U. S. NAVAL
PERSONNEL RESEARCH ACTIVITY
SAN DIEGO, CALIFORNIA 92152

TECHNICAL BULLETIN 65-15 MAY 1965

A COMPUTER PROGRAM FOR THE MAXIMUM LIKELIHOOD ANALYSIS OF TYPES

John H. Wolfe

NPRA
AN ACTIVITY OF THE BUREAU OF NAVAL PERSONNEL

AD-620-026

U. S. NAVAL PERSONNEL RESEARCH ACTIVITY

SAN DIEGO, CALIFORNIA 92152

TECHNICAL BULLETIN 65-15

MAY 1965

A COMPUTER PROGRAM FOR THE MAXIMUM LIKELIHOOD ANALYSIS OF TYPES

John H. Wolfe

NPRA

SAN DIEGO CALIFORNIA

AN ACTIVITY OF THE BUREAU OF NAVAL PERSONNEL

A COMPUTER PROGRAM FOR THE MAXIMUM LIKELIHOOD ANALYSIS OF TYPES

JOHN H. WOLFE

U.S. NAVAL PERSONNEL RESEARCH ACTIVITY
SAN DIEGO, CALIFORNIA 92152

Submitted by

W. J. Moonan, Ph.D., Director, Statistical Department

Approved by

E. E. Dudek, Ph.D., Technical Director

G. W. Watson, Commander, USN
Officer in Charge

U.S. Naval Personnel Research Activity
San Diego, California 92152

BRIEF

This report contains a description of a computer program for estimating the parameters of a mixture of multivariate normal distributions with unknown frequencies, means, and covariances. The basic equations for the procedure are presented for the first time here, with their derivation omitted. An example with the results of the computer printout is described for an artificially constructed mixture of three bivariate normal distributions. The method of using the program and the Fortran listing are detailed in this report.

CONTENTS

A COMPUTER PROGRAM FOR THE
MAXIMUM LIKELIHOOD ANALYSIS OF TYPES

I Identification

A. TYPE

B. Written by John H. Wolfe, January 1963. Revised June, 1964.

C. U.S. Naval Personnel Research Activity, San Diego, California.

D. Coded entriely in FORTRAN II.

II Purpose

Given m scores on each of N individuals drawn from an unknown mixture of multivariate normal distributions, the program gives maximum-likelihood estimates of the means and covariances within each type, the relative frequency of each type, and the maximum likelihood of the sample for a given number of types.

III Restrictions

A. The program uses $14{,}554_8$ words in the main program and $47{,}256_8$ words in COMMON. One input magnetic tape (Unit #3), one output magnetic tape (Unit #2) and one tape for temporary binary storage (Unit #4) are required.

B. Restrictions on Parameters

Numbers of Variables $\leq$ 5

Number of Types $\leq$ 6

IV Method

The program solves the maximum-likelihood equations by one of four alternative iteration schemes, depending on a control card option.

Suppose that m measurements have been made on N individuals. Let X_{ik} be the i^{th} variable for individual k. Suppose the population from which the sample of N individuals is drawn is a mixture of r multivariate normal distributions. That is, the probability density, $f(\bar{x})$, is given by

$$f(x) = \sum_{s=1}^{r} \lambda_s \alpha_s (x) \quad , \text{ where } \sum_{s=1}^{r} \lambda_s = 1, \lambda_s \geq 0, \text{ and}$$

$$\alpha_s (x) = \left(\frac{|\sigma_s^{ij}|}{(2\pi)^m} \right)^{1/2} e^{-1/2 \sum_{ij} (x_i - M_i^s) \sigma_s^{ij} (x_j - M_j^s)}$$

Here λ_s is the relative proportion of type s in population, $|\sigma_s^{ij}|$ is the determinant of the inverse of the covariance matrix for type s, and M_i^s is the mean of the i^{th} variable for type s.

Let us define $g_s(x) = \alpha_s(x) / f(x)$. SUBROUTINE DENSITY calculates α_s, f, and g_s for each individual. Define the "generalized sample moments" $\{\mu^{ps}_{ijab}\}$ as follows:

$$\mu^{ps}_{ijab} = \frac{1}{N} \sum_{k=1}^{N} x_{ik} x_{jk} x_{ak} x_{bk} \; g_s(x_k) \; g_p(x_k)$$

If a subscript is omitted or set to o, the corresponding term on the right hand side of the equation is omitted. For example

$$\mu^{po}_{ijao} = \mu^{p}_{ija} = \frac{1}{N} \sum_{k=1}^{N} x_{ik} x_{jk} x_{ak} \; g_p(x_k)$$

and $\mu^o_o = \frac{1}{N} \sum_{k=1}^{N} 1 = 1$.

SUBROUTINE MOMENT computes a table of $\{\mu^{ps}_{ijab}\}$ with $b \leq a \leq j \leq i$ and $p \leq s$.

FUNCTION U (II, IJ, IA, IB, IP, IS) looks up the correct μ^{ps}_{ijab} from the two-dimensional table created by MOMENT, even when the inequalities on the indices are not satisfied.

The maximum likelihood estimates of the parameters are those values which maximize the function

$$L = \sum_{k=1}^{N} \log f(x_k) - \omega\left(\sum_{s=1}^{r} \lambda_s - 1\right) .$$

Setting the partial derivatives of the likelihood to zero results in the following equations:

$$f^{\,s}_{oo} = \mu^{\,s}_{o} - 1 = o$$

$$f^{\,s}_{oi} = \mu^{\,s}_{i} - \mu^{\,s}_{o} M^s_i = o$$

$$f^{\,s}_{ij} = \mu^{\,s}_{ij} - M^s_i \mu^s_j - M^s_j \mu^s_i - \mu^{\,s}_{o} (\sigma^{\,s}_{ij} - M^s_i M^s_j) = o$$

The moments $\{\mu^{\,s}_{ij}\}$ can also be differentiated as follows:

$$\frac{\partial \mu^s_{ij}}{\partial \lambda_p} = -\mu^{ps}_{ij}$$

$$\frac{\partial \mu^s_{ij}}{\partial M^p_a} = -\lambda_p \sum_{b=1}^{m} \sigma^{ab}_p (\mu^{ps}_{ijb} - M^p_b \mu^{ps}_{ij}) + \delta_{ps} \sum_{b=1}^{m} \sigma^{ab}_p (\mu^p_{ijb} - M^p_b \mu^p_{ij})$$

$$\frac{\partial \mu^s_{ij}}{\partial \sigma^{ab}_p} = (1 - \frac{\delta^{ab}}{2})\{-\lambda_p [\mu^{ps}_{ij}(\sigma^p_{ab} - M^p_a M^p_b) + M^p_a \mu^{ps}_{ijb} + M^p_b \mu^{ps}_{ija} - \mu^{ps}_{ijab}]$$

$$+\delta_{ps} [\mu^p_{ij}(\sigma^p_{ab} - M^p_a M^p_b) + M^p_a \mu^p_{ijb} + M^p_b \mu^p_{ija} - \mu^p_{ijab}]\}$$

Where δ_{ps} = 1 if p=s
0 otherwise

The derivatives of the moments are computed by

FUNCTION DERV (II, IJ, IS, IA, IB, IP) = $\frac{\partial \mu_{ij}}{\partial \theta^p_{ab}}$,

Where $\theta^p_{oa} = M^p_a$, $\theta^p_{oo} = \lambda_p$, and $\theta^p_{ab} = \sigma^{ab}_p$ for a≠o≠b.

DERV calls on three functions, ONE, EM, SIG defined as follows:

ONE (IA, IB) = δ_{ab}

EM (II, IJ, IA, IS, IP) = $\sum_{b=1}^{m} \sigma^{ab}_p (\mu^{ps}_{ijb} - M^p_b \mu^{ps}_{ij})$

$$\text{SIG (II, IJ, IA, IB, IS, IP)} = \mu_{ij}^{ps} (\sigma_{ab}^{p} - M_a^p M_b^p) + M_a^p \mu_{ijb}^{ps}$$

$$+ M_b^p \mu_{ija}^{ps} - \mu_{ijab}^{ps}$$

The function EM used with s=0 gives the second term in $\dfrac{\partial \mu_{ij}^{s}}{\partial M_a^{p}}$

and SIG used with s=0 gives the second term in $\dfrac{\partial \mu_{ij}^{s}}{\partial \sigma_p^{ab}}$.

The maximum likelihood equations $\{f_{ij}^{s} = 0\}$ can be solved in several ways. One iterative scheme is
Newton-Raphson iteration, which solves the linear equations

$$\sum_{p=1}^{r} \sum_{b=1}^{m} \sum_{a=b}^{m} \frac{\partial f_{ij}^{s}}{\partial \theta_{ab}^{p}} \qquad \Delta\theta_{ab}^{p} = -f_{ij}^{s}$$

for $\Delta\theta_{ab}^{p}$.

On the next iteration $\theta_{ab}^{p\prime} = \theta_{ab}^{p} + \Delta\theta_{ab}^{p}$.

SUBROUTINE NEWTON computes the vector

$B(IAT) = -f_{ij}^{s}$ and the matrix of coefficients

$A(IAT, JAT) = \partial f_{ij}^{s} / \partial \theta_{ab}^{p}$.

SUBROUTINE MATINV, a standard SHARE routine, solves the linear equations for $\Delta\theta_{ab}^{p}$ and stores the result in the vector B (IAT).

SUBROUTINE RAPHSON computes the values of the parameters for the next iteration. First it determines if any of the increments are so large that $\lambda_p + \Delta\lambda_p < o$ or >1. If so, the increment vector B is shortened until no $\Delta\lambda_p$ moves λ_p more than half the interval between λ_p and a boundary point.

That is, if $\Delta\lambda_p < o$, $\Delta\lambda_p > -\dfrac{\lambda_p}{2}$, and if $\Delta\lambda_p > o$, $\Delta\lambda_p < \dfrac{1-\lambda_p}{2}$. After the $\Delta\theta_{ab}^{p}$

are shortened, they are added to the old estimates of the parameters to obtain new estimates for the next iterations.

The main routine also plays a role in shortening the increment vector. If the new likelihood is less than the previous one, or if the determinant of one of the covariance matrices as determined by RAPHSON is negative, then the increment vector B is shortened to half its previous value and subroutine RAPHSON is entered again.

Several alternative versions of $\{f_{ij}^{s}\}$ can be written.

At the maximum-likelihood points

$$f_{oo}^{s} = \mu_{o}^{s} - 1 = 0, \text{ hence } \mu_{o}^{s} = 1.$$

$$f_{oi}^{s} = \mu_{i}^{s} - \mu_{o}^{s} M_{i}^{s} = \mu_{i}^{s} - M_{i}^{s} = 0, \text{ hence } \mu_{i}^{s} = M_{i}^{s} .$$

Substituting for μ_{i}^{s} and μ_{o}^{s} in f_{ij}^{s}, we have

$$f_{ij}^{s} = \mu_{ij}^{s} - M_{i}^{s} M_{j}^{s} .$$

When METH = 1 on the control card, the subroutine NEWTON iteratively solves the equations

$$_{1}f_{oo}^{s} = \mu_{o}^{s} - 1 = 0 .$$

$$_{1}f_{oi}^{s} = \mu_{i}^{s} - M_{i}^{s} = 0.$$

$$_{1}f_{ij}^{s} = \mu_{ij}^{s} - M_{i}^{s} M_{j}^{s} = 0 .$$

When METH = 2, subroutine NEWTON solves the original set of equations, hereafter referred to as $\{_{2}f_{ij}^{s} = 0\}$

When METH = 3, the equations used are

$$_{3}f_{oc}^{s} = 1/\mu_{o}^{s} \, (_{2}f_{oo}^{s}) = 1 - 1/\mu_{o}^{s} = 0.$$

$$_{3}f_{oi}^{s} = 1/\mu_{o}^{s} \, (_{2}f_{oi}^{s}) = \mu_{i}^{s}/\mu_{o}^{s} - M_{i}^{s} = 0.$$

$$_{3}f_{ij}^{s} = 1/\mu_{o}^{s} \, (_{2}f_{ij}^{s}) = \mu_{ij}^{s} - M_{i}^{s} \, \mu_{j}^{s}/\mu_{o}^{s} - M_{j}^{s} \, \mu_{i}^{s}/\mu_{o}^{s} - (\sigma_{ij}^{s} - M_{i}^{s} M_{j}^{s}) = 0.$$

All three types of equations have the same solutions but their radii of convergence may differ.

For METH = 2,

the function $\{{}_2f_{ij}^{s}\}$ can be summarized in the following formula:

$$ {}_2f_{ij}^{s} = \mu_{ij}^{s} - \delta_{io} - (1 - \delta_{io})\{\theta_i^s\mu_j^s + (1 - \delta_{jo})[\mu_o^s(\sigma_{ij}^{s} - \theta_i^s\,\theta_j^s) + \theta_j^s\mu_i^s\,]\}\,. $$

The partial derivatives of f_{ij}^{s} are then easily written as

$$ \frac{\partial\, {}_2f_{ij}^{s}}{\partial\,\theta_{ab}^{p}} = \frac{\partial\,\mu_{ij}^{s}}{\partial\,\theta_{ab}^{p}} - (1 - \delta_{io})\{\theta_i^s\,\frac{\partial\,\mu_j^s}{\partial\,\theta_{ab}^{p}} + (1 - \delta_{jo})[\frac{\partial\,\mu_o^s}{\partial\,\theta_{ab}^{p}}\,(\sigma_{ij}^{s} - \theta_i^s\theta_j^s) + \theta_j^s\,\frac{\partial\,\mu_i^s}{\partial\,\theta_{ab}^{p}}]\} $$

$$ -\delta_{ps}(1 - \delta_{io})\{\delta_{bo}[\delta_{ia}\mu_j^s + (1 - \delta_{jo})(\delta_{ja}\mu_i^s - \delta_{ia}\mu_o^s\theta_j^s - \delta_{ja}\mu_o^s\theta_i^s)] $$

$$ - (1 - \frac{\delta_{ab}}{2})(1 - \delta_{bo})(1 - \delta_{jo})\;\mu_o^s\;(\sigma_{ia}^{p}\sigma_{jb}^{p} + \sigma_{ib}^{p}\,\sigma_{ja}^{p})\} $$

where $o \le j \le i \le m$ and $o \le b \le a \le m$.

The above formulas are used in METH = 2. If METH = 1, the functions

$$ {}_1f_{oo}^{s} = \mu_o^s - 1 $$

$$ {}_1f_{oi}^{s} = \mu_i^s - M_i^s $$

$$ {}_1f_{ij}^{s} = \mu_{ij}^{s} - (\sigma_{ij}^{s} + M_i^sM_j^s) $$

are summarized by the formulas

$$ {}_1f_{ij}^{s} = \mu_{ij}^{s} - \delta_{io} - (1 - \delta_{io})\{\delta_{jo}\theta_i^s + (1 - \delta_{jo})(\sigma_{ij}^{s} + \theta_i^s\theta_j^s)\}, \text{ and} $$

$$ \frac{\partial\, {}_1f_{ij}^{s}}{\partial\,\theta_{ab}^{p}} = \frac{\partial\,\mu_{ij}^{s}}{\partial\,\theta_{ab}^{p}} - (1 - \delta_{io})\delta_{ps}\{\delta_{bo}\;\delta_{jo}\delta_{ia} + (1 - \delta_{jo})(\delta_{ia}\theta_j^s + \delta_{ja}\theta_i^s)] $$

$$ - (1 - \delta_{ab}/2)(1 - \delta_{bo})[1 - \delta_{jo}](\sigma_{ia}^{p}\sigma_{jb}^{p} + \sigma_{ib}^{p}\sigma_{ja}^{p})\}\,. $$

The functions $\{{}_3f_{ij}^s\} = \{{}_2f_{ij}^s/\mu_o^s\}$.

Hence $\dfrac{\partial\, {}_3f_{ij}^s}{\partial\, \theta_{ab}^p} = \dfrac{1}{\mu_o^s} \dfrac{\partial\, {}_2f_{ij}^s}{\partial\, \theta_{ab}^p} - \dfrac{{}_2f_{ij}^s}{(\mu_o^s)^2} \dfrac{\partial\, \mu_o^s}{\partial\, \theta_{ab}^p}$.

The maximum likelihood Newton-Raphson iteration equations give:

$$\sum_{abp} \frac{\partial\, {}_3f_{ij}^s}{\partial\, \theta_{ab}^p} \Delta\, \theta_{ab}^p = -\, {}_3f_{ij}^s .$$

Multiplying by μ_o^s and substituting, we have

$$\sum_{abp} \left[\frac{\partial\, {}_2f_{ij}^s}{\partial\theta_{ab}^p} - {}_2f_{ij}^s\left(\frac{1}{\mu_o^s} \frac{\partial\, \mu_o^s}{\partial\, \theta_{ab}^p}\right)\right] \Delta\theta_{ab}^p = -{}_2f_{ij}^s .$$

Thus the equations for METH = 3 are readily calculated from the equations for METH = 2. Only the matrix of coefficients has to be changed, simply by subtracting ${}_2f_{ij}^s\left(\dfrac{1}{\mu_o^s} \dfrac{\partial\, \mu_o^s}{\partial\theta_{ab}^p}\right)$.

Instead of Newton-Raphson iteration, a method of successive substitutions may be used for finding M_i^s and σ_{ij}^s.

Since $f_{oi}^s = 0 = \mu_i^s - \mu_o^s M_i^s$, the value of M_i^s for the next iteration is defined as $M_i^s = \mu_i^s/\mu_o^s$.

Since $f_{ij}^s = 0 = \mu_{ij}^s - M_i^s\mu_i^s - M_j^s\mu_i^s - \mu_o^s(\sigma_{ij}^s - M_i^s M_j^s)$ we can solve for σ_{ij}^s after first substituting $M_i^s\mu_o^s$ for μ_i^s:

$\sigma_{ij}^s = \mu_{ij}^s/\mu_o^s - M_i^s M_j^s$, where M_i^s are the new values $= \mu_i^s/\mu_o^s$.

The new values of λ_p can be determined by Newton-Raphson iteration using only the equations $\{f_{oo}^s = \mu_o^s - 1 = 0\}$

Differentiation of these equations leads to the system

$$\sum_{p=1}^{r} \frac{\partial\, \mu_o^s}{\partial\, \lambda_p} \Delta\, \lambda_p = -\, \mu_o^s + 1 .$$

$$\text{or } \sum_{p=1}^{r} \mu_o^{ps} \Delta\lambda_p = \mu_o^s - 1$$

When the control card METH = 0, subroutine NEWTON determines the increments associated with successive substitutions.

Experience with the program seems to indicate that the NEWTON-Raphson iteration schemes have very small radii of convergence as compared with the successive substitution methods. Theoretically, however, the Newton-Raphson iteration should converge more rapidly once within its radius. Therefore, provision has been made in the program for running a fixed number of iterations with successive substitutions so as to get improved initial estimates and then switching to Newton-Raphson methods for the remaining iterations. If the control card option is -1, -2, or -3, then a certain number of iterations by successive substitution will be used before Newton-Raphson iteration by methods 1, 2, 3 respectively. The control card number IDUMP specifies the number of preliminary iterations.

The subroutine INITIAL determines the initial values of parameters preliminary to iteration. The proper determination of initial values is crucial to the successful convergence of any iteration method. The initial values determined by INITIAL are quite crude, and the researcher may wish to write his own version of INITIAL after some experimentation. The present version also allows the user to specify his own guesses of the initial values of the parameters of certain types.

The subroutine INITIAL takes a sample of 100 individuals and subjects them to a crude clustering procedure. For each individual, a count is made of the number of other individuals within a "box" two standard deviations on a side around it. That is, for individual k, the number of individuals j is counted such that $| x_{ik} - x_{ij}| \leq \sigma_i$ for all $i = 1, 2, \ldots, m$. The individual with the highest count is made the centroid of the first cluster--that is, his scores are the initial estimates of the means for the first type.

The individuals in the first cluster are erased from the sample and the procedure is repeated with the remaining individuals.

The initial estimates of the λ_s are all equal to 1/r, where r is the number of types.

The initial estimates of the covariances are the same for each type and are equal to the covariances computed from the sample of 100 taken as a whole.

After each iteration the subroutine RESULT prints the current estimates of the parameter for each type. At the end of the last iteration, the program PLACE gives the probabilities of membership in each type for each individual.

These probabilities are:

$$P(\text{individual } k \text{ \& Type } S) = \lambda_s \, g_s(x_k).$$

A few words should be said about the indexing used within the program. First of all, the indices of the moments do not range from o to m and o to r as in our equations, but from 1 to m+1 and 1 to r+1. Thus the value of $U(2, 3, 1, 1, 1, 4) = \mu_{12}^{3}$. The moments are conveniently calculated by

setting $z_{i+1} = x_i$ for each x_{ik}

and $z_1 = 1.0$. Similarly $G(1) = 1.0$

and G(2) is the relative density for type 1, $g_1(x_k)$.
The values for $PERS(K) = \lambda_{k-1}$

$COV(I, J, K) = \sigma_{ij}^{k-1}$ and $COVIN(I, J, K) = \sigma_{k-1}^{ij}$.

also $AV(I,K) = M_i^{k-1}$.

The routine MOMENT collapses the $\{\mu_{ijab}^{ps}\}$ into a two dimensional array.

The single index KL is uniquely related to p and s and the single index IJMN is uniquely related to the indices i, j, a, and b.

In general, suppose we have an array indexed as follows:

There are M indices. The first index varies from 1 to N. Each succeeding index varies from 1 to the preceding index.

$$1 \geq IX(1) \geq \ldots \geq IX(M)$$

Let S(N, M) = number of elements in this array.
Then $S(N,M) = \sum_{I=1}^{N} S(I, M-1)$
and $S(N,M) = \dfrac{(N+M-1)!}{M!\,(N-1)!} = \binom{N+M-1}{M}$.

Let $IX(1) \geq IX(2) \geq \ldots \geq IX(M)$ be a sequence of indices for a particular element of the array. The one-dimensional index of the element is

$$K = 1 + \sum_{I=1}^{M} \prod_{J=1}^{I} (IX(M-I+1)-2+J)/J$$

$$\text{or } K = 1 + \sum_{I=1}^{M} S(IX(M-I+1)-1, I).$$

These formulas are used by the Function U to look up values of μ^{ps}_{ijab} .

V Usage

A. Input (TAPE Unit 3, BCD-card images)
   1. Title Card in columns 1-72, any alphanumeric characters.
   2. Control Card

COLS:	NAME	DEFINITION
1-4	MX	Number of variables
5-12	NX	Sample size
13-16	IRM	Number of Types Assumed. If IRM = 0, 6 analyses will be done assuming 1, 2, 3, 4, 5, 6 types.
17-20	ITERM	Maximum Number of iterations. If blank, ITERM is set to 50.
21-28	CONV	Criterion of Convergence which all parameters must satisfy between successive iterations. If blank, CONV is set to .0001.
29-32	IRUN	=1 if every iteration is printed, = 0 if only the last iteration is printed
33-36	METH	=0 if successive substitutions is used ±1, ±2, ±3, if various Newton-Raphson methods are used.
37-40	IDUMP	The number of preliminary iterations by successive substitutions before Newton-Raphson iteration for METH = -1, -2, or -3.

3. Variable Format Card. This is an ordinary FORTRAN Variable Format Card according to which the data will be read.

4. Data Deck
   N sets of one or more cards per individual.

5. Initial Estimates of Parameters (optional)
   (a) Estimate control card
       Cols 1-4 = K = TYPE # (1 through 6)
       Cols 5-8 = λ_k = Proportion of population of type K (all 4 digits assumed after decimal point)
   (b) Means for type K
       8 digits per mean, last 4 digits assumed after decimal point.
   (c) Standard deviations for type K (same format as (b))
   (d) Correlation matrix for type K (1 row per card, same format as (b))
   (e) Estimate control card for another type, etc.

A blank card terminates the reading of initial estimates. If no initial estimates are to be read, one blank card must be read. The sets of initial estimates do not have to be present for all types. For example, a set of initial estimates for type 5 may be followed by a set for type 3 followed by a blank card. If any initial parameters are read for a given type, all parameters must be read for that type. For example, initial estimates of means for type 3 must be followed by initial estimates of standard deviations and correlations for type 3.

If no initial estimates of parameters are read in, the computer will generate its own by a clustering procedure.

Multiple runs may be made at one time by placing one batch of input cards [A(1) to A(5)] followed by another batch. The last batch of data must be followed by two blank cards.

B. Recommended typical usage.

Under most conditions, the control card A-2 will be blank in all but the first 12 columns. The composition of the input will be:

Title Card
Control Card (Cols 1-12 only)
Variable Format Card
Data Cards
3 blank cards.

Another highly successful set of control parameters is METH = -1 or -2 with IDUMP = 40.

C. Output (Tape No 2).

The natural logarithm of the likelihood is printed for each iteration: $\sum_{k=1}^{N} \log f(x_k) - N(\sum_{s=1}^{r} \lambda_s - 1)$. If the likelihood on an iteration is not greater than that on all previous iterations, the computer prints out"Iteration--diverges". The parameters for each type are printed out on either the last iteration or on every iteration, depending on the value of IRUN. At the end of the last iteration, the probabilities of type membership for each individual, $\lambda_s g_s(x_k)$, are printed. If sense switch 3 is down, the computer dumps out the moments $\{\mu^{ps}_{ijab}\}$ and the matrices $\{\sigma^{s}_{ij}\}$ and $\{\sigma^{ij}_{s}\}$ following subroutine MOMENT, the matrix A of coefficients of the Newton-Raphson iteration and the vector $\{-f^{s}_{ij}\}$ following subroutine NEWTON, and the matrices A^{-1} and the vector $\{\Delta\ \theta^{p}_{ab}\}$ following the subroutine MATINV.

D. Time estimates.

For METH = 0, time per iteration in seconds is

$$T = .14\ N\ (m+1)(m+2)(r+1)$$

For METH ≠ 0, time per iteration in seconds is

$$T = .0075\ (m+1)(m+2)(m+3)(m+4)(r+1)(r+2)N + .00002[(m+1)(m+2)r]^3 N$$

A typical run with N = 225, r = 1, 2, 3, 4, 5, 6, m = 2 required 50 minutes by METH = 0 on the CDC 1604.

E. Suggestions for reducing running time.

1. By intuition, cluster analysis of variables, factor analysis, or any other means, reduce the number of variables to those that are really important for discriminating types.

2. Use any good prior information from theoretical hypotheses or any other classification-clustering procedures to provide initial estimates.

3. If you have a large sample, say 10,000, don't run it all at once. Take a small sample, say 100 to 200 cases at random and run an analysis using a weak convergence criterion, say .01. Get an idea of how many types there are and some initial estimates from the small sample. Run the entire set of data using these initial estimates on specified numbers of types. The hypothesis H_r that there are r types can be tested against the alternative, H_{r-1} that there are r-1 types by the χ^2 test

$$\chi^2_{d.f.} = -2 \log \lambda = 2(L_r - L_{r-1})$$

with d.f. = (m+1)(m+2)/2

where L_r = likelihood printed for r types. Thus the χ^2 test of the likelihood ratio on the small sample may give a good guess as to how many types there are. A final test on the entire sample could be performed specifying r, r+1, and r-1 types on the runs, where r is the hypothesized number of types.

VI Example - Artifical Clusters

To test this program, an example was constructed consisting of three artifical clusters in two dimensions. The points in each cluster were generated by a pseudo-random normal deviate generator. The characteristics of each cluster are given in Table 1 below. The results of the computer printout are summarized in Table 2. The points are plotted in Figure 1. The 75 points in Cluster 1 are designated by triangles in Figure 1, the 50 points in Cluster 2 are designated by squares, and the 100 points in Cluster 3 are designated by circles. Drawn around some of the squares, circles, and triangles are larger squares, circles and triangles. The larger symbols give the classification assigned by the computer program with 3 specified types. If a point does not have two symbols around it, it was correctly classified by the computer. It is evident from the figure that most points were correctly classified, and the computer's types have clear cut boundaries whereas the actual clusters overlap to some degree.

TABLE 1
Characteristics of Artificial Clusters

	Cluster 1		Cluster 2		Cluster 3	
Number/225	.333		.222		.444	
MEANS	.05	-1.33	-.83	1.64	1.41	.85
S.D. 's	1.26	.49	.87	1.04	.92	.97
r_{12}	.1590		.4040		.4743	

TABLE 2
Characteristics of Types from Computer Program

	Type 2		Type 3		Type 1	
λ_p	.346		.170		.484	
Means	.20	-1.31	-1.11	1.79	1.19	.93
S.D. 's	1.28	.50	.83	1.12	1.04	.91
r_{12}	.2462		.7168		.5231	

FIGURE 1

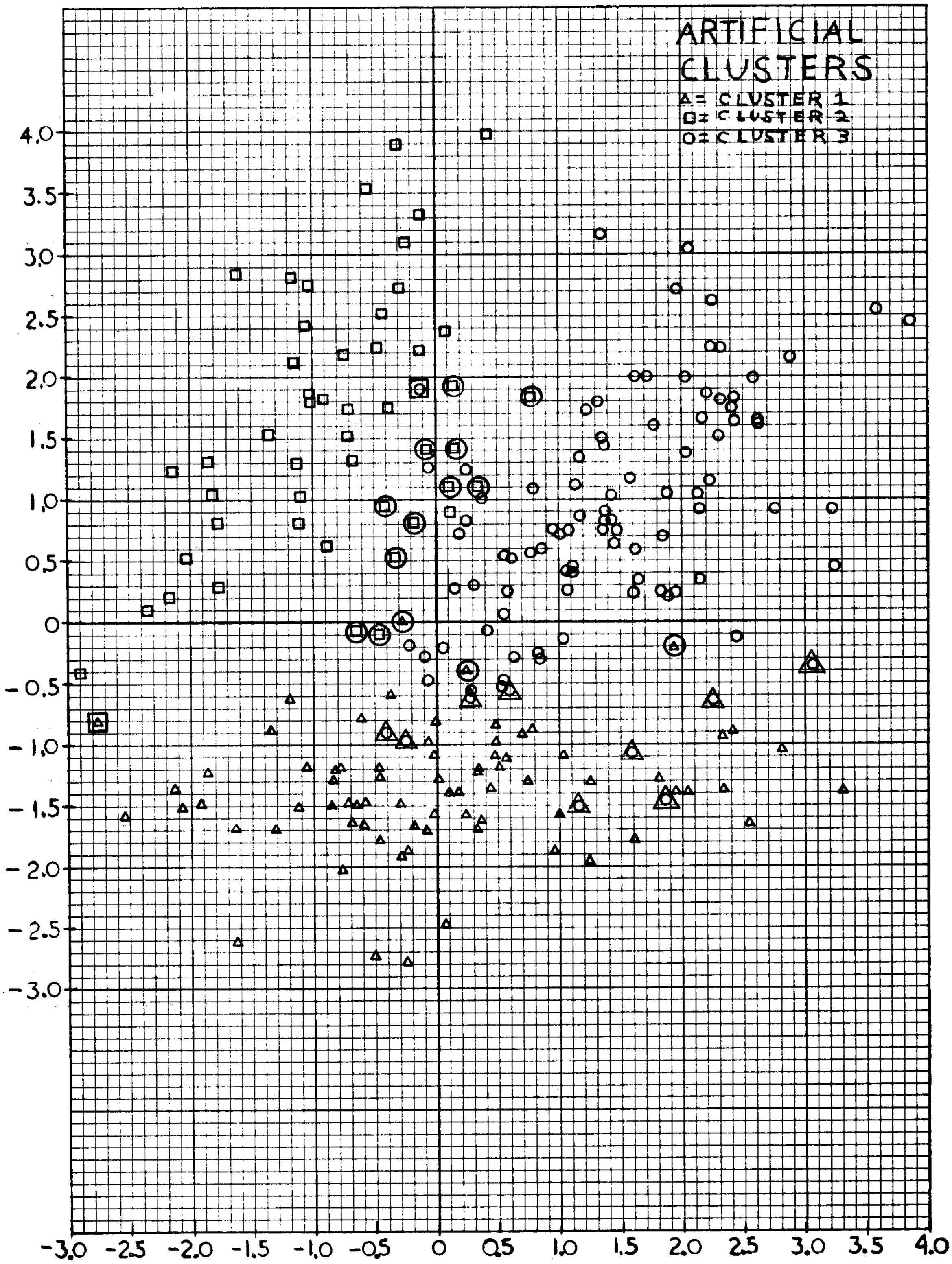
ARTIFICIAL CLUSTERS
△= CLUSTER 1
□= CLUSTER 2
○= CLUSTER 3
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.0 -2.5 -2.0 -1.5 -1.0 -0.5 0 0.5 1.0 1.5 2.0 2.5 3.0 3.5 4.0

TABLE 3

Likelihoods and χ^2 for Numbers of Types

Number of Types	Natural Logarithm of Likelihood	χ^2(with 6d.f.) $=2(L_R - L_{R-1})$	P
1	-380.48930		
2	-358.96468	43.04924	$.114 \times 10^{-6}$
3	-340.36400	37.20136	$.161 \times 10^{-5}$
4	-334.83078	11.06644	$.863 \times 10^{-1}$
5	-325.63847	18.38462	$.534 \times 10^{-2}$
6	-318.02872	15.21950	$.186 \times 10^{-1}$

The data cards for input are listed in Section B and the output is given in Section C.

A previous run with METH = 0, IRM = 0 (not given here) gave the likelihoods for 1 to 6 types. These are presented in Table 3, along with associated χ^2 values. The results indicate that the hypothesis that there are only two types can be rejected against the hypothesis that there are three types; but the hypothesis that there are three types cannot be rejected against the alternative that there are four types.

In order to save space in this report, only a run with IRM = 3 is given in Sections B and C.

The method used was METH = -1, which has 10 preliminary iterations by successive substitutions followed by Newton-Raphson iteration. The previous unpublished run with METH = 0 took 45 iterations to converge, while the one presented here took only 17. The difference can be attributed to the superior convergence rate of Newton-Raphson methods. However, other computer runs with IRM = 0, METH = -1 and IDUMP = 10, 30, or 40 sometimes failed to converge at all, once the Newton-Raphson procedure was started. If the likelihoods have not coverged to 0.1 by the successive substitutions, then Newton-Raphson iteration often fails. Thus the initial estimates must be quite accurate if Newton-Raphson iteration is to work. So far the various methods -1, -2, -3, appear to work equally well. All three converged in exactly 17 iterations in the present example. The run reported here took 13 minutes on the CDC 1604.

SECTION VI

A. Listing of Input for Example

```
      ARTIFICIAL CLUSTERS  METHOD-1  10PRELIMINARY ITERATIONS
   2     225   3                  -1  10
(2F4.2)
-058-141                                                                        001
-008-167                                                                        002
 193-017                                                                        003
-027 002                                                                        004
-002-151                                                                        005
-016-165                                                                        006
 024-157                                                                        007
-070-165                                                                        008
-110-150                                                                        009
 000-075                                                                        010
-214-131                                                                        011
 006-243                                                                        012
 281-100                                                                        013
 232-090                                                                        014
-047-173                                                                        015
-082-118                                                                        016
-251-157                                                                        017
 204-134                                                                        018
 240-089                                                                        019
 330-138                                                                        020
 001-125                                                                        021
-083-129                                                                        022
-207-149                                                                        023
 103-108                                                                        024
 183-122                                                                        025
 034-114                                                                        026
 035-160                                                                        027
-194-146                                                                        028
-084-148                                                                        029
 082-088                                                                        030
-077-200                                                                        031
 198-134                                                                        032
-136-085                                                                        033
-130-168                                                                        034
 121-191                                                                        035
-048-113                                                                        036
 122-129                                                                        037
 253-161                                                                        038
-036-055                                                                        039
-080-119                                                                        040
-069-162                                                                        041
-003-104                                                                        042
-072-141                                                                        043
-276-077                                                                        044
 049-107                                                                        045
 050-115                                                                        046
 011-139                                                                        047
```

-162-257	048
-186-120	049
034-116	050
-164-166	051
-024-272	052
077-122	053
043-144	054
-030-144	055
-049-121	056
-108-115	057
078-086	058
-061-073	059
-050-270	060
016-135	061
232-133	062
-066-146	063
-120-060	064
-024-184	065
032-169	066
049-093	067
025-034	068
056-107	069
048-080	070
094-185	071
100-152	072
161-174	073
-007-096	074
-026-189	075
-070 134	076
-047 229	077
-030 278	078
077 186	079
-073 177	080
-107 245	081
-235-012	082
-219 025	083
-059 357	084
-286 167	085
-179 081	086
-111 105	087
013 145	088
-074 220	089
-071 155	090
-046 261	091
-068-004	092
-014 337	093
-115 292	094
-114 215	095
-179 031	096
-025 312	097

```
-041 098                                                    098
-035 055                                                      099
-103 280                                                      100
-292-038                                                      101
-092 184                                                      102
-205 053                                                      103
-186 108                                                      104
-018 084                                                      105
-215 128                                                      106
 045 400                                                      107
-049-007                                                      108
-161 288                                                      109
-007 143                                                      110
-111 082                                                      111
 011 111                                                      112
-090 066                                                      113
 034 112                                                      114
-113 133                                                      115
 013 196                                                      116
 011 091                                                      117
-186 130                                                      118
 106 240                                                      119
-103 187                                                      120
-040 178                                                      121
-138 157                                                      122
-011 226                                                      123
-032 391                                                      124
-101 185                                                      125
 063-027                                                      126
 131 180                                                      127
 062 052                                                      128
 275 092                                                      129
 107 042                                                      130
 208 306                                                      131
 038 105                                                      132
 218 137                                                      133
 113-150                                                      134
 139 147                                                      135
 027 110                                                      136
-014 190                                                      137
 060-054                                                      138
 240 179                                                      139
 161 200                                                      140
 141 084                                                      141
 086 061                                                      142
-007 129                                                      143
 134 151                                                      144
 172 200                                                      145
 230 153                                                      146
 160 025                                                      147
```

013 030	148
017 071	149
214 107	150
135 078	151
224-063	152
204 224	153
103-012	154
-009-049	155
056 056	156
181 025	157
163 060	158
189 106	159
261 163	160
205 139	161
193 126	162
359 255	163
118 187	164
109 074	165
211 038	166
023 083	167
322 093	168
011 109	169
054-042	170
111 112	171
-010-028	172
112 040	173
106 029	174
211 093	175
148 118	176
023 128	177
065 027	178
136 083	179
079 059	180
002-020	181
-031-089	182
139 092	183
287 217	184
044-002	185
241 185	186
179 160	187
096 079	188
058 009	189
221 188	190
232 180	191
222 117	192
083-027	193
148 078	194
135 316	195
197 270	196
-025-095	197

```
 203 200                                                                     198
 080 110                                                                     199
 175 122                                                                     200
 233 225                                                                     201
 245 167                                                                     202
 261 161                                                                     203
-023-018                                                                     204
 189 022                                                                     205
 218 169                                                                     206
 226 261                                                                     207
 096 018                                                                     208
 259 197                                                                     209
 031 032                                                                     210
 112 141                                                                     211
 385 246                                                                     212
 306-035                                                                     213
 242-010                                                                     214
 142 106                                                                     215
 158-102                                                                     216
 107 076                                                                     217
 024-054                                                                     218
 144 064                                                                     219
 324 046                                                                     220
 082-022                                                                     221
 163 037                                                                     222
 051-050                                                                     223
 182 071                                                                     224
 175-143                                                                     225
                                                                        BLANK001
                                                                        BLANK002
                                                                        BLANK003
```

SECTION VI

B. Computer Printout for Example

```
ITERATION  1, LIKELIHOOD OF  3 TYPES IN THIS SAMPLE =    -.42371292E+03
ITERATION  2, LIKELIHOOD OF  3 TYPES IN THIS SAMPLE =    -.37195046E+03
ITERATION  3, LIKELIHOOD OF  3 TYPES IN THIS SAMPLE =    -.36744158E+03
ITERATION  4, LIKELIHOOD OF  3 TYPES IN THIS SAMPLE =    -.36440563E+03
ITERATION  5, LIKELIHOOD OF  3 TYPES IN THIS SAMPLE =    -.36094806E+03
ITERATION  6, LIKELIHOOD OF  3 TYPES IN THIS SAMPLE =    -.35693192E+03
ITERATION  7, LIKELIHOOD OF  3 TYPES IN THIS SAMPLE =    -.35270142E+03
ITERATION  8, LIKELIHOOD OF  3 TYPES IN THIS SAMPLE =    -.34901014E+03
ITERATION  9, LIKELIHOOD OF  3 TYPES IN THIS SAMPLE =    -.34638251E+03
ITERATION 10, LIKELIHOOD OF  3 TYPES IN THIS SAMPLE =    -.34476925E+03
ITERATION 11, LIKELIHOOD OF  3 TYPES IN THIS SAMPLE =    -.34381568E+03
ITERATION 12, LIKELIHOOD OF  3 TYPES IN THIS SAMPLE =    -.34657528E+03
ITERATION 11 DIVERGES
ITERATION 13, LIKELIHOOD OF  3 TYPES IN THIS SAMPLE =    -.34173161E+03
ITERATION 14, LIKELIHOOD OF  3 TYPES IN THIS SAMPLE =    -.34064823E+03
ITERATION 15, LIKELIHOOD OF  3 TYPES IN THIS SAMPLE =    -.34037172E+03
ITERATION 16, LIKELIHOOD OF  3 TYPES IN THIS SAMPLE =    -.34036401E+03
ITERATION 17, LIKELIHOOD OF  3 TYPES IN THIS SAMPLE =    -.34036400E+03
```

MAXIMUM-LIKELIHOOD ANALYSIS OF TYPES

ARTIFICIAL CLUSTERS METHOD-1 10PRELIMINARY ITERATIONS

SAMPLE SIZE =   225
NUMBER OF VARIABLES = 2
NUMBER OF TYPES =   3

ITERATION NUMBER 17

LIKELIHOOD OF 3 TYPES IN THIS SAMPLE =   -.34036400E+03

CHARACTERISTICS OF THE WHOLE SAMPLE

MEANS

.46	.30

STANDARD DEVIATIONS

1.38	1.47

CORRELATIONS

1.0000	.1849
.1849	1.0000

CHARACTERISTICS OF TYPE   1

THE PROPORTION OF THE POPULATION FROM THIS TYPE = .484

MEANS

1.19	.93

STANDARD DEVIATIONS

1.04	.91

CORRELATIONS

1.0000	.5231
.5231	1.0000

CHARACTERISTICS OF TYPE 2

THE PROPORTION OF THE POPULATION FROM THIS TYPE = .346

MEANS

.20	-1.31

STANDARD DEVIATIONS

1.28	.50

CORRELATIONS

1.0000	.2462
.2462	1.0000

CHARACTERISTICS OF TYPE 3

THE PROPORTION OF THE POPULATION FROM THIS TYPE = .170

MEANS

-1.11	1.79

STANDARD DEVIATIONS

.83	1.12

CORRELATIONS

1.0000	.7168
.7168	1.0000

PROBABILITIES OF TYPE MEMBERSHIP

	1	2	3
1	.041	.959	.000
2	.021	.979	.000
3	.739	.261	.000
4	.957	.040	.003
5	.029	.971	.000
6	.023	.977	.000
7	.022	.978	.000
8	.025	.975	.000
9	.031	.969	.000
10	.284	.716	.000
11	.023	.945	.032
12	.009	.991	.000
13	.006	.994	.000
14	.025	.975	.000
15	.021	.979	.000
16	.072	.928	.000
17	.010	.954	.037
18	.006	.994	.000
19	.023	.977	.000
20	.000	1.000	.000
21	.057	.943	.000
22	.054	.946	.000
23	.018	.971	.012
24	.056	.944	.000
25	.013	.987	.000
26	.071	.929	.000
27	.020	.980	.000
28	.021	.971	.008
29	.034	.966	.000
30	.138	.862	.000
31	.016	.984	.000
32	.007	.993	.000
33	.150	.834	.015
34	.021	.979	.000
35	.005	.995	.000
36	.087	.913	.000
37	.022	.978	.000
38	.001	.999	.000
39	.496	.504	.000
40	.070	.930	.000
41	.026	.974	.000
42	.111	.889	.000
43	.041	.959	.000
44	.016	.344	.640
45	.084	.916	.000
46	.064	.936	.000
47	.037	.963	.000
48	.015	.985	.000
49	.039	.942	.019
50	.067	.933	.000
51	.019	.980	.001
52	.013	.987	.000
53	.042	.958	.000
54	.028	.972	.000
55	.037	.963	.000
56	.069	.931	.000
57	.072	.927	.001
58	.153	.847	.000
59	.292	.707	.001
60	.014	.986	.000

61	.041	.959	.000
62	.004	.996	.000
63	.036	.964	.000
64	.342	.626	.031
65	.017	.983	.000
66	.016	.984	.000
67	.137	.863	.000
68	.746	.254	.000
69	.081	.919	.000
70	.217	.783	.000
71	.007	.993	.000
72	.014	.986	.000
73	.004	.996	.000
74	.144	.856	.000
75	.016	.984	.000
76	.186	.000	.814
77	.020	.000	.980
78	.006	.000	.994
79	.992	.000	.008
80	.042	.000	.958
81	.001	.000	.999
82	.023	.016	.961
83	.017	.001	.982
84	.000	.000	1.000
85	.000	.000	1.000
86	.015	.000	.985
87	.089	.000	.911
88	.927	.000	.073
89	.009	.000	.991
90	.094	.000	.906
91	.006	.000	.994
92	.906	.066	.028
93	.001	.000	.999
94	.000	.000	1.000
95	.003	.000	.997
96	.054	.001	.944
97	.002	.000	.998
98	.788	.000	.212
99	.963	.001	.036
100	.000	.000	1.000
101	.008	.051	.941
102	.016	.000	.984
103	.014	.000	.986
104	.006	.000	.994
105	.960	.000	.040
106	.002	.000	.998
107	.001	.000	.999
108	.919	.073	.008
109	.000	.000	1.000
110	.810	.000	.190
111	.169	.000	.831
112	.979	.000	.021
113	.477	.000	.523
114	.995	.000	.005
115	.036	.000	.964
116	.594	.000	.406
117	.991	.000	.009
118	.004	.000	.996
119	.984	.000	.016
120	.009	.000	.991
121	.160	.000	.840
122	.007	.000	.993
123	.106	.000	.894
124	.000	.000	1.000
125	.011	.000	.989
126	.800	.200	.000

127	1.000	.000	.000
128	.999	.001	.000
129	1.000	.000	.000
130	.997	.003	.000
131	1.000	.000	.000
132	.997	.000	.003
133	1.000	.000	.000
134	.013	.987	.000
135	1.000	.000	.000
136	.993	.000	.007
137	.304	.000	.696
138	.477	.523	.000
139	1.000	.000	.000
140	1.000	.000	.000
141	1.000	.000	.000
142	.999	.001	.000
143	.883	.000	.117
144	1.000	.000	.000
145	1.000	.000	.000
146	1.000	.000	.000
147	.987	.013	.000
148	.994	.006	.001
149	.997	.000	.003
150	1.000	.000	.000
151	1.000	.000	.000
152	.105	.895	.000
153	1.000	.000	.000
154	.893	.107	.000
155	.575	.425	.000
156	.999	.001	.000
157	.984	.016	.000
158	.999	.001	.000
159	1.000	.000	.000
160	1.000	.000	.000
161	1.000	.000	.000
162	1.000	.000	.000
163	1.000	.000	.000
164	1.000	.000	.000
165	1.000	.000	.000
166	.992	.008	.000
167	.997	.000	.003
168	1.000	.000	.000
169	.981	.000	.019
170	.638	.362	.000
171	1.000	.000	.000
172	.805	.195	.000
173	.997	.003	.000
174	.993	.007	.000
175	1.000	.000	.000
176	1.000	.000	.000
177	.980	.000	.020
178	.993	.007	.000
179	1.000	.000	.000
180	.999	.001	.000
181	.868	.132	.000
182	.184	.816	.000
183	1.000	.000	.000
184	1.000	.000	.000
185	.951	.049	.000
186	1.000	.000	.000
187	1.000	.000	.000
188	1.000	.000	.000
189	.975	.025	.000
190	1.000	.000	.000
191	1.000	.000	.000
192	1.000	.000	.000

193	.786	.214	.000
194	1.000	.000	.000
195	.897	.000	.103
196	1.000	.000	.000
197	.151	.849	.000
198	1.000	.000	.000
199	1.000	.000	.000
200	1.000	.000	.000
201	1.000	.000	.000
202	1.000	.000	.000
203	1.000	.000	.000
204	.876	.123	.001
205	.979	.021	.000
206	1.000	.000	.000
207	1.000	.000	.000
208	.985	.015	.000
209	1.000	.000	.000
210	.995	.005	.000
211	1.000	.000	.000
212	1.000	.000	.000
213	.135	.865	.000
214	.709	.291	.000
215	1.000	.000	.000
216	.041	.959	.000
217	1.000	.000	.000
218	.506	.494	.000
219	.999	.001	.000
220	.982	.018	.000
221	.833	.167	.000
222	.995	.005	.000
223	.538	.462	.000
224	1.000	.000	.000
225	.007	.993	.000

VII Fortran Program Listing

```
      PROGRAM TYPE                                                      TYPE0000
      DIMENSION X(5,3000),Z(6),PER(7),PERS(7),AV(5,7),COV(5,5,7),COVIN(5TYPE0001
     1,5,7),ALPHA(7),DETERM(7),G(7),IX(4),ID(4),V(126,28),A(126,126),B(1TYPE0002
     226),BV(126),RECORD(12),SD(5),XSAM(5,100),MAT(100,100)            TYPE0003
      DIMENSION FMT(12)                                                 TYPE0004
      COMMON X,Z,PER,PERS,AV,COV,COVIN,ALPHA,DETERM,G,V,A,B,BV,RECORD,SDTYPE0005
     1,MX,MX1,IR,XIR,IR1,IJMNX,KLX,NX,XNX,JATMX,CONV,ITERM,LX,XSAM,MAT,ETYPE0006
     2PH,IX,ID,ITER,PROB,METH,IDUMP                                     TYPE0007
      EQUIVALENCE (XSAM,V),(A,MAT,X)                                    TYPE0008
    1 READ INPUT TAPE 3,101,(RECORD(I),I=1,12),MX,NX,IRM,ITERM,CONV,IRUNTYPE0009
     1,MOTH,IDUMP                                                       TYPE0010
  101 FORMAT(12A6/I4,I8,2I4,F8.8,3I4)                                   TYPE0011
      IF(MX) 23,23,2                                                    TYPE0012
C     MX=NUMBER OF VARIABLES.                                           TYPE0013
C     THE PROGRAM READS BATCHES OF DATA UNTIL 2 BLANK CARDS ARE ENCOUNTETYPE0014
C    1RED.                                                              TYPE0015
    2 IF(ITERM)3,3,4                                                    TYPE0016
C     ITERM=MAXIMUM NUMBER OF PERMISSABLE ITERATIONS.                   TYPE0017
    3 ITERM=50                                                          TYPE0018
    4 IF(CONV)5,5,6                                                     TYPE0019
C     CONV=CRITERION OF CONVERGENCE WHICH ALL PARAMETERS MUST SATISFY BETYPE0020
C    1TWEEN SUCCESSIVE ITERATIONS.                                      TYPE0021
    5 CONV= 0.0001                                                      TYPE0022
    6 MX1=MX+1                                                          TYPE0023
      REWIND 4                                                          TYPE0024
      XNX=NX                                                            TYPE0025
      NEV=XINTF(XNX/100.)+1                                             TYPE0026
      IJMNX=(MX1*(MX1+1)*(MX1+2)*(MX1+3))/24                            TYPE0027
      KRMN=0                                                            TYPE0028
      LX=0                                                              TYPE0029
      READ INPUT TAPE 3,102,(FMT(I),I=1,12)                             TYPE0030
  102 FORMAT(12A6)                                                      TYPE0031
    7 CALL BLOCK(NX,3000,KRMN,KST,KEND,LONG)                            TYPE0032
      DO 8 J=1,LONG                                                     TYPE0033
    8 READ INPUT TAPE 3,FMT,(X(I,J),I=1,MX)                             TYPE0034
      DO 9 J=1,LONG,NEV                                                 TYPE0035
      LX=LX+1                                                           TYPE0036
      DO 9 I=1,MX                                                       TYPE0037
    9 XSAM (I,LX)=X(I,J)                                                TYPE0038
C     XSAM=SAMPLE OF UP TO 100 CASES USED FOR INITIAL ESTIMATES OF PARAMTYPE0039
C    1ETERS.                                                            TYPE0040
      WRITE TAPE 4,((X(I,J),I=1,MX),J=1,LONG)                           TYPE0041
      IF (KRMN)10,10,7                                                  TYPE0042
   10 CALL INITIAL                                                      TYPE0043
      IR=IRM                                                            TYPE0044
C     IRM=NUMBER OF TYPES ASSUMED.  IF IRM=0, 6 DIFFERENT ANALYSES ARE  TYPE0045
C     DONE ASSUMING 1 TO 6 TYPES.                                       TYPE0046
      IF (IRM)11,11,12                                                  TYPE0047
   11 IR=IR+1                                                           TYPE0048
   12 IR1=IR+1                                                          TYPE0049
      METH=MOTH                                                         TYPE0050
```

```
      XIR=IR                                                            TYPE0051
      KLX=(IR1*(IR1+1))/2                                               TYPE0052
      JATMX = (MX1*(MX1+1)*IR)/2                                        TYPE0053
C     PERS(K)=PROPORTION OF POPULATION OF TYPE K.                       TYPE0054
      DO 913 K=2,IR1                                                    TYPE0055
      IF(IRM)13,13,912                                                  TYPE0056
  912 IF(PERS(K))13,13,913                                              TYPE0057
   13 PERS(K)=1./XIR                                                    TYPE0058
  913 CONTINUE                                                          TYPE0059
      ITER=0                                                            TYPE0060
      IDIV=0                                                            TYPE0061
      PROBA= -10E30                                                     TYPE0062
   14 ITER=ITER+1                                                       TYPE0063
      IF(DETERM(1)) 3500,3501,3501                                      TYPE0064
 3500 DETERM(1)=0.0                                                     TYPE0065
      GO TO 346                                                         TYPE0066
 3501 CALL MOMENT                                                       TYPE0067
      WRITE OUTPUT TAPE 2,9916,ITER,IR,PROB                             TYPE0068
 9916 FORMAT(10H ITERATION I3,15H, LIKELIHOOD OF I3,23H TYPES IN THIS SATYPE0069
     1MPLE =E18.8)                                                      TYPE0070
      IF(PROB-PROBA)346,347,347                                         TYPE0071
  346 IDIV=IDIV+1                                                       TYPE0072
      ITERA=ITER-1                                                      TYPE0073
      WRITE OUTPUT TAPE 2,9915,ITERA                                    TYPE0074
 9915 FORMAT(10H ITERATION I3,9H DIVERGES)                              TYPE0075
      DO 3445 K=1,JATMX                                                 TYPE0076
 3445 B(K)=  0.5*B(K)                                                   TYPE0077
      IF(IDIV-1)115,3446,115                                            TYPE0078
 3446 DO 3447 K=1,JATMX                                                 TYPE0079
 3447 B(K)= -B(K)                                                       TYPE0080
      GO TO 115                                                         TYPE0081
  347 IDIV=0                                                            TYPE0082
      IF(METH)344,348,348                                               TYPE0083
  344 IF(ITER-IDUMP) 348,348,3444                                       TYPE0084
 3444 METH=-METH                                                        TYPE0085
C     IDUMP= THE NUMBER OF PRELIMINARY ITERATIONS BY SUCCESSIVE         TYPE0086
C     SUBSTITUTIONS BEFORE NEWTON-RAPHSON ITERATION FOR METH=-1,-2,OR -3TYPE0087
  348 PROBA=PROB                                                        TYPE0088
      IF(SENSE SWITCH 3)349,249                                         TYPE0089
  349 WRITE OUTPUT TAPE 2,9910                                          TYPE0090
 9910 FORMAT(7H MOMENT)                                                 TYPE0091
      WRITE OUTPUT TAPE 2,9914,((V(I,J),J=1,KLX),I=1,IJMNX)             TYPE0092
      WRITE OUTPUT TAPE 2,9914,(((COV(I,J,K),I=1,MX),J=1,MX),K=1,IR1)   TYPE0093
      WRITE OUTPUT TAPE 2,9914,(((COVIN(I,J,K),I=1,MX),J=1,MX),K=1,IR1) TYPE0094
  249 CONTINUE                                                          TYPE0095
  114 CALL NEWTON                                                       TYPE0096
      IF(SENSE SWITCH3) 350,250                                         TYPE0097
  350 WRITE OUTPUT TAPE 2,9911                                          TYPE0098
 9911 FORMAT(7H NEWTON)                                                 TYPE0099
      WRITE OUTPUT TAPE 2,9914,((A(I,J),I=1,JATMX),J=1,JATMX)           TYPE0100
```

```
      WRITE OUTPUT TAPE 2,9914,(B(I),I=1,JATMX)                         TYPE0101
 9914 FORMAT(6F12.6)                                                    TYPE0102
  250 CONTINUE                                                          TYPE0103
      IF(METH) 115,115,248                                              TYPE0104
C     IF METH = 0,SUCCESSIVE SUBSTITUTION IS USED,OTHERWISE             TYPE0105
C     NEWTON-RAPHSON ITERATION.                                         TYPE0106
  248 CALL MATINV(A,JATMX,B,1,DA)                                       TYPE0107
      IF(SENSE SWITCH3) 351,251                                         TYPE0108
  351 WRITE OUTPUT TAPE 2,9912                                          TYPE0109
      WRITE OUTPUT TAPE 2,9914,((A(I,J),I=1,JATMX),J=1,JATMX)           TYPE0111
 9912 FORMAT(7H MATINV)                                                 TYPE0110
      WRITE OUTPUT TAPE 2,9914,(B(I),I=1,JATMX)                         TYPE0112
  251 CONTINUE                                                          TYPE0113
  115 CALL RAPHSON                                                      TYPE0114
      IF(IRUN) 15,16,15                                                 TYPE0115
C     IF IRUN=0,ONLY THE LAST ITERATION IS PRINTED.                     TYPE0116
   15 CALL RESULT                                                       TYPE0117
   16 IF(ITER-ITERM)17,19,19                                            TYPE0118
   17 DO 18 K=1,JATMX                                                   TYPE0119
      TA=B(K)                                                           TYPE0120
      TA=ABSF(TA)                                                       TYPE0121
      IF(TA-CONV)18,18,14                                               TYPE0122
   18 CONTINUE                                                          TYPE0123
   19 IF(IRUN) 21,20,21                                                 TYPE0124
   20 CALL RESULT                                                       TYPE0125
   21 CALL PLACE                                                        TYPE0126
      IF(IRM)22,22,1                                                    TYPE0127
   22 IF(IR-6)11,1,1                                                    TYPE0128
   23 END FILE 2                                                        TYPE0129
      CALL EXIT                                                         TYPE0130
      END(0,1,0,0,0)                                                    TYPE0131
```

```
      SUBROUTINE INITIAL                                                TYPE0132
      DIMENSION X(5,3000),Z(6),PER(7),PERS(7),AV(5,7),COV(5,5,7),COVIN(5TYPE0133
     1,5,7),ALPHA(7),DETERM(7),G(7),IX(4),ID(4),V(126,28),A(126,126),B(1TYPE0134
     226),BV(126),RECORD(12),SD(5),XSAM(5,100),MAT(100,100)             TYPE0135
      COMMON X,Z,PER,PERS,AV,COV,COVIN,ALPHA,DETERM,G,V,A,B,BV,RECORD,SDTYPE0136
     1,MX,MX1,IR,XIR,IR1,IJMNX,KLX,NX,XNX,JATMX,CONV,ITERM,LX,XSAM,MAT,ETYPE0137
     2PH,IX,ID,ITER,PROB,METH,IDUMP                                     TYPE0138
      EQUIVALENCE (XSAM,V),(A,MAT,X)                                    TYPE0139
C     THIS SUBROUTINE DETERMINES INITIAL ESTIMATES OF THE MEANS AND COVATYPE0140
C    1RIANCES OF THE TYPES BY APPLYING A CLUSTERING PROCEDURE TO A SAMPLTYPE0141
C    2E OF UP TO 100 CASES.                                             TYPE0142
C     FIRST THE SAMPLE MEANS AND COVARIANCES ARE DETERMINED.            TYPE0143
      XLX=LX                                                            TYPE0144
      DO 22 I=1,MX                                                      TYPE0145
      AV(I,1)=0.                                                        TYPE0146
      DO 1 K=1,LX                                                       TYPE0147
    1 AV(I,1)=AV(I,1)+XSAM(I,K)                                         TYPE0148
   22 AV(I,1)=AV(I,1)/XLX                                               TYPE0149
      DO 3 I=1,MX                                                       TYPE0150
      DO 3 J=I,MX                                                       TYPE0151
      COV(I,J,1)=0.                                                     TYPE0152
      DO 2 K=1,LX                                                       TYPE0153
    2 COV(I,J,1)=COV(I,J,1)+XSAM(I,K)*XSAM(J,K)                         TYPE0154
      COV(I,J,1)=COV(I,J,1)/XLX-AV(I,1)*AV(J,1)                         TYPE0155
    3 COV(J,I,1)=COV(I,J,1)                                             TYPE0156
      DO 4 I=1,MX                                                       TYPE0157
C     AN LX BY LX MATRIX IS COMPUTED. AN ELEMENT CORRESPONDING TO A PAIRTYPE0158
C    1 OF POINTS IS 1 IF AND ONLY IF BOTH POINTS LIE WITHIN A BOX WHOSE TYPE0159
C    2 SIDES ARE ONE STANDARD DEVIATION LONG.                           TYPE0160
      SDV=COV(I,I,1)                                                    TYPE0161
    4 SD(I)=SQRTF(SDV)                                                  TYPE0162
      DO 7 K=1,LX                                                       TYPE0163
      DO 7 L=K,LX                                                       TYPE0164
      DO 5 I=1,MX                                                       TYPE0165
      R=XSAM(I,K)-XSAM(I,L)                                             TYPE0166
      R=ABSF(R)                                                         TYPE0167
      IF(R-SD(I))5,6,6                                                  TYPE0168
    5 CONTINUE                                                          TYPE0169
      MAT(L,K)=1                                                        TYPE0170
      MAT(K,L)=1                                                        TYPE0171
      GO TO 7                                                           TYPE0172
    6 MAT(L,K)=0                                                        TYPE0173
      MAT(K,L)=0                                                        TYPE0174
    7 CONTINUE                                                          TYPE0175
      DO 14 M=2,7                                                       TYPE0176
C     THE CENTROID OF A CLUSTER IS THE POINT WITH THE GREATEST NUMBER OFTYPE0177
C    1OTHER POINTS IN A BOX AROUND IT.                                  TYPE0178
C     FIRST THE INDEX OF THE POINT OF GREATEST DENSITY IS FOUND. THE TYPTYPE0179
C    1E MEANS ARE THE CO-ORDINATES OF THE DENSEST POINT.                TYPE0180
      MAX=0                                                             TYPE0181
      DO 10 K=1,LX                                                      TYPE0182
      LB=0                                                              TYPE0183
      DO 8 L=1,LX                                                       TYPE0184
```

```
    8 LB=LB+MAT(L,K)                                                   TYPE0185
      IF(LB-MAX)10,10,9                                                 TYPE0186
    9 MAX=LB                                                            TYPE0187
      LAK=K                                                             TYPE0188
   10 CONTINUE                                                          TYPE0189
      DO 11 I=1,MX                                                      TYPE0190
      AV(I,M)=XSAM(I,LAK)                                               TYPE0191
      DO 11 J=I,MX                                                      TYPE0192
      COV(J,I,M)=COV(J,I,1)                                             TYPE0193
   11 COV(I,J,M)=COV(J,I,M)                                             TYPE0194
      DO 14 K=1, LX                                                     TYPE0195
      IF(MAT(K,LAK))14,14,12                                            TYPE0196
   12 DO 13 L=1,LX                                                      TYPE0197
   13 MAT(K,L)=0                                                        TYPE0198
C     ALL PREVIOUSLY CLUSTERED POINTS ARE ERASED. THE CENTROID OF THE NETYPE0199
C    1XT CLUSTER THUS WILL BE THE POINT WITH THE GREATEST NUMBER OF PREVTYPE0200
C    2IOUSLY UNCLUSTERED POINTS WITHIN A BOX AROUND IT.                 TYPE0201
   14 CONTINUE                                                          TYPE0202
      DO 15 I=1,MX                                                      TYPE0203
      DO 15 J=1,MX                                                      TYPE0204
   15 A(I,J)= COV(I,J,1)                                                TYPE0205
      CALL MATINV(A,MX,B,0,DA)                                          TYPE0206
      AD=ABSF(DA)/DA                                                    TYPE0207
      DA=(SQRTF(AD/DA))*AD                                              TYPE0208
      DO 16  K=1,7                                                      TYPE0209
      PERS(K)=-1.0                                                      TYPE0210
      DETERM(K)= DA                                                     TYPE0211
      DO 16  I=1,MX                                                     TYPE0212
      DO 16  J=1,MX                                                     TYPE0213
   16 COVIN(I,J,K)=A(I,J)                                               TYPE0214
C     THE ROUTINE ALSO READS IN INITIAL ESTIMATES UNTIL A BLANK CARD    TYPE0215
C     IS ENCOUNTERED.                                                   TYPE0216
   17 READ INPUT TAPE 3,122,K,DA                                        TYPE0217
  122 FORMAT(I4,F4.4)                                                   TYPE0218
      IF(K) 18,21,18                                                    TYPE0219
   18 K=K+1                                                             TYPE0220
      PERS(K)=DA                                                        TYPE0221
      READ INPUT TAPE 3,123,(AV(I,K),I=1,MX)                            TYPE0222
      READ INPUT TAPE 3,123,(SD(I),I=1,MX)                              TYPE0223
  123 FORMAT(5F8.4)                                                     TYPE0224
      DO 19 J=1,MX                                                      TYPE0225
      READ INPUT TAPE 3,123,(COV(I,J,K),I=1,MX)                         TYPE0226
      DO 19 I=1,MX                                                      TYPE0227
      COV(I,J,K)=COV(I,J,K)*SD(I)*SD(J)                                 TYPE0228
   19 A(I,J)=COV(I,J,K)                                                 TYPE0229
      CALL MATINV(A,MX,B,0,DA)                                          TYPE0230
      AD=ABSF(DA)/DA                                                    TYPE0231
      DA=(SQRTF(AD/DA))*AD                                              TYPE0232
      DETERM(K)=DA                                                      TYPE0233
      DO 20 I=1,MX                                                      TYPE0234
      DO 20 J=1,MX                                                      TYPE0235
   20 COVIN(I,J,K)=A(I,J)                                               TYPE0236
      GO TO 17                                                          TYPE0237
   21 RETURN                                                            TYPE0238
      END(0,1,0,0,0)                                                    TYPE0239
```

```
      SUBROUTINE DENSITY                                                TYPE0240
      DIMENSION X(5,3000),Z(6),PER(7),PERS(7),AV(5,7),COV(5,5,7),COVIN(5TYPE0241
     1,5,7),ALPHA(7),DETERM(7),G(7),IX(4),ID(4),V(126,28),A(126,126),B(1TYPE0242
     226),BV(126),RECORD(12),SD(5),XSAM(5,100),MAT(100,100)             TYPE0243
      COMMON X,Z,PER,PERS,AV,COV,COVIN,ALPHA,DETERM,G,V,A,B,BV,RECORD,SDTYPE0244
     1,MX,MX1,IR,XIR,IR1,IJMNX,KLX,NX,XNX,JATMX,CONV,ITERM,LX,XSAM,MAT,ETYPE0245
     2PH,IX,ID,ITER,PROB,METH,IDUMP                                     TYPE0246
      EQUIVALENCE (XSAM,V),(A,MAT,X)                                    TYPE0247
C     FOR CURRENT ESTIMATES OF THE PARAMETERS, THIS ROUTINE COMPUTES-   TYPE0248
C     ALPHA(K)=MULTIVARIATE NORMAL DENSITY FOR TYPE K AT POINT Z.       TYPE0249
C     EPH=DENSITY AT Z FOR THE MIXTURE OF DISTRIBUTIONS                 TYPE0250
C     G(K)=ALPHA/EPH AT POINT Z.                                        TYPE0251
      DO 3 K=2,IR1                                                      TYPE0252
      AL=0.                                                             TYPE0253
      DO 2 I=1,MX                                                       TYPE0254
      IA=I+1                                                            TYPE0255
      DO 1 J=I,MX                                                       TYPE0256
    1 AL=AL-COVIN(J,I,K)*(Z(IA)-AV(I,K))*(Z(J+1)-AV(J,K))               TYPE0257
    2 AL=AL+0.5*COVIN(I,I,K)*(Z(IA)-AV(I,K))**2                         TYPE0258
    3 ALPHA (K)= DETERM (K)*EXPF(AL)                                    TYPE0259
      EPH=0.                                                            TYPE0260
      DO 4 K=2,IR1                                                      TYPE0261
    4 EPH=EPH+PERS(K)*ALPHA(K)                                          TYPE0262
      DO 5 K=2,IR1                                                      TYPE0263
    5 G(K)=ALPHA(K)/EPH                                                 TYPE0264
      G(1)=1.                                                           TYPE0265
C     THE ADDITION OF A DUMMY TYPE OF DENSITY G=1 SIMPLIFIES THE COMPUTATYPE0266
C    1TION OF LOWER ORDER MOMENTS WITH ONE OR MORE TYPES OMITTED.       TYPE0267
      RETURN                                                            TYPE0268
      END(0,1,0,0,0)                                                    TYPE0269
```

```
      SUBROUTINE MOMENT                                                 TYPE0270
      DIMENSION X(5,3000),Z(6),PER(7),PERS(7),AV(5,7),COV(5,5,7),COVIN(5TYPE0271
     1,5,7),ALPHA(7),DETERM(7),G(7),IX(4),ID(4),V(126,28),A(126,126),B(1TYPE0272
     226),BV(126),RECORD(12),SD(5),XSAM(5,100),MAT(100,100)             TYPE0273
      COMMON X,Z,PER,PERS,AV,COV,COVIN,ALPHA,DETERM,G,V,A,B,BV,RECORD,SDTYPE0274
     1,MX,MX1,IR,XIR,IR1,IJMNX,KLX,NX,XNX,JATMX,CONV,ITERM,LX,XSAM,MAT,ETYPE0275
     2PH,IX,ID,ITER,PROB,METH,IDUMP                                     TYPE0276
      EQUIVALENCE (XSAM,V),(A,MAT,X)                                    TYPE0277
C     THIS ROUTINE COMPUTES THE GENERALIZED MOMENTS V(IJMN,KL) FOR VARIATYPE0278
C    1BLES I,J,M, AND N, AND TYPES K AND L. THE MOMENTS REALLY HAVE SIX TYPE0279
C    2INDICES BUT ARE STORED AS A 2-DIMENSIONAL MATRIX. TRIANGULAR INDEXTYPE0280
C    3ING IS USED TO ELIMINATE DUPLICATION.                             TYPE0281
      DO 1 IJMN=1,IJMNX                                                 TYPE0282
      DO 1 KL=1,KLX                                                     TYPE0283
    1 V(IJMN,KL)=0.                                                     TYPE0284
      KRMN=0                                                            TYPE0285
C     INITIALIZE                                                        TYPE0286
      REWIND 4                                                          TYPE0287
      PROB=XNX                                                          TYPE0288
      DO 10 K=2,IR1                                                     TYPE0289
   10 PROB=PROB-XNX*PERS(K)                                             TYPE0290
    2 CALL BLOCK(NX,3000,KRMN,KST,KEND,LONG)                            TYPE0291
      READ TAPE 4,((X(I,J),I=1,MX),J=1,LONG)                            TYPE0292
C     TAPE 4 IS READ IN BLOCKS OF 3000 CASES.                           TYPE0293
      DO 17 KA=1,LONG                                                   TYPE0294
      Z(1)=1.0                                                          TYPE0295
C     THE ADDITION OF A DUMMY VARIABLE=1 SIMPLIFIES THE COMPUTATION OF LTYPE0296
C    1OWER ORDER MOMENTS OMITTING ONE OR MORE VARIABLES.                TYPE0297
      DO 3 I=1,MX                                                       TYPE0298
    3 Z(I+1)=X(I,KA)                                                    TYPE0299
      CALL DENSITY                                                      TYPE0300
C     PROB=LIKELIHOOD OF SAMPLE                                         TYPE0301
      IF(EPH)7,7,8                                                      TYPE0302
    7 INDEX=KST+KA-1                                                    TYPE0303
      WRITE OUTPUT TAPE 2,120,INDEX,EPH                                 TYPE0304
  120 FORMAT(45H NEGATIVE PROBABILITY DENSITY FOR OBSERVATION I5,3H  =  TYPE0305
     1F16.9)                                                            TYPE0306
      GO TO 9                                                           TYPE0307
    8 PROB=PROB+LOGF(EPH)                                               TYPE0308
    9 KL=0                                                              TYPE0309
      IF(METH) 12,12,11                                                 TYPE0310
   11 DO 4  K=1,IR1                                                     TYPE0311
      DO 4 L=1,K                                                        TYPE0312
      KL=KL+1                                                           TYPE0313
      GT1=G(K)*G(L)                                                     TYPE0314
      IJMN=0                                                            TYPE0315
      DO 4 I=1,MX1                                                      TYPE0316
      GT2=GT1*Z(I)                                                      TYPE0317
      DO 4 J=1,I                                                        TYPE0318
      GT3=GT2*Z(J)                                                      TYPE0319
```

```
      DO 4 M=1,J                                                        TYPE0320
      GT4  =GT3*Z(M)                                                    TYPE0321
      DO 4 N=1,M                                                        TYPE0322
      IJMN=IJMN+1                                                       TYPE0323
    4 V(IJMN,KL)=V(IJMN,KL)+GT4*Z(N)                                    TYPE0324
   17 CONTINUE                                                          TYPE0325
   15 IF(KRMN)5,5,2                                                     TYPE0326
    5 DO 6 IJMN=1,IJMNX                                                 TYPE0327
      DO 6 KL=1,KLX                                                     TYPE0328
    6 V(IJMN,KL)=V(IJMN,KL)/XNX                                         TYPE0329
      RETURN                                                            TYPE0330
   12 KL=1                                                              TYPE0331
      DO 13 K=1,IR1                                                     TYPE0332
C     FOR  METH=0,WE DO NOT NEED ALL THE MOMENTS                        TYPE0333
      IJMN=2                                                            TYPE0334
      DO 14 I=2,MX1                                                     TYPE0335
      GT1= Z(I)* G(K)                                                   TYPE0336
      JS=0                                                              TYPE0337
      DO 14 J=1,I                                                       TYPE0338
      V(IJMN,KL)= V(IJMN,KL) +   GT1* Z(J)                              TYPE0339
      JS=JS+J                                                           TYPE0340
   14 IJMN=IJMN+JS                                                      TYPE0341
      DO 13 L=1,K                                                       TYPE0342
      V(1,KL)=V(1,KL)+G(K)*G(L)                                         TYPE0343
   13 KL=KL+1                                                           TYPE0344
      GO TO 17                                                          TYPE0345
      END(0,1,0,0,0)                                                    TYPE0346
```

```
      FUNCTION U(IA,JA,MA,NA,KA,LA )                                    TYPE0347
      DIMENSION X(5,3000),Z(6),PER(7),PERS(7),AV(5,7),COV(5,5,7),COVIN(5TYPE0348
     1,5,7),ALPHA(7),DETERM(7),G(7),IX(4),ID(4),V(126,28),A(126,126),B(1TYPE0349
     226),BV(126),RECORD(12),SD(5),XSAM(5,100),MAT(100,100)             TYPE0350
      COMMON X,Z,PER,PERS,AV,COV,COVIN,ALPHA,DETERM,G,V,A,B,BV,RECORD,SDTYPE0351
     1,MX,MX1,IR,XIR,IR1,IJMNX,KLX,NX,XNX,JATMX,CONV,ITERM,LX,XSAM,MAT,ETYPE0352
     2PH,IX,ID,ITER,PROB,METH,IDUMP                                     TYPE0353
      EQUIVALENCE (XSAM,V),(A,MAT,X)                                    TYPE0354
C     THIS ROUTINE FINDS THE MOMENT CORRESPONDING TO 6 INDICES BY COMPUTTYPE0355
C    1ING THE 2 INDICES OF THE MOMENT IN THE MATRIX V.                  TYPE0356
      IX(1)=IA                                                          TYPE0357
      IX(2)=JA                                                          TYPE0358
      IX(3)=MA                                                          TYPE0359
      IX(4)=NA                                                          TYPE0360
C     THE FIRST 4 INDICES ARE PUT IN ORDER FROM SMALLEST TO LARGEST     TYPE0361
      DO 3 J=1,4                                                        TYPE0362
      KAT=100                                                           TYPE0363
      DO 2 I=1,4                                                        TYPE0364
      IF(KAT-IX(I)) 2,2,1                                               TYPE0365
    1 KAT=IX(I)                                                         TYPE0366
      K=I                                                               TYPE0367
      GO TO 2                                                           TYPE0368
    2 CONTINUE                                                          TYPE0369
      IX(K)=100                                                         TYPE0370
    3 ID(J)=KAT                                                         TYPE0371
C     THE LAST 2 INDICES ARE PUT IN ORDER FROM LARGER TO SMALLER        TYPE0372
      IF (KA-LA)4,5,5                                                   TYPE0373
    4 KAT=LA                                                            TYPE0374
      LAT=KA                                                            TYPE0375
      GO TO 6                                                           TYPE0376
    5 KAT=KA                                                            TYPE0377
      LAT=LA                                                            TYPE0378
    6 IJMN=1                                                            TYPE0379
C     THE APPROPRIATE INDICES OF THE MOMENT MATRIX V ARE COMPUTED.      TYPE0380
      DO 8 I=1,4                                                        TYPE0381
      LPR=1                                                             TYPE0382
      DO 7 J=1,I                                                        TYPE0383
    7 LPR=(LPR*(ID(I)-2+J))/J                                           TYPE0384
    8 IJMN=IJMN+LPR                                                     TYPE0385
      KL=(KAT*(KAT-1))/2+LAT                                            TYPE0386
      U=V(IJMN,KL)                                                      TYPE0387
      RETURN                                                            TYPE0388
      END(0,1,0,0,0)                                                    TYPE0389
```

```
      FUNCTION ONE(IW,IU)                                               TYPE0390
C     THIS IS JUST THE KRONECKER DELTA                                  TYPE0391
      IF(IW-IU)1,2,1                                                    TYPE0392
    1 ONE=0.                                                            TYPE0393
      GO TO 3                                                           TYPE0394
    2 ONE=1.                                                            TYPE0395
    3 RETURN                                                            TYPE0396
      END                                                               TYPE0397
```

```
      FUNCTION EM(II,IJ,IA,IS,IP)                                       TYPE0398
      DIMENSION X(5,3000),Z(6),PER(7),PERS(7),AV(5,7),COV(5,5,7),COVIN(5TYPE0399
     1,5,7),ALPHA(7),DETERM(7),G(7),IX(4),ID(4),V(126,28),A(126,126),B(1TYPE0400
     226),BV(126),RECORD(12),SD(5),XSAM(5,100),MAT(100,100)             TYPE0401
      COMMON X,Z,PER,PERS,AV,COV,COVIN,ALPHA,DETERM,G,V,A,B,BV,RECORD,SDTYPE0402
     1,MX,MX1,IR,XIR,IR1,IJMNX,KLX,NX,XNX,JATMX,CONV,ITERM,LX,XSAM,MAT,ETYPE0403
     2PH,IX,ID,ITER,PROB,METH,IDUMP                                     TYPE0404
      EQUIVALENCE (XSAM,V),(A,MAT,X)                                    TYPE0405
C     EM IS A TERM WHICH APPEARS IN THE PARTIAL DERIVATIVE OF THE MAXIMUTYPE0406
C    1M LIKELIHOOD EQUATIONS WITH RESPECT TO THE TYPE MEANS.            TYPE0407
      EM=0.                                                             TYPE0408
      IA1=IA-1                                                          TYPE0409
      DO 1 I=1,MX                                                       TYPE0410
      IB=I                                                              TYPE0411
    1 EM=EM+COVIN(IA1,IB,IP)*(U(1,II,IJ,IB+1,IP,IS)-AV(IB,IP)*U(1,1,II,ITYPE0412
     1J,IP,IS))                                                         TYPE0413
      RETURN                                                            TYPE0414
      END(0,1,0,0,0)                                                    TYPE0415
```

```
      FUNCTION SIG(II,IJ,IA,IB,IS,IP)                                   TYPE0416
      DIMENSION X(5,3000),Z(6),PER(7),PERS(7),AV(5,7),COV(5,5,7),COVIN(5TYPE0417
     1,5,7),ALPHA(7),DETERM(7),G(7),IX(4),ID(4),V(126,28),A(126,126),B(1TYPE0418
     226),BV(126),RECORD(12),SD(5),XSAM(5,100),MAT(100,100)             TYPE0419
      COMMON X,Z,PER,PERS,AV,COV,COVIN,ALPHA,DETERM,G,V,A,B,BV,RECORD,SDTYPE0420
     1,MX,MX1,IR,XIR,IR1,IJMNX,KLX,NX,XNX,JATMX,CONV,ITERM,LX,XSAM,MAT,ETYPE0421
     2PH,IX,ID,ITER,PROB,METH,IDUMP                                     TYPE0422
      EQUIVALENCE (XSAM,V),(A,MAT,X)                                    TYPE0423
C     SIG IS A TERM WHICH APPEARS IN THE PARTIAL DERIVATIVE OF THE MAXIMTYPE0424
C    1UM LIKELIHOOD EQUATIONS WITH RESPECT TO THE ELEMENTS OF THE INVERSTYPE0425
C    2E OF THE TYPE COVARIANCE MATRIX.                                  TYPE0426
      IA1=IA-1                                                          TYPE0427
      IB1=IB-1                                                          TYPE0428
      SIG=U(1,1,II,IJ,IP,IS)*(COV(IA1,IB1,IP)-AV(IA1,IP)*AV(IB1,IP))+AV(TYPE0429
     1IB1,IP)*U(1,II,IJ,IA,IP,IS)+AV(IA1,IP)*U(1,II,IJ,IB,IP,IS)-U(II,IJTYPE0430
     2,IA,IB,IP,IS)                                                     TYPE0431
      RETURN                                                            TYPE0432
      END(0,1,0,0,0)                                                    TYPE0433
```

```
      FUNCTION DERV(II,IJ,IS,IA,IB,IP)                                  TYPE0434
      DIMENSION X(5,3000),Z(6),PER(7),PERS(7),AV(5,7),COV(5,5,7),COVIN(5TYPE0435
     1,5,7),ALPHA(7),DETERM(7),G(7),IX(4),ID(4),V(126,28),A(126,126),B(1TYPE0436
     226),BV(126),RECORD(12),SD(5),XSAM(5,100),MAT(100,100)             TYPE0437
      COMMON X,Z,PER,PERS,AV,COV,COVIN,ALPHA,DETERM,G,V,A,B,BV,RECORD,SDTYPE0438
     1,MX,MX1,IR,XIR,IR1,IJMNX,KLX,NX,XNX,JATMX,CONV,ITERM,LX,XSAM,MAT,ETYPE0439
     2PH,IX,ID,ITER,PROB,METH,IDUMP                                     TYPE0440
      EQUIVALENCE (XSAM,V),(A,MAT,X)                                    TYPE0441
C     THIS ROUTINE COMPUTES THE DERIVATIVE OF THE IJS MOMENT WITH       TYPE0442
C     RESPECT TO THE ABP PARAMETER.                                     TYPE0443
      IF(IA-1)1,1,2                                                     TYPE0444
    1 DERV = -U(II,IJ,1,1,IS,IP)                                        TYPE0445
      GO TO 10                                                          TYPE0446
    2 IF(IB-1)3,3,6                                                     TYPE0447
    3 DERV=-PERS(IP)*EM(II,IJ,IA,IS,IP)                                 TYPE0448
      IF(IP-IS)5,4,5                                                    TYPE0449
    4 DERV=DERV+EM(II,IJ,IA,1,IP)                                       TYPE0450
    5 GO TO 10                                                          TYPE0451
    6 DERV=-PERS(IP)*SIG(II,IJ,IA,IB,IS,IP)                             TYPE0452
      IF(IP-IS)8,7,8                                                    TYPE0453
    7 DERV=DERV+SIG(II,IJ,IA,IB,1,IP)                                   TYPE0454
    8 IF(IA-IB)10,9,10                                                  TYPE0455
    9 DERV=0.5*DERV                                                     TYPE0456
   10 RETURN                                                            TYPE0457
      END                                                               TYPE0458
```

```
      SUBROUTINE NEWTON                                                 TYPE0459
      DIMENSION X(5,3000),Z(6),PER(7),PERS(7),AV(5,7),COV(5,5,7),COVIN(5TYPE0460
     1,5,7),ALPHA(7),DETERM(7),G(7),IX(4),ID(4),V(126,28),A(126,126),B(1TYPE0461
     226),BV(126),RECORD(12),SD(5),XSAM(5,100),MAT(100,100)             TYPE0462
      COMMON X,Z,PER,PERS,AV,COV,COVIN,ALPHA,DETERM,G,V,A,B,BV,RECORD,SDTYPE0463
     1,MX,MX1,IR,XIR,IR1,IJMNX,KLX,NX,XNX,JATMX,CONV,ITERM,LX,XSAM,MAT,ETYPE0464
     2PH,IX,ID,ITER,PROB,METH,IDUMP                                     TYPE0465
      EQUIVALENCE (XSAM,V),(A,MAT,X)                                    TYPE0466
C     THIS ROUTINE SETS UP THE NEWTON-RAPHSON MATRIX EQUATION FOR THE CHTYPE0467
C    1ANGES IN THE PARAMETERS FOR THE NEXT ITERATION.                   TYPE0468
      MEK=XABSF(METH)                                                   TYPE0469
C     METH = +1 OR -1 IMPLIES SIMPLIFIED MAXIMUM-LIKELIHOOD EQUATIONS   TYPE0470
C     METH = +2 OR -2 IMPLIES COMPLETE MAXIMUM-LIKELIHOOD EQUATIONS     TYPE0471
C     METH= +3 OR -3 IMPLIES COMPLETE EQUATIONS DIVIDED BY UOS          TYPE0472
      IF(METH)42,42,8                                                   TYPE0473
    8 IAT=0                                                             TYPE0474
      DO 27 K=2,IR1                                                     TYPE0475
      UOS=U(1,1,1,1,K,1)                                                TYPE0476
      DO 27 J=1,MX1                                                     TYPE0477
      UJS=U(1,1,1,J,K,1)                                                TYPE0478
      DO 27 I=J,MX1                                                     TYPE0479
      UIS=U(1,1,1,I,K,1)                                                TYPE0480
      JAT=0                                                             TYPE0481
      IAT=IAT+1                                                         TYPE0482
      DO 27 L=2,IR1                                                     TYPE0483
      DO 27 N=1,MX1                                                     TYPE0484
      DO 27 M=N,MX1                                                     TYPE0485
      JAT=JAT+1                                                         TYPE0486
      TEMP=DERV(I,J,K,M,N,L)                                            TYPE0487
      IF(I-1) 27,27,9                                                   TYPE0488
    9 GO TO (12,10,10),MEK                                              TYPE0489
   10 TEMP=TEMP-AV(I-1,K)*DERV(J,1,K,M,N,L)                             TYPE0490
      IF(J-1)12,12,11                                                   TYPE0491
   11 TEMP=TEMP-(COV(I-1,J-1,K)-AV(I-1,K)*AV(J-1,K))*DERV(1,1,K,M,N,L)  TYPE0492
     1-AV(J-1,K)*DERV(I,1,K,M,N,L)                                      TYPE0493
   12 IF(K-L)27,13,27                                                   TYPE0494
   13 IF(N-1)14,14,17                                                   TYPE0495
   14 GO TO (16,15,15),MEK                                              TYPE0496
   15 TEMP=TEMP-ONE(I,M)*UJS-(1.0-ONE(J,1))*(ONE(J,M)*UIS-UOS*          TYPE0497
     1(ONE(I,M)*AV(J-1,K)+ONE(J,M)*AV(I-1,K)))                          TYPE0498
      GO TO 27                                                          TYPE0499
   16 TEMP= TEMP-ONE(J,1)*ONE(I,M)-(1.0- ONE(J,1))*(ONE(I,M)*AV(J-1,K)+ TYPE0500
     1ONE(J,M)*AV(I-1,K))                                               TYPE0501
      GO TO 27                                                          TYPE0502
   17 IF(J-1)27,27,18                                                   TYPE0503
   18 TEM=COV(I-1,M-1,L)*COV(J-1,N-1,L)+COV(I-1,N-1,L)*COV(J-1,M-1,L)   TYPE0504
      GO TO (20,19,19),MEK                                              TYPE0505
   19 TEM=TEM*UOS                                                       TYPE0506
   20 IF(M-N)22,21,22                                                   TYPE0507
   21 TEM=0.5*TEM                                                       TYPE0508
   22 TEMP=TEMP+TEM                                                     TYPE0509
   27 A(IAT,JAT)=TEMP                                                   TYPE0510
      L=0                                                               TYPE0511
      DO 33 K=2,IR1                                                     TYPE0512
      L=L+1                                                             TYPE0513
```

```
      UOS=U(1,1,1,1,K,1)                                                TYPE0514
      B(L)=1.0-UOS                                                      TYPE0515
      DO 29 I=2,MX1                                                     TYPE0516
      L=L+1                                                             TYPE0517
      B(L)=AV(I-1,K)                                                    TYPE0518
      GO TO (29,28,28),MEK                                              TYPE0519
   28 B(L)=B(L)*UOS                                                     TYPE0520
   29 B(L)=B(L)-U(1,1,1,I,K,1)                                          TYPE0521
      DO 33 J=2,MX1                                                     TYPE0522
      DO 33 I=J,MX1                                                     TYPE0523
      L=L+1                                                             TYPE0524
      GO TO (31,30,30),MEK                                              TYPE0525
   30 B(L)=UOS*(COV(I-1,J-1,K)-AV(I-1,K)*AV(J-1,K))+AV(I-1,K)*          TYPE0526
     1U(1,1,1,J,K,1)+AV(J-1,K)*U(1,1,1,I,K,1)                           TYPE0527
      GO TO 33                                                          TYPE0528
   31 B(L)=COV(I-1,J-1,K)+AV(I-1,K)*AV(J-1,K)                           TYPE0529
   33 B(L)=B(L)-U(1,1,I,J,K,1)                                          TYPE0530
      GO TO (37,37,34),MEK                                              TYPE0531
   34 IAT=0                                                             TYPE0532
      DO 36 K=2,IR1                                                     TYPE0533
      JAT=0                                                             TYPE0534
      DO 35 L=2,IR1                                                     TYPE0535
      DO 35 N=1,MX1                                                     TYPE0536
      DO 35 M=N,MX1                                                     TYPE0537
      JAT=JAT+1                                                         TYPE0538
   35 BV(JAT)= DERV(1,1,K,M,N,L)/U(1,1,1,1,K,1)                         TYPE0539
      DO 36 J=1,MX1                                                     TYPE0540
      DO 36 I=J,MX1                                                     TYPE0541
      IAT=IAT+1                                                         TYPE0542
      JAT=0                                                             TYPE0543
      DO 36 L=2,IR1                                                     TYPE0544
      DO 36 N=1,MX1                                                     TYPE0545
      DO 36 M=N,MX1                                                     TYPE0546
      JAT=JAT+1                                                         TYPE0547
   36 A(IAT,JAT)= A(IAT,JAT)+B(IAT)*BV(JAT)                             TYPE0548
   37 RETURN                                                            TYPE0549
   42 DO 44 I=1,IR                                                      TYPE0550
      DO 43 J=1,I                                                       TYPE0551
      A(I,J)=U(1,1,1,1,I+1,J+1)                                         TYPE0552
   43 A(J,I)=A(I,J)                                                     TYPE0553
   44 B(I)= U(1,1,1,1,1,I+1)-1.0                                        TYPE0554
      CALL MATINV(A,IR,B,1,DA)                                          TYPE0555
      L=IR                                                              TYPE0556
      DO 52 K=2,IR1                                                     TYPE0557
      PERT= U(1,1,1,1,1,K)                                              TYPE0558
      DO 52 I=1,MX                                                      TYPE0559
      L=L+1                                                             TYPE0560
      B(L)=U(1,1,1,I+1,1,K)/PERT -AV(I,K)                               TYPE0561
      BV(I)= AV(I,K)+B(L)                                               TYPE0562
      DO 52 J=1,I                                                       TYPE0563
      L=L+1                                                             TYPE0564
   52 B(L)=U(1,1,J+1,I+1,1,K)/PERT -BV(I)*BV(J)-COV(I,J,K)              TYPE0565
      GO TO 37                                                          TYPE0566
      END                                                               TYPE0567
```

```
      SUBROUTINE RAPHSON                                                TYPE0568
      DIMENSION X(5,3000),Z(6),PER(7),PERS(7),AV(5,7),COV(5,5,7),COVIN(5TYPE0569
     1,5,7),ALPHA(7),DETERM(7),G(7),IX(4),ID(4),V(126,28),A(126,126),B(1TYPE0570
     226),BV(126),RECORD(12),SD(5),XSAM(5,100),MAT(100,100)            TYPE0571
      COMMON X,Z,PER,PERS,AV,COV,COVIN,ALPHA,DETERM,G,V,A,B,BV,RECORD,SDTYPE0572
     1,MX,MX1,IR,XIR,IR1,IJMNX,KLX,NX,XNX,JATMX,CONV,ITERM,LX,XSAM,MAT,ETYPE0573
     2PH,IX,ID,ITER,PROB,METH,IDUMP                                     TYPE0574
      EQUIVALENCE (XSAM,V),(A,MAT,X)                                    TYPE0575
C     THIS ROUTINE ADDS THE APPROPRIATE CHANGES TO THE PARAMETERS TO OBTTYPE0576
C    1AIN THEIR VALUES FOR THE NEXT ITERATION.                          TYPE0577
      AKAT=2.0                                                          TYPE0578
      IF(METH)30,30,23                                                  TYPE0579
C     NEWTON-RAPHSON ITERATION                                          TYPE0580
   23 IAT=1                                                             TYPE0581
      MXT=(MX1*(MX1+1))/2                                               TYPE0582
C     SHORTEN   INCREMENT VECTOR UNTIL ALL PERCENTAGES ARE WITHIN BOUNDSTYPE0583
      DO 10 K=2,IR1                                                     TYPE0584
      AKAZ=-PERS(K)/(2.0*B(IAT))                                        TYPE0585
      IF(AKAZ)7,7,8                                                     TYPE0586
    7 AKAZ =AKAZ + 0.5/B(IAT)                                           TYPE0587
    8 IF(AKAZ-AKAT)9,10,10                                              TYPE0588
    9 AKAT=AKAZ                                                         TYPE0589
   10 IAT=IAT+MXT                                                       TYPE0590
      IF(AKAT-1.0)11,13,13                                              TYPE0591
   11 DO 12 K=1,JATMX                                                   TYPE0592
   12 B(K)=AKAT*B(K)                                                    TYPE0593
   13 IAT =0                                                            TYPE0594
      DO 26 K=2,IR1                                                     TYPE0595
      IAT=IAT+1                                                         TYPE0596
      PERS(K)=PERS(K)+B(IAT)                                            TYPE0597
      DO 24I=1,MX                                                       TYPE0598
      IAT=IAT+1                                                         TYPE0599
   24 AV(I,K)=AV(I,K)+B(IAT)                                            TYPE0600
      DO 25 J=1,MX                                                      TYPE0601
      DO 25I=J,MX                                                       TYPE0602
      IAT=IAT+1                                                         TYPE0603
      COVIN(I,J,K)=COVIN(I,J,K)+B(IAT)                                  TYPE0604
      COVIN(J,I,K)=COVIN(I,J,K)                                         TYPE0605
      A(I,J)=COVIN(I,J,K)                                               TYPE0606
   25 A(J,I)=A(I,J)                                                     TYPE0607
      CALL MATINV(A,MX,BV,0,DA)                                         TYPE0608
      AD=ABSF(DA)/DA                                                    TYPE0609
      DETERM(K)=(SQRTF(AD*DA))*AD                                       TYPE0610
      IF(AD)27,27,28                                                    TYPE0611
   27 INDEX=K-1                                                         TYPE0612
      WRITE OUTPUT TAPE 2,121,INDEX,DA                                  TYPE0613
  121 FORMAT(44H DETERMINANT OF COVARIANCE INVERSE FOR TYPE I2,1H=F16.9)TYPE0614
      DETERM(1)= -1.0                                                   TYPE0615
   28 DO 26 J=1,MX                                                      TYPE0616
      DO 26 I=1,MX                                                      TYPE0617
   26 COV(I,J,K)=A(I,J)                                                 TYPE0618
   29 RETURN                                                            TYPE0619
```

```
C     SUCCESSIVE SUBSTITUTIONS.                                         TYPE0620
   30 L=IR                                                              TYPE0621
      MXT=1                                                             TYPE0622
C     SHORTEN   INCREMENT VECTOR UNTIL ALL PERCENTAGES ARE WITHIN BOUNDSTYPE0623
      DO 36 K=2,IR1                                                     TYPE0624
      AKAZ=-PERS(K)/(2.0*B(K-1))                                        TYPE0625
      IF(AKAZ)33,33,34                                                  TYPE0626
   33 AKAZ = AKAZ +0.5/B(K-1)                                           TYPE0627
   34 IF(AKAZ-AKAT)35,36,36                                             TYPE0628
   35 AKAT=AKAZ                                                         TYPE0629
   36 CONTINUE                                                          TYPE0630
      IF(AKAT-1.0)37,39,39                                              TYPE0631
   37 DO 38 K=1,IR                                                      TYPE0632
   38 B(K)=AKAT*B(K)                                                    TYPE0633
   39 DO 55 K= 2,IR1                                                    TYPE0634
      PERS(K)=PERS(K)+ B(K-1)                                           TYPE0635
      DO 52 I=1,MX                                                      TYPE0636
      L=L+1                                                             TYPE0637
   47 AV(I,K)=AV(I,K)+B(L)                                              TYPE0638
      DO 52 J=1,I                                                       TYPE0639
      L=L+1                                                             TYPE0640
   49 COV(I,J,K)=COV(I,J,K)+B(L)                                        TYPE0641
   51 COV(J,I,K)=COV(I,J,K)                                             TYPE0642
      A(I,J)=COV(I,J,K)                                                 TYPE0643
      A(J,I)=A(I,J)                                                     TYPE0644
   52 CONTINUE                                                          TYPE0645
   53 CALL MATINV(A,MX,BV,0,DA)                                         TYPE0646
      AD=ABSF(DA)/DA                                                    TYPE0647
      DETERM(K)=(SQRTF(AD/DA))*AD                                       TYPE0648
      DO 54 I=1,MX                                                      TYPE0649
      DO 54 J=1,MX                                                      TYPE0650
   54 COVIN(I,J,K)=A(I,J)                                               TYPE0651
   55 CONTINUE                                                          TYPE0652
      GO TO 29                                                          TYPE0653
      END                                                               TYPE0654
```

```
      SUBROUTINE RESULT                                                 TYPE0655
      DIMENSION X(5,3000),Z(6),PER(7),PERS(7),AV(5,7),COV(5,5,7),COVIN(5TYPE0656
     1,5,7),ALPHA(7),DETERM(7),G(7),IX(4),ID(4),V(126,28),A(126,126),B(1TYPE0657
     226),BV(126),RECORD(12),SD(5),XSAM(5,100),MAT(100,100)             TYPE0658
      COMMON X,Z,PER,PERS,AV,COV,COVIN,ALPHA,DETERM,G,V,A,B,BV,RECORD,SDTYPE0659
     1,MX,MX1,IR,XIR,IR1,IJMNX,KLX,NX,XNX,JATMX,CONV,ITERM,LX,XSAM,MAT,ETYPE0660
     2PH,IX,ID,ITER,PROB,METH,IDUMP                                     TYPE0661
      EQUIVALENCE (XSAM,V),(A,MAT,X)                                    TYPE0662
C     THIS ROUTINE PRINTS OUT THE PARAMETERS DESCRIBING EACH TYPE.      TYPE0663
    1 FORMAT(1H1,15X,36HMAXIMUM-LIKELIHOOD ANALYSIS OF TYPES//4X,11A6)  TYPE0664
    2 FORMAT(1H1,15X,35HCHARACTERISTICS OF THE WHOLE SAMPLE///3X)       TYPE0665
    3 FORMAT(3X//////16X,23HCHARACTERISTICS OF TYPE I4//3X)             TYPE0666
    4 FORMAT(11X,49HTHE PROPORTION OF THE POPULATION FROM THIS TYPE =F6.TYPE0667
     13/3X)                                                             TYPE0668
    5 FORMAT(30X,5HMEANS/5F12.2)                                        TYPE0669
    6 FORMAT(3X/23X,19HSTANDARD DEVIATIONS /5F12.2)                     TYPE0670
    7 FORMAT(3X/28X,12HCORRELATIONS)                                    TYPE0671
    8 FORMAT (5F12.4)                                                   TYPE0672
    9 FORMAT(3X/////10X,13HSAMPLE SIZE =I10/10X,21HNUMBER OF VARIABLES =TYPE0673
     1I2/10X,17HNUMBER OF TYPES =I6//25X,16HITERATION NUMBER I3//3X,    TYPE0674
     213HLIKELIHOOD OF I2,23H TYPES IN THIS SAMPLE =E18.8)              TYPE0675
      DO 10 J=1,MX                                                      TYPE0676
      JA=J+1                                                            TYPE0677
   10 AV(J,1)=U(1,1,1,JA,1,1)                                           TYPE0678
      DO 20 J=1,MX                                                      TYPE0679
      JA=J+1                                                            TYPE0680
      DO 20 I=J,MX                                                      TYPE0681
      IA=I+1                                                            TYPE0682
      COV(I,J,1)=U(1,1,JA,IA,1,1)-AV(J,1)*AV(I,1)                       TYPE0683
   20 COV(J,I,1)=COV(I,J,1)                                             TYPE0684
      WRITE OUTPUT TAPE 2,1,(RECORD(I),I=1,11)                          TYPE0685
      WRITE OUTPUT TAPE 2,9,NX,MX,IR,ITER,IR,PROB                       TYPE0686
      DO 16 K=1,IR1                                                     TYPE0687
      K1=K-1                                                            TYPE0688
      IF (K1) 11,11,12                                                  TYPE0689
   11 WRITE OUTPUT TAPE 2,2                                             TYPE0690
      GO TO 13                                                          TYPE0691
   12 WRITE OUTPUT TAPE 2,3,K1                                          TYPE0692
      WRITE OUTPUT TAPE 2,4,PERS(K)                                     TYPE0693
   13 WRITE OUTPUT TAPE 2,5,(AV(I,K),I=1,MX)                            TYPE0694
      DO 14 I=1,MX                                                      TYPE0695
      SDV=COV(I,I,K)                                                    TYPE0696
      AD=ABSF(SDV)/SDV                                                  TYPE0697
      SDV=(SQRTF(AD*SDV))*AD                                            TYPE0698
   14 SD(I)=SDV                                                         TYPE0699
      DO 15 I=1,MX                                                      TYPE0700
      DO 15J=I,MX                                                       TYPE0701
      A(I,J)=COV(I,J,K)/(SD(I)*SD(J))                                   TYPE0702
   15 A(J,I)=A(I,J)                                                     TYPE0703
      WRITE OUTPUT TAPE 2,6,(SD(I),I=1,MX)                              TYPE0704
      WRITE OUTPUT TAPE 2,7                                             TYPE0705
      DO 16 J=1,MX                                                      TYPE0706
   16 WRITE OUTPUT TAPE 2,8,(A(I,J),I=1,MX)                             TYPE0707
      RETURN                                                            TYPE0708
      END(0,1,0,0,0)                                                    TYPE0709
```

```
      SUBROUTINE PLACE                                                  TYPE0710
      DIMENSION X(5,3000),Z(6),PER(7),PERS(7),AV(5,7),COV(5,5,7),COVIN(5TYPE0711
     1,5,7),ALPHA(7),DETERM(7),G(7),IX(4),ID(4),V(126,28),A(126,126),B(1TYPE0712
     226),BV(126),RECORD(12),SD(5),XSAM(5,100),MAT(100,100)             TYPE0713
      COMMON X,Z,PER,PERS,AV,COV,COVIN,ALPHA,DETERM,G,V,A,B,BV,RECORD,SDTYPE0714
     1,MX,MX1,IR,XIR,IR1,IJMNX,KLX,NX,XNX,JATMX,CONV,ITERM,LX,XSAM,MAT,ETYPE0715
     2PH,IX,ID,ITER,PROB,METH,IDUMP                                     TYPE0716
      EQUIVALENCE (XSAM,V),(A,MAT,X)                                    TYPE0717
C     THIS ROUTINE PRINTS OUT THE PROBABILITIES OF TYPE MEMBERSHIP FOR ETYPE0718
C    1ACH OBSERVATION                                                   TYPE0719
    1 FORMAT(1H1,9X,32HPROBABILITIES OF TYPE MEMBERSHIP//7X,7I8)        TYPE0720
    2 FORMAT(I9,7F8.3)                                                  TYPE0721
      WRITE OUTPUT TAPE 2,1,(I,I=1,IR)                                  TYPE0722
      KB=0                                                              TYPE0723
      KRMN=0                                                            TYPE0724
      REWIND 4                                                          TYPE0725
    3 CALL BLOCK(NX,3000,KRMN,KST,KEND,LONG)                            TYPE0726
      READ TAPE 4, ((X(I,J),I=1,MX),J=1,LONG)                           TYPE0727
      DO 6 K=1,LONG                                                     TYPE0728
      DO 4 I=1,MX                                                       TYPE0729
    4 Z(I+1)=X(I,K)                                                     TYPE0730
      CALL DENSITY                                                      TYPE0731
      KB=KB+1                                                           TYPE0732
      DO 5 I=2,IR1                                                      TYPE0733
    5 G(I)=G(I)*PERS(I)                                                 TYPE0734
    6 WRITE OUTPUT TAPE 2,2,KB,(G(I),I=2,IR1)                           TYPE0735
      IF(KRMN)7,7,3                                                     TYPE0736
    7 RETURN                                                            TYPE0737
      END(0,1,0,0,0)                                                    TYPE0738
```

```
      SUBROUTINE BLOCK(NX,LBLK,KRMN,KST,KEND,LONG)                      TYPE0739
C     THIS ROUTINE COMPUTES NUMBERS USEFUL IN THE CONTROL OF THE INPUT  TYPE0740
C     AND OUTPUT OF LISTS OF LENGTH NX IN BLOCKS OF LENGTH LBLK.        TYPE0741
C     KRMN=NUMBER OF ITEMS REMAINING IN THE LIST                        TYPE0742
C     LONG=LENGTH OF CURRENT BLOCK                                      TYPE0743
C     KST AND KEND ARE THE STARTING AND ENDING INDEXES FOR THE ITEMS IN TYPE0744
C     THE CURRENT BLOCK.                                                TYPE0745
      IF (KRMN)2,1,2                                                    TYPE0746
    1 KRMN=NX                                                           TYPE0747
      KEND=0                                                            TYPE0748
    2 LONG=LBLK                                                         TYPE0749
      KST=KEND+1                                                        TYPE0750
      IF(KRMN-LBLK)3,4,4                                                TYPE0751
    3 LONG=KRMN                                                         TYPE0752
    4 KEND=KEND+LONG                                                    TYPE0753
      KRMN=KRMN-LONG                                                    TYPE0754
      RETURN                                                            TYPE0755
      END                                                               TYPE0756
```

```
      SUBROUTINE MATINV(A,N,B,M,DETERM)                                 TYPE0757
      DIMENSION IPIVOT(126),A(126,126),B(126,1),INDEX(126,2),PIVOT(126) TYPE0758
      EQUIVALENCE (IROW,JROW), (ICOLUM,JCOLUM), (AMAX, T, SWAP)         TYPE0759
C                                                                       TYPE0760
C          PROGRAMMED BY BURTON S. GARBOW, ARGONNE NATIONAL LABORATORY, TYPE0761
C     AND REPORTED IN IBM 704-709 SHARE LIBRARY AS AN F402.             TYPE0762
C          THIS SUBROUTINE COMPUTES THE INVERSE AND DETERMINANT OF      TYPE0763
C     MATRIX A, OF ORDER N, BY THE GAUSS-JORDAN METHOD. A-INVERSE       TYPE0764
C     REPLACES A, AND THE DETERMINANT OF A IS PLACED IN DETERM. IF      TYPE0765
C     M = 1 THE VECTOR B CONTAINS THE CONSTANT VECTOR WHEN MATINV IS    TYPE0766
C     CALLED, AND THIS IS REPLACED WITH THE SOLUTION VECTOR.  IF M = 0, TYPE0767
C     NO SIMULTANEOUS EQUATION SOLUTIONS ARE CALLED FOR, AND B IS NOT   TYPE0768
C     PERTINENT. N IS NOT TO EXCEED 50.                                 TYPE0769
C     A, N, B, M, AND DETERM IN THE ARGUMENT LIST ARE DUMMY VARIABLES.  TYPE0770
C                                                                       TYPE0771
C                                                                       TYPE0772
C     INITIALIZATION                                                    TYPE0773
   10 DETERM=1.0                                                        TYPE0774
   15 DO 20 J=1,N                                                       TYPE0775
   20 IPIVOT(J)=0                                                       TYPE0776
   30 DO 550 I=1,N                                                      TYPE0777
C     SEARCH FOR PIVOT ELEMENT                                          TYPE0778
   40 AMAX=0.0                                                          TYPE0779
   45 DO 105 J=1,N                                                      TYPE0780
   50 IF (IPIVOT(J)-1) 60, 105, 60                                      TYPE0781
   60 DO 100 K=1,N                                                      TYPE0782
   70 IF (IPIVOT(K)-1) 80, 100, 740                                     TYPE0783
   80 IF (ABSF(AMAX)-ABSF(A(J,K))) 85, 100, 100                         TYPE0784
   85 IROW=J                                                            TYPE0785
   90 ICOLUM=K                                                          TYPE0786
   95 AMAX=A(J,K)                                                       TYPE0787
  100 CONTINUE                                                          TYPE0788
  105 CONTINUE                                                          TYPE0789
  110 IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1                                   TYPE0790
C     INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL                 TYPE0791
  130 IF (IROW-ICOLUM) 140, 260, 140                                    TYPE0792
  140 DETERM=-DETERM                                                    TYPE0793
  150 DO 200 L=1,N                                                      TYPE0794
  160 SWAP=A(IROW,L)                                                    TYPE0795
  170 A(IROW,L)=A(ICOLUM,L)                                             TYPE0796
  200 A(ICOLUM,L)=SWAP                                                  TYPE0797
```

```
  205 IF(M) 260, 260, 210                                               TYPE0798
  210 DO 250 L=1, M                                                     TYPE0799
  220 SWAP=B(IROW,L)                                                    TYPE0800
  230 B(IROW,L)=B(ICOLUM,L)                                             TYPE0801
  250 B(ICOLUM,L)=SWAP                                                  TYPE0802
  260 INDEX(I,1)=IROW                                                   TYPE0803
  270 INDEX(I,2)=ICOLUM                                                 TYPE0804
  310 PIVOT(I)=A(ICOLUM,ICOLUM)                                         TYPE0805
  320 DETERM=DETERM*PIVOT(I)                                            TYPE0806
C     DIVIDE PIVOT ROW BY PIVOT ELEMENT                                 TYPE0807
  330 A(ICOLUM,ICOLUM)=1.0                                              TYPE0808
  340 DO 350 L=1,N                                                      TYPE0809
  350 A(ICOLUM,L)=A(ICOLUM,L)/PIVOT(I)                                  TYPE0810
  355 IF(M) 380, 380, 360                                               TYPE0811
  360 DO 370 L=1,M                                                      TYPE0812
  370 B(ICOLUM,L)=B(ICOLUM,L)/PIVOT(I)                                  TYPE0813
C     REDUCE NON-PIVOT ROWS                                             TYPE0814
  380 DO 550 L1=1,N                                                     TYPE0815
  390 IF(L1-ICOLUM) 400, 550, 400                                       TYPE0816
  400 T=A(L1,ICOLUM)                                                    TYPE0817
  420 A(L1,ICOLUM)=0.0                                                  TYPE0818
  430 DO 450 L=1,N                                                      TYPE0819
  450 A(L1,L)=A(L1,L)-A(ICOLUM,L)*T                                     TYPE0820
  455 IF(M) 550, 550, 460                                               TYPE0821
  460 DO 500 L=1,M                                                      TYPE0822
  500 B(L1,L)=B(L1,L)-B(ICOLUM,L)*T                                     TYPE0823
  550 CONTINUE                                                          TYPE0824
C     INTERCHANGE COLUMNS                                               TYPE0825
  600 DO 710 I=1,N                                                      TYPE0826
  610 L=N+1-I                                                           TYPE0827
  620 IF (INDEX(L,1)-INDEX(L,2)) 630, 710, 630                          TYPE0828
  630 JROW=INDEX(L,1)                                                   TYPE0829
  640 JCOLUM=INDEX(L,2)                                                 TYPE0830
  650 DO 705 K=1,N                                                      TYPE0831
  660 SWAP=A(K,JROW)                                                    TYPE0832
  670 A(K,JROW)=A(K,JCOLUM)                                             TYPE0833
  700 A(K,JCOLUM)=SWAP                                                  TYPE0834
  705 CONTINUE                                                          TYPE0835
  710 CONTINUE                                                          TYPE0836
  740 RETURN                                                            TYPE0837
      END                                                               TYPE0838
      END(0,1,0,0,0)                                                    TYPE0839
```

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* U. S. NAVAL PERSONNEL RESEARCH ACTIVITY SAN DIEGO, CALIFORNIA, 92152	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP

3. REPORT TITLE

A COMPUTER PROGRAM FOR THE MAXIMUM LIKELIHOOD ANALYSIS OF TYPES

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

FINAL

5. AUTHOR(S) *(Last name, first name, initial)*

WOLFE, JOHN H.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
MAY 1965	52	NONE

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO.	TECHNICAL BULLETIN 65-15
c.	9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
d.	

10. AVAILABILITY/LIMITATION NOTICES

Release to CFSTI authorized

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	CHIEF OF NAVAL PERSONNEL (Pers-A3) NAVY DEPARTMENT WASHINGTON, D. C., 20370

13. ABSTRACT

This report contains a description of a computer program for estimating the parameters of a mixture of multivariate normal distributions with unknown frequencies, means, and covariances. The basic equations for the procedure are presented for the first time here, with their derivation omitted. An example with the results of the computer printout is described for an artificially constructed mixture of three bivariate normal distributions. The method of using the program and the Fortran listing are detailed in this report.

DD FORM 1 JAN 64 1473 0101-807-6800

Security Classification

14. KEY WORDS	LINK A ROLE	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
Computer Techniques Maximum Likelihood Type Analysis Multivariate Analysis Cluster Analysis Multivariate Normal Distribution Mixtures						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through
____________________________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through
____________________________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through
____________________________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.