on Multicategory Pattern Recognition," Journal of the Association for Computing Machinery, Vol. 13, pp. 304-316, April 1966.

40. A. C. Wolff, "The Estimation of the Optimum Linear Decision Function with a Sequential Random Method," IEEE Transactions on Information Theory, Vol. IT-12, pp. 312-315, July 1966.

41. D. F. Specht, "Generation of Polynomial Discriminant Functions for Pattern Recognition," IEEE Transactions on Electronic Computers, Vol. EC-16, June 1967.

APPENDIX A

```
C                     OS/360 FORTRAN H
C
C      IMPLEMENTATION AND COMPUTER SIMULATION OF THE ADAPTIVE
C      HYPERSPHERE DECISION THRESHOLD CLASSIFIER. THE PATTERN
C      RECOGNITION PROBLEM CONTAINS NINE PATTERN CLASSES,
C      CONSISTING OF EIGHT DESIGNED PATTERNS AND A NOISE
C      PATTERN. THE SIMULATION REQUIRES THE SUBROUTINES MINV
C      AND RANDU FROM THE IBM SYSTEM/360 SCIENTIFIC
C      SUBROUTINE PACKAGE.
C
       DIMENSION Y(25,9),FBA(25,25),FA(25),BD(25,25),TZ(9),
      1BF(9),XE(9),XD(25),X(25),WB(25),VD(9,9),CB(9),VG(9),
      2WA(9,20,9),EE(9,9),AHR(9,9),UB(9),PSID(9),PSI(9,9,9),
      3ERR(9),EMLC(9),UNC(9),UNR(9),NOTJ(9),DB(25,25,9),PATC(9)
 1000 FORMAT(T20,'PATTERN CLASS MEAN VECTOR'/)
 1001 FORMAT(T10,'CLASS',T24,'1',T35,'2',T46,'3',T57,'4',
      1T68,'5',T79,'6',T90,'7',T101,'8',T112,'9'/)
 1002 FORMAT(T20,9E11.4)
 1003 FORMAT(T20,'COVARIANCE MATRIX'/)
 1004 FORMAT(T20,'SINGULAR MATRIX'/)
 1005 FORMAT(T20,'INVERSE MATRIX'/)
 1006 FORMAT(T20,'DETERMINANT'/)
 1007 FORMAT(T20,'PATTERN CLASS CENTROID'/)
 1008 FORMAT(T20,'MEAN LIKELIHOOD VECTOR'/)
 1009 FORMAT(T20,'TRAINING PATTERNS'/)
 1010 FORMAT(T20,'FIRST LEVEL HYPERSPHERE THRESHOLD'/)
 1011 FORMAT(T20,'NUMBER OF TRAINING PATTERNS FALLING WITHIN',
      1T63,'THE FIRST LEVEL HYPERSPHERE THRESHOLD'/)
 1012 FORMAT(T20,'SECOND LEVEL HYPERSPHERE THRESHOLD'/)
 1013 FORMAT(T20,9I11)
 1014 FORMAT(T20,'PATTERN CLASS CENTROID'/)
 1015 FORMAT(T20,'PATTERN CLASS',I2/)
 1016 FORMAT(T20,'MAXIMUM LIKELIHOOD RATIO CLASSIFIER ERROR'/)
 1017 FORMAT(T20,'AHDT CLASSIFIER ERROR'/)
 1018 FORMAT(T20,'AHDT UNCLASSIFIABLE PATTERNS'/)
 1019 FORMAT(T20,'TRAINING PATTERNS SECOND LEVEL HYPERSPHERE',
      1T63,'THRESHOLD CLASS TO CLASS SEPARATION MATRIX'/)
 1020 FORMAT(1H1,T90,'PAGE',F5.1)
 1021 FORMAT(T10,'CB(J)',T20,9E11.4)
 1022 FORMAT(T10,'XE(J)',T20,9E11.4)
 1023 FORMAT(T10,I3,T20,9E11.4)
 1024 FORMAT(T10,'JJ',T25,'EJJ',T40,'JT',T55,'EJT',T70,'PXK2',
      1T95,'XK2')
 1025 FORMAT(T10,I3,T25,F6.1,T40,I3,T55,F6.1,T70,E10.3,T95,E10.3)
 1026 FORMAT(T20,'SOC=',E10.3,T60,'POI=',E10.3)
 1027 FORMAT(T10,'GB=',E10.3,T30,'XK1=',E10.3,T50,'XLIM=',E10.3)
 1028 FORMAT(T10,'RA2=',E10.3)
 1029 FORMAT(T9,10I11)
 1030 FORMAT(T20,'THE LAST VALUE OF IX WAS',I70/)
```

```
 1031 FORMAT(T10,'VAR=',E20.10)
 1032 FORMAT(T20,'UNKNOWN PATTERNS'/)
      XPAGE=1.
      WRITE(6,1020)XPAGE
      XPAGE=XPAGE+1.
      PI=3.14159
      T=100.
      A1=4.
      A2=8.
      A3=SQRT(5./3.)
      C2=T*((6.)**(-.5))
      C3=((20./3.)**.5)/T
      C4=T*((PI/(18.*PI-48.))**.5)
      C5=T*((7.5+6.*EXP(-A1)-1.5*EXP(-A2))**(-.5))
      C6=T*((18.)**(-.5))
      C7=C2
      C8=T*((5./12.)**.5)
      T1=.111
      T2=.222
      T3=.333
      T4=.444
      T5=.555
      T6=.666
      T7=.777
      T8=.888
      T9=.999
C
C     THE SIGNAL TO NOISE POWER RATIO IS OBTAINED BY
C     DIVIDING 2500./3. BY THE NOISE VARIANCE VAR.
C
      VAR=250./3.
      IX=1
C
C     CALCULATE THE PATTERN CLASS MEAN VECTOR.
C
      Do 20 I=1,25
      J=1
      IF(I.GT.12) GO TO 2
      Y(I,J)=T*I/25.
      GO TO 3
    2 Y(I,J)=T*(1.-I/25.)
    3 J=2
      Y(I,J)=C2*SIN(4.*PI*I/T)
      IF(I.EQ.25) Y(I,J)=0.
      J=3
      IF(I.GT.12) GO TO 4
      Y(I,J)=C3*((4.*I)**2)
      GO TO 5
    4 Y(I,J)=C3*((T-4.*I)**2)
```

```
      IF(I.EQ.25) Y(I,J)=0.
    5 J=4
      IF(I.GT.12) GO TO 6
      Y(I,J)=C4*(1.-COS(4.*PI*I/T))
      GO TO 7
    6 Y(I,J)=C4*(1.+COS(4.*PI*I/T))
      IF(I.EQ.25) Y(I,J)=0.
    7 J=5
      IF(I.GT.12) GO TO 8
      Y(I,J)=C5*(1.-EXP(-A1*I/12.5))
      GO TO 9
    8 Y(I,J)=C5*(1.-EXP(-(T-A1*I)/12.5))
    9 J=6
      Y(I,J)=C6*(1.-COS(8.*PI*I/T))
      IF(I.EQ.25) Y(I,J)=0.
      J=7
      IF(I.GT.6) GO TO 10
      Y(I,J)=0.
      GO TO 12
   10 IF(I.GT.18) GO TO 11
      Y(I,J)=C7
      GO TO 12
   11 Y(I,J)=0.
   12 J=8
      IF(I.GT.9) GO TO 13
      Y(I,J)=0.
      GO TO 15
   13 IF(I.GT.15) GO TO 14
      Y(I,J)=C8
      GO TO 15
   14 Y(I,J)=0.
   15 J=9
   20 Y(I,J)=0.
      WRITE(6,1000)
      WRITE(6,1001)
      WRITE(6,1002) ((Y(I,J),J=1,9),I=1,25)
      WRITE(6,1020) XPAGE
      XPAGE=XPAGE+1.
C
C     CALCULATE THE PATTERN CLASS COVARIANCE MATRIX.
C
      IMN=0
      DO 40 J=1,9
   21 DO 22 I=1,25
      DO 22 K=1,25
   22 FBA(I,K)=0.
   24 DO 28 L=1,500
      DO 26 I=1,25
      RN=-10.
```

```
C
C     USE THE CENTRAL LIMIT THEOREM TO APPROXIMATE THE
C     GAUSSIAN DISTRIBUTION.
C
      DO 25 IK=1,20
      CALL RANDU(IX,IY,RNN)
      IX=IY
   25 RN=RN+RNN
   26 FA(I)=RN/A3
      DO 28 I=1,25
      DO 28 K=1,25
      IF(I.GT.K) GO TO 28
      FBA(I,K)=FBA(I,K)+FA(I)*FA(K)
   28 CONTINUE
      DO 30 I=1,25
      DO 30 K=1,25
      IF(I.GT.K) GO TO 30
      BD(I,K)=FBA(I,K)/499.
      BD(K,I)=BD(I,K)
   30 CONTINUE
      WRITE(6,1015)J
      WRITE(6,1003)
      DO 31 I=1,25
      IF(I.NE.18) GO TO 31
      WRITE(6,1020) XPAGE
      XPAGE=XPAGE+1.
   31 WRITE(6,1002) (BD(I,K),K=1,25)
C
C     THE PATTERN CLASS COVARIANCE MATRIX HAS BEEN CONSTRUCTED
C     FROM THE ADDITIVE GAUSSIAN NOISE. NOW CALCULATE THE
C     DETERMINANT AND THE INVERSE COVARIANCE MATRIX.
C
      CALL MINV(BD,25,DET,FA,WB)
      IF(DET.NE.0.) GO TO 32
      WRITE(6,1004)
      IMN=IMN+1
      IF(IMN.LT.4) GO TO 21
      CALL EXIT
   32 TZ(J)=DET
      DO 34 I=1,25
      DO 34 K=1,25
   34 DB(I,K,J)=BD(I,K)
      WRITE(6,1005)
      DO 36 I=1,25
      IF(I.NE.7) GO TO 36
      WRITE(6,1020) XPAGE
      XPAGE=XPAGE+1.
   36 WRITE(6,1002) (BD(I,K),K=1,25)
      WRITE(6,1020) XPAGE
```

```
      XPAGE=XPAGE+1.
   40 CONTINUE
      WRITE(6,1006)
      WRITE(6,1002) (TZ(J),J=1,9)
C
C     CALCULATE THE PATTERN CLASS CENTROID.
C
      DO 50 JS=1,9
      BF(JS)=0.
      DO 50 J=1,9
      BE=0.
      IF(J.EQ.JS) GO TO 50
      DO 42 I=1,25
   42 BE=BE+(Y(I,JS)-Y(I,J))**2
   50 BF(JS)=BF(JS)+BE
      JS=1
      DO 54 J=2,9
      IF(BF(J).GT.BF(JS)) GO TO 54
      JS=J
   54 CONTINUE
      WRITE(6,1007)
      WRITE(6,1002) (BF(J),J=1,9)
      JC=JS
C
C     THE PATTERN CLASS CENTROID IS CLASS JC.
C
      WRITE(6,1014)
      WRITE(6,1013) JC
C
C     CALCULATE THE MEAN LOGARITHM OF THE LIKELIHOOD RATIO
C     VECTOR CONTRIBUTION OF THE COVARIANCE MATRIX DETERMINANT.
C
      DO 56 JS=1,9
   56 CB(JS)=.5*LOG(TZ(JC)/TZ(JS))
C
C     CALCULATE THE MEAN LOGARITHM OF THE LIKELIHOOD RATIO
C     VECTOR FOR EACH CLASS.
C
      ICON=1
  500 DEV=SQRT(VAR)
      WRITE(6,1031) VAR
      DO 64 J=1,9
      DO 62 JS=1,9
      XE(JS)=0.
      DO 58 I=1,25
   58 WB(I)=Y(I,J)-Y(I,JS)
      DO 62 K=1,25
      XD(K)=0.
      DO 60 I=1,25
```

```
   60 XD(K)=XD(K)+WB(I)*DB(I,K,JS)
   62 XE(JS)=XE(JS)+XD(K)*WB(K)
      WRITE(6,1022) (XE(JS),JS=1,9)
      DO 64 JS=1,9
   64 VD(JS,J)=CB(JS)-(XE(JS)-XE(JC))/2./VAR
      WRITE(6,1001)
      WRITE(6,1021) (CB(J),J=1,9)
      WRITE(6,1008)
      WRITE(6,1002) ((VD(JS,J),J=1,9),JS=1,9)
      WRITE(6,1020) XPAGE
      XPAGE=XPAGE+1.
C
C     NOW GENERATE THE LEARNING PATTERNS.
C
      DO 74 J=1,9
      DO 74 L=1,20
      DO 66 I=1,25
      RN=-10.
C
C     USE THE CENTRAL LIMIT THEOREM TO APPROXIMATE THE
C     GAUSSIAN DISTRIBUTION.
C
      DO 65 IK=1,20
      CALL RANDU(IX,IY,RNN)
      IX=IY
   65 RN=RN+RNN
   66 X(I)=RN*DEV/A3+Y(I,J)
      DO 72 JS=1,9
      XE(JS)=0.
      DO 68 I=1,25
   68 WB(I)=X(I)-Y(I,JS)
      DO 72 K=1,25
      XD(K)=0.
      DO 70 I=1,25
   70 XD(K)=XD(K)+WB(I)*DB(I,K,JS)
   72 XE(JS)=XE(JS)+XD(K)*WB(K)
C
C     CALCULATE THE TRAINING PATTERN LOGARITHM OF THE
C     LIKELIHOOD RATIO VECTOR.
C
      DO 74 JS=1,9
   74 WA(JS,L,J)=CB(JS)-(XE(JS)-XE(JC))/2./VAR
      WRITE(6,1009)
      IJKL=3
      DO 75 J=1,9
      IF(J.EQ.IJKL) WRITE(6,1020) XPAGE
      IF(J.EQ.IJKL) XPAGE=XPAGE+1.
      IF(J.EQ.IJKL) IJKL=IJKL+2
      WRITE(6,1015) J
```

```
   75 WRITE(6,1002) ((WA(JS,L,J),JS=1,9),L=1,20)
C
C     DETERMINE THE FIRST LEVEL HYPERSPHERE THRESHOLD.
C
      DO 80 J=1,9
      DO 76 L=1,20
      VG(L)=0.
      DO 76 JS=1,9
   76 VG(L)=VG(L)+(WA(JS,L,J)-VD(JS,J))**2
      WRITE(6,1015) J
      WRITE(6,1002) (VG(L),L=1,20)
      LL=1
      DO 78 L=2,20
      IF(VG(L).LE.VG(LL)) GO TO 78
      LL=L
   78 CONTINUE
   80 UB(J)=VG(LL)
      WRITE(6,1010)
      WRITE(6,1002) (UB(J),J=1,9)
C
C     CALCULATE THE NUMBER OF TRAINING PATTERNS FALLING
C     WITHIN THE FIRST LEVEL HYPERSPHERE THRESHOLD.
C
      DO 82 JD=1,9
      DO 82 J=1,9
   82 EE(JD,J)=0.
      DO 86 J=1,9
      DO 86 L=1,20
      DO 86 JD=1,9
      VF=0.
      DO 84 JS=1,9
   84 VF=VF+(WA(JS,L,J)-VD(JS,JD))**2
      IF(VF.GT.UB(JD)) GO TO 86
      EE(JD,J)=EE(JD,J)+1.
   86 CONTINUE
C
C     THE MATRIX EE IS READ AS THE MISCLASSIFICATION IN CLASS
C     JD GIVEN THE TRAINING PATTERN ORIGINATED IN CLASS J.
C
      WRITE(6,1011)
      WRITE(6,1001)
      WRITE(6,1002) ((EE(JD,J),J=1,9),JD=1,9)
      DO 87 J=1,9
      DO 87 JD=1,9
      AHR(JD,J)=0.
      DO 87 JS=1,9
   87 PSI(JS,JD,J)=0.
C
C     NOW GENERATE THE SECOND LEVEL HYPERSPHERE THRESHOLDS.
C
```

```
      DO 120 J=1,9
      DO 120 JD=1,9
      IF(JD.LE.J) GO TO 120
      GB=0.
C
C     DECIDE WHICH CLASS HAS THE SMALLER FIRST LEVEL
C     HYPERSPHERE THRESHOLD.
C
      IF(UB(JD).LT.UB(J)) GO TO 88
      JJ=J
      JT=JD
      GO TO 89
   88 JJ=JD
      JT=J
C
C     IS A SECOND LEVEL HYPERSPHERE THRESHOLD REQUIRED TO
C     SEPARATE CLASS JD AND J
C
   89 IF((EE(JT,JJ)+EE(JJ,JT)).EQ.0.) GO TO 120
      DO 90 JS=1,9
   90 GB=GB+(VD(JS,JT)-VD(JS,JJ))**2
C
C     CALCULATE THE LIMITING CASE VALUE FOR A HYPERSPHERE
C     WITHIN A HYPERSPHERE.
C
      XK1=-SQRT(UB(JJ)/GB)
      XLIM=(1.-2.*XK1)*GB+UB(JJ)
      WRITE(6,1027) GB,XK1,XLIM
      XK2=0.
      PXK2=0.
      NI=0
      EE(JT,JJ)=0.
      II=0
      ISIN=0
      BETA=.5
      WRITE(6,1024)
      WRITE(6,1025) JJ,EE(JT,JJ),JT,EE(JJ,JT),PXK2,XK2
      IF(UB(JT).LT.XLIM) GO TO 105
      IF(EE(JJ,JT).EQ.0.) RA2=UB(JJ)
      IF(EE(JJ,JT).EQ.0.) GO TO 93
C
C     CLASS JJ HYPERSPHERE IS WITHIN CLASS JT HYPERSPHERE.
C
      XK2=.05
   91 XK2=XK2*2.
   92 RA2=((XK1+XK2)**2)*GB
   93 DO 94 JS=1,9
   94 PSID(JS)=(1.+XK2)*VD(JS,JJ)-XK2*VD(JS,JT)
      EEJJ=0.
```

```
      EEJT=0.
      DO 101 L=1,20
      HR=0.
C
C     DETERMINE THE NUMBER OF CLASS JJ AND JT TRAINING PATTERNS
C     MISCLASSIFIED BY THE SECOND LEVEL HYPERSPHERE THRESHOLD.
C
      DO 96 JS=1,9
   96 HR=HR+(WA(JS,L,JJ)-PSID(JS))**2
      IF(HR.LE.RA2) GO TO 98
      EEJJ=EEJJ+1
   98 HR=0.
      DO 100 JS=1,9
  100 HR=HR+(WA(JS,L,JT)-PSID(JS))**2
      IF(HR.GT.RA2) GO TO 101
      EEJT=EEJT+1.
  101 CONTINUE
      WRITE(6,1025) JJ,EEJJ,JT,EEJT,PXK2,XK2
      NI=NI+1
      IF(NI.GT.100) GO TO 103
C
C     PREDICT THE VALUE OF XK2 GIVING A MINIMUM ERROR.
C
      IF(EEJJ.EQ.EEJT) GO TO 103
      IF(EEJJ.GT.EEJT) ISIN=1
      IF(ISIN.EQ.1) GO TO 102
      SOC=(EEJJ-EEJT)-(EE(JT,JJ)-EE(JJ,JT))
      IF(SOC.EQ.0.) GO TO 91
      POI=(PXK2*(EEJJ-EEJT)-XK2*(EE(JT,JJ)-EE(JJ,JT)))/SOC
      IF(POI.LT.XK2) GO TO 102
      PXK2=XK2
      WRITE(6,1026) SOC,POI
      EE(JT,JJ)=EEJJ
      EE(JJ,JT)=EEJT
      XK2=POI
      IF(EEJJ.NE.EEJT) GO TO 92
      GO TO 103
C
C     STORE THE LIMITS OF THE CROSSOVER AREA.
C
  102 IF(II.EQ.0) XA=XK2
      IF(II.EQ.0) XB=PXK2
C
C     ADJUST THE CROSSOVER AREA AFTER EACH ITERATION.
C
      II=1
      ISIN=1
      IF(EEJJ.GT.EEJT) XA=XK2
      IF(EEJJ.LT.EEJT) XB=XK2
```

```
      XK2=.5*(XA+XB)
      PXK2=XB
      IF(EEJJ.NE.EEJT) GO TO 92
  103 DO 104 JS=1,9
      AHR(JT,JJ)=RA2
  104 PSI(JS,JT,JJ)=PSID(JS)
      EE(JT,JJ)=EEJJ
      EE(JJ,JT)=EEJT
      GO TO 120
  105 XK1=.5*(1.-SQRT((UB(JT)-UB(JJ))/GB))
      IF(XK1.GT.(.5)) XK1=.5
      WRITE(6,1027) GB,XK1,XLIM
      IF(EE(JJ,JT).EQ.0.) RA2=UB(JJ)
      IF(EE(JJ,JT).EQ.0.) GO TO 108
C
C     THE UNION OF THE JJ AND JT HYPERSPHERE THRESHOLDS DOES
C     NOT INCLUDE ALL OF THE JJ HYPERSPHERE.
C
      XK2=.05
  106 XK2=XK2*10.
  107 XKM=1.-2.*XK1
      RA2=(1.-XK2/XKM)*UB(JJ)+UB(JT)*XK2/XKM+XK2*(XK2-XKM)*GB
  108 DO 109 JS=1,9
  109 PSID(JS)=(1.+XK2)*VD(JS,JJ)-XK2*VD(JS,JT)
      EEJJ=0.
      EEJT=0.
C
C     DETERMINE THE NUMBER OF CLASS JJ AND JT TRAINING PATTERNS
C     MISCLASSIFIED BY THE SECOND LEVEL HYPERSPHERE THRESHOLD.
C
      DO 115 L=1,20
      HR=0.
      DO 110 JS=1,9
  110 HR=HR+(WA(JS,L,JJ)-PSID(JS))**2
      IF(HR.LE.RA2) GO TO 112
      EEJJ=EEJJ+1.
  112 HR=0.
      DO 114 JS=1,9
  114 HR=HR+(WA(JS,L,JT)-PSID(JS))**2
      IF(HR.GT.RA2) GO TO 115
      EEJT=EEJT+1.
  115 CONTINUE
      WRITE(6,1025) JJ,EEJJ,JT,EEJT,PXK2,XK2
      WRITE(6,1028) RA2
      NI=NI+1
      IF(NI.GT.100) GO TO 117
C
C     PREDICT THE VALUE OF XK2 GIVING A MINIMUM ERROR.
C
```

```
      IF(EEJJ.EQ.EEJT) GO TO 117
      IF(EEJJ.GT.EEJT) ISIN=1
      IF(ISIN.EQ.1) GO TO 116
      SOC=(EEJJ-EEJT)-(EE(JT,JJ)-EE(JJ,JT))
      IF(SOC.EQ.0.) GO TO 106
      POI=(PXK2*(EEJJ-EEJT)-XK2*(EE(JT,JJ)-EE(JJ,JT)))/SOC
      IF(POI.LT.XK2) GO TO 116
      PXK2=XK2
      WRITE(6,1026) SOC,POI
      EE(JT,JJ)=EEJJ
      EE(JJ,JT)=EEJT
      XK2=POI
      IF(EEJT.NE.EEJT) GO TO 107
      GO TO 117
C
C     STORE THE LIMITS OF THE CROSSOVER AREA.
C
  116 IF(II.EQ.0) XA=XK2
      IF(II.EQ.0) XB=PXK2
C
C     ADJUST THE CROSSOVER AREA AFTER EACH ITERATION.
C
      II=1
      ISIN=1
      IF(EEJJ.GT.EEJT) XA=XK2
      IF(EEJJ.LT.EEJT) XB=XK2
      XK2=BETA*(XA+XB)
      PXK2=XB
      IF(XB.GT.(.999*XA)) ISIN=0
      IF(ISIN.EQ.0) XK2=5.*XA
      IF(ISIN.EQ.0) PXK2=0.
      IF(ISIN.EQ.0) EE(JT,JJ)=0.
      IF(ISIN.EQ.0) EE(JJ,JT)=0.
      IF(ISIN.EQ.0) GO TO 107
      IF(EEJJ.NE.EEJT) GO TO 107
  117 DO 118 JS=1,9
      AHR(JT,JJ)=RA2
  118 PSI(JS,JT,JJ)=PSID(JS)
      EE(JJ,JT)=EEJT
      EE(JT,JJ)=EEJJ
  120 CONTINUE
      WRITE(6,1012)
      WRITE(6,1002) ((AHR(JT,JJ),JJ=1,9),JT=1,9)
      WRITE(6,1020) XPAGE
      XPAGE=XPAGE+1.
      DO 121 JJ=1,9
      IF(JJ.EQ.6) WRITE(6,1020) XPAGE
      WRITE(6,1013) JJ
      WRITE(6,1001)
```

```
  121 WRITE(6,1002) ((PSI(JS,JT,JJ),JT=1,9),JS=1,9)
      XPAGE=XPAGE+1.
      WRITE(6,1019)
      WRITE(6,1001)
C
C     THE MATRIX EE IS READ AS THE MISCLASSIFICATION IN CLASS
C     JD GIVEN THE TRAINING PATTERN ORIGINATED IN CLASS J. AT
C     THIS POINT IN THE SIMULATION THE MATRIX CONTAINS THE
C     TRAINING PATTERNS SECOND LEVEL HYPERSPHERE THRESHOLD
C     CLASS TO CLASS SEPARATION DATA..
C
      WRITE(6,1002) ((EE(JD,J),J=1,9),JD=1,9)
      WRITE(6,1020) XPAGE
      XPAGE=XPAGE+1.
      DO 122 J=1,9
      ERR(J)=0.
      EMLC(J)=0.
      UNC(J)=0.
  122 UNR(J)=0.
      IPA=0
      IPB=0
      IPC=0
      IPD=0
      IPE=0
      IPF=0
      IPG=0
      IPH=0
      IPI=0
C
C     NOW GENERATE SOME PATTERNS (UNKNOWN) AND LET THE
C     MACHINE CLASSIFY THEM. THE THRESHOLD LEVELS T1,T2,T3,
C     T4,T5,T6,T7,T8, AND T9 ARE SET TO MAKE ALL PATTERN
C     CLASSES EQUALLY LIKELY.
C
      IJKL=0
      IJK=55
      DO 200 NP=1,5000
      CALL RANDU(IX,IY,SS)
      IX=IY
      IF(SS.GE.T1) GO TO 124
      J=1
      IPA=IPA+1
      IF(IPA.GT.100) GO TO 200
      GO TO 140
  124 IF(SS.GE.T2) GO TO 126
      J=2
      IPB=IPB+1
      IF(IPB.GT.100) GO TO 200
      GO TO 140
```

```
  126 IF(SS.GE.T3) GO TO 128
      J=3
      IPC=IPC+1
      IF(IPC.GT.100) GO TO 200
      GO TO 140
  128 IF(SS.GE.T4) GO TO 130
      J=4
      IPD=IPD+1
      IF(IPD.GT.100) GO TO 200
      GO TO 140
  130 IF(SS.GE.T5) GO TO 132
      J=5
      IPE=IPE+1
      IF(IPE.GT.100) GO TO 200
      GO TO 140
  132 IF(SS.GE.T6) GO TO 134
      J=6
      IPF=IPF+1
      IF(IPF.GT.100) GO TO 200
      GO TO 140
  134 IF(SS.GE.T7) GO TO 136
      J=7
      IPG=IPG+1
      IF(IPG.GT.100) GO TO 200
      GO TO 140
  136 IF(SS.GE.T8) GO TO 138
      J=8
      IPH=IPH+1
      IF(IPH.GT.100) GO TO 200
      GO TO 140
  138 IF(SS.GE.T9) GO TO 200
      J=9
      IPI=IPI+1
      IF(IPI.GT.100) GO TO 200
  140 DO 142 I=1,25
      RN=-10.
C
C     USE THE CENTRAL LIMIT THEOREM TO APPROXIMATE THE
C     GAUSSIAN DISTRIBUTION.
C
      DO 141 IK=1,20
      CALL RANDU(IX,IY,RNN)
      IX=IY
  141 RN=RN+RNN
  142 X(I)=RN*DEV/A3+Y(I,J)
      DO 148 JS=1,9
      XF(JS)=0.
      DO 144 I=1,25
  144 WB(I)=X(I)-Y(I,JS)
```

```
      DO 148 K=1,25
      XD(K)=0.
      DO 146 I=1,25
  146 XD(K)=XD(K)+WB(I)*DB(I,K,JS)
  148 XE(JS)=XE(JS)+XD(K)*WB(K)
C
C     CALCULATE THE LOGARITHM OF THE LIKELIHOOD RATIO VECTOR.
C
      IJKL=IJKL+1
      IF(IJKL.NE.IJK) GO TO 149
      WRITE(6,1020) XPAGE
      XPAGE=XPAGE+1.
      IJK=IJK+54
      WRITE(6,1032)
  149 DO 150 JS=1,9
  150 WB(JS)=CB(JS)-(XE(JS)-XE(JC))/2./VAR
      WRITE(6,1023) J,(WB(JS),JS=1,9)
C
C     CALCULATE THE MAXIMUM LIKELIHOOD RATIO CLASSIFIER ERROR.
C
      JD=1
      IMLR=0
      DO 151 JS=2,9
      IF(WB(JS).LT.WB(JD)) GO TO 151
      JD=JS
  151 CONTINUE
      DO 152 JS=1,9
      IF(JS.EQ.JD) GO TO 152
      IF(WB(JS).EQ.WB(JD)) IMLR=IMLR+4
  152 CONTINUE
      IF(IMLR.GT.1) GO TO 153
      IF(J.EQ.JD) GO TO 154
  153 EMLC(J)=EMLC(J)+1.
C
C     CAN THE PATTERN BE CLASSIFIED BY THE FIRST LEVEL
C     HYPERSPHERE THRESHOLD
C
  154 DO 156 JD=1,9
      NOTJ(JD)=0
      PATC(JD)=0.
      VG(JD)=0.
      DO 156 JS=1,9
  156 VG(JD)=VG(JD)+((WB(JS)-VD(JS,JD))**2)
      JNOT=0
      DO 158 JD=1,9
      NOT=0
      IF(VG(JD).GT.UB(JD)) GO TO 158
      NOT=1
      NOTJ(JD)=1
```

```
      PATC(JD)=1.
      JTS=JD
  158 JNOT=JNOT+NOT
      WRITE(6,1023) J,(PATC(JD),JD=1,9)
      IF(JNOT.NE.0) GO TO 160
C
C     THE PATTERN WAS OUTSIDE THE FIRST LEVEL HYPERSPHERE
C     THRESHOLD DECISION SPACE.
C
      UNC(J)=UNC(J)+1.
      GO TO 200
  160 IF(JNOT.GT.1) GO TO 162
C
C     THE PATTERN WAS WITHIN THE DECISION SPACE BOUNDED
C     BY A HYPERSPHERE THRESHOLD.
C
      IF(J.NE.JTS) ERR(J)=ERR(J)+1.
      GO TO 200
C
C     CAN THE PATTERN BE SEPARATED BY THE SECOND LEVEL
C     HYPERSPHERE THRESHOLD.
C
  162 DO 170 JA=1,9
C
C     DID THE PATTERN FALL WITHIN CLASS JA
C
      IF(PATC(JA).EQ.0) GO TO 170
      DO 170 JD=1,9
C
C     DID THE PATTERN FALL WITHIN CLASS JD
C
      IF(PATC(JD).EQ.0.) GO TO 170
      IF(JD.EQ.JA) GO TO 170
C
C     DID THE TRAINING PATTERNS PROVIDE A SECOND LEVEL
C     SEPARATION OF CLASS JA AND JD
C
      IF(AHR(JD,JA).EQ.0) GO TO 170
      SB=AHR(JD,JA)
      SD=0.
      DO 164 JS=1,9
  164 SD=SD+((WB(JS)-PSI(JS,JD,JA))**2)
      IF(SD.GT.SB) GO TO 166
      NOTJ(JD)=0
      GO TO 170
  166 NOTJ(JA)=0
  170 CONTINUE
      WRITE(6,1029) J,(NOTJ(JD),JD=1,9)
      INUM=0
```

```
      DO 172 JD=1,9
      IF(NOTJ(JD).EQ.0) GO TO 172
      IPAT=JD
      INUM=INUM+1
  172 CONTINUE
C
C     CAN THE PATTERN BE CLASSIFIED IN MORE THAN ONE CLASS
C
      IF(INUM.NE.1) GO TO 174
C
C     IS THE PATTERN CORRECTLY CLASSIFIED
C
      IF(IPAT.NE.J) ERR(J)=ERR(J)+1.
      GO TO 200
  174 UNR(J)=UNR(J)+1.
  200 CONTINUE
      WRITE(6,1016)
      WRITE(6,1002) (EMLC(J),J=1,9)
      WRITE(6,1017)
      WRITE(6,1002) (ERR(J),J=1,9)
      WRITE(6,1018)
      WRITE(6,1002) (UNC(J),J=1,9)
      WRITE(6,1002) (UNR(J),J=1,9)
C
C     WRITE THE LAST VALUE OF IX
C
      WRITE(6,1030) IX
C
C     IF THE SIGNAL TO NOISE POWER RATIOS ARE SET AT THE VALUES
C          S/N=10.,2.,1.,.5,.2,.AND .1
C     AND ALLOWED TO RUN IN SERIES, THEN APPROXIMATELY
C     THIRTY-ONE MINUTES OF IBM 360-75 TIME IS REQUIRED FOR THE
C     COMPUTATIONS. THE AVERAGE TIME FOR ONE S/N LEVEL IS
C     APPROXIMATELY NINE MINUTES.
C
      ICON=ICON+1
      IF(ICON.EQ.2) VAR=1250./3.
      IF(ICON.EQ.3) VAR=2500./3.
      IF(ICON.EQ.4) VAR=12500./3.
      IF(ICON.EQ.5) VAR=5000./3.
      IF(ICON.EQ.6) VAR=25000./3.
      IF(ICON.EQ.7) GO TO 202
      GO TO 500
  202 STOP
      END
```

VITA

The author was born on 7 August 1940 in Neosho, Missouri. He received his primary and secondary education in the Missouri and California school systems. He received a B.S. in Electrical Engineering in 1962 and a M.S. in Electrical Engineering in 1968 from the University of Missouri at Rolla.