

1967

Comparison of the markov mesh assumption with other statistical assumptions in the classification of binary data

James David Womer
Lehigh University

Follow this and additional works at: <https://preserve.lehigh.edu/etd>

 Part of the [Applied Mathematics Commons](#)

Recommended Citation

Womer, James David, "Comparison of the markov mesh assumption with other statistical assumptions in the classification of binary data" (1967). *Theses and Dissertations*. 3570.
<https://preserve.lehigh.edu/etd/3570>

This Thesis is brought to you for free and open access by Lehigh Preserve. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of Lehigh Preserve. For more information, please contact preserve@lehigh.edu.

COMPARISON OF THE MARKOV MESH ASSUMPTION WITH OTHER
STATISTICAL ASSUMPTIONS IN THE CLASSIFICATION
OF BINARY DATA

by
James David Womer

A THESIS

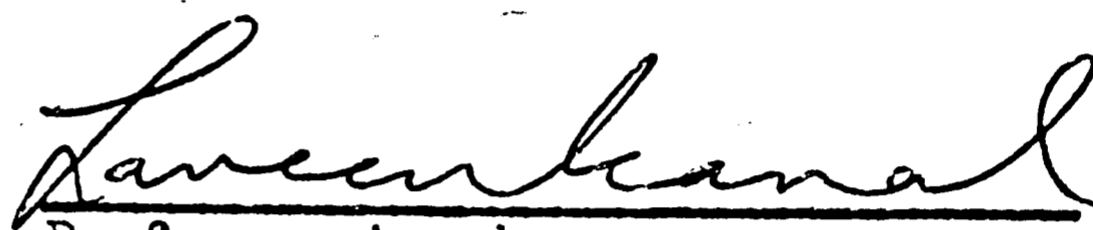
Presented to the Graduate Faculty
of Lehigh University
in Candidacy for the Degree of
Master of Science

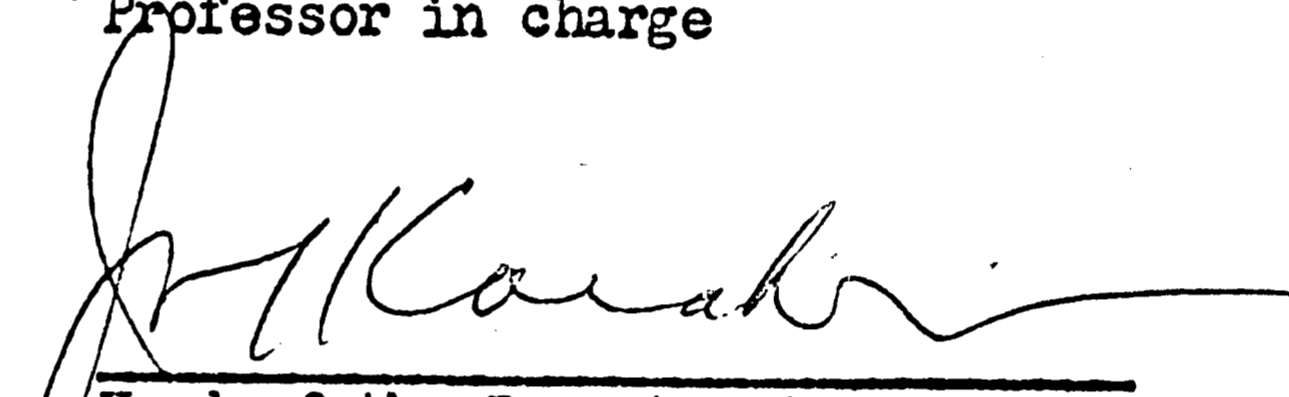
Lehigh University

1967

This thesis is accepted and approved in partial fulfillment of the requirements for the degree of Master of Science.

Sept. 19, 1967


Professor in charge


Head of the Department

Acknowledgments

I would like to express my thanks to Dr. Laveen N. Kanal for the help and guidance he afforded while this thesis was being prepared. I would also like to thank Philco-Ford Corporation, Blue Bell, Pa., for the use of their computing facilities and Dr. Kenneth Abend, a Philco-Ford employee, for his help. I also wish to thank the National Science Foundation which provided a Fellowship which enabled me to undertake this work and Lehigh University which provided the opportunity for my work.

Table of Contents

	page
Abstract	1
Chapter 1. Introduction	3
Chapter 2. Theoretical Considerations	7
Classification Theory	7
Discriminant Functions for Normal Populations	11
Binary Variables	13
Independent Binary	14
Dependent Binary Variables	15
Markov Chain	16
Markov Mesh	21
Markov Tree	25
Discrimination Without Known Probability Distributions	30
Four Layer Process	32
Chapter 3. Experimental Work	36
The Nature of the Samples	36
Previous Work	37

. Table of Contents (cont.)

Experimental Work	38
Results	48
Chapter 4. Conclusions	52
Appendix	53
Markov Mesh Discriminant Function	54
Linear Normal Discriminant Function with Estimated Parameters	58
Derivation of Number of Inputs for Optimum Multilevel Decision Process	59
Sample output of Markov mesh program (Format 3)	61
Sample output of four level process program	62
References	63
Vita	65

List of Figures

	page
Fig. 1 n-space Universe Consisting of Two Populations	7
Fig. 2 Definitions of Arrays Used With the Markov Mesh	23
Fig. 3 Spatial Configuration of Some Simple $U_{a,b}$ and Corresponding $Y_{a,b}$	24
Fig. 4 Examples of Markov Tree	29
Fig. 5 Format 1	39
Fig. 6 Format 2	40
Fig. 7 Format 3	41
Fig. 8 Format 4	42
Fig. 9 Format 5	43
Fig. 10 Flow Diagram for Markov Mesh Program	45
Fig. 11 Flow Diagram for Four Layer Process	47

List of Tables

Table I Performance of the Various Methods page 49

ABSTRACT

The thesis deals primarily with the study of Markov mesh statistical dependence and the use of this mesh assumption in pattern classification. The patterns classified were binary random variables and came from one of two populations. The Markov mesh assumption was used to derive a discriminant function with the aid of design samples. Test samples were classified. The population from which the test samples came were not known a priori. The test samples were independent of the design samples. The number of errors of classification were noted.

The performance of the discriminant function based on the Markov assumption was compared to the performance of other discriminant functions obtained under other statistical assumptions. The Markov assumption performed as well as but not better than the other methods.

The sample size problem is noted and discussed. This problem is as yet unsolved.

A four-level linear normal discriminant function was obtained and evaluated. All evaluations were made using identical data. The four level decision process was used

because the restriction of small sample size is not as severe.
The number of errors was very much greater than with those
discriminant functions using other assumptions about the
probability.

Chapter 1. Introduction

In the course of each day, everyone is called upon to make many decisions. Some of these decisions are unique; the occasion for the decision occurs only seldom, while the occasions for some other decisions occur frequently.

A man working on an assembly line, inspecting a product and deciding if it is faulty or not is an example of the latter type of decision. Another example would be a sonar operator deciding if he was hearing a ship or a school of fish (if he is good enough, the type of fish could also be decided).

Many of these routine decisions are simply classifications.

In the above examples a sound is classified into Group 1 (ships), Group 2 (submarines), or Group 3 (fish) and a product is classified as a good or a bad product.

Instead of having man make these routine classifications, it would be desirable to have a machine make this type of classification and thus liberate men to make the more unique decisions. This has been done. Work done in the field of "pattern recognition" or "pattern classification" has accomplished this aim to a limited extent. For example, Philco-Ford Corporation has built a machine which scans an

aerial photograph and locates M-48 tanks (army military tanks) on the photograph. The discrimination between tanks and non-tanks occurs with a high degree of accuracy.

In classifying something one must first determine the characteristics of the object to be classified and then use these characteristics to decide to which group the object belongs. In determining the characteristics, various measurements are made on the object. These measurements are then variables in a "discriminant function". The value of the discriminant function determines in which group the object is to be placed. The problem in classification is to choose the correct discriminant function and to choose the proper measurements to make. Choosing the best measurements to make on an object is a difficult and unsolved problem. In this paper we assume that the measurements made are properly chosen and thus ignore this problem. The paper is concerned with examining various discriminant functions.

There are two types of discriminant functions. One type of discriminant function is based on assumptions concerning the statistical distributions of the populations from which the objects come. Since each object of a given population

is not identical the measurement of a certain trait will vary from object to object within the group. There will be some statistical distribution of the measurements. This is termed a parametric approach. Different statistical distributions of the measurements may yield different forms (i.e. linear, quadratic, cubic, etc.) of the discriminant functions. However, sometimes the form stays the same but different coefficients result. The second type of discriminant function is one derived without assuming anything about the probability distribution of the variables. This type is termed non-parametric. This paper is concerned primarily with the parametric type of discriminant function. An assumption of Markov mesh¹ statistical dependence is made and the discriminant function calculated. The discriminant function is used to classify objects. The number of misclassifications is a measure of the performance of the discriminant function. The discrimination based on the Markov mesh assumption is compared with previous work done using a different assumption concerning the statistical distribution of the variables. Some work is also done with multilayer discrimination. The sample size problem is

also noted and discussed.

Chapter 2. Theoretical Considerations

Classification Theory

We consider the case where the individual (denoting either a person or an object) to be classified can come from one of two populations (groups). Population 1 will be denoted as G_1 and population 2 as G_2 . An extension of the following theory to more than two populations can be made.²

Let X (a vector) denote the measurements made on the individual.

$$X = (x_1, x_2, \dots, x_n)$$

X is a point in n -space. We divide the n -space into two parts, R_1 and R_2 (Fig. 1).

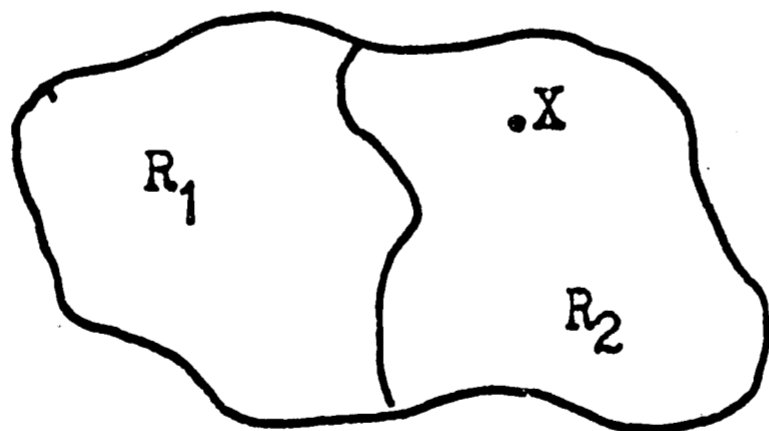


Fig. 1

n -space Universe Consisting of Two Populations

If X falls in R_1 we say that the individual belongs to G_1 and if X is in R_2 we say that he belongs to G_2 . One must accordingly determine the boundaries of the region R_1 and R_2 . The boundaries may be decided by using the criterion that the total loss incurred in the classification should be a minimum. We consider the following costs of misclassification

		decision	
		1	2
Real World	1	$C(1/1)$	$C(2/1)$
	2	$C(1/2)$	$C(2/2)$

Here $C(2/1)$ is the cost of saying that the individual belongs to G_2 when in reality it belongs to G_1 . A similar interpretation is given to $C(1/2)$, $C(2/2)$, and $C(1/1)$.

Assume X is distributed as $P_1(X)$ in G_1 and $P_2(X)$ in G_2 .

($P_1(X)$ and $P_2(X)$ are called probability density functions

of X .) Let q_1 denote the proportion of the universe occupied

by G_1 and q_2 the proportion occupied by G_2 . A measure of

the total loss is then the cost of a classification multiplied

by the probability that the classification occurred. Thus

the total loss is

$$\begin{aligned}
 L = & C(1/2) \cdot [\text{probability that } X \text{ is in } G_2 \text{ and classified as } G_1] \\
 & + C(2/1) \cdot [\text{probability that } X \text{ is in } G_1 \text{ and classified as } G_2] \\
 & + C(1/1) \cdot [\text{probability that } X \text{ is in } G_1 \text{ and classified as } G_1] \\
 & + C(2/2) \cdot [\text{probability that } X \text{ is in } G_2 \text{ and classified as } G_2]
 \end{aligned}$$

$$\begin{aligned}
 \text{Where } & [\text{probability that } X \text{ is in } G_i \text{ and classified as } G_j] \\
 & = [\text{probability of classifying } X \text{ in } G_j / X \text{ comes from } G_i] \cdot q_i \\
 & = q_i P(j/i).
 \end{aligned}$$

Therefore the expected loss becomes:

$$L = C(1/2)P(1/2)q_2 + C(2/1)P(2/1)q_1 + C(1/1)P(1/1)q_1 + C(2/2)P(2/2)q_2.$$

$$\begin{aligned}
 L = & C(1/2)q_2 \int_{R_1} P_2(X) dX + C(2/1)q_1 \int_{R_2} P_1(X) dX \\
 & + C(1/1)q_1 \int_{R_1} P_1(X) dX + C(2/2)q_2 \int_{R_2} P_2(X) dX.
 \end{aligned}$$

$$\begin{aligned}
 \int_{R_1} P_2(X) dX & = \int_R P_2(X) dX - \int_{R_2} P_2(X) dX \\
 \int_{R_1} P_1(X) dX & = \int_R P_1(X) dX - \int_{R_2} P_1(X) dX
 \end{aligned}$$

$$L = c(1/2)q_2 \left[\int_R P_2(x) dx - \int_{R_2} P_2(x) dx \right] \\ + q_1 c(2/1) \int_{R_2} P_1(x) dx + c(2/2)q_2 \int_{R_2} P_2(x) dx \\ + c(1/1)q_1 \left[\int_R P_1(x) dx - \int_{R_2} P_1(x) dx \right]$$

$$L = c(1/2)q_2 + c(1/1)q_1 \\ + \int_{R_2} \left\{ [c(2/1)q_1 P_1(x) - c(1/1)q_1 P_1(x)] - [c(1/2)q_2 P_2(x) - c(2/2)q_2 P_2(x)] \right\} dx$$

We want to minimize L. To do this choose R_2 so that

$$P_1(x)q_1 [c(2/1) - c(1/1)] < P_2(x)q_2 [c(1/2) - c(2/2)]$$

$$\frac{P_1(x)}{P_2(x)} < \frac{c(1/2) - c(2/2)}{c(2/1) - c(1/1)} \cdot \frac{q_2}{q_1} = t$$

R_1 is chosen so that

$$P_1(x)/P_2(x) \geq \frac{c(1/2) - c(2/2)}{c(2/1) - c(1/1)} \cdot \frac{q_2}{q_1} = t$$

$P_1(x)/P_2(x)$ is called the likelihood ratio. The value of the likelihood ratio may be used to discriminate between

G_1 and G_2 . The likelihood ratio may then be used as a discriminant function. Any monotonic function of the likelihood ratio which preserves the above inequalities may be used instead. The monotonic function is used if it provides a discriminant function easier to work with than the likelihood ratio.

One particularly useful monotonic function is the logarithm. Taking the log of the likelihood ratio, we get the following discriminant function:

choose G_1 if $\log P_1(X) - \log P_2(X) \geq \log t$

choose G_2 if $\log P_1(X) - \log P_2(X) < \log t$.

The discriminant function is a function of the measurements.

The form depends on the nature of P_1 and P_2 .

Discriminant Functions for Normal Populations³

The multivariate normal density function is

$$P(X) = P(x_1, x_2, \dots, x_n) \\ = \frac{1}{(2\pi)^{n/2} |V|^{1/2}} \cdot \exp\left(-\frac{1}{2}(X-M)' V^{-1} (X-M)\right)$$

Here V is the covariance matrix.

$$V = \begin{bmatrix} v_{11} & v_{12} & \dots & v_{1n} \\ v_{21} & v_{22} & \dots & v_{2n} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ v_{n1} & v_{n2} & \dots & v_{nn} \end{bmatrix}$$

$v_{ij} = E[(x_i - m_i)(x_j - m_j)]$ and $m_i = E(x_i)$. $E(y)$ denotes the expected value of y . $M = (m_1, m_2, \dots, m_n)$ is the vector of means. $X = (x_1, x_2, \dots, x_n)$ is the vector of the random variables x_i . Let M_1 and M_2 be the mean vectors of G_1 and G_2 respectively and V_1 and V_2 be the covariance of G_1 and G_2 respectively. Then

$$P_1(X)/P_2(X) = \frac{|V_2|^{\frac{1}{2}}}{|V_1|^{\frac{1}{2}}} \cdot \frac{\exp(-\frac{1}{2}(X-M_1)'V_1^{-1}(X-M_1))}{\exp(-\frac{1}{2}(X-M_2)'V_2^{-1}(X-M_2))}$$

Taking the logarithm, the discriminant function is obtained

as

$$-\frac{1}{2}X'(V_1^{-1} - V_2^{-1})X + X'(V_1^{-1}M_1 - V_2^{-1}M_2)$$

and the threshold is

$$\log \left| \frac{V_2}{V_1} \right|^{\frac{1}{2}} + \frac{1}{2}(M_2'V_2^{-1}M_2 - M_1'V_1^{-1}M_1) + \log t$$

If $V_1 = V_2 = V$ the discriminant function becomes

$$x'V^{-1}(M_1 - M_2)$$

and the threshold is

$$\log t + \frac{1}{2}(M_1 + M_2)'V^{-1}(M_1 - M_2)$$

M_1 , M_2 , V_1 , and V_2 are called parameters of the distribution.

Binary Variables

Instead of having continuous variables, as in the case of the normal distribution, one may obtain binary variables. Binary variables may be the natural result of the measurements. The individual does or does not have brown hair; he does or does not have blue eyes. These are all examples of the measurements yielding binary results. Binary variables may also be the result of a preprocessing of the measurements. For example a photograph may be converted from a gray scale picture to a two-color, black and white picture. In any case binary variables do arise and discriminant functions based on them will now be looked at.

Independent Binary

Let $X=(x_1, x_2, \dots, x_n)$ and assume each x_i $i=1, \dots, n$ are binary and independent. Then $P(X)=P(x_1)P(x_2)\dots P(x_n)$

Let $P(x_i = 1) = \alpha_i$, then $P(x_i=0)=1-\alpha_i$ and

$$P(x_i) = \alpha_i^{x_i} (1 - \alpha_i)^{1-x_i}$$

$$P(X) = \prod_{i=1}^n \alpha_i^{x_i} (1 - \alpha_i)^{1-x_i}$$

$$P(X) = \prod_{i=1}^n \left(\frac{\alpha_i}{1 - \alpha_i} \right)^{x_i} (1 - \alpha_i)$$

$$= a_0 \cdot a_1^{x_1} \cdot a_2^{x_2} \dots a_n^{x_n}$$

where

$$a_0 = \prod_{i=1}^n (1 - \alpha_i)$$

$$a_i = \frac{\alpha_i}{1 - \alpha_i}$$

Suppose $P(X)$ is the distribution of X in G_1 . Then we may have a $Q(X)$ which is the distribution in G_2 .

$$Q(X) = b_0 \cdot b_1^{x_1} \cdot b_2^{x_2} \dots b_n^{x_n}$$

The b's are defined in a manner similar to the definition of the a's. The log of the likelihood ratio is

$$\sum_{i=1}^n \left(\log \frac{a_i}{b_i} \right) x_i + \log (a_0/b_0)$$

This is the discriminant function for independent binary variables. The threshold is $\log t$.

Dependent Binary Variables³

In general if the x_i 's are not independent $P(X)$ is a product involving 2^n exponents.³

$$P(X) = a_0 \cdot a_1^{x_1} \cdot a_2^{x_2} \dots a_n^{x_n} \cdot a_{12}^{x_1 x_2} \cdot a_{13}^{x_1 x_3} \dots (a_{n-1,n})^{x_{n-1} x_n} \dots (a_{1,2,3,\dots,n})^{x_1 x_2 \dots x_n}$$

There are 2^n exponents because the n binary variables can take on 2^n states. One exponent corresponds to one state.

For example, if $n = 3$, there are 8 states the variables

x_1, x_2, x_3 can assume, namely

000

001

010

011

100

101

110

111.

$$P(X) = a_0 a_1^{x_1} a_2^{x_2} a_3^{x_3} a_{12}^{x_1 x_2} a_{13}^{x_1 x_3} a_{23}^{x_2 x_3} a_{123}^{x_1 x_2 x_3}$$

A special case of dependent binary variables is Markovian dependency.

Markov Chain¹

Consider n variables x_1, x_2, \dots, x_n . A Markov chain dependency is one in which the probability of any variable, say x_k , given all preceding variables is equal to the probability of the variable given a predetermined, finite number of variables immediately preceding the variable.

Thus for a first order Markov chain

$$P(x_k/x_1 x_2 \dots x_{k-1}) = P(x_k/x_{k-1})$$

The probability of x_k depends only on the preceding variable.

For the r -th order Markov chain

$$P(x_k/x_1 \dots x_{k-1}) = P(x_k/x_{k-r} \dots x_{k-1}).$$

The probability of x_k depends only on the preceding r variables. If $k-r$ is zero or negative, the probability of x_k depends only on the $k-1$ variables.

For the first order Markov chain, a well known property is that $P(x_k/x_1 x_2 \dots x_j) = P(x_k/x_j)$. This is shown as follows:

By definition

$$P(x_k/x_1 \dots x_{k-1}) = P(x_k/x_{k-1})$$

$$\frac{P(x_j \dots x_k)}{P(x_j \dots x_{k-1})} = \frac{P(x_k/x_j \dots x_{k-1}) P(x_j \dots x_{k-1})}{P(x_j \dots x_{k-1})}$$

$$= P(x_k/x_j \dots x_{k-1}) = P(x_k/x_{k-1})$$

$$\sum_{x_{j+1} \dots x_{k-1}} \frac{P(x_1 \dots x_k)}{P(x_1 \dots x_{k-1})} = \sum_{x_{j+1} \dots x_{k-1}} \frac{P(x_j \dots x_k)}{P(x_j \dots x_{k-1})}$$

$$= \frac{P(x_1 \dots x_j, x_k)}{P(x_1 \dots x_j)} = \frac{P(x_j, x_k)}{P(x_j)}$$

$$= P(x_k/x_1 \dots x_j) = P(x_k/x_j) \quad \text{q.e.d.}$$

Also for $k < n$

$$\begin{aligned}
 P(x_k/x_1 x_2 \dots x_{k-1} x_{k+1} \dots x_n) &= \frac{P(x_1, \dots, x_n)}{P(x_1 \dots x_{k-1} x_{k+1} \dots x_n)} \\
 &= \frac{P(x_1)P(x_2/x_1) \dots P(x_k/x_{k-1})P(x_{k+1}/x_k) \dots P(x_n/x_{n-1})}{P(x_1)P(x_2/x_1) \dots P(x_{k+1}/x_{k-1}) \dots P(x_n/x_{n-1})} \\
 &= \frac{P(x_k/x_{k-1})P(x_{k+1}/x_k)}{P(x_{k+1}/x_{k-1})} = \frac{P(x_{k-1})P(x_k/x_{k-1})P(x_{k+1}/x_k)}{P(x_{k-1})P(x_{k+1}/x_{k-1})} \\
 &= \frac{P(x_{k-1} x_k x_{k+1})}{P(x_{k-1} x_{k+1})} = P(x_k/x_{k-1} x_{k+1})
 \end{aligned}$$

Thus any point is dependent only on its two nearest neighbors.

Similarly for the r -th order Markov chain,

$$P(x_k/x_1 x_2 \dots x_{k-1} x_{k+1} \dots x_n) = P(x_k/x_{k-r} \dots x_{k-1} x_{k+1} \dots x_{k+r}).$$

x_k is dependent only on its $2r$ nearest neighbors.

We develop the discriminant function for the first order Markov chain in the following manner. Let

$\alpha_i = P(x_i=1/x_{i-1}=0)$ and $\beta_i = P(x_i=1/x_{i-1}=1)$. Also define $x_i=0$ for $i < 1$ and $i > n$. Then $\alpha_1 = P(x_1=1)$, $\alpha_{n+1} = \beta_{n+1} = 0$.

$$P(x_1 x_2 \dots x_n) = P(x_1)P(x_2/x_1) \dots P(x_n/x_{n-1})$$

$$P(x_1 \dots x_n) = \alpha_1^{x_1} (1 - \alpha_1)^{1 - x_1} \prod_{i=2}^n \left[\beta_i^{x_{i-1} x_i} (1 - \beta_i)^{x_{i-1} (1 - x_i)} \right] \\ \times \left[\alpha_i^{(1 - x_{i-1}) x_i} (1 - \alpha_i)^{(1 - x_{i-1}) (1 - x_i)} \right]$$

Taking logarithms:

$$\log P(x_1 \dots x_n) = A_0 + \sum_{i=1}^n A_i x_i + \sum_{i=2}^n B_i x_{i-1} x_i$$

where

$$A_0 = \sum_{i=1}^n \log (1 - \alpha_i)$$

$$A_i = \log \frac{\alpha_i}{1 - \alpha_i} + \log \frac{1 - \beta_{i+1}}{1 - \alpha_{i+1}}$$

$$B_i = \log \frac{\beta_i}{1 - \beta_i} - \log \frac{\alpha_i}{1 - \alpha_i}$$

Likewise for the second order Markov chain

$$\log P(x_1 \dots x_n) = A_0 + \sum_{i=1}^n A_i x_i + \sum_{i=2}^n B_i x_{i-1} x_i \\ + \sum_{i=3}^n C_i x_{i-2} x_i + \sum_{i=3}^n D_i x_{i-2} x_{i-1} x_i$$

$$\text{where } A_0 = \sum_{i=1}^n \log (1 - \alpha_i)$$

$$A_i = \log \frac{\alpha_i}{1-\alpha_i} + \log \frac{1-\beta_{i+1}}{1-\alpha_{i+1}} + \log \frac{1-\gamma_{i+2}}{1-\alpha_{i+2}}$$

$$B_i = \log \frac{\beta_i}{1-\beta_i} + \log \frac{1-\delta_{i+1}}{1-\gamma_{i+1}} - \log \frac{\alpha_i}{1-\alpha_i} - \log \frac{1-\beta_{i+1}}{1-\alpha_{i+2}}$$

$$C_i = \log \frac{\gamma_i}{1-\gamma_i} - \log \frac{\alpha_i}{1-\alpha_i}$$

$$D_i = \log \frac{1-\delta_i}{1-\delta_i} - \log \frac{\gamma_i}{1-\gamma_i} - \log \frac{\beta_i}{1-\beta_i} + \log \frac{\alpha_i}{1-\alpha_i}$$

Here $\alpha_i = P(x_i=1/x_{i-2}=0, x_{i-1}=0)$

$$\beta_i = P(x_i=1/x_{i-2}=0, x_{i-1}=1)$$

$$\gamma_i = P(x_i=1/x_{i-2}=1, x_{i-1}=0)$$

$$\delta_i = P(x_i=1/x_{i-2}=1, x_{i-1}=1)$$

In like manner the joint probability may be determined for other order Markov chains. There are $2^r(n-r+1) - 1$ coefficients for the r-th order chain.

The discriminant function is obtained from the expression for the joint probability. Use a superscript 1 to identify coefficients pertaining to population 1 and a superscript 2 to identify coefficients pertaining to population 2.

Then the discriminant function becomes (for the first order

Markov chain)

$$A_0^{(1)} - A_0^{(2)} + \sum_{i=1}^n (A_i^{(1)} - A_i^{(2)}) x_i + \sum_{i=2}^n (B_i^{(1)} - B_i^{(2)}) x_{i-1} x_i$$

Markov Mesh¹

Instead of having a linear or temporal sequence of variables, as in the Markov chain, we consider a two dimensional array of binary variables as in Fig. 2a. Such an array could result from a gray scale picture being processed to yield a black and white matrix. (The picture is divided into small squares. The squares are then made either black or white depending on the characteristics of the gray scale picture.) By making certain assumptions similar to the assumptions made for the Markov chain, one obtains the Markov mesh distribution wherein the probability of a given element is dependent only on certain of its nearest neighbors.. The development follows.

We make the following definitions:

$\chi_{m,n}$ is an $m \times n$ matrix of binary variables (Fig. 2a.)

$x_{a,b}$ is the variable in row a and column b

$Z_{m,n}^{a,b}$ is the non-rectangular array of all variables $x_{i,j}$ with $i < a$ or $j < b$ (i.e. all variables to the left of or above $x_{a,b}$) (Fig. 2b.)

Similar to the Markov chain, the Markov mesh yields the following defining equation:

$$P(x_{a,b} / Z_{m,n}^{a,b}) = P(x_{a,b} / U_{a,b})$$

where $U_{a,b}$ is some array of variables adjacent to but to the left of or above $x_{a,b}$. It may be shown that

$$P(\chi_{m,n}) = \prod_{i=1}^m \prod_{j=1}^n P(x_{i,j} / U_{i,j})$$

and $P(x_{a,b} / \chi_{m,n}^{a,b}) = P(x_{a,b} / Y_{a,b})$. Various $U_{a,b}$ and the corresponding $Y_{a,b}$ are shown in Fig. 3. ($\chi_{m,n}^{a,b}$ is the array $\chi_{m,n}$ with the element $x_{a,b}$ deleted.)

Thus for the third order Markov mesh

$$P(\chi_{m,n}) = \prod_{i=1}^m \prod_{j=1}^n P(x_{i,j} / x_{i-1,j}, x_{i,j-1}, x_{i-1,j-1})$$

$$\begin{array}{cccccc}
 x_{1,1} & x_{1,2} & \dots & x_{1,b-1} & x_{1,b} & \dots & x_{1,n} \\
 x_{2,1} & x_{2,2} & \dots & x_{2,b-1} & x_{2,b} & \dots & x_{2,n} \\
 \cdot & \cdot & & \cdot & \cdot & & \cdot \\
 \cdot & \cdot & & \cdot & \cdot & & \cdot \\
 \cdot & \cdot & & \cdot & \cdot & & \cdot \\
 x_{a-1,1} & x_{a-1,2} & \dots & x_{a-1,b-1} & x_{a-1,b} & \dots & x_{a-1,n} \\
 x_{a,1} & x_{a,2} & \dots & x_{a,b-1} & x_{a,b} & \dots & x_{a,n} \\
 \cdot & \cdot & & \cdot & \cdot & & \cdot \\
 \cdot & \cdot & & \cdot & \cdot & & \cdot \\
 x_{m,1} & x_{m,2} & \dots & x_{m,b-1} & x_{m,b} & \dots & x_{m,n}
 \end{array}$$

$\chi_{m,n}$
Fig. 2a.

$$\begin{array}{cccccc}
 x_{1,1} & x_{1,2} & \dots & x_{1,b-1} & x_{1,b} & \dots & x_{1,n} \\
 x_{2,1} & x_{2,2} & \dots & x_{2,b-1} & x_{2,b} & \dots & x_{2,n} \\
 \cdot & \cdot & & \cdot & \cdot & & \cdot \\
 \cdot & \cdot & & \cdot & \cdot & & \cdot \\
 \cdot & \cdot & & \cdot & \cdot & & \cdot \\
 x_{a-1,1} & x_{a-1,2} & \dots & x_{a-1,b-1} & x_{a-1,b} & \dots & x_{a-1,n} \\
 x_{a,1} & x_{a,2} & \dots & x_{a,b-1} & & & \\
 \cdot & \cdot & & \cdot & & & \\
 \cdot & \cdot & & \cdot & & & \\
 \cdot & \cdot & & \cdot & & & \\
 x_{m,1} & x_{m,2} & \dots & x_{m,b-1} & & &
 \end{array}$$

Fig. 2b. $Z_{m,n}^{a,b}$

Fig. 2 Definitions of Arrays Used With the Markov Mesh

$U_{a,b}$	$Y_{a,b}$
$\begin{array}{c} X \\ X_{a,b} \end{array}$	$\begin{array}{c} X \ X \\ X_{a,b} \ X \\ X \ X \end{array}$
$\begin{array}{c} X \ X \\ X_{a,b} \end{array}$	$\begin{array}{c} X \ X \ X \\ X_{a,b} \ X \\ X \ X \ X \end{array}$
$\begin{array}{c} X \\ X \ X \\ X \ X_{a,b} \end{array}$	$\begin{array}{c} X \ X \ X \\ X \ X \ X \ X \\ X \ X_{a,b} \ X \ X \\ X \ X \ X \ X \\ X \ X \ X \end{array}$
$\begin{array}{c} X \ X \\ X \ X \ X \\ X \ X_{a,b} \end{array}$	$\begin{array}{c} X \ X \ X \ X \\ X \ X \ X \ X \ X \\ X \ X_{a,b} \ X \ X \\ X \ X \ X \ X \ X \\ X \ X \ X \ X \ X \end{array}$
$\begin{array}{c} X \ X \ X \\ X \ X \ X \\ X \ X_{a,b} \end{array}$	$\begin{array}{c} X \ X \ X \ X \ X \\ X \ X \ X \ X \ X \\ X \ X_{a,b} \ X \ X \\ X \ X \ X \ X \ X \\ X \ X \ X \ X \ X \end{array}$

Fig. 3 Spatial Configuration of Some Simple $U_{a,b}$ and Corresponding $Y_{a,b}$

This is developed into a discriminant function by following the procedure used for the Markov chain. Let

$$a_{ij} = P(x_{ij} = 1 / x_{i-1,j-1} = 0, x_{i-1,j} = 0, x_{i,j-1} = 0)$$

$$b_{ij} = P(x_{ij} = 1 / x_{i-1,j-1} = 0, x_{i-1,j} = 0, x_{i,j-1} = 1)$$

$$c_{ij} = P(x_{ij} = 1 / x_{i-1,j-1} = 0, x_{i-1,j} = 1, x_{i,j-1} = 0)$$

$$d_{ij} = P(x_{ij} = 1 / x_{i-1,j-1} = 0, x_{i-1,j} = 1, x_{i,j-1} = 1)$$

$$e_{ij} = P(x_{ij} = 1 / x_{i-1,j-1} = 1, x_{i-1,j} = 0, x_{i,j-1} = 0)$$

$$f_{ij} = P(x_{ij} = 1 / x_{i-1,j-1} = 1, x_{i-1,j} = 0, x_{i,j-1} = 1)$$

$$g_{ij} = P(x_{ij} = 1 / x_{i-1,j-1} = 1, x_{i-1,j} = 1, x_{i,j-1} = 0)$$

$$h_{ij} = P(x_{ij} = 1 / x_{i-1,j-1} = 1, x_{i-1,j} = 1, x_{i,j-1} = 1)$$

$P(\chi_{m,n})$ can be written in terms of the eight above parameters.

As in the case of the Markov chain the logarithm of this probability is taken. The logarithm of the probability of a second population is subtracted to give the discriminant function. This is shown in the Appendix.

Markov Tree

The above Markov mesh and chain are special cases of the Markov tree.^{4,5} A Markov tree dependency is one in which a variable is dependent on one or more other variables which

have no special spatial or temporal relationship to the primary variable. For example, in the first order tree

$$P(x_1 \dots x_n) = \prod_{i=1}^n P(x_i / x_{j(i)}), \quad 0 \leq j(i) < i. \quad j(i) \text{ is an integer the value of which is dependent on } i. \text{ If } j(i)=0, P(x_i / x_{j(i)}) = P(x_i). \text{ It is noted that } j(k) \text{ may equal } j(l). \text{ Likewise for the second order tree}$$

$$P(x_1 x_2 \dots x_n) = \prod_{i=1}^n P(x_i / x_{j(i)} x_{m(i)}).$$

Here $0 \leq j(i) < i$ and $0 \leq m(i) < i$. Again if $j(i)=0$ or if $m(i)=0$, the probability of x_i is not dependent on $x_{j(i)}$ or $x_{m(i)}$ respectively. It is obvious that the Markov chain is a special case of the Markov tree with (for first order case) $j(i)=i-1$. Likewise the Markov mesh is a special case of the third order Markov tree.

From $P(x_1 \dots x_n)$ the discriminant function can be obtained as in the preceding section. Here there is a slight difference however. Not only must the parameters be known (or estimated), but the functional relationship $j(i)$ must also be determined. A method of obtaining this relationship from a probabilistically known population or from samples of the population is presented⁴.

If x_j is dependent on x_i then we say that there is a branch of the tree connecting x_i and x_j . Consider only the first order tree. To determine the tree consider all possible branches (first order dependencies) and assign a weight to each branch. This weight is defined as

$$I(x_i, x_j) = \sum_{x_i, x_j} P(x_i, x_j) \log \frac{P(x_i, x_j)}{P(x_i)P(x_j)}$$

The branch weights are ordered and numbered so that branch b_i has a higher weight than b_j if $i < j$. The tree is formed by selecting branches b_1 and b_2 . Then the next b which does not form a loop in the tree is added. This process of adding branches of next lower weights under the restriction that no loops are formed (this restriction is needed because if a loop is formed, we would no longer have a tree) continues until all variables are included and the tree is formed. If there are branches with identical weights this method does not lead to a unique result. As an example⁴ let

$$I(x_1, x_2) = 0.079$$

$$I(x_1, x_3) = 0.00005$$

$$I(x_1, x_4) = 0.0051$$

$$I(x_2, x_3) = 0.189$$

$$I(x_2, x_4) = 0.0051$$

$$I(x_3, x_4) = 0.0051$$

Thus the first two branches are (x_2, x_3) and (x_1, x_2) . The third and final branch may be either (x_1, x_4) , (x_2, x_4) , or (x_3, x_4) . If we choose (x_1, x_4) , the joint probability is $P(x_1, x_2, x_3, x_4) = P(x_1)P(x_2/x_1)P(x_3/x_2)P(x_4/x_1)$. The tree is shown in Fig. 4a. The other possible trees are also shown in Fig. 4. An arrow from x_j to x_i means that x_j is dependent on x_i .

For the case where the population is not probabilistically known $I(x_i, x_j)$ must be estimated. $I(x_i, x_j)$ is estimated by

$$\hat{I}(x_i, x_j) = \sum_{u,v} f_{u,v}(i,j) \log \frac{f_{u,v}(i,j)}{f_u(i) f_u(j)}$$

$$\text{where } f_{u,v}(i,j) = \frac{n_{u,v}(i,j)}{\sum_{u,v} n_{u,v}(i,j)} \quad \text{and} \quad f_u(i) = \sum_v n_{u,v}(i,j).$$

$f_{u,v}(i,j)$ denotes $f(x_i = u, x_j = v)$ and $f_u(i)$ denotes $f(x_i = u)$. $n_{u,v}(i,j)$ is the number of samples such that their i -th and j -th components assume the values of u and v respectively.

A method of obtaining an optimum tree has been developed.⁵

This was done by using a computer. The program selected the

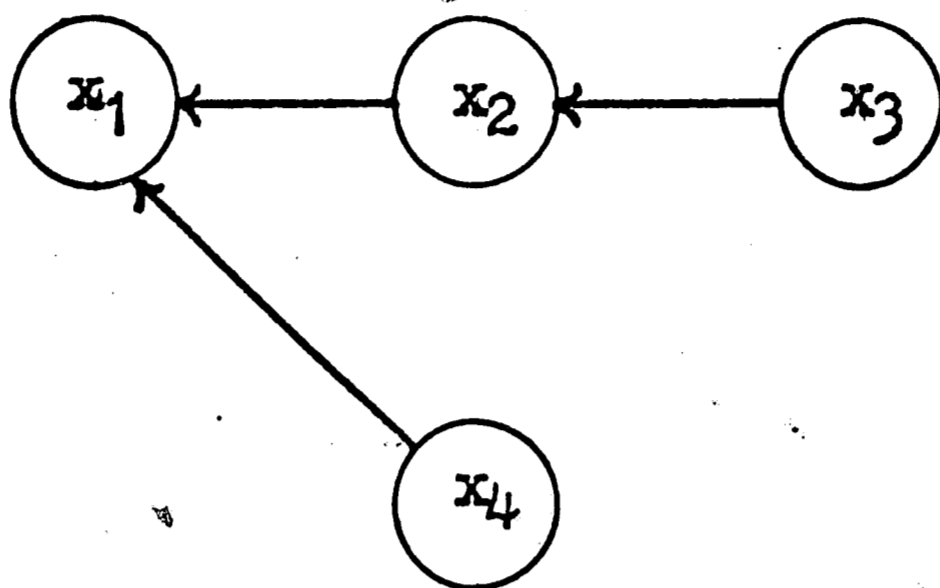


Fig. 4a.

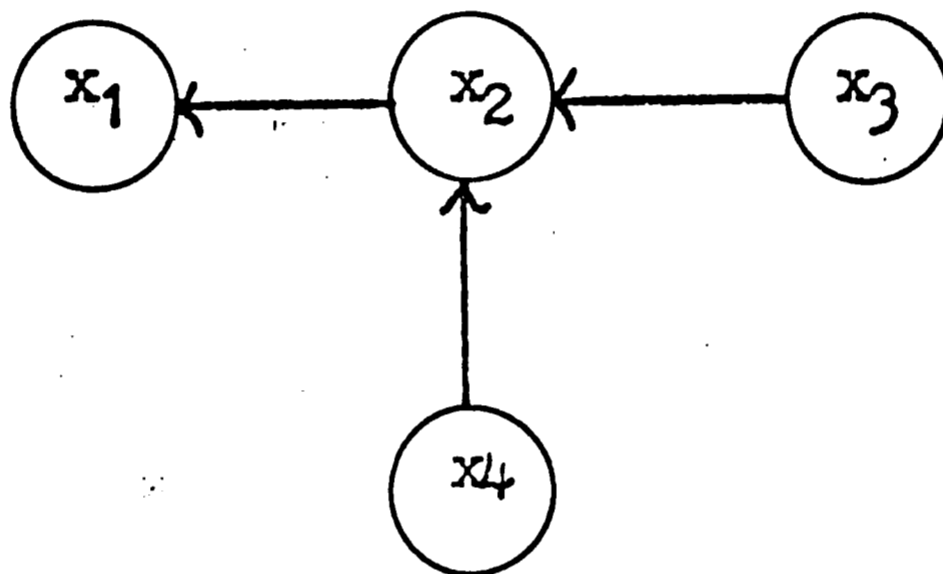


Fig. 4b.

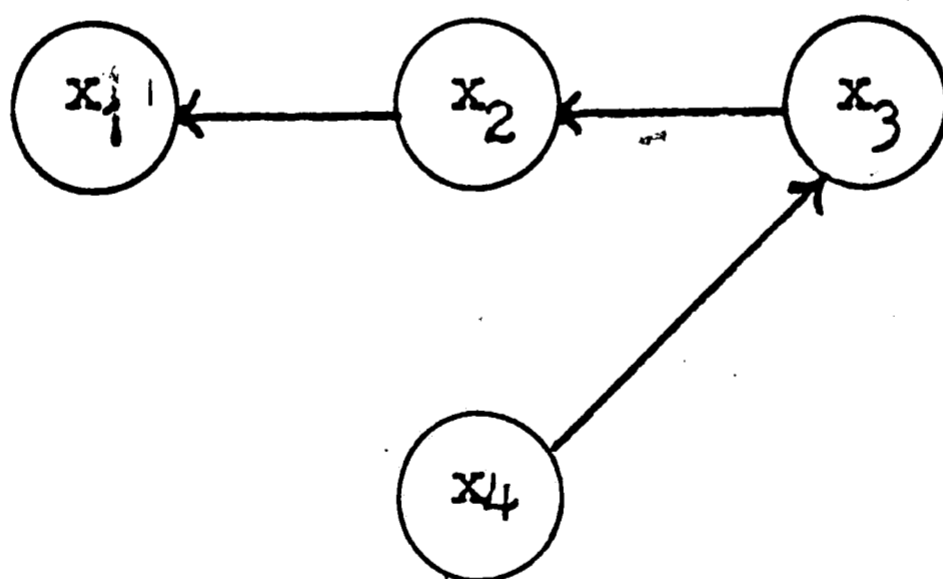


Fig. 4c.

Fig. 4 Examples of Markov Tree

first link in the chain. Then in an iterative procedure an optimum chain was arrived at. Another iterative procedure was then applied to the optimum chain to obtain an optimum tree. The criterion of optimality was a minimum error rate.

Experiments⁴ were performed with the Markov tree and it was found that for those particular experiments the first order tree gives an error rate one half that of the first order chain.

Discrimination Without Known Probability Distributions

In the case where the probability distribution of the population is not known the underlying form of the probability is assumed (Markov, linear normal, quadratic normal, etc.) and the parameters are estimated. These estimates for the parameters are obtained from measurements made on samples (design samples) from known populations. The parameters (such as the mean and variance of the variables) are calculated from the design samples. Thus the parameters are functions of the samples used. Changing the samples may change the estimates. Changing the number of design samples (sample size) also may change

the estimates.

In estimating a parameter one desires an unbiased consistent estimate. That is, an estimate is desired whose expected value is equal to the value of the estimated parameter and, as the number of design samples becomes large, an estimate that approaches the true value with probability one. We then have an estimate which has a mean equal to the true value and a spread (variance of the estimate) which is a function of the number of design samples. For a large number of design samples the spread is small while for a small number of design samples the spread is large and the estimate may be very far from the true value.

For example, let r_1, r_2, \dots, r_n be n design samples drawn from a one-dimensional normal population with mean m and variance v^2 . Let the estimate for m be

$$\bar{r} = (r_1 + r_2 + \dots + r_n)/n$$

and the estimate for v^2 be

$$s^2 = \frac{(r_1 - \bar{r})^2 + (r_2 - \bar{r})^2 + \dots + (r_n - \bar{r})^2}{n-1}$$

Then⁶ $E(\bar{r})=m$ and the variance of \bar{r} is v^2/n . Also the expected

value of s^2 is v^2 and the variance of s^2 is $a(r)(n-1)/n^2 - b(r)(n-3)/n^2$. $a(r)$ and $b(r)$ are parameters of the distribution of r and are finite. As may be seen as the number of samples, n , becomes very large the spread of s^2 and \bar{r} becomes zero and the value of the estimate approaches the true value with probability one.

In many cases the number of design samples are too small and good estimates are not obtained. This gives rise to a major problem in classification theory - the sample size problem. The difficulty arises when too many parameters are to be estimated with too few samples. A second difficulty is determining the optimum number of design samples to be used, and the relation between the number of design samples and independent test samples. These problems are as yet unsolved.

Four Layer Process

For reasonably good results, the number of design samples should be at least as large as the number of parameters to be estimated. This may be seen since with n variables, only n unique equations are possible. Thus, if there are more

parameters to be estimated than the number of samples, the estimated parameters will not be independent of each other.

Consider the case of 240 variables in each sample and consider 100 design samples. For the linear case (p.13) 240 coefficients must be estimated. If, however, a two level process which consists of 10 subsets of 24 variables each is used, the number of coefficients in the first layer to be estimated is 24 for each subset; the assumption is of course that the subsets are independent. In the second layer 10 coefficients must be estimated. Effectively 34 coefficients must be estimated with 100 samples. (Essentially the design samples are broken up into 10 parts. Each part must estimate 24 coefficients and the complete design sample must estimate 10 for the second layer. This effectively makes the 34 coefficients that must be estimated.) For a three layer process, consider breaking it up so that the first level consists of 30 discriminant functions with 8 variables, the second level consists of 5 discriminant functions with 6 variables, and the third consists of a discriminant function with 5 variables. This will be denoted as (5,6,8). Here $5 + 6 + 8 = 19$ coefficients must be evaluated. An n

level decision process is denoted as (N_1, N_2, \dots, N_n) .
 Here $\prod_{i=1}^n N_i = K =$ number of variables in the samples. The
 number of coefficients to be estimated is given by $\sum_{i=1}^n N_i = S$.
 It is desired to minimize S . For the case $n=2$, it should
 be obvious that $N_1=N_2 = \sqrt{240}$ for minimum S . Likewise for
 minimum S in the general case $N_i=N_j$. The problem is to
 minimize $\sum_{i=1}^n N_i$ given that $N_i^n = K$. This sum is minimized
 if $N_i = e = 2.72$ (p. 59). Thus to avoid the sample size
 problem, the decision process should consist of $n = \ln K$
 number of layers with e variables input into each discriminant
 function.

The number of inputs must be an integer and cannot be
 equal to e . If instead of e we would use $N_i = 3$, we would
 obtain the following decision process:

(3,3,3,3,3).

One dummy variable would have to be added to achieve this
 process. Fifteen coefficients would have to be estimated.
 If the process (3,5,4,4) were used, 16 coefficients would
 have to be estimated. For the data used in the experiment
 conducted in order to write this paper, the (3,5,4,4) proved

315

easier to work with than the (3,3,3,3,3) process. Since 16 is not significantly greater than 15, the four level process was evaluated using the assumption of linear normality to achieve the discriminant function.

It is reasoned that the multi-level process may give more accurate estimates of the parameters than the single level because there are less parameters to estimate for the given design sample size. Although in each stage the discriminant function is linear, the overall discriminant function is not linear. It is quite complex. So in addition to possibly providing better estimates for the parameters, the multi-level process implements a complex decision which should give better performance than the single plane of the one level linear process.

Chapter 3. Experimental Work

The Nature of the Samples

The individuals that were to be classified were aerial photographs of tanks. These gray scale pictures were processed so that the picture consisted of 240 black or white areas. Black was made to correspond to a binary 1 while white corresponded to a binary 0. The elements were in the form of a 20 by 12 matrix. This lent itself quite naturally to the Markov mesh assumption. The pictures either contained a tank or they did not. The design samples consisted of 50 tanks and 50 non-tanks. In addition there were 100 test samples (50 tanks and 50 non-tanks) which were used to determine how well the discriminant function would discriminate. It was desired to determine how well a third order Markov mesh assumption would perform on the samples. The number of errors of classification of the test samples provides a measure of the "goodness" of the assumption.

Previous Work

Some previous work has been done with the samples.⁷

Assumptions of binary independence, multivariate normal with equal covariance (linear multi-norm), and multivariate normal with unequal covariance matrices (quadratic normal) were used to derive a discriminant function.

A non-parametric procedure was also used to classify the pictures. This non-parametric method was one developed by Fix and Hodges. All design samples were stored in the computer. The unknown test sample is compared with the stored samples. The stored samples that are most similar to the test sample are selected. A classification based on the stored sample most like the test sample, the three closest stored samples, the five closest and the seven closest, was made. In these four comparisons, if any of the selected stored samples was a non-tank, the test sample was classified as a non-tank; otherwise, it was classified as a tank.

The 20 by 12 matrix was divided into submatrices. These submatrices are of five different types as shown in Fig. 5 thru Fig. 9. A discriminant function and

a threshold is determined for each submatrix of a particular format with the use of the design samples. The output of each submatrix (a 0 or 1) is then summed in a second layer. The output of the second layer provides another 0 or 1 which tells us to which population the sample belongs. In our case a 1 meant that the sample came from population 1 (tanks) and a zero that it came from population 2 (non-tanks). In the case of Format 1 (Fig. 5) a two layer network was not possible.

Experimental Work

A third order Markov mesh distribution was assumed for the probability in each population. The same design and test samples as used in the previous work were used. A computer program was written to estimate the values of the coefficients of the discriminant function.

A Philco S-2000 computer was used to perform the calculations. The computer programs were written in Fortran IV language. The programs are available from the author.

The coefficients were estimated by using the design

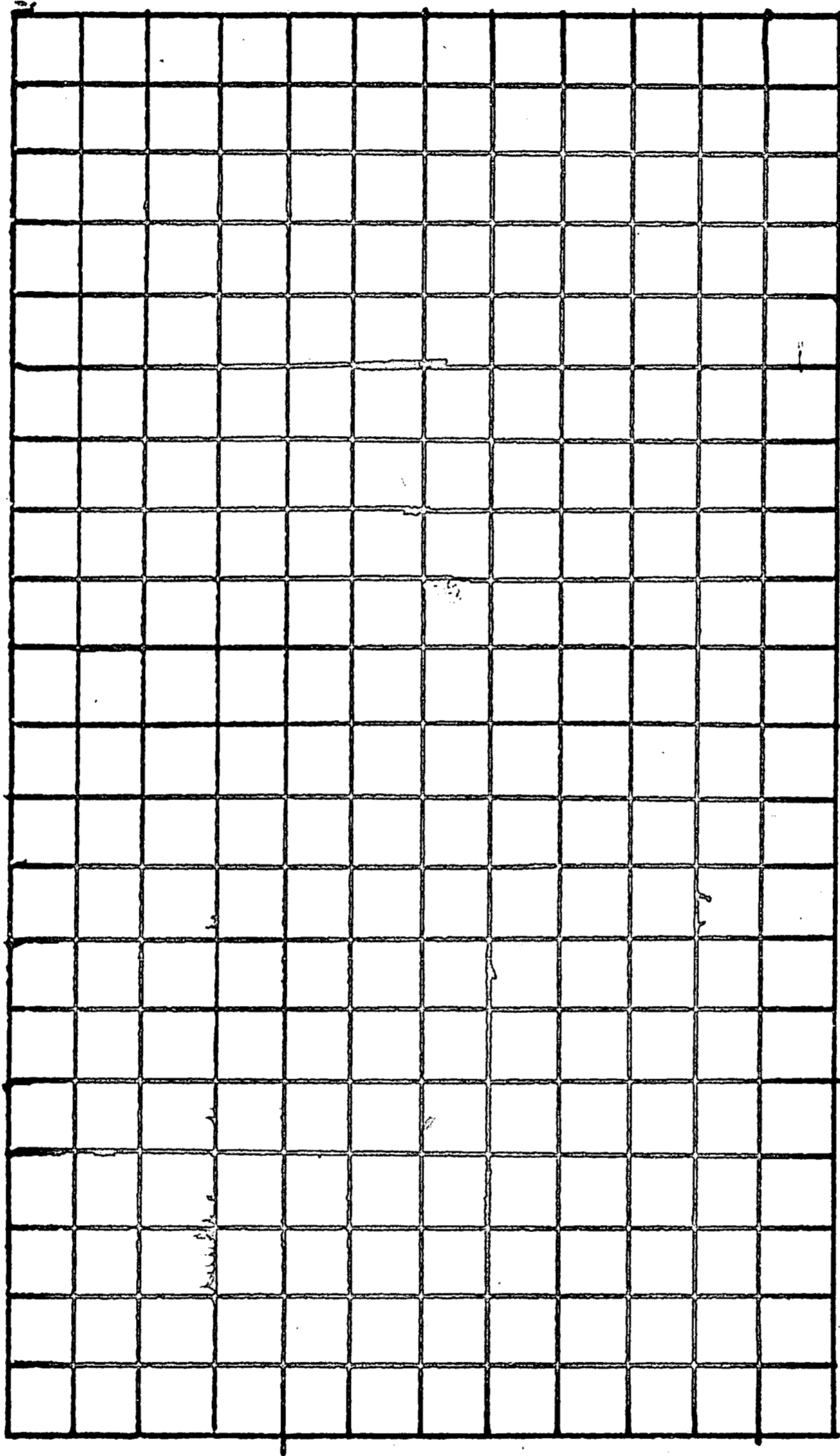
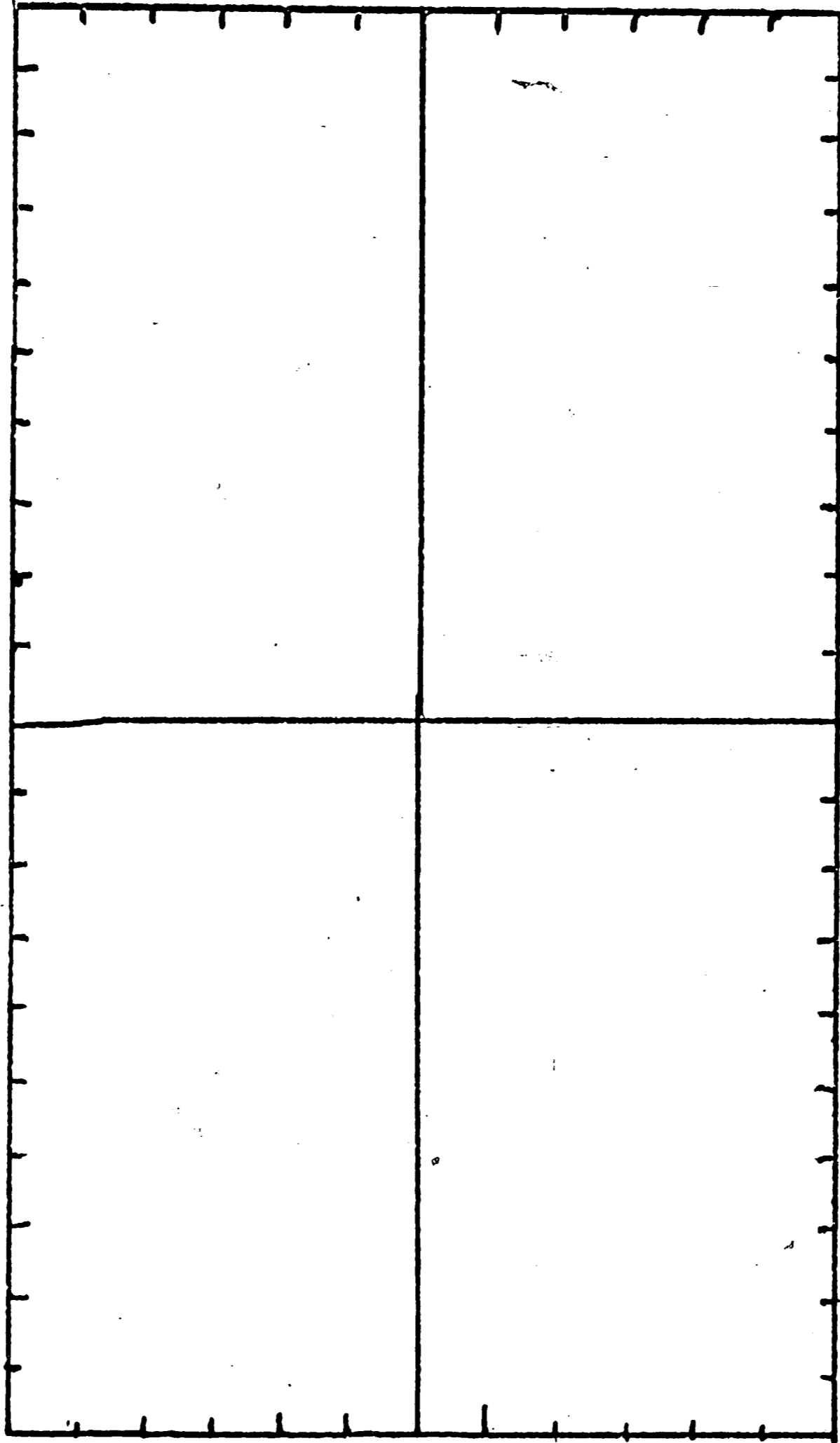
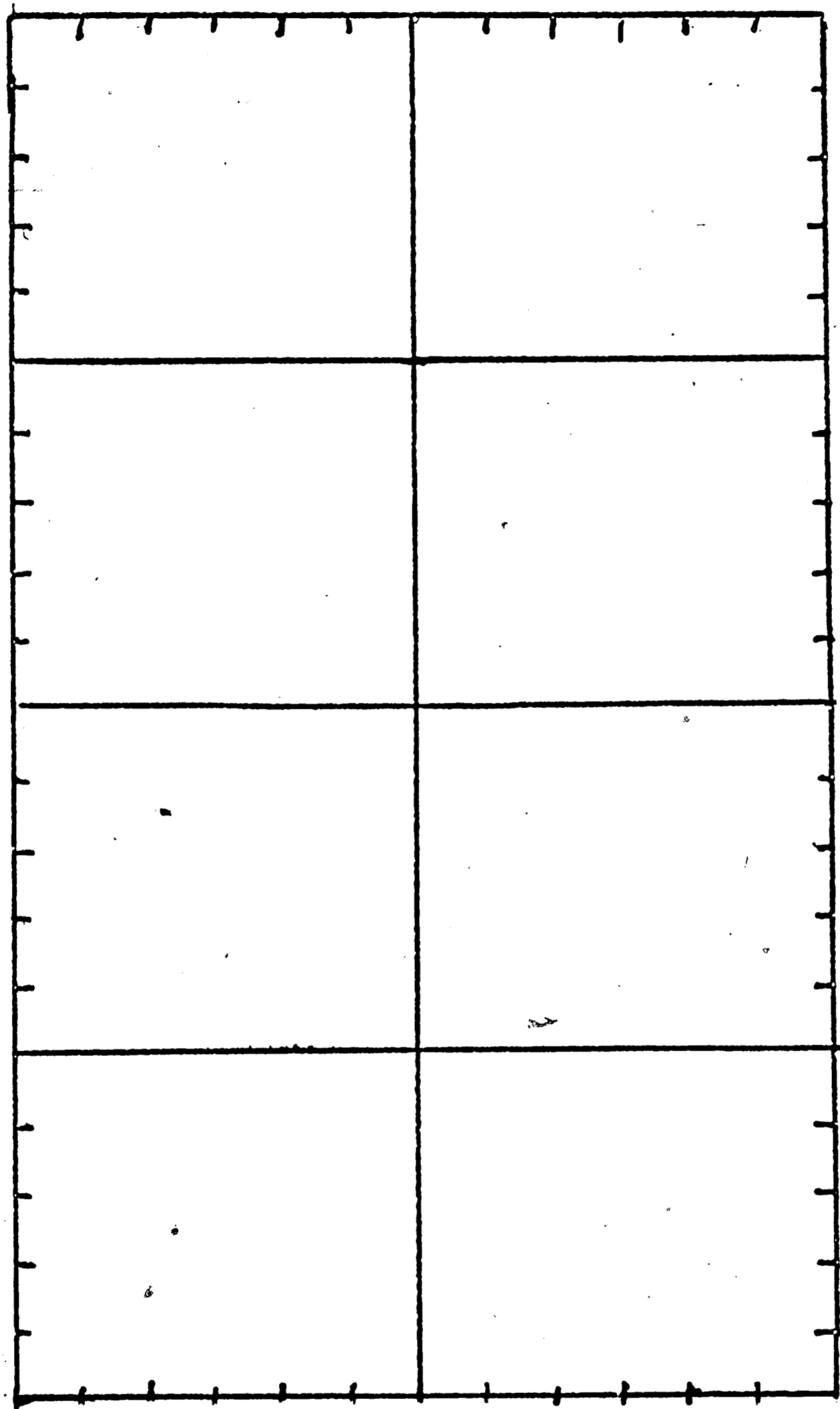


Figure 5. Format 1



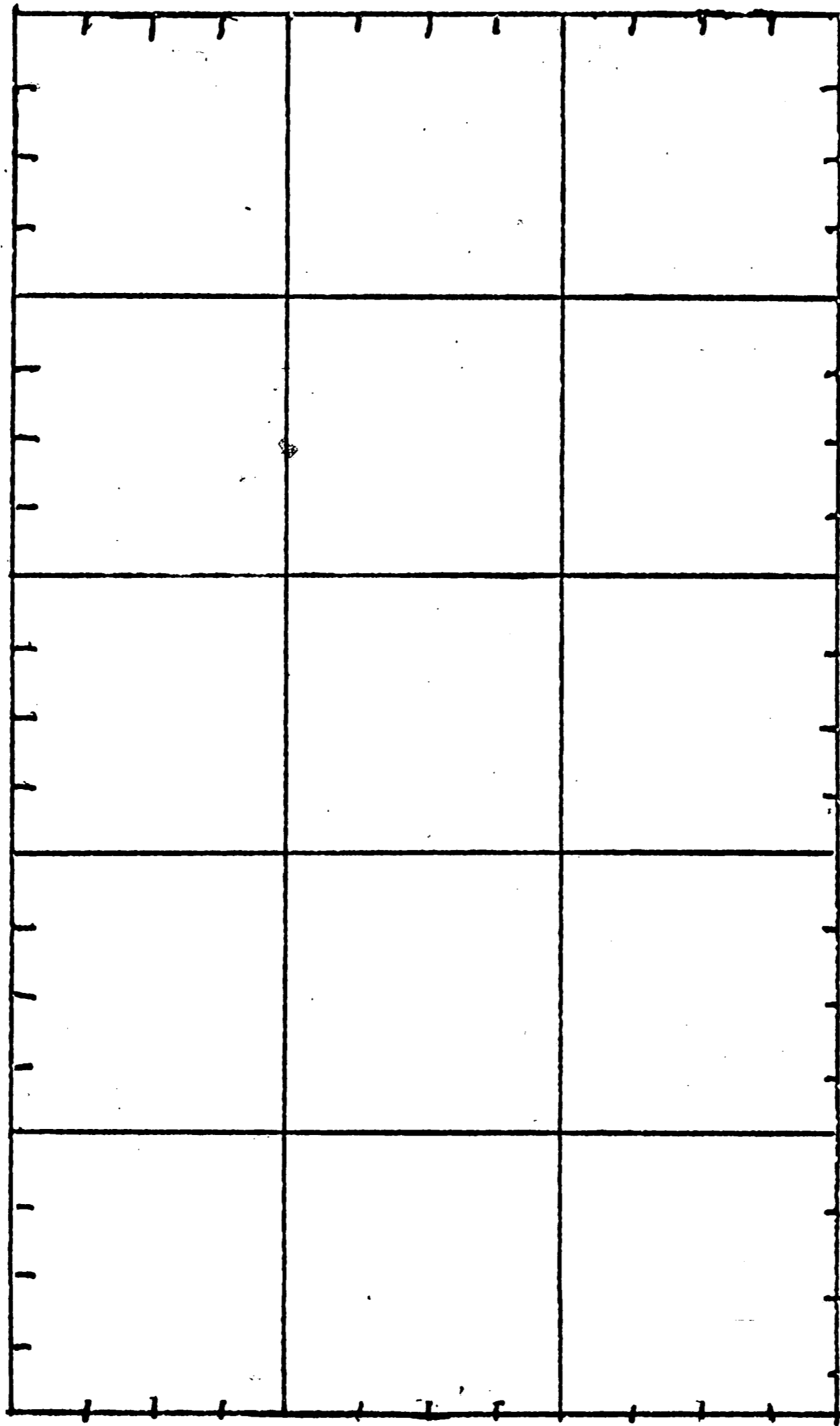
4 6 x 10 blocks

Fig. 6 Format 2



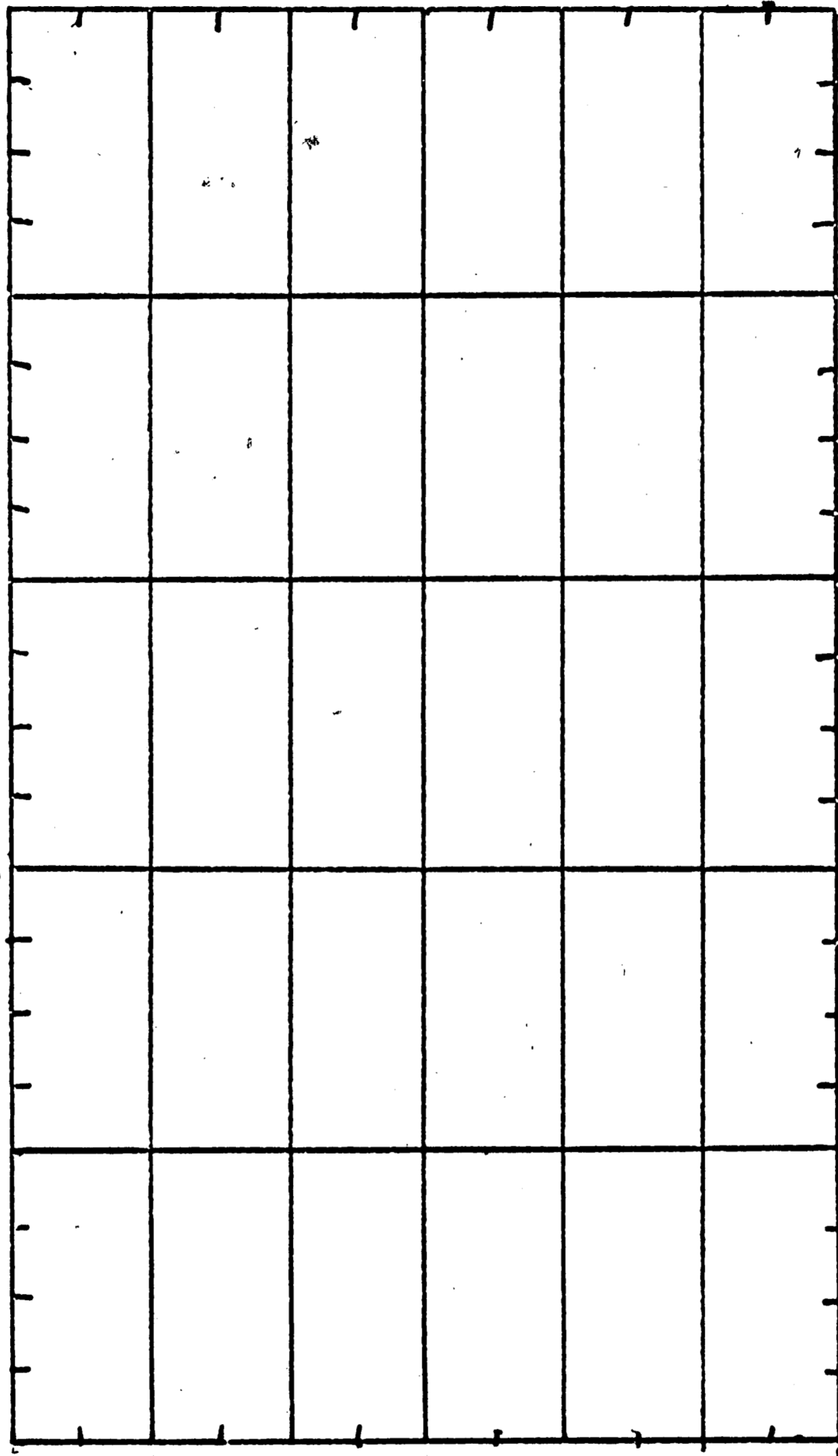
8 5 x 6 blocks

Fig. 7. Format 3



15 4 x 4 blocks

Fig. 8. Format 4



30 2 x 4 blocks

Fig. 9. Format 5

samples to estimate a, b, \dots, h (p.25). The estimate for a is $a_{i,j} = (r+1)/(n+2)$ where r is the number of times that $x_{i,j}$ is 1 when $x_{i,j-1}=0, x_{i-1,j}=0, x_{i-1,j-1}=0$ and n is the number of times that $x_{i,j-1}=0, x_{i-1,j}=0, x_{i-1,j-1}=0$. For large numbers of samples this approximation approaches the true value of $a_{i,j}$. The values of b thru h were estimated in a similar manner. After the program derived these estimates, the coefficients of the discriminant function could be calculated. Once having the coefficients the value of the discriminant function was calculated and printed for both design and test samples. A flow diagram is given in Fig. 10. A sample of the printout is given on page 61.

A threshold was manually chosen as follows. The scores (values of the discriminant function) were examined for the 100 design samples. A threshold was selected so as to give the least number of errors in classification of the design samples. In practice this threshold could be in a certain range. The threshold selected was in the middle of the range. The scores of the test samples were then compared with the threshold and a 1 or 0 output was

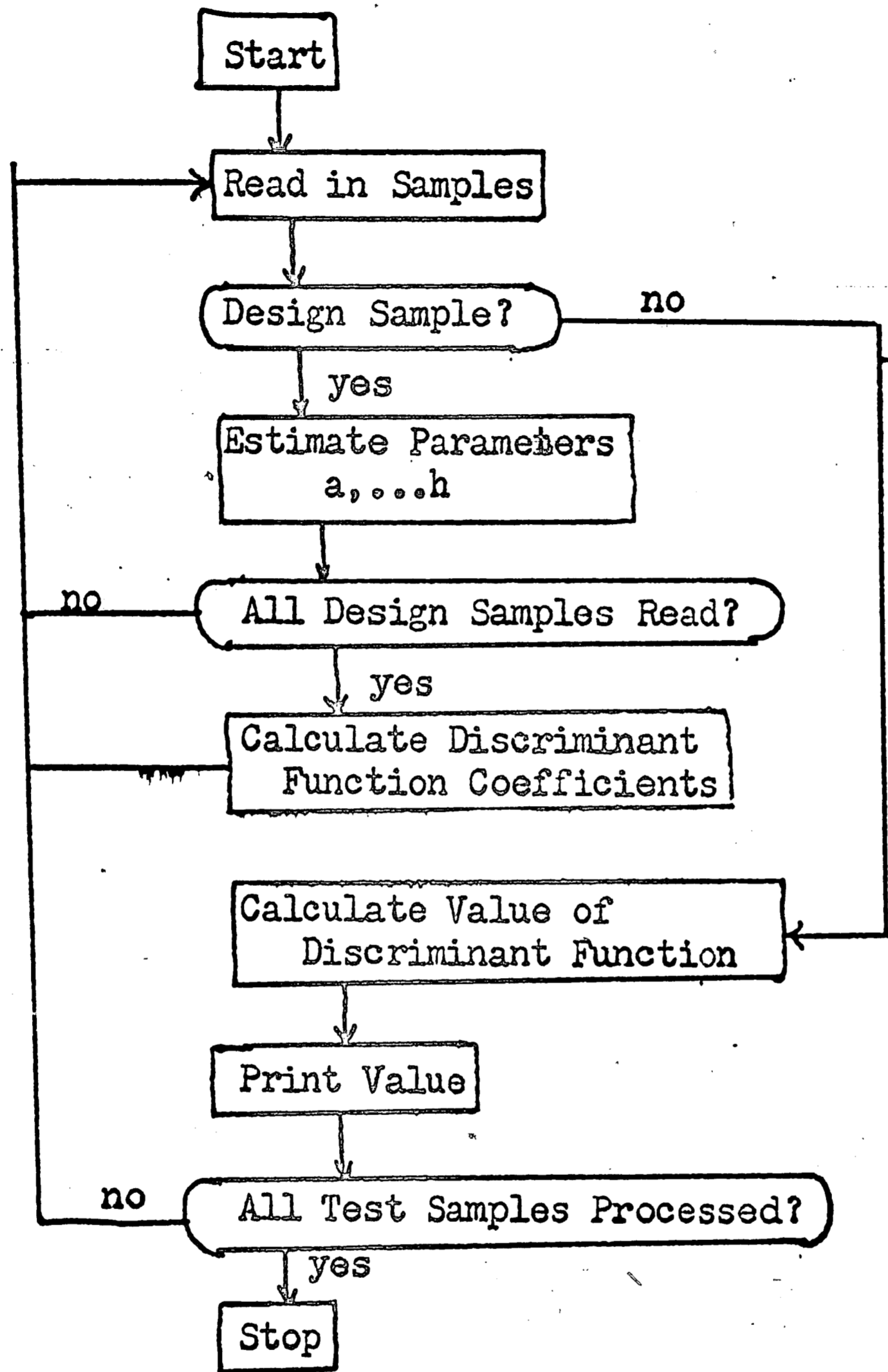


Fig. 10. Flow Diagram for Markov Mesh Program

assigned.

In the second level, the test samples were examined. A second level threshold was selected so as to provide for the least number of errors in classification of the test samples. The total number of test samples classified incorrectly is given in Table I. The results of previous work are also given.

A FORTRAN IV program was also written to evaluate the four level decision process. This involved assuming a linear multivariate normal population for each decision. The output of the first level decisions were made a 1 if the discriminant function is above or equal to the threshold and a zero otherwise. The same procedure is followed for the 2nd and 3rd levels of decision. The program provides a print-out of the value of the discriminant function of the fourth level. If this is positive or zero, the test sample is classified as a tank, otherwise it is classified as a non-tank.

The same 100 design and 100 test samples are used as used in the program for the Markov assumption. A flow diagram of the program is given in Fig. 11. The formulas

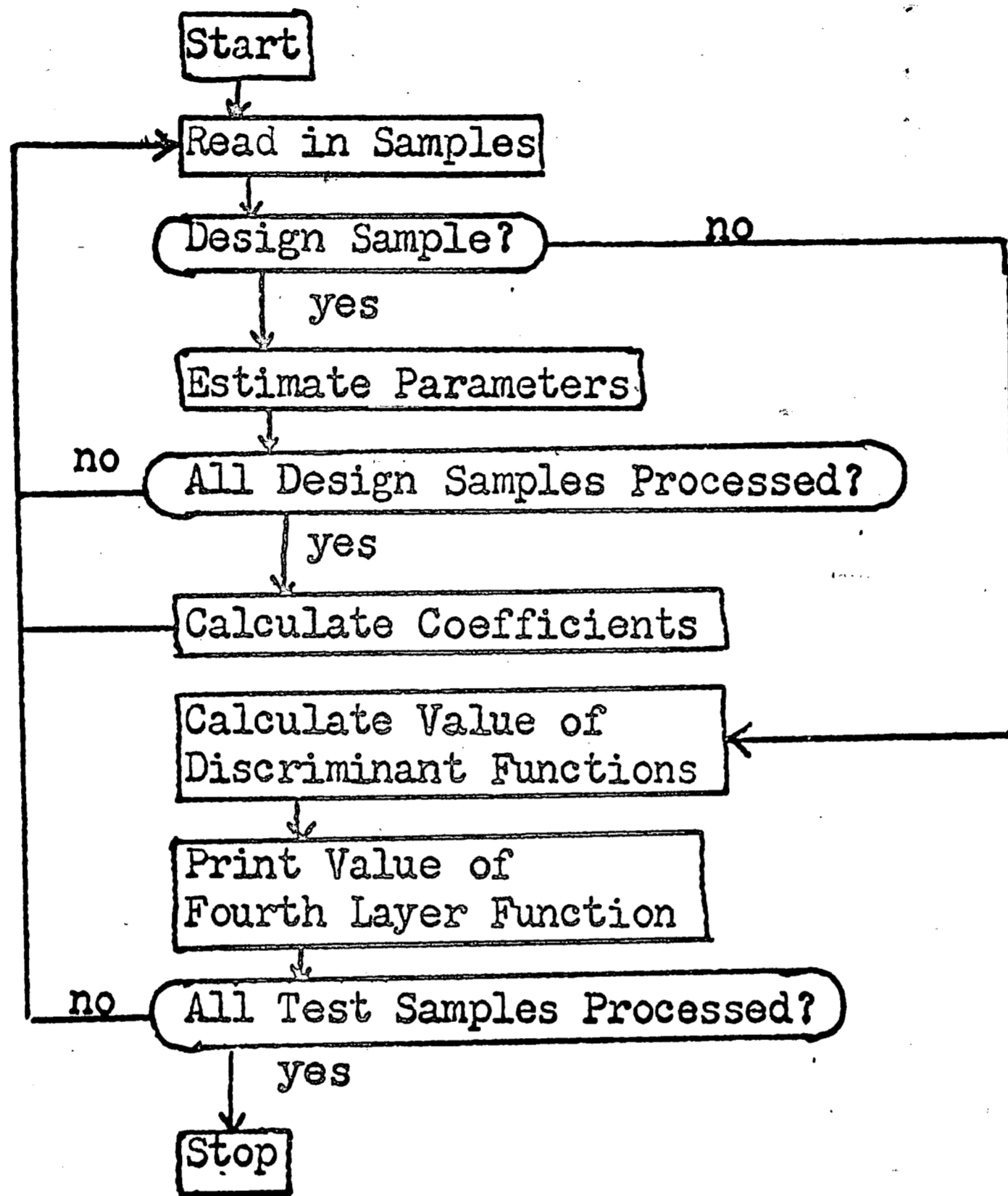


Fig. 11. Flow Diagram for Four Layer Process

used in the calculations of the program are given in the Appendix (p. 58). A sample printout is shown on page 62 .

Results

As can be seen from Table I, the Markov mesh assumption did not produce better results than the other methods. The number of errors, to a certain extent is comparable with the linear multinorm.

The number of errors occurring in the Markov mesh is greater than anticipated. It was thought that the Markov mesh assumption would prove better than the others because the Markov mesh is based on binary variables and takes into account dependency among the variables. A possible explanation of this poorer performance is that the number of parameters that had to be estimated (indicated in Table I) for the discriminant function based on the Markov mesh may be too many for the number of design samples. A good estimate may not have been obtained. This is the sample size problem. In Format 2, 3, and 4, as the number of parameters to be estimated decreased, the number of errors also tended to decrease. The difference in the number of errors between Formats 1 and 2 may

Table I Performance of the Various Methods

Method	Format	Number of Errors	Number of Parameters
Independent Binary	1	16	-
Independent Binary	2	16	-
Independent Binary	3	14	-
Independent Binary	4	11	-
Independent Binary	5	13	-
Multinorm Quadratic	2	9	-
Multinorm Quadratic	3	8	-
Multinorm Quadratic	4	8	-
Multinorm Quadratic	5	10	-
Multinorm Linear	2	14	64
Multinorm Linear	3	11	38
Multinorm Linear	4	5	31
Multinorm Linear	5	9	38
Parametric (1 neighbor)	1	20	-
Parametric (3 neighbors)	1	12	-
Parametric (5 neighbors)	1	10	-

Table I cont.

Method	Format	Number of Errors	Number of Parameters
Parametric (7 neighbors)	1	11	-
Markov Mesh	1	11	1733
Markov Mesh	2	14	389
Markov Mesh	3	14	179
Markov Mesh	4	6	85
Four Level Process	-	31	16

be due to the change from a one level decision process in Format 1 to a two level decision process in Format 2.

The four level decision process provided the fewest number of parameters to be estimated. Thus a small number of design samples should not affect the operation as severely as in the other schemes. Thus the number of errors should be lower. The number of errors of classification was 31 - much higher than expected. This shouldn't be due to the sample size problem since the number of design samples exceeded the number of parameters to be estimated. The large number of errors may be due to the change from a one or two level process to a four level process. As inputs into the linear multivariate normal discriminant function, there were 4, 5, or 3 binary variables. For such a small number of binary variables the normal distribution is probably not a good approximation. We are assuming normality when such an assumption is apparently not warranted. Also with such a few variables in each subset, it is possible that the assumption that the subsets are independent is not warranted.

Chapter 4 Conclusions

A Markov mesh assumption is used in a pattern classification problem and compared with classifications using other probability assumptions. The Markov mesh assumption worked no better than the other methods. This may be due to the lack of sufficient design samples for the Markov mesh. In this experiment one of the more elementary methods (linear multinorm) performed just as well as the complex Markov mesh method.

A four level decision process was implemented in order to avoid the sample size problem. The number of errors of classification were much larger here than expected. This is because the assumption of linear multinorm was not a good assumption and because the assumption that the subsets are independent may not be warranted. The problem of sample size was probably solved but further problems were introduced by a bad assumption about the statistical make-up of the populations.

Some of the results indicate that the results depend somewhat on the number of layers of the decision process. All other things constant, as the number of layers increase, the number of errors also tend to increase.

Appendix

Markov Mesh Discriminant Function

Using the coefficients a, b, ..., h as defined on page 25,
the probability may be written as follows:

$$\begin{aligned}
 P(\lambda_{m,n}) &= \prod_{i=1}^m \prod_{j=1}^n P(x_{ij} / x_{i-1,j-1}, x_{i-1,j}, x_{i,j-1}) \\
 &= a_{ij} (1-x_{i-1,j-1})(1-x_{i-1,j})(1-x_{i,j-1})x_{ij} \\
 &\quad \cdot (1-a_{ij})(1-x_{i-1,j-1})(1-x_{i-1,j})(1-x_{i,j-1})(1-x_{ij}) \\
 &\quad \cdot b_{ij} x_{ij}(1-x_{i-1,j-1})(1-x_{i-1,j})(x_{i,j-1}) \\
 &\quad \cdot (1-b_{ij})(1-x_{ij})(1-x_{i-1,j-1})(1-x_{i-1,j})(x_{i,j-1}) \\
 &\quad \cdot c_{ij} x_{ij}(1-x_{i-1,j-1})(x_{i-1,j})(1-x_{i,j-1}) \\
 &\quad \cdot (1-c_{ij})(1-x_{ij})(1-x_{i-1,j-1})(x_{i-1,j})(1-x_{i,j-1}) \\
 &\quad \cdot d_{ij} x_{ij}(1-x_{i-1,j-1})(x_{i-1,j})(x_{i,j-1}) \\
 &\quad \cdot (1-d_{ij})(1-x_{ij})(1-x_{i-1,j-1})(x_{i-1,j})(x_{i,j-1}) \\
 &\quad \cdot e_{ij} x_{ij}(x_{i-1,j-1})(1-x_{i-1,j})(1-x_{i,j-1}) \\
 &\quad \cdot (1-e_{ij})(1-x_{ij})(x_{i-1,j-1})(1-x_{i-1,j})(1-x_{i,j-1})
 \end{aligned}$$

$$\cdot f_{ij} x_{ij} (x_{i-1,j-1}) (1-x_{i-1,j}) (x_{i,j-1})$$

$$\cdot (1-f_{ij}) (1-x_{ij}) (x_{i-1,j-1}) (1-x_{i-1,j}) (x_{i,j-1})$$

$$\cdot g_{ij} x_{ij} (x_{i-1,j-1}) (x_{i-1,j}) (1-x_{i,j-1})$$

$$\cdot (1-g_{ij}) (1-x_{ij}) (x_{i-1,j-1}) (x_{i-1,j}) (1-x_{i,j-1})$$

$$\cdot h_{ij} x_{ij} x_{i-1,j-1} x_{i-1,j} x_{i,j-1}$$

$$\cdot (1-h_{ij}) (1-x_{ij}) x_{i-1,j-1} x_{i-1,j} x_{i,j-1}$$

Let

$$A_{ij} = \log a_{ij} - \log (1-a_{ij})$$

$$B_{ij} = \log (1-e_{ij}) - \log (1-a_{ij})$$

$$C_{ij} = \log (1-e_{ij}) - \log (1-a_{ij})$$

$$D_{ij} = \log (1-b_{ij}) - \log (1-a_{ij})$$

$$E_{ij} = \log (1-a_{ij}) - \log a_{ij} + \log e_{ij} - \log (1-e_{ij})$$

$$F_{ij} = \log (1-a_{ij}) - \log a_{ij} + \log c_{ij} - \log (1-c_{ij})$$

$$G_{ij} = \log a_{ij} + \log (1-a_{ij}) + \log b_{ij} - \log (1-b_{ij})$$

$$H_{ij} = \log(1-a_{ij}) - \log(1-c_{ij}) + \log(1-g_{ij}) \\ - \log(1-e_{ij})$$

$$L_{ij} = \log(1-a_{ij}) - \log(1-b_{ij}) - \log(1-c_{ij}) \\ + \log(1-d_{ij})$$

$$M_{ij} = \log a_{ij} + \log(1-c_{ij}) - \log(1-a_{ij}) - \log c_{ij} \\ + \log g_{ij} + \log(1-e_{ij}) - \log(1-g_{ij}) - \log e_{ij}$$

$$N_{ij} = \log \frac{(1-b_{ij})(1-c_{ij})d_{ij}}{b_{ij}e_{ij}(1-f_{ij})a_{ij}(1-a_{ij})}$$

$$O_{ij} = \log \frac{(1-b_{ij})(1-c_{ij})d_{ij}}{b_{ij}c_{ij}(1-d_{ij})a_{ij}(1-a_{ij})}$$

$$R_{ij} = \log \frac{a_{ij}(1-a_{ij})b_{ij}c_{ij}(1-d_{ij})e_{ij}(1-f_{ij})(1-g_{ij})h_{ij}}{(1-b_{ij})(1-c_{ij})d_{ij}(1-e_{ij})f_{ij}g_{ij}(1-h_{ij})}$$

Taking the log of $P(\mathcal{X}_{m,n})$, we get

$$\sum_{i=1}^m \sum_{j=1}^n A_{ij} x_{ij} + \sum_{i=2}^m \sum_{j=2}^n B_{ij} x_{i-1,j-1}$$

$$\sum_{i=2}^m \sum_{j=1}^n C_{ij} x_{i-1,j} + \sum_{i=1}^m \sum_{j=2}^n D_{ij} x_{i,j-1}$$

$$\sum_{i=2}^m \sum_{j=2}^n E_{ij} x_{ij} x_{i-1,j-1} + \sum_{i=2}^m \sum_{j=1}^n F_{ij} x_{ij} x_{i-1,j}$$

$$\begin{aligned}
& + \sum_{i=1}^m \sum_{j=2}^n G_{ij} x_{ij} x_{i,j-1} + \sum_{i=1}^m \sum_{j=2}^n H_{ij} x_{i-1,j-1} x_{i-1,j} \\
& + \sum_{i=2}^m \sum_{j=2}^n K_{ij} x_{i-1,j-1} x_{i,j-1} + \sum_{i=2}^m \sum_{j=2}^n L_{ij} x_{i,j-1} x_{i-1,j} \\
& + \sum_{i=2}^m \sum_{j=2}^n M_{ij} x_{ij} x_{i-1,j} x_{i-1,j-1} \\
& + \sum_{i=2}^m \sum_{j=2}^n N_{ij} x_{ij} x_{i-1,j-1} x_{i,j-1} \\
& + \sum_{i=2}^m \sum_{j=2}^n O_{ij} x_{ij} x_{i-1,j} x_{i,j-1} \\
& + \sum_{i=2}^m \sum_{j=2}^n Q_{ij} x_{i-1,j-1} x_{i-1,j} x_{i,j-1} \\
& + \sum_{i=2}^m \sum_{j=2}^n R_{ij} x_{ij} x_{i-1,j-1} x_{i,j-1} x_{i-1,j}
\end{aligned}$$

where

$$Q_{ij} = \log \frac{(1-b_{ij})(1-c_{ij})(1-e_{ij})(1-h_{ij})}{(1-a_{ij})(1-d_{ij})(1-f_{ij})(1-g_{ij})}$$

If we use a superscript to distinguish the coefficients from the different populations, we obtain identical equations for the log of the probability in the two populations except for the coefficients. Subtracting the two equations, we

obtain the discriminant function. It is

$$\sum_{i=1}^m \sum_{j=1}^n (A_{ij}^{(1)} - A_{ij}^{(2)}) x_{ij} + \dots$$

$$\sum_{i=2}^m \sum_{j=2}^n (R_{ij}^{(1)} - R_{ij}^{(2)}) x_{ij} x_{i-1,j-1} x_{i,j-1} x_{i-1,j}$$

Linear Normal Discriminant Function with Estimated Parameters

The discriminant function is $X'V^{-1}(M_1 - M_2)$ and the threshold is $\frac{1}{2}(M_1 + M_2)'V^{-1}(M_1 - M_2)$. This assumes that $t=1$. M and V are defined on page 12.

M and V are not known but must be estimated. Assume that there are R design samples from population 1 and S from population 2. Each design sample gives us the variable X ($X^{(1)}$ if from G_1 , or $X^{(2)}$ if from G_2), $X = (x_1, x_2, \dots, x_n)$. The k -th sample from G_1 denote as $X_k^{(1)} = (x_{1k}^{(1)}, \dots, x_{nk}^{(1)})$ and likewise for G_2 . Let $M_1 = (m_1^{(1)}, \dots, m_n^{(1)})$ and $M_2 = (m_1^{(2)}, \dots, m_n^{(2)})$. Approximate $m_i^{(1)}$ by

$$(1/R) \sum_{k=1}^R x_{ik}^{(1)}$$

and $m_i^{(2)}$ by $(1/S) \sum_{k=1}^S x_{ik}^{(2)}$

Also approximate v_{ij} by s_{ij} , where

$$s_{ij} = \left[\sum_{k=1}^R (x_{ik}^{(1)} - m_i^{(1)})(x_{jk}^{(1)} - m_j^{(1)}) + \sum_{k=1}^S (x_{ik}^{(2)} - m_i^{(2)})(x_{jk}^{(2)} - m_j^{(2)}) \right] / (R+S-2)$$

These are the formulas used in the computer program to evaluate the four level decision process.

Derivation of Number of Inputs for Optimum

Multilevel Decision Process

As given on page 34 for optimality we need to minimize

$$\sum_{i=1}^n N_i \text{ given that } \prod_{i=1}^n N_i = K. \text{ For optimality we expect to}$$

have the same number of inputs to all levels. That is,

$N_i = N_j = N$. Then

$$N^n = K$$

$$N = K^{1/n}$$

We want to minimize nN

$$N = K^{1/n}$$

$$\ln N = (1/n) \ln K$$

$$n = (\ln K) / \ln N$$

We then want to minimize

$$(\ln K) N / (\ln N)$$

$$N / \ln N = \text{constant}$$

Differentiating

$$(1 / \ln N)^2 ((1 / \ln N) - 1) = 0$$

$$(1 / \ln N) - 1 = 0$$

$$\ln N = 1$$

$$N = e$$

$$n = \ln K \quad (K = \text{the number of variables} \\ \text{in a sample})$$

SAMPLE DISCRIMINANT FUNCTION

1	1	0.81273+001
1	2	0.10290+002
1	3	0.22996+002
1	4	0.35186+001
1	5	0.20263+002
1	6	0.93743+000
1	7	0.33840+001
1	8	0.10695+002
11001110000011100111011001110000111001100000111001000100011001000000011001011110		
11100100111001100110010011100100110010100101110001101100110001100100010001100100		
01000100010110010100010100000110011010100110011011101110011001100110000000000000		
2	1	0.76197+001
2	2	0.31993+001
2	3	0.43537+001
2	4	0.28532+001
2	5	0.44727+001
2	6	0.16658+001
2	7	0.43164+001
2	8	0.84085+001
10000100000011001100110011001100100010001101000110001111000111001000100011001000100		
010001000000011001100100011001100110011001100100011001100100101100100101100100100101		
000100110100000000100101010001100101111101100110011101100110011001100000000000111		
3	1	0.10754+002
3	2	0.58813+001
3	3	0.12970+002
3	4	0.41555+001
3	5	0.15244+002
3	6	0.15880+001
3	7	0.52589+001
3	8	0.97355+001
00000000110001000100000011100100011001110000111001100000111001110000111001110100		
01100100000001100100010001101101111001101100110100101100010000101101110000101101		
11100010110001100010110001100000110100100101110010100100110011101110010001100111		
4	1	0.60120+001
4	2	0.21732+001
4	3	0.90510+001
4	4	0.74947+001
4	5	0.50275+001
4	6	0.42718+001
4	7	0.37320+001
4	8	0.79888+001
01111101110101000010000011110111011110110111000110000100000111000111010101100		
11100010100011100010000000100110011100100110011100101010000000101010010000101010		
11000010001001000110001101001110000101000010001100110010001100100001001101101001		
5	1	0.10178+002
5	2	0.60603+001
5	3	0.19243+002
5	4	0.33024+000

Sample output of Markov mesh program (Format 3)

FOURTH LEVEL COEFF AND THRESHOLD

0,27862+001 0,20515+001 =0,16295+001 0,16640+001
01110000001101110000101101110000001101100000001101100010001101101000011101101110
00110100011100110100001000110100010000110100110010110100011100100100011000100110
01100011011011000110011010010010011001100110011001100110001001100110000000000000000
SCORE

0,31538+001 1
00000100000011001110011011001000111011011000111011000000111011001100011011001100
01101100000001101110011001101100011000101100011000101100111100101101100100101101
00010011100001000110100111100110110111101101110111101101110011001100000000000101
SCORE

0,15243+001 2
0000000000000110011001100110001011100111000011000110010011100011001100110011001101000
0110011011110110110001110110110001000010010010000010010010010010010000000000000100100
011001100010111001100110110111100110100100100111011001100110011101101101100000000100
SCORE

0,15243+001 3
110000000000000011000100111011101100111000011100111000011100110000111001100100011001100110
01100110111001100110111101101110010001100100010000100110101100100100100000100110
01000110001011100110001011010110001000010110001011010110001101110110011000100010
SCORE

=0,12420+001 4
000000000000001100010000001110000011000111000111000110000111001111000111001100000
01100110000001100100111100100100011100100100000000100100011000100100111000100110
01100010011000110000011110000000011010100110011000100110001000000110001100100010
SCORE

0,11023+001 5
011111111001111011110011110001110011100000110011001100110011001111110011001101110
001101100100001101101100001101101110001100101111001100100111100110010011100110011
00110111001110001111001100000111001100010111001100111011001110110011001000011001
SCORE

0,38752+000 6
000011001100110011011100111000011100011000001100011001001110001100100111001111110
01100001111001101100110001101100110000100100111000100100111000100100111000100100
01100110001100111110001101000010001101110111001100110111001100110011000000000000
SCORE

0,31538+001 7
010011100110011001111110011100001110011000001110011001101110011001100110011001101110
0110010111100110011011001100100001000100100111000100100111100100110000100100010
10110010001011011010001011000010001101110011001101110011101100111011001100110011000000100
SCORE

0,38752+000 8
11000000001001000110011001110110111001110000111001111000011001111000011001101110
011001101111011001100110001001100100001001101100001001001110000100010011000100010
01100010001001100110001101000110001101110110001101110111001100110011100100100000
SCORE

0,11023+001 9
1111110110001011111000011110011100001001001110001111000111001100000110001000000
111101000000111101000110011101000010011101000110010000100011001100100110101110110

Sample output of four level process program

References

- ¹Abend, K., Harley, T. J., and Kanal, L. N. "Classification of Binary Random Variables," IEEE Transactions on Information Theory, Vol. IT-11 (1965), pp. 538-544
- ²Nilsson, N. J., Learning Machines, New York: McGraw-Hill, Inc., 1965
- ³Abend, K. and Kanal, L. N. "Adaptive Modelling of Likelihood Classification", Report No. RADC-TR-66-190 prepared by Philco-Ford Corporation under Air Force contract No. AF 30(602)-3623, Blue Bell, Pa.: Philco-Ford Corporation, 1966
- ⁴Chow, C. K. and Liu, C. N. "Approximating Discrete Distributions with Dependence Tree," unpublished paper, IBM Watson Research Center, Yorktown Heights, New York, 1967
- ⁵Chow, C. K. and Liu, C. N. "An Approach to Structure Adaptation in Pattern Recognition", IEEE Transactions on Systems Science and Cybernetics, Vol. SSC-2 (1966), pp. 73-80

6 Freeman, H. Introduction to Statistical Inference

Reading, Mass: Addison-Wesley Publishing Company,
Inc., 1963

7 Richards, Jerry R. "Comparison of Statistical Techniques
for the Classification of Complex Targets in
Photographic Data," Unpublished Master's Thesis,
University of Pennsylvania, 1966

Vita

- Born - Nov. 14, 1944
Lewisburg, Pa.
- Parents - Geraldine May Howell Womer
and William Paul Womer
- 1962-1966 - Undergraduate student at Lehigh
University, Bethlehem, Pa.
- June 13, 1966 - Bachelor of Scienc degree in
Electrical Engineering from Lehigh
University, Bethlehem, Pa.
Degree received with "high honors".
- 1966- - Graduate Student at Lehigh
University, Bethlehem, Pa.