DANIEL F. MERRIAM, Editor

FORTRAN IV PROGRAM FOR THE GENERALIZED STATISTICAL DISTANCE AND ANALYSIS OF COVARIANCE MATRICES FOR THE CDC 3600 COMPUTER

By

R. A. REYMENT
University of Uppsala

HANS-AKE RAMDEN
University of Uppsala

and

W. J. WAHLSTEDT
Cities Service Oil Company

COMPUTER CONTRIBUTION 39
State Geological Survey
The University of Kansas, Lawrence
1969

in cooperation with the
American Association of Petroleum Geologists
Tulsa, Oklahoma

EDITORIAL STAFF

D.F. Merriam* Editor

Technical Editors

John C. Davis* Owen T. Spitz°

Associate Editors

John R. Dempsey
Richard W. Fetzner*
James M. Forgotson, Jr.*
John C. Griffiths
John W. Harbaugh*
R.G. Hetherington
Sidney N. Hockens*
J. Edward Klovan
William C. Krumbein*
R.H. Lippert
William C. Pearn+
Max G. Pitcher*
Floyd W. Preston
Walther Schwarzacher
Peter H.A. Sneath

Editor's Remarks

The "FORTRAN IV program for the generalized statistical distance and analysis of covariance matrices for the CDC 3600 computer", by R.A. Reyment, Hans-Ake Ramden, and W.J. Wahlstedt will be made available on magnetic tape for a limited time for $25.00 (U.S.). An extra charge of $10.00 will be made if punched cards are required.

An up-to-date list of COMPUTER CONTRIBUTIONS can be obtained by writing the Editor, COMPUTER CONTRIBUTIONS, Kansas Geological Survey, The University of Kansas, Lawrence, Kansas 66044, U.S.A.

COMPUTER CONTRIBUTIONS

Kansas Geological Survey
University of Kansas
Lawrence, Kansas

Computer Contribution

1. Mathematical simulation of marine sedimentation with IBM 7090/7094 computers, by J.W. Harbaugh, 1966 . . . . (out of print)
2. A generalized two-dimensional regression procedure, by J.R. Dempsey, 1966 . . . . $0.50
3. FORTRAN IV and MAP program for computation and plotting of trend surfaces for degrees 1 through 6, by Mont O'Leary, R.H. Lippert, and O.T. Spitz, 1966 . . . . . . $0.75
4. FORTRAN II program for multivariate discriminant analysis using an IBM 1620 computer, by J.C. Davis and R.J. Sampson, 1966 . . . . $0.50
5. FORTRAN IV program using double Fourier series for surface fitting of irregularly spaced data, by W.R. James, 1966 . . . . $0.75
6. FORTRAN IV program for estimation of cladistic relationships using the IBM 7040, by R.L. Bartcher, 1966 . . . . $1.00
7. Computer applications in the earth sciences: Colloquium on classification procedures, edited by D.F. Merriam, 1966 . . . . $1.00
8. Prediction of the performance of a solution gas drive reservoir by Muskat's Equation, by Apolonio Baca, 1967 . . . . $1.00

(continued on inside back cover)

---

*Active Member, °Associate Member, +Junior Member, American Association of Petroleum Geologists.

FORTRAN IV PROGRAM FOR GENERALIZED STATISTICAL DISTANCE AND ANALYSIS OF COVARIANCE MATRICES FOR THE CDC 3600 COMPUTER

by

R.A. Reyment, Hans-Ake Ramden and Warren J. Wahlstedt

ABSTRACT

Heterogeneity in sample covariance matrices, deriving from differences in the orientation of major axes of the ellipsoids of scatter, may be of common occurrence. The generalized test of equality of covariance matrices will give a significant result in instances where the scatter ellipsoids are (1) unequally inflated, although identicially oriented, (2) identically inflated but differently oriented and (3) a combination of these conditions. Equations to the major axes of a scatter ellipsoid of morphologic variables represent growth patterns in the variables. An approximate application of the asymptotic test developed by T.A. Anderson is used here to identify structure of the heterogeneity between two covariance matrices where such exists. The foregoing procedures are preliminary to a treatment of generalized distances in which the path taken by the computer program is decided by structure of the covariance matrices of the samples. Depending on the nature of the covariance matrices either the Mahanolobis' generalized distance is computed or the Anderson-Bahadur distance for heterogeneous covariance matrices. Tests of significance of the results are provided and the linear discriminant function coefficients produced as a by-product.

INTRODUCTION

The generalized statistical distance is probably the best known and most widely used of the multivariate techniques employed in taxonomic work. It is used also widely in other connections.

Calculations performed by this program may be conceived in terms of the geometric properties of two ellipsoids, in this situation ellipsoids of scatter. For the strict application of statistical theory involved, it is necessary that the variables be distributed normally. The ellipsoid of scatter associated with a multivariate normal distribution provides a representation of the variability of the population, directly analogous to the variance of univariate statistics. In univariate statistics one will wish to make sure that in a comparison of two statistical populations on the basis of samples drawn from them the variances of the variable being analyzed are statistically identical for both populations. One also will want to know if the variables are distributed normally. The same reasoning applies in multivariate analysis. It is not difficult to test for homogeneity of univariate variances but the multivariate analog is associated with certain drawbacks and difficulties.

If the variables are distributed as a multivariate normal, the shape of a three-dimensional plot of many points will approximate a football flattened equally on two opposite sides. For a two-sample statistical comparison to be valid one would require the footballs to be of the same size and to be oriented in the same direction with respect to all axes. It is possible to employ a large-sample test to ascertain whether each principal axis of the ellipsoid of one sample is collinear with the corresponding principal axis of the other ellipsoid. In the bivariate situation there are four categories of difference to be recognized. For three and higher dimensions the possibility of rotation about axes has to be taken into consideration.

How serious a matter is it if the covariance matrices are not equal? Experience shows that in most neontological investigations, heterogeneity in covariance matrices, although common, usually does not cause serious inaccuracies in the generalized distance. Geologic materials pose a more intricate problem. Owing to the action of geologic (nonbiologic) agencies, such as transport, considerable heterogeneity factors may be introduced into a sample. These factors may be of such an order as to cause serious inaccuracy in generalized distance computations.

Acknowledgments.- The original research work for this paper was done while R.A.R. was Visiting Research Scientist with the Kansas Geological Survey at The University of Kansas (1966-1967). The work was supported by Computing Grant 104104 at the University of Uppsala and Research Contracts 2320-26-7819G and -24-7540G.

GEOMETRICAL INTERPRETATION

Suppose that the N vectors of a sample, $x_1, \ldots x_N$ (N observation vectors), with mean vector $Ex = \mu$, and with covariance matrix $E(x - \mu)(x - \mu)' = \Sigma$ are multivariate normally distributed and form an ellipsoid.

If one considers two p-variate normally distributed populations with the mean vectors μ_1 and μ_2, not necessarily different (i.e., 0_1 and 0_2 in the diagrams of Figure 1 may be coincident), and the covariance matrices, Σ_1 and Σ_2, then if the test of equality of these matrices be applied (Kullback, 1959) and it be concluded, that Σ_1 and Σ_2, the following situation may prevail. In Figure 1a, the ellipsoid Ω_1 is merely a translation of ellipsoid Ω_2. Hence, AB = EF and CD = GH, and AB is parallel to EF.

If the homogeneity test for covariance matrices leads to the conclusion, that $\Sigma_1 \neq \Sigma_2$, then one of the following conditions may prevail (for two dimensions). Figure 1b has AB parallel to EF and CD parallel to GH; EF > AB and GH > CD. Hence, ellipsoid Ω_2 is a translation of ellipsoid Ω_1 with magnification. Figure 1c has AB = EF, CD = GH, AB is not parallel to EF and CD is not parallel to GH. Ellipsoids Ω_1 and Ω_2 have the same shape, but are rotated in relation to each other. Figure 1d has AB $\neq$ EF, CD $\neq$ GH, AB is not parallel to EF and CD is not parallel to GH. Ellipsoids Ω_1 and Ω_2 are thus differently inflated and their axes are rotated in relation to each other.

For three or more dimensions, the situation becomes more complicated. Figure 1e illustrates the position for three dimensions. Here, two axes of ellipsoids Ω_1 and Ω_2 are parallel to each other, AB is parallel to EF, but not the remaining axes, owing to rotation about AB and EF. Hense, CD is not paralled to GH and IJ is not parallel to LK. In this example, it has been taken that the ellipsoids have the same shape, thus, AB = EF, CD = GH, and IJ = KL. Figure 1f indicates the positions if the second axes of Ω_1 and Ω_2 are parallel and the first and third axes are rotated about the second axes. A numerical example will illustrate the foregoing. Consider the sample covariance matrices ($N_1 = N_2 = 51$), S_1, S_2

$$S_1 = \begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix}, \; S_2 = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}, \; S = S_1 + S_2 = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}.$$

Here, S is the pooled covariance matrix. Both of the samples have the same mean vector, $\bar{x}_1 = \bar{x}_2 = (1,1)'$. The determinants of these matrices are, det $S_1 = 1$, det $S_2 = 1$, det $S = 2$. The generalized test for homogeneity of covariance matrices, in the form presented in Kullback (1959, p. 317), is

$$2I(H_1 : H_2(*)) = n_1 \log_e(\det S_1) + n_2 \log_e(\det S/\det S_2), \quad (1)$$

where $nS = n_1S_1 + n_2S_2$ and $n = n_1 + n_2$ and $n_i = N_i - 1$ $(i = 1,2)$. This is approximately distributed as χ^2 with $p(p+1)/2$ degrees of freedom, where p is the number of variables. A better approximation available, however, for only a few dimensions, is given by the B-distribution (Kullback, 1959, p. 317).

Applying (1) to the present example, one has,

$$B^2 = 50 \log_e (2/1) + 50 \log_e (2/1) = 69.315,$$

which is significant. The problem of homogeneity in covariance matrices is given detailed study in Chapter 10 of Anderson (1958).

Inasmuch as the determinants of the covariance matrices S_1 and S_2 are the same, both of these must have the same volume, because $\det A = (n-1)^p \det S$ is the squared volume of a parallelotope (Anderson, 1958, p. 167). (Here, A is the matrix of sums and products, S the sample covariance matrix, n the sample size and p the number of variables.) S_1 has the eigenvalues $d_1 = 2.618$, $d_2 = 0.382$ and the normalized corresponding eigenvectors are

$$b_1 = \begin{bmatrix} .8507 \\ -.5257 \end{bmatrix}, \; b_2 = \begin{bmatrix} .5257 \\ .8507 \end{bmatrix}.$$

Matrix S_2 has the eigenvalues $g_1 = 2.618$, $g_2 = 0.382$ and the corresponding, normalized eigenvectors are

$$c_1 = \begin{bmatrix} .8507 \\ .5257 \end{bmatrix}, \; c_2 = \begin{bmatrix} -.5257 \\ .8507 \end{bmatrix}.$$

The ellipsoids have the same shape, but are orientated in relation to each other at an angle of

$$\cos \nu = 0.44733, \; \nu = 63^\circ 25'.$$

APPROXIMATE TEST OF ORIENTATION OF ELLIPSOIDS

In order to disclose relative heterogeneity in the orientation of the axes of two ellipsoids of scatter, if the generalized test (1) indicates $\Sigma_1 \neq \Sigma_2$, it is sufficient to demonstrate that the first two axes are parallel to each other in order to localize the inequality in the covariance matrices to condition (b) of Figure 1. This demonstrates that the heterogeneity is due solely to a greater degree of scatter in the points of the observational vectors forming the ellipsoids.

Some interest may attach to comparing the relative differences in the inflations of ellipsoids Ω_1 and Ω_2. This may be gauged approximately by comparing the eigenvalues of matrices Σ_1 and Σ_2. Let

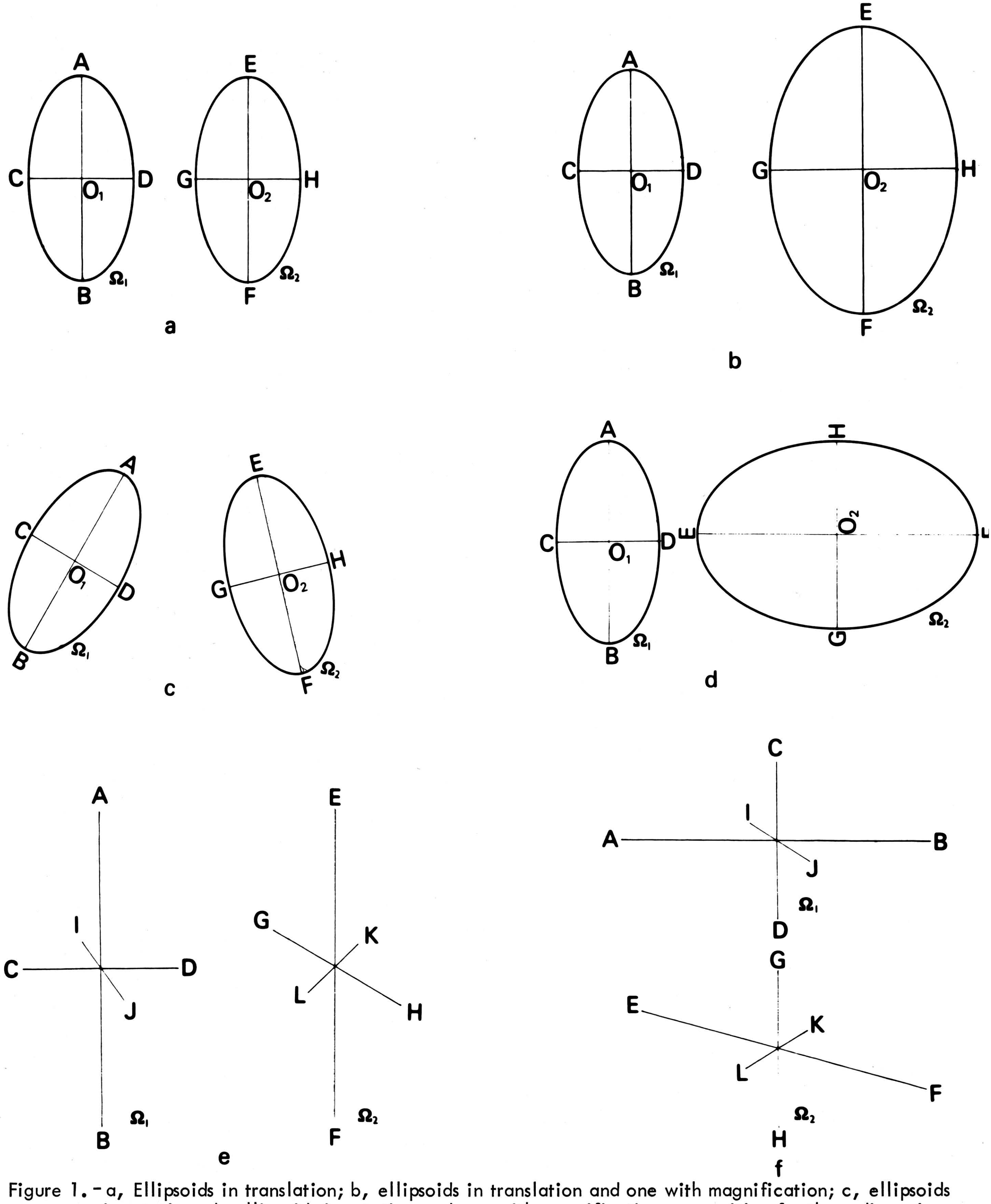


Figure 1. - a, Ellipsoids in translation; b, ellipsoids in translation and one with magnification; c, ellipsoids in rotation; d, ellipsoids in rotation and one with magnification; e, position for three dimensions; f, three-dimensional position with two axes rotated.

$\delta_1 > \ldots > \delta_p > 0$ be the p eigenvalues of the positive definite matrix Σ_1; then $|\Sigma_1 - \delta I| = 0$, and $\gamma_1, \ldots, \gamma_p$ are the corresponding normalized eigenvectors which satisfy $\Sigma_1 \gamma_i$ and $\delta_i' \gamma$ and $\gamma_i' \gamma_i = 1$. Likewise let $\lambda_1 > \ldots > \lambda_p > 0$ be the p eigenvalues of the positive definite matrix Σ_2; $|\Sigma_2 = \lambda I| = 0$ and $\beta_1, \ldots, \beta_p$ the corresponding normalized eigenvectors which satisfy $\Sigma_2 \beta_i = \lambda_i \beta_i$ and $\beta_i' \beta_i = 1$. If the δ_i and λ_i are different, $\gamma_i' \gamma_i = 0$ and $\beta_i' \beta_i - 0$. The length of the i-th principal axis of the ellipsoid of scatter is related to the magnitude of the corresponding eigenvalue, in the sense that it constitutes the variance of the i-th principal component.

For the first eigenvalue and eigenvectors of covariance matrix Σ_1 we have that

$$\delta_1 \gamma_1' \Sigma_1^{-1} \gamma_1 = \delta_1 \delta_1^{-1} = 1,$$

and,

$$(1/\delta_1)\gamma_1' \Sigma_1 \gamma_1 = (1/\delta_1)\delta_1 = 1,$$

and similarly for Σ_2. If, say, the first eigenvector of Σ_2, β_1, has the same direction cosines as the first eigenvector of Σ_1, γ_1, then $\delta_1 \beta_1' \Sigma_1^{-1} \beta_1 = 1$ and $(1/\delta_1)\beta_1' \Sigma_1 \beta_1 = 1$. Anderson (1963, p. 144) has put forward a procedure for testing the null hypothesis, that a given eigenvector (principal component) is a specified vector. The procedure is based on the limiting normal distribution possessed by $\sqrt{n}(c_1 - \gamma_1)$, where γ_1 is the first eigenvector of a population covariance matrix and c_1 a sample estimate thereof, based on a sample of size $(n+1)$. As an approximate test of the hypothesis that a given eigenvector is the i-th eigenvector of a large sample estimate of a covariance matrix, it is suggested, that the following version of Anderson's test criterion might be applicable

$$n(d_i b_i' S_1^{-1} b_i + (1/d_i) b_i' S_1 b_i - 2). \qquad (2)$$

Here, (n+1) is the sample size of S_1, the estimate of Σ_1, d_i is the sample estimate of δ_i and b_i that of β_i, the latter being based on a large sample. If $\Sigma_1 = \Sigma_2$, then $\gamma_i = \beta_i$. This criterion has, of course, more variables than the original form given by Anderson and must, therefore, be more susceptible to fluctuations.

Test (1) may then be computed. The quadratic forms of (2) may be calculated separately and the test worked out as a subsequent step. The first test criterion is asymptotically distributed as χ^2 with p (p+1)/2 degrees of freedom, where p is the number of variables. The Anderson test criterion is distributed as χ^2 with (p-1) degrees of freedom. In judging the significance of a result, it seems advisable to take the 1% level as the highest level of risk, rather than the customary 5%.

GENERALIZED DISTANCE CALCULATIONS

The Mahanalobis' generalized statistical distance squared is yielded by the relationship

$$D^2 = d'S^{-1}d, \qquad (3)$$

where d is the vector of differences between the sample mean vectors of the two samples and S is the pooled covariance matrix for the two samples. The statistical significance of this result is assessed by means of the Hotelling T^2, which is related to D^2 as follows

$$T^2 = D^2 \frac{N_1 N_2}{N_1 + N_2}, \qquad (4)$$

where N_1 and N_2 are the sizes of the two samples involved. The final step in finding the significance of the generalized distance is the calculation of the variance ratio (F) by means of the following formula

$$F = \frac{T^2 (N_1 + N_2 - p - 1)}{p(N_1 + N_2 - 2}. \qquad (5)$$

Here p represents the number of variables. Anderson and Bahadur (1962) considered a generalized statistical distance for unequal covariance matrices of the following form

$$D_H = \frac{2b'd}{(b'S_1 b) + (b'S_2 b)}. \qquad (6)$$

Here, d is again the difference mean vector, b is an analog of the vector of coefficients of the Fisherian discriminant function and S_1 and S_2 are the respective samples covariance matrices. If the covariance matrices are equal this formula reduces to formula (3). Formula (6) is not connected with a test of significance. In order to obtain a significance test based on T^2 where the covariance matrices are unequal one may proceed by means of a distance method proposed by Reyment (1962).

DESCRIPTION OF PROGRAM

Operating Instructions

The following INPUT cards are required.

Card 1: The number of JOBS occurring in the current sequence in FORMAT (3X, I2).

Card 2: The following information is punched on this card in FORMAT (25I3), in the order given.

Column 1-3. M = number of variables.
4-6. NA = size of the larger sample.
7-9. NB = size of the smaller sample.
10-12. MATRIX = 0 if observation vectors are read in and = 1 for data in matrix form.
13-15. L = 1 for cross-products matrices and = 0 for covariance.
16-18. LOGDEC = 1 if the data are transformed to the base ten logarithms and = 0 for raw data.
19-21. LDISCR = 1 if the Mahanalobis' generalized distance is computed in addition to the heterogeneous distance if this branch has been chosen by the course of the program, otherwise put = 0.
22-24. INVST (1)
25-27. INVST (2)
28-30. INVST (3) } intervals for histogram subroutines
31-33. KREYD. Supply here the number of times program REYDST is to operate.

Card 3: The TITLE card.
Card 4: The FORMAT card for reading the data.
Card 5 and subsequent cards: either the data x(I) in matrix form in accordance with the format given by Card 4 or the data as vectors of observations. Sample A is always to be located before sample B, the smaller sample. Thus, for the matrix option, matrices A (I, J) and B (I, J) and vectors SA (I) and SB (I) are read in at this point. The larger sample should be greater than 100 observations for the theory used in this program to have full validity.

Mainline Program ORNTDIST

Calculations either are made directly on the covariance matrices (or cross-products matrices) of samples A and B and the corresponding mean vectors or such are computed from the observations. The main program produces the basic information required for the statistical tests and calculations. In addition to the covariance matrices and mean vectors the program computes the eigenvalues and eigenvectors of matrices A and B as well as those of matrix C, obtained by pooling A and B.

Subroutines BAHADU and HETDST

The subroutines perform the heterogeneous distance calculations of Anderson and Bahadur (1962). The value of the expression

$$\frac{2b'd}{\sqrt{(b'S_1b)} + \sqrt{(b'S_2b)}}$$

is found, where d is the difference vector between sample means, S_1 and S_2 are the sample covariance matrices and b is a vector estimated iteratively in the two subroutines by trial and error solutions of

$$b = \{tS_1 + (1-t)S_2\}^{-1}d, \qquad (7)$$

t being a scalar term located between zero and unity. The value of t, initially guessed, is improved iteratively from

$$b'\{t^2S_1 - (1-t)^2S_2\}b = 0. \qquad (8)$$

Convergence is checked for in HETDST. For each iteration the output from BAHADU consists of the heterogeneous generalized distance, its square; t and the iteration factors are given from HETDST. The analog of the coefficients of the discriminant function SB(I) also are printed out.

Subroutine DISTFN

The subroutine computes the Mahanalobis' generalized distance for homogeneous covariance matrices. The output consists of the discriminant function coefficients, the generalized distance, its square, Hotelling's T^2, the corresponding variance ratio and degrees of freedom for it. The calculations are based essentially on the expansion of a quadratic form.

Subroutine HAFCOV

The subroutine computes a heterogeneous generalized distance by averaging the covariance matrices S_1 and S_2. The calculations then proceed as in DISTFN. The output comprises a generalized distance and its square. HAFCOV provides a valuable check on the calculations in other parts of the program but it is not a vital section and may be excluded from the program by punching C in column 1 of CALL HAFCOV in subroutine HETDST.

Subroutine ORIENT

The subroutine is called from the main program. It operates on eigenvectors of the comparison matrix, in conjunction with the reference matrix, to test the significance in differences in orientation of the scatter ellipsoids. The computations are based on formula (2). The subroutine does not treat eigenvectors whose eigenvalues are less than one percent of the trace.

Subroutine REYDST

The role of the subroutine is to calculate a formal test of significance for the differences in means if the covariance matrices are not equal. It is based on an iterative randomization procedure and is operated in conjunction with subroutines RANKRT and CLEAR.

REYDST is called from HETDST. The details of the calculations are the same as for DISTFN. The subroutine is erratic if deviations from multivariate normality occur and tends to overestimate the generalized distance. This becomes particularly noticeable for many dimensions. The calculations are time-consuming owing to the randomization procedure. For four variables and 30 randomizations the entire program took 5.36 minutes of which about 85% was taken by REYDST. Clearly, the user of the program may consider this to be too uneconomical to make the significance test worthwhile. The instruction KREYD if put = 0 causes REYDST to be omitted. Otherwise the number of iterations required is specified here.

Subroutine RANKRT

This selects cards randomly from the stored decks of data cards. The use of RANKRT requires two random access (RA) control cards on the CDC 3600 - e.g.
{7
{9EQUIP, 10 = RA(10000). CLEAR must be called to initiate the use of RANKRT.

Subroutine HOUSER

This is a standard subroutine for computing eigenvalues and eigenvectors of a square symmetric matrix by the Householder method. The matrix is reduced to tridiagonal form, the eigenvalues of the tridiagonal form are computed then by the Sturm sequence process and finally the eigenvectors are obtained by the Wilkinson inverse iteration method.

Subroutine PUTMAT

This subroutine writes a matrix on a scratch unit as one logical record. The matrices A, B, and C are stored in this manner.

Subroutine GETMAT

This reads one of the matrices A, B, or C from the scratch unit generated by PUTMAT.

Subroutine MATOUT

This prints out a matrix on the printer unit. Columns and rows are marked.

Subroutine JZUP2

This subroutine carries out a substitution of the original measurements, or the logarithmically transformed measurements, into the vector of discriminant function coefficients. The values are plotted as a histogram by PLTHS1.

Subroutine MATINV

This is a standard subroutine for carrying out the inversion of square, symmetric matrices.

Subroutine DISTRB

This calculates the coefficients of skewness and kurtosis for each of the variables for both samples. The large-sample standard deviations of each coefficient are found and an approximate t-value calculated. The foregoing results as well as the means and standard deviations for each variable are printed out.

Subroutine PLTHS1

This is a standard subroutine of the Computation Center of The University of Kansas for printing histograms. It is called from the subroutines DISTFN, and HETDST through JZUP2. The resulting histogram of substituted values gives a valuable pictorial representation of the efficiency of the discriminant function in separating the two samples.

NOTE ON PROBLEM EXAMPLE

The example used to illustrate the program consists of measurements on male and female grasshoppers from Gotland (<u>Omocestus haemorrhoidalis L</u>). The variables are X_1 = length of hind femur, X_2 = pronotal length, X_3 = elytron length and X_4 = the least width between the ridges of the pronotus. The example used here is an entomological one, but any type of normally distributed multivariate data are amenable to distance analysis by this program, and it may equally as well be applied to sedimentologic, anthropometric, geochemical and sociologic studies just to mention a few possibilities (Reyment, 1969).

REFERENCES

Anderson, T.W., 1958, Introduction to multivariate statistical analysis: John Wiley & Sons, New York, 374 p.

Anderson, T.W., 1963, Asymptotic theory for principal components: Ann. Math. Statist., v. 34, p. 122-148.

Anderson, T.W., and Bahadur, R.R., 1962, Classification into two multivariate normal distributions with different covariance matrices: Ann. Math. Statist., v. 33, p. 420-431.

Kullback, S. 1959, Information theory and statistics: John Wiley & Sons, New York, 395 p.

Reyment, R.A., 1962, Observations on homogeneity of covariance matrices in paleontologic biometry: Biometrics, v. 18, p. 1-11.

Reyment, R.A., 1969, A multivariate paleontological growth problem: Biometrics, v. 25, p. 1-8.

APPENDIX A. -Program Listing.

```
LISTABLE OUTPUT FROM SEQUENCE    003027
UDAC DRUM SCOPE 2,020, TWO DRUMS
SEQUENCE NUMBER 003027 STARTED AT TIME  063358  DATED  28/03-69
JOB,104104,REYMENT,3, BUD PALEONTOLOGEN
DEMAND,23000
EQUIP,10=RA(25000)
EQUIP,11=RA(25000)
EQUIP,26=(104104 ORNTDIST,,,999)
FTN,L,X=26,*

      PROGRAM ORNTDIST
C                        LIST OF SUBROUTINES
C     HOUSER - COMPUTES EIGENVALUES AND EIGENVECTORS
C     ORIENT - COMPARES ORIENTATIONS OF EIGENVECTORS
C     HETDST - COMPUTES ANDERSON/BAHADUR DISTANCE
C     BAHADU - PERFORMS STEPS IN THE ITERATIVE CALCULATION OF HETDST
C     REYDST - COMPUTES SIGNIFICANCE OF A HETEROGENEOUS DISTANCE
C     HAFCOV - GIVES A HETEROGENEOUS DISTANCE BY AVERAGING COVARIANCES
C     DISTFN - FINDS THE MAHANOLOBIS DISTANCE
C     RANKRT                   MATINV                         JZUP2
C     PUTMAT                   GETMAT                         MATOUT
C     PLTHS1 - PLOTS HISTOGRAMS
C     DISTRB - TESTS THE DISTRIBUTIONS OF EACH OF THE VARIABLES
C
C***********************************************************************
C
C     EXPLANATION OF CONTROL CARDS
C
C     CARD 1     JOBS     = NUMBER OF JOBS BEING RUN IN SEQUENCE
C     CARD 2
C       COL 1-3    M = NUMBER OF VARIABLES (MAXIMUM = 45)
C       COL 4-6    NA = IS SIZE OF THE LARGER SAMPLE (MAXIMUM=500)
C       COL 7-9    NB = SIZE OF SMALLER SAMPLE (MAXIMUM=500)
C       COL 10-12  MATRIX = 0 FOR SERIES OF MEASUREMENTS AND 1 FOR
C                  DATA READ IN IN MATRIX FORM ,I.E. PARTIALLY PROCESSED)
C       COL 13-15  L = 1 FOR INPUT DATA AS CROSS-PRODUCTS MATRICES
C                  L = 0 FOR COVARIANCE MATRICES AND SERIES OF
C                  MEASUREMENTS
C       COL 16-18  LOGDEC = 1 IF LOGARITHMIC TRANSFORMATION OF RAW DATA
C                  REQUIRED AND 0 IF THE DATA ARE NOT TO BE TRANSFORMED
C       COL 19-21  LDISC = 1 IF THE NORMAL FORM OF THE GENERALIZED
C                  DISTANCE IS TO BE COMPUTED IN ADDITION TO THE
C                  DECISION MADE BY THE PROGRAM - THIS IS A USEFUL
C                  CONTROL STEP
C       COL 22-24, 25-27, 28-30   INVST = NUMBER OF INTERVALS FOR EACH
C                  HISTOGRAM,  IF NO SELECTION ALTERNATIVELY ZEROES
C                  PROGRAM PICKS INTERVAL OF TEN
C       COL 31-33  KREYD = NUMBER OF TIMES IN THE REYDST=LOOP
C      CARD 3    THE TITLE CARD WITH THE HEADING IN 72 SPACES
C      CARD 4    THE VARIABLE FORMAT CARD FOR THE DATA
C      CARD 5    THE DATA IN ONE OF THE PRESELECTED FORMS
C      THE LARGER SAMPLE MUST ALWAYS BE READ FIRST
      DIMENSION E(45,45),F(45,45),AEGVAL(45),BEGVAL(45),CEGVAL(45),
     1 TDATA(30),FMT(12),TITLE(12)
      COMMON A(45,45),B(45,45),C(45,45),SA(45),SB(45),SC(45),DIFF(45),
     1 T(45),X(45),Z(45),M,NA,NB,K,MW,MATRIX,LDISCR,LOGDEC,INVST(3),IT,
     2 MATPOS,KREYD
```

```
      COMMON /UNITS/ LIN,LUT,MTA,MTB,MTMABC,MTARA,MTBRA
      DATA(TDATA= 0.0,9.21,11,34,13.28,15.09,16.81,18.48,20.09,21.67,23.
     121,24.72,26.22,27.69,29.14,30.58,32.00,33.41,34.81,36.19,37.57,38.
     293,40.29,41.64,42.98,44.31,45.64,46.96,48.28,49.59,50.89)
C=F IV- DATA TDATA/.0,9.21,11,34,13.28,15.09,16.81,18.48,20.09,21.67,23.
C    121,24.72,26.22,27.69,29.14,30.58,32.00,33.41,34.81,36.19,37.57,38.
C    293,40.29,41.64,42.98,44.31,45.64,46.96,48.28,49.59,50.89/
C
C
C======= LIN IS THE STANDARD INPUT UNIT (CARD READER)
      LIN=60
C======= LUT IS THE STANDARD OUTPUT UNIT (PRINTER)
      LUT=61
C======= MTA,MTB,MTMABC,MTARA AND MTBRA ARE SCRATCH UNITS
C======= ******** SCRATCH TAPE USAGE ********
C======= TAPE MTA CONTAINS SAMPLE A
      MTA=51
C======= TAPE MTB CONTAINS SAMPLE B
      MTB=52
C======= MTMABC CONTAINS MATRICES A,B AND C IN THIS ORDER,ONE PER RECORD
      MTMABC=50
C======= RANDOM ACCES UNITS ARE USED ONLY IN KEYDST AND RANKRT
C======= MTARA IS A RANDOM ACCESS UNIT WHERE SAMPLE A IS STORED
      MTARA=10
C======= MTBRA IS A RANDOM ACCESS UNIT WHERE SAMPLE B IS STORED
      MTBRA=11
      READ (LIN,380) JOBS
      KLOBS = 0
      MW=45
C======= HERE BEGINS A NEW CYCLE
 1111 CONTINUE
      REWIND MTA
      REWIND MTB
      REWIND MTMABC
      READ (LIN,3) M,NA,NB,MATRIX,L,LOGDEC,LDISCR,INVST,KREYB
      IF (M .LE. 0) CALL EXIT
      IF (M .LE. MW .AND. NA .LE. 500 .AND. NB .LE. 500) GO TO 33
      WRITE (LUT ,133) M,NA,NB
   33 DO 10101 IX = 1,3
10101 IF(INVST(IX).LE.0)INVST(IX) = 10
      DO 1620 I=1,M
      SA(I)=0.0
      SB(I)=0.0
      DO 1620 J=1,M
      A(I,J)=0.0
      B(I,J)=0.0
      C(I,J)=0.0
 1620 CONTINUE
      WRITE (LUT,207)
      IF (LOGDEC .EQ. 1) WRITE (LUT,490)
      EM=M
      ENA=NA
      ENB=NB
      READ (LIN,1) TITLE
      WRITE (LUT,333) TITLE
      READ (LIN,1) FMT
      WRITE (LUT,400) FMT
      IF(MATRIX.GT.0) GO TO 5000
      WRITE (LUT,710) NA,NB
      WRITE (LUT,5100)
```

```
      NAB=NA
      DO 6009 IREP=1,2
C======= WHEN IREP .EQ. 1  STORE SAMPLE A ON MTA
C======= WHEN IREP .EQ. 2  STORE SAMPLE B ON MTB
      LUN=MTA
      IF (IREP .EQ. 2) LUN=MTB
      DO 6008 IV=1,NAB
      READ (LIN,FMT) (X(IX),IX=1,M)
      WRITE (LUT,5020) (X(IX),IX=1,M)
      IF (LOGDEC .EQ. 0) GO TO 5051
      DO 5050 IX=1,M
 5050 X(IX)=ALOG10(X(IX))
 5051 WRITE (LUN) (X(IX),IX=1,M)
  625 CONTINUE
  627 IF(IREP.EQ.2) GO TO 635
      DO 626 I=1,M
      SA(I)=SA(I)+X(I)
      DO 626 J=1,M
  626 A(I,J)=A(I,J)+X(I)*X(J)
      GO TO 6008
  635 DO 628 I=1,M
      SB(I)=SB(I)+X(I)
      DO 628 J=1,M
  628 B(I,J)=B(I,J)+X(I)*X(J)
 6008 CONTINUE
      IF(IREP.EQ.2) GO TO 6009
      NAB=NB
      WRITE (LUT,5105)
 6009 CONTINUE
  655 DO 660 I=1,M
      DO 660 J=1,M
      A(I,J)=(A(I,J)-SA(I)*SA(J)/ENA)/(ENA-1.)
  660 B(I,J)=(B(I,J)-SB(I)*SB(J)/ENB)/(ENB-1.)
      DO 199 IX=1,M
      SA(IX)=SA(IX)/ENA
  199 SB(IX)=SB(IX)/ENB
      CALL DISTRB
      GO TO 1117
 5000 WRITE (LUT,5100)
      DO 10 I=1,M
   10 READ (LIN,FMT) (A(I,J),J=1,M)
      CALL MATOUT (A)
      WRITE (LUT,5105)
      DO 30 I=1,M
   30 READ (LIN,FMT) (B(I,J),J=1,M)
      CALL MATOUT (B)
      READ (LIN,FMT) (SA(IX),IX=1,M)
      READ (LIN,FMT) (SB(IX),IX=1,M)
      IF(L.LE.0) GO TO1117
 1116 DO 1121 I=1,M
      DO 1121 J=1,M
 1121 A(I,J)=A(I,J)/(ENA-1.)
      DO 1122 I=1,M
      DO 1122 J=1,M
 1122 B(I,J)=B(I,J)/(ENB-1.)
 1117 DO 40 I=1,M
      DO 40 J=1,M
   40 C(I,J)=((ENA-1.)*A(I,J)+(ENB-1.)*B(I,J))/(ENA+ENB-2.)
      WRITE (LUT,1120)
      CALL MATOUT (A)
C     MATRIX A IS REFERENCE
```

```
      WRITE (LUT,1119)
      CALL MATOUT (B)
      WRITE (LUT,1118)
      CALL MATOUT (C)
      CALL PUTMAT(A)
      DO 5432 IX=1,M
 5432 SC(IX)=SB(IX)-SA(IX)
      WRITE (LUT,3333)
      WRITE (LUT,3333)
      CALL HOUSER (M,M,A,AEGVAL,1.,E)
      WRITE (LUT,716)
      WRITE (LUT,1492)
      WRITE (LUT,20) (AEGVAL(IX),IX=1,M)
      WRITE (LUT,1493)
      CALL MATOUT (E)
      DETA=0.
      DO 80 IX=1,M
      IF (AEGVAL(IX) .LE. 1.E-09) GO TO 561
   80 DETA=DETA+ALOG(AEGVAL(IX))
      CALL PUTMAT(B)
C     MATRIX  B  IS STORED
      WRITE (LUT,3333)
      CALL HOUSER (M,M,B,BEGVAL,1.,F)
      WRITE (LUT,1494)
      WRITE (LUT,20) (BEGVAL(IX),IX=1,M)
      WRITE (LUT,1495)
      CALL MATOUT (F)
      DETB=0.
      DO 100 IX=1,M
      IF (BEGVAL(IX) .LE. 1.E-09) GO TO 561
  100 DETB=DETB+ALOG(BEGVAL(IX))
      CALL PUTMAT(C)
      MATPOS=3
      WRITE (LUT,3333)
      WRITE (LUT,3333)
      CALL HOUSER (M,M,C,CEGVAL,1.,F)
      WRITE (LUT,1497)
      WRITE (LUT,20) (CEGVAL(IX),IX=1,M)
      WRITE (LUT,1498)
      CALL MATOUT (F)
      DETC=0.
      DO 120 IX=1,M
      IF (CEGVAL(IX) .LE. 1.E-09) GO TO 561
  120 DETC=DETC+ALOG(CEGVAL(IX))
      WRITE (LUT,90)
      WRITE (LUT,110)  DETA,DETB,DETC
      BSQ=(ENA-1.)*(DETC-DETA)+(ENB-1.)*(DETC-DETB)
      BETASQ=((2.*FLOAT(M)**3+3.*FLOAT(M)**2-FLOAT(M))/12.)*
     1(1./(ENA-1.)+1./(ENB-1.)-1./(ENA+ENB-2.))
      WRITE (LUT,716)
      WRITE (LUT,140) BSQ,BETASQ
      KT = M*(M+1)/2
      WRITE (LUT,151) KT
      TRACE = 0.
      DO 170 IX=1,M
  170 TRACE=TRACE+BEGVAL(IX)
      WRITE (LUT,180) TRACE
      DO 200 IX=1,M
  200 Z(IX)=(BEGVAL(IX)*100.)/TRACE
      WRITE (LUT,220) (Z(IX),IX=1,M)
```

```
      CALL GETMAT (B,2)
      DO 500 I=1,M
      DO 500 J=1,M
  500 A(I,J)=B(I,J)
C     MATRIX  A  IS FOR INVERSICN
      CALL MATINV (A,M,CBGVAL,0,DETERM,MW)
      KOUNT = 1
      WRITE (LUT,160)
      DO 230 KL = 1,M
      IF(Z(KL).LE,1.) GO TO 10000
      CALL ORIENT (E,B,M,X,A,KOUNT,ENB,BEGVAL)
      KOUNT = KOUNT + 1
  230 CONTINUE
10000 CALL GETMAT(A,1)
      CALL GETMAT(B,2)
      IF(KT.GT.30) GO TO 5551
      TEST = TDATA(KT)
      WRITE (LUT,10598) TEST
      GO TO 5552
 5551 TEST=0.5*(2.*(ENA+ENB-1.)+0.96)
      WRITE (LUT,425) TEST
 5552 IF(TEST.GT.BSQ) CALL DISTFN
      IF(TEST.LE.BSQ) CALL BETDST
      GO TO 1000
  561 WRITE (LUT,562)
 1000 KLOBS = KLOBS + 1
      IF(JOBS-KLOBS) 365,365,1111
  365 WRITE (LUT,360)
      CALL EXIT
C
C
    1 FORMAT(12A6)
    3 FORMAT (25I3)
   20 FORMAT(1X,11F12.5)
   90 FORMAT(27H0          LOG DETERMINANTS /41H  REFERENCE      COMPARISO
     1N          POOLED  )
  110 FORMAT (2X,F7.3,8X,F7.3,9X,F7.3)
  133 FORMAT (30H0NUMBER OF VARIABLES TOO HIGH= ,I4,5H, NA=,I4,4H,NB=I4)
  140 FORMAT(17H       B SQUARE = ,F14.7/20H      BETA SQUARE = ,F14.7)
  151 FORMAT(27H       DEGREES OF FREEDOM = I3)
  160 FORMAT(31H1     ORIENTATION OF ELLIPSOIDS                  )
  180 FORMAT(13H0  TRACE B = F18.7)
  207 FORMAT(50H1   ANALYSIS OF HOMOGENEITY OF COVARIANCE MATRICES)
  220 FORMAT(23H0  PERCENTAGES FOR B(I)/(X,8F14.7))
  333 FORMAT(10X12A6)
  360 FORMAT(18H1  CYCLE COMPLETED/12H0  RETURNING)
  380 FORMAT(3X,I2)
400   FORMAT (38H0DATA CARDS READ WITH VARIABLE FORMAT ,12A6)
  425 FORMAT(23H0  THE FORKING VALUE = F14.7)
  490 FORMAT(29H0  DATA LOGARITHM TRANSFORMED)
  562 FORMAT(50H0NEGATIVE EIGENVALUE, PROCEEDING TO NEXT DATA SET )
  710 FORMAT(48H0  REFERENCE SAMPLE SIZE BASED ON POPULATION OF ,I3/29H0
     1  COMPARISON SAMPLE SIZE IS ,I3)
  716 FORMAT(1H0)
 1118 FORMAT(20H0 ROW  POOLED MATRIX)
 1119 FORMAT(47H0 ROW  COVARIANCE MATRIX 2 ( COMPARISON MATRIX)  )
 1120 FORMAT(47H0 ROW  COVARIANCE MATRIX 1 ( REFERENCE MATRIX )  )
 1492 FORMAT(35H0  EIGENVALUES FOR REFERENCE MATRIX)
 1493 FORMAT(36H0  EIGENVECTORS FOR REFERENCE MATRIX)
 1494 FORMAT(36H0  EIGENVALUES FOR COMPARISON MATRIX)
```

```
 1495 FORMAT(37H0  EIGENVECTORS FOR COMPARISON MATRIX)
 1497 FORMAT(32H0  EIGENVALUES FOR POOLED MATRIX)
 1498 FORMAT(33H0  EIGENVECTORS FOR POOLED MATRIX)
 3333 FORMAT(6H0*****)
 5020 FORMAT(1X10F13.7)
 5100 FORMAT(23H0  FIRST DATA INPUT SET)
 5105 FORMAT(24H0  SECOND DATA INPUT SET)
10598 FORMAT(10H0  TEST = F14.7)
      END


      SUBROUTINE GETMAT (STMAT,MATNR)
C======= GETMAT READS THE MATNR.TH MATRIX FROM LOGICAL UNIT MTMABC TO
C======= THE PARAMETER STMAT.
C======= MTMABC CONTAINS 3 RECORDS WHICH CONTAINS RESP MATRIX A,
C======= MATRIX B AND MATRIX C.
C======= IN THE VARIABLE MATPOS IS INDICATED WHICH MATRIX WAS READ LAST.
C======= GETMAT MUST NOT BE CALLED UNTIL PUTMAT HAS WRITTEN ALL 3
C======= MATRICES E.G. PUTMAT HAS BEEN CALLED THREE TIMES.
      DIMENSION STMAT(45,45)
      COMMON A(45,45),B(45,45),C(45,45),SA(45),SB(45),SC(45),DIFF(45),
     1 T(45),X(45),Z(45),M,NA,NB,K,MW,MATRIX,LDISCR,LOGDEC,INVST(3),IT,
     2 MATPOS,KREYD
      COMMON /UNITS/ LIN,LUT,MTA,MTB,MTMABC,MTARA,MTBRA
      L=MATNR
C======= J IS THE NUMBER OF RECORDS TO SKIP FORWARD (IF J .GT. 0) OR
C======= BACKWARD (IF J .LT. 0)
      J=L-1-MATPOS
C======= IF J .EQ. 0 MATMABC IS ALREADY POSITIONED.
      IF (J) 10,20,30
   10 J=-J
C======= BACK OVER J RECORDS
      DO 11 J1=1,J
   11 BACKSPACE MTMABC
      GO TO 20
C======= SKIP OVER J RECORDS
   30 DO 31 J1=1,J
   31 READ (MTMABC) J2
C======= READ THE MATRIX
   20 READ (MTMABC) ((STMAT(I,JO),JO=1,M),I=1,M)
C======= UPDATE MATPOS
      MATPOS=L
      RETURN
      END


      SUBROUTINE PUTMAT(STMAT)
C======= PUTMAT WRITES THE MATRIX STMAT ON LOGICAL UNIT MTMABC.
C======= PUTMAT EXPECTS THAT MTMABC IS REWOUND BEFORE FIRST MATRIX IS
C======= STORED.
C======= FIRST CALL IS FOR STORING MATRIX A, SECOND FOR MATRIX B AND
C======= THIRD FOR MATRIX C.
C======= EACH MATRIX WILL OCCUPY ONE RECORD.
      DIMENSION STMAT(45,45)
      COMMON A(45,45),B(45,45),C(45,45),SA(45),SB(45),SC(45),DIFF(45),
     1 T(45),X(45),Z(45),M,NA,NB,K,MW,MATRIX,LDISCR,LOGDEC,INVST(3),IT,
     2 MATPOS,KREYD
      COMMON /UNITS/ LIN,LUT,MTA,MTB,MTMABC,MTARA,MTBRA
      WRITE (MTMABC)((STMAT(I,J),J=1,M),I=1,M)
      RETURN
      END
```

```
      SUBROUTINE HOUSER (N,NEV,C,EV,VEC,V)
C------- COMPUTES EIGENVALUES-EIGENVECTORS---SYMMETRIC MATRIX
      DIMENSION A(45),B(50),C(45,45),EV(45),P(45),TA(50),TB(50),W(50),
     1 Y(45),V(45,45)
      NN=N-1
      DO 16 I=1,NN
      SUM1=0
      B(I)=0
       JI=I+1
      DO 14  J =JI,N
 14   SUM1 = SUM1 + (C(I,J)*C(I,J))
      S=SQRT(SUM1)
      IF(S)15,16,15
   15 SGN=SIGN(1.,C(I,I+1))
      TEMP = SGN*(C(I, I+1))
      W(I+1)=SQRT(.5*(1.0+(TEMP/S)))
      C(I, I+1) = W(I+1)
       II=I+2
      IF(II-N)250,250,260
 250  TEMP=SGN/(2.*W(I+1)*S)
      DO 20 J=II,N
      W(J) = TEMP*C(I,J)
 20   C(I,J) = W(J)
 260  B(I) = -SGN*S
      DO 22  L = JI,N
      SUM2 = 0.0
      DO 21  M = JI,N
 21   SUM2 = SUM2 + (C(L,M) * W(M))
 22   P(L) = SUM2
      XKAP = 0.0
      DO 23  K = JI,N
 23   XKAP = XKAP + (W(K) * P(K))
      DO 24  L = JI,N
 24   P(L) = P(L) - (XKAP * W(L))
      DO 26  J = JI,N
      DO 25  K = J,N
      C(J,K) = C(K,J) - (2.0 * ((P(J) * W(K)) + (P(K) * W(J))))
 25   C(K,J)=C(J,K)
 26   CONTINUE
 16   CONTINUE
 17   DO 18  K = 1,N
 18   A(K) = C(K,K)
      B(N-1)=-B(N-1)
      B(N) = 0.0
      DO 500 I = 1,N
 500  W(I) = B(I)
C------- STURM METHOD (MODIFIED FROM SAND HCUSESR)
   29 U=ABS(A(1))+ABS(B(1))
      DO 30 I=2,N
      BD=ABS(A(I))+ABS(B(I))+ABS(B(I-1))
   30 IF (BD .GT. U) U=BD
      U2=U**2
      DO 32 I=1,N
      B(I)=B(I)**2/U2
      A(I)=A(I)/U
      EV(I)=-1.
 32   CONTINUE
      BD=U
      U=1.
      DO 1500 K=1,NEV
```

```
      EL=EV(K)
38    ELAM=.5*(U+EL)
      IF (ABS(ELAM-U) .LT. 1.E-20) GO TO 34
   33 IF (ABS(ELAM-EL) .LT. 1.E-20) GO TO 34
   35 P0=1.
      P1=A(1)-ELAM
      ZNSIG2=1.
      B2=0.
      B1=B(1)
      IF(P1)1051,1052,1052
1051  ZNSIG1=-1.
      IAG=0
      GO TO 1053
1052  ZNSIG1=1.
      IAG=1
1053  DO 100 I=2,N
      ALPH=A(I)-ELAM
      IF(B1)115,111,115
  111 P2=ZNSIG1*ALPH
      GO TO 114
115   IF(B2)116,117,116
  116 IF (ABS(P1)+ABS(P0) .GE. 1.E-20) GO TO 152
  151 P1=1.E20*P1
      P0=1.E20*P0
  152 P2=ALPH*P1-B1*P0
      GO TO 114
117   P2=ALPH*P1-ZNSIG2*B1
  114 P0=P1
      B2=B1
      B1=B(I)
      P1=P2
      ZNSIG2=ZNSIG1
      IF(P2) 121,125,122
  121 ZNSIG1=-1.
      GO TO 123
122   ZNSIG1=1.
  123 IF (ZNSIG1*ZNSIG2 .LE. 0) GO TO 100
125   IAG=IAG+1
100   CONTINUE
      IF (IAG .GE. K) GO TO 40
   42 U=ELAM
      GO TO 38
40    M=K+1
      DO 41 MM=M,IAG
41    EV(MM)=ELAM
      EL=ELAM
      GO TO 38
34    EV(K)=ELAM
1500  CONTINUE
      DO 1060 I=1,N
      A(I)=A(I)*BD
      EV(I)=EV(I)*BD
1060  CONTINUE
43    IF(VEC) 44,700,44
44    L = NEV - 1
      DO 502 K = 1,L
      IF (EV(K) .GT. EV(K+1)) GO TO 502
501   EV(K+1)=.99999999*EV(K)
502   CONTINUE
      DO 600 I = 1,NEV
      II = 0
```

```
      DO 503 J=1,N
503   Y(J)=1.
601   DO 504 K=1,N
      P(K) = 0.0
      TB(K) = W(K)
504   TA(K) = A(K) - EV(I)
      L = N-1
      DO 505 J = 1,L
      IF (ABS(TA(J))-ABS(W(J))) 507,506,506
506   F = W(J) / TA(J)
      GO TO 509
507   F = TA(J)/W(J)
      TA(J) = W(J)
      T = TA(J+1)
      TA(J+1) = TB(J)
      TB(J) = T
      P(J) = TB(J+1)
      TB(J+1)=0.0
      IF(II-1)  509,508,509
508   T = Y(J)
      Y(J) = Y(J+1)
      Y(J+1) = T
509   TB(J+1) = TB(J+1) -F*P(J)
      TA(J+1) = TA(J+1)-F*TB(J)
      IF(II-1)  505,510,505
510   Y(J+1) = Y(J+1) - F*Y(J)
505   CONTINUE
      IF(TA(N)) 511,512,511
512   TA(N) = 10.E-30
511   IF(TA(N-1)) 513,514,513
514   TA(N-1) = 10.E-30
513   Y(N) = Y(N)/TA(N)
      Y(N-1) = (Y(N-1)-Y(N)*TB(N-1))/TA(N-1)
      L = N-2
      DO 515 J = 1,L
      K = N-J-1
      IF(TA(K))  515,516,515
516   TA(K) = 10.E-30
515   Y(K) = (Y(K) - Y(K+1)*TB(K)-Y(K+2)*P(K))/TA(K)
      IF(II) 517,518,517
518   II = 1
      GO TO 601
517   DO 521 J=1,L
      T = 0.0
      K=N-J-1
      M = K+1
      DO 519 KK=M,N
519   T = T+C(K,KK)*Y(KK)
      DO 520 KK=M,N
520   Y(KK) = Y(KK) - 2.*T*C(K,KK)
521   CONTINUE
      T=0.0
      DO 523 J = 1,N
523   T = T + Y(J)**2
      XNORM=SQRT(T)
      DO 524 J = 1,N
524   V(J,I) = Y(J) / XNORM
600   CONTINUE
700   RETURN
      END
```

```
      SUBROUTINE ORIENT(E,H,M,X,O,KOUNT,ENB,EV)
      DIMENSION E(45,45),H(45,45),O(45,45),X(45),EV(45)
      COMMON /UNITS/ LIN,LUT,MTA,MTB,MTMABC,MTARA,MTBRA
      DO 170 IX=1,M
  170 X(IX)=0.
      WRITE (LUT,100)
      DO 200 I=1,M
      DO 200 J=1,M
  200 X(I) = X(I) + H(I,J)*E(J,KOUNT)
      AQ=0.
      DO 230 I=1,M
  230 AQ = AQ + X(I)*E(I,KOUNT)
      DO 260 I=1,M
      X(I)=0.
      DO 260 J=1,M
  260 X(I) = X(I) + O(I,J)*E(J,KOUNT)
      AP=0.
      DO 271 I=1,M
  271 AP = AP +E(I,KOUNT)*X(I)
      HLAMB = EV(KOUNT)
      CHISQ=ENB*(AP*HLAMB+AQ/HLAMB-2.)
      CHISQ = ABS(CHISQ)
      K=M-1
      WRITE (LUT,280) KOUNT,CHISQ,K
      RETURN
C
C
  100 FORMAT (33H0   VECTOR        CHISQUARE      DF   )
  280 FORMAT (6X,I2, 9X,F9.3,4X,I3)
      END


      SUBROUTINE HETDST
      COMMON A(45,45),B(45,45),C(45,45),SA(45),SB(45),SC(45),DIFF(45),
     1 T(45),X(45),Z(45),M,NA,NB,K,MW,MATRIX,LDISCR,LOGDEC,INVST(3),IT,
     2 MATPOS,KREYD
      COMMON /UNITS/ LIN,LUT,MTA,MTB,MTMABC,MTARA,MTBRA
C     PROGRAM HETDST
C     ANDERSON AND BAHADURS GENERALIZED DISTANCE (R.REYMENT,JULY,1966)
C     THE PROGRAM COMPUTES THE GENERALIZED DISTANCE FOR UNEQUAL
C     COVARIANCE MATRICES ACCORDING TO ANDERSON/BAHADUR(1962).
C
C
C     INPUT = ON FIRST CARD...NO. VARIABLES, NO.SPECIMENS IN SAMPLE A,
C     AND NO. SPECIMENS,IN SAMPLE B...INDIVIDUALS OF A AND INDIVIDUALS
C
C     OUTPUT = COMPUTED COVARIANCE MATRICES,SAMPLE SIZES,FOR EACH
C     ITERATION,THE NUMBER OF ITERATION,VALUE OF ITERATION FACTOR T,
C     COMPUTED Z DERIVING FROM THIS,INVERSE MATRIX AND ITS DETERMINANT,
C     THE ESTIMATION VECTORS,INTERMEDIATE VALUES FOR DISTANCE,DISTANCE,
C     SQUARE OF DISTANCE.
      WRITE (LUT,1)
      ENA=NA
      ENB=NB
      WRITE (LUT,18) NA,NB
C     REQUIRES COVARIANCE MATRICES,A,B,A VECTOR,SB(I) TO BE DETERMINED.
      IT=0                                                              HETDST
      K=1
      T(K)=0.5
      NV=M
```

```
C     VECTOR SC CONTAINS THE DIFFERENCES OF MEANS
      WRITE (LUT,993)
      DO 997 IX=1,NV
  997 WRITE (LUT,304) SC(IX)
      WRITE (LUT,733)
      DO 69 KK=1,4
      CALL BAHADU
      IF(KK.EQ.4) GO TO 720
      I=1.+(Z(K)*10.)
      IF (I .EQ. 1) GO TO 161
      K=K+1
      IF (I .GT. 1) GO TO (162,2001,2003),KK
      IF (KK-2) 30,31,32
   30 T(K)=0.75
      GO TO 69
   31 T(K)=0.875
      GO TO 69
   32 T(K)=1.0
      GO TO 69
 2003 T(K)=.8125
   69 CONTINUE
 2001 T(K)=.625
      DO 2004 KK=1,2
      CALL BAHADU
      IF(KK.EQ.2) GO TO 720
      I=1.+(Z(K)*10.)
      IF (I .EQ. 1) GO TO 161
      K=K+1
      IF (I .GT. 1) GO TO 2005
      T(K)=.6875
      GO TO 2004
 2005 T(K)=.5625
 2004 CONTINUE
  162 T(K)=.25
      DO 2169 KK=1,3
      CALL BAHADU
      IF(KK.EQ.3) GO TO 720
      I=1.+(Z(K)*10.)
      IF (I .EQ. 1) GO TO 161
      K=K+1
      IF (I .GT. 1) GO TO (736,2007),KK
      T(K)=.375
      IF(KK.EQ.2) T(K)=.4375
      GO TO 2169
 2007 T(K)=.3125
 2169 CONTINUE
  736 T(K)=.125
      DO 2008 KK=1,2
      CALL BAHADU
      IF(KK.EQ.2) GO TO 720
      I=1.+(Z(K)*10.)
      IF (I .EQ. 1) GO TO 161
      K=K+1
      IF (I .GT. 1) GO TO 2009
      T(K)=.1875
      GO TO 2008
 2009 T(K)=.0625
 2008 CONTINUE
  720 IT = K-1
      I=1.+(Z(K)*10.)
```

```
      IF (I-1) 2710,161,2711
 2710 SIGNTM=1.0
      GO TO 2703
 2711 SIGNTM=-1.0
 2703 IT=IT+1
      K=K+1
      DIFF(K)=ABS(T(K-1)-T(K-2))
      IF (DIFF(K) .LE. DIFF(K-1))GO TO  2705
      T(K)=T(K-1)+SIGNTM*DIFF(K)/FLOAT(IT-2*IT/3)
 2706 IF (DIFF(K) .LT. 0.00001) GO TO 161
      CALL BAHADU
      IF(IT-19)720,720,169
 2705 T(K)=T(K-1)+SIGNTM*DIFF(K)*0.5
      GO TO 2706
  169 WRITE (LUT,184)
C
  161 WRITE (LUT,174)
      DO 735 I=1,NV
  735 WRITE (LUT,734) SB(I)
      IF(MATRIX.EQ.1) CALL EXIT
      CALL JZUP2
      CALL HAFCOV
      CALL REYDST
      IF(LDISCR.EQ.1) CALL DISTFN
C     REYDST COMPUTES T SQUARE AND VARIANCE RATIO FOR HET MATRICES
      RETURN
C
C
    1 FORMAT(81H1 STATISTICAL DISTANCE FOR HETEROGENEOUS COVARIANCE MATR
     1ICES                           )                                   HETDSI
   18 FORMAT(8H0    MA =,I3,7H    NB =,I3/38H0  ITERATIONS FOR DISTANCE CA
     1LCULATION)
  174 FORMAT(33H0ESTIMATE OF DISCRIMINANT VECTOR          )
  184 FORMAT(19H0WOULD NOT CONVERGE)
  304 FORMAT(1X,7F12.5)
  733 FORMAT(48H0ITERATION      Z         D         DSQ       DIFF T           )
  734 FORMAT(2X,F10.5)
  993 FORMAT(31H  SAMPLE DIFFERENCE MEAN VECTOR)
      END


      SUBROUTINE BAHADU
      DIMENSION ASC(45)
      COMMON A(45,45),B(45,45),C(45,45),SA(45),SB(45),SC(45),DIFF(45),
     1 T(45),X(45),Z(45),M,NA,NE,K,MW,MATRIX,LDISCR,LOGDEC,INVST(3),IT,
     2 MATPOS,KREYD
      COMMON /UNITS/ LIN,LUT,MTA,MTB,MTMABC,MTARA,MTBRA
C     SC IS DIFFERENCE OF MEANS
      NV=M
      TNEW=T(K)
      DO 151 I=1,NV
      DO 151 J=1,NV
  151 C(I,J)=TNEW*A(I,J)+(1.-TNEW)*B(I,J)
      CALL MATINV (C,NV,ASC,0,DETERM,MW)
C     SOLUTION OF EQUATION FOR VECTOR SB(I) THRU MATINV AND MEAN VECTOR
      DO 153 I=1,NV
      SB(I)=0.
      DO 153 J=1,NV
  153 SB(I)=C(I,J)*SC(J)+SB(I)
C     THIS GIVES THE VECTOR ESTIMATION DESIRED
```

```
C     DISCRIMINANT VECTOR IS SB(I)
      DO 154 I=1,NV
      DO 154 J=1,NV
  154 C(I,J)=(A(I,J)*TNEW**2)-((1.-TNEW)**2)*B(I,J)
      DO 155 I=1,NV
      X(I)=0.
      DO 155 J=1,NV
  155 X(I)=X(I)+C(I,J)*SB(J)
      ZNEW=0.
      DO 156 I=1,NV
  156 ZNEW=ZNEW+X(I)*SB(I)
C     EXPANSION OF THE QUADRATIC FORM
      Z(K)=ZNEW
      TNEW=0.
      Q=0.0
      DO 167 I=1,NV
      X(I)=0.0
  167 Q=Q+SC(I)*SB(I)
      Q=2.0*Q
      DO 168 I=1,NV
      ASC(I) = 0.
      DO 168 J=1,NV
  168 ASC(I) = ASC(I) + A(I,J)*SB(J)
      R=0.0
      DO 170 I=1,NV
  170 R = R + ASC(I)*SB(I)
      R=SQRT(ABS(R))
      DO 171 I=1,NV
      ASC(I) = 0.
      DO 171 J=1,NV
  171 ASC(I) = ASC(I) + B(I,J)*SB(J)
      AS=0.0
      DO 172 I=1,NV
  172 AS = AS + ASC(I)*SB(I)
      AS=SQRT(ABS(AS))
      D = Q/(R+AS)
      DSQ = D**2
      WRITE (LUT,159) IT,Z(K),D,DSQ,DIFF(K)
      RETURN
C
C
  159 FORMAT(2X,I2,7X,F8.5,2X,F6.3,3X,F6.3,3X,F10.7)
      END


      SUBROUTINE HAFCOV
      COMMON A(45,45),B(45,45),C(45,45),SA(45),SB(45),SC(45),DIFF(45),
     1 T(45),X(45),Z(45),M,NA,NE,K,MW,MATRIX,LDISCR,LOGDEC,INVST(3),IT,
     2 MATPOS,KREYD
      COMMON /UNITS/ LIN,LUT,MTA,MTB,MTMABC,MTARA,MTBRA
      WRITE (LUT,407)
      CALL GETMAT(A,1)
      CALL GETMAT(B,2)
      DO 408 I=1,M
      DO 408 J=1,M
  408 C(I,J)=(A(I,J)+B(I,J))/2.
      WRITE (LUT,200)
      CALL MATOUT (C)
      CALL MATINV(C,M,Z,0,DETERM,MW)
      DO 410 I=1,M
```

```
      Z(I)=0.
      DO 410 J=1,M
  410 Z(I) = Z(I) + C(I,J)*SC(J)
      DSQ=0.
      DO 411 I=1,M
  411 DSQ = DSQ + SC(I)*Z(I)
      DSQ=ABS(DSQ)
      DS=SQRT(DSQ)
      WRITE (LUT,222) DS,DSC
      RETURN
C
C
  200 FORMAT (24H0 ROW  COVARIANCE MATRIX )
  222 FORMAT( 33H0      DISTANCE, DISTANCE SQUARED  /7X,F7.3,8X,F9.3/)
  407 FORMAT(61H1    HETEROGENEOUS DISTANCE BY AVERAGING COVARIANCE MATR
     1ICES             )
      END

      SUBROUTINE REYDST
      DIMENSION XA(45),XB(45),SLM(45),DISC(45),XC(45)
      COMMON A(45,45),B(45,45),C(45,45),SA(45),SB(45),SC(45),DIFF(45),
     1 T(45),X(45),Z(45),M,NA,NB,K,MW,MATRIX,LDISCR,LOGDEC,INVST(3),IT,
     2 MATPOS,KREYD
      COMMON /UNITS/ LIN,LUT,MTA,MTB,MTMABC,MTARA,MTBRA
C------- RETURN IF KREYD IS LE 0
      IF (KREYD .LE. 0) RETURN
      WRITE (LUT,4165)
      DSUM = 0.
      ENB = NB
      ENA = NA
      NAB = NA
      TK = M
      AN = SQRT(ENB/ENA)
      WRITE (LUT,3) AN
C      REMEMBER THAT A IS THE LARGER SAMPLE AND NA IS GRT TH NB
C     REMEMBER THAT VECTOR SC CONTAINS DIFFERENCE OF MEANS
C***********************************************************************
C------- THE STATEMENTS FROM  DC 6008 UNTIL 200 CONTINUE ARE MACHINE
C------- DEPENDENT. THEY USE CD 3600-FORTRAN STATEMENTS TO STORE SAMPLE
C------- A AND SAMPLE B ON A RANDOM ACCESS DRUM STORAGE UNIT.
C------- IF YOU HAVE NO RANDOM ACCESS UNIT YOU CAN SKIP THIS STATEMENTS
C------- AND MAKE SOME CHANGES IN ROUTINE  RANKRT.
      REWIND MTARA
      REWIND MTA
      DO 6008 IV = 1,NAB
      READ (MTA) (XA(IX),IX=1,M)
      BUFFER OUT (MTARA,1) (XA(1),XA(M))
 6011 IF (UNIT,MTARA) 6011,6008
 6008 CONTINUE
C------- SAMPLE A STORED NOW ON DRUM IN RANDOM ACCESS
      REWIND MTBRA
      REWIND MTB
      DO 200 IV=1,NB
      READ (MTB) (XB(IX),IX=1,M)
      BUFFER OUT (MTBRA,1) (XB(1),XB(M))
  205 IF (UNIT,MTBRA) 205,200
  200 CONTINUE
C------- SAMPLE B IS NOW STORED ON DRUM IN RANDOM ACCESS
C***********************************************************************
 6009 NAB = NB
```

```
      DO 500 KK=1,KREYD
      DO 6050 I=1,M
      XC(I) = 0.
 6050 X(I) = 0.
      DO 6007 I=1,M
      SUM(I) = 0.
      DO 6007 J=1,M
 6007 A(I,J) = 0.
      CALL CLEAR
      DO 40 IV=1,NAB
      CALL RANKRT(XA,0)
      CALL RANKRT(XB,1)
      DO 20 I=1,M
   20 XC(I) = XA(I)*AN
      DO 22 I=1,M
   22 X(I) = ABS(XB(I)-XC(I))
C     NEW OBSERVATIONS STORED IN X (I)
      DO 23 I=1,M
   23 SUM(I) = SUM(I) + X(I)
      DO 25 I=1,M
      DO 25 J=1,M
   25 A(I,J) = A(I,J) + X(I)*X(J)
   40 CONTINUE
      DO 45 I=1,M
      DO 45 J=1,M
   45 A(I,J)=(A(I,J) - SUM(I)*SUM(J)/ENB)/(ENB-1.0)
C     COVARIANCE MATRIX OF THE NEW VARIABLES
      CALL MATINV(A,M,B,0,DETERM,MW)
      DO 60 I=1,M
      DISC(I)=0.
      DO 60 J=1,M
   60 DISC(I) = DISC(I) + A(I,J)*SC(J)
      DS = 0.
      DO 70 I=1,M
   70 DS = DS + DISC(I)*SC(I)
      DS = DS*2.0
      DSUM = DSUM + DS
      TEST = DSUM/FLOAT(KK)
      WRITE (LUT,76) KK,DS
  500 CONTINUE
      WRITE (LUT,80)
      DS=DSUM/FLOAT(KREYD+1)
      TT = DS*ENB/2.
      WRITE (LUT,85) DS,TT
      F = (TT*(ENB-FLOAT(M)))/((ENB-1.)*FLOAT(M))
      NAA = NB-1
      WRITE (LUT,95) F,NAA,M
      RETURN
C
C
    3 FORMAT(24H0  MULTIPLICATION FACTOR /8H0   AN = ,F14.7/32H0   ITERATI
     1ON   REYMENTS DSQ   TEST   )
   76 FORMAT(6X,I2,8X,F14.4,8X,F14.4)
   80 FORMAT(28H0  SIGNIFICANCE COMPUTATIONS /1H0)
   85 FORMAT (25H0           AVE.DSQ         TSQ /2(3X,F10.3))
   95 FORMAT(23H0  VARIANCE RATIO IS = F14.2,4H FORI4,5H AND I2,19H DEGR
     1EES OF FREEDOM     )
 4165 FORMAT(30H1  HETEROGENEOUS SIGNIFICANCES        )
      END
```

```
      SUBROUTINE CLEAR
C======= CLEAR IS A ROUTINE WHICH MUST BE CALLED BEFORE FIRST CALLRANKRT
C======= IT CLEARS THE INDICATOR ARRAY KAT.
      COMMON  /RANKO/ KAT(500)
      DO 40 I=1,500
   40 KAT(I)=0
      RETURN
C***********************************************************************
C======= IF YOU HAVE NO RANDOM ACCESS UNIT YOU MUST ADD THE FOOLOWING
C======= CODE.
C     COMMON /UNITS/ .....
C     REWIND MTA
C     REWIND MTB
C***********************************************************************
      END


      SUBROUTINE RANKRT(XD,KABSW)
C======= RANKRT PICKS RANDOMLY ONE RECORD FROM SAMPLE A (IF KABSW=0 )
C======= OR FROM SAMPLE B ( IF KABSW=1 ), THIS VERSION ASSUMES THAT
C======= SAMPLE A AND SAMPLE B IS PLACED EARLIER ON RANDOM ACCESS UNITS
C======= MTARA RESP MTBRA. BEFORE FIRST CALL RANKRT THE INDICATOR ARRAY
C======= KAT MUST BE CLEARED BY CALL CLEAR.
C======= VERSION 2, WHICH CAN HANDLE TWO PARALLEL RANDOM FILES
      DIMENSION XD(45)
      COMMON A(45,45),B(45,45),C(45,45),SA(45),SB(45),SC(45),DIFF(45),
     1 T(45),X(45),Z(45),M,NA,NE,K,MW,MATRIX,LDISCR,LOGDEC,INVST(3),IT,
     2 MATPOS,KREYD
      COMMON /UNITS/ LIN,LUT,MTA,MTB,MTMABC,MTARA,MTBRA
      COMMON  /RANKO/ KAT(500)
      KAB=KABSW
      IF (KAB) 45,45,47
C======= PICK UP A CARD FROM UNIT MTARA
   45 NT=NA
      LUN=MTARA
      GO TO 48
C======= PICK UP A CARD FROM UNIT MTBRA
   47 NT=NB
      LUN=MTBRA
   48 CONTINUE
C======= GET A RANDOM NUMBER FROM 1 TO NT+1
C======= X=RANF(-1) WILL GIVE A NEW FLOATING POINT RANDOM NUMBER IN THE
C======= INTERVAL 0.0 UNTIL 1.0 EVERY TIME RANF IS CALLED.
   20 I=RANF(-1)*FLOAT(NT)+1.0
C======= DON*T ACCEPT I .GT. NT
      IF (I .GT. NT ) GO TO 20
C======= GET INDICATOR FOR UNIT MTBRA
      IK=KAT(I)/10
C======= GET INDICATOR FOR UNIT MTARA
      IR=KAT(I)-IK*10
C======= TEST IF WE JUST NOW ARE USING MTARA
      IF (KAB .EQ. 0) IK=IR
C======= IF THIS CARD ALREADY USED TRY TO GET A NEW RANDOM NUMBER
      IF (IK .NE. 0) GO TO 20
C======= POSITION THE DRUM
      CALL LOCATE (LUN,(I-1)*M)
      BUFFER IN (LUN,1) (XD(1),XD(M))
   11 IF (UNIT,LUN) 11,12
C======= INDICATE THAT THIS CARD IS USED
   12 KAT(I)=KAT(I)+KAB*10
      RETURN
```

```
C*****************************************************************************
C=------- IF YOU HAVE NO RANDOM ACCES UNIT YOU CAN USE THE FOLLOWING CODE
C      COMMON  ......
C      DIMENSION   ........
C      DIMENSION LUNPOS(2)
C      DATA (LUNPOS=0,0)      OR      DATA LUNPOS/0,0/
C      KAB=KABSW
C      IF (KAB) 45,45,47
C   45 NT=NA
C      LUN=MTA
C      GO TO 48
C   47 NT=NB
C      LUN=MTB
C   48 CONTINUE
C   20 I=RANF(-1)*FLOAT(NT)+1.0
C      IF (I .GT. NT) GO TO 20
C      IK=KAT(I)/10
C      IR=KAT(I)-IK*10
C      IF (KAB.EQ. 0) IK=IR
C      IF (IK .NE. 0) GO TO 20
C      J=I-1-LUNPOS(KAB+1)
C      IF (J) 50,60,70
C   50 J=-J
C      DO 51 J1=1,J
C   51 BACKSPACE LUN
C      GO TO 60
C   70 DO 71 J1=1,J
C   71 READ(LUN) J2
C   60 LUNPOS(KAB+1)=I
C      READ (LUN) (XD(I)X),IX=1,M)
C   12 KAT(I)=KAT(I)+KAB*10
C      RETURN
C*****************************************************************************
       END

      SUBROUTINE DISTFN
      COMMON A(45,45),B(45,45),C(45,45),SA(45),SB(45),SC(45),DIFF(45),
     1 T(45),X(45),Z(45),M,NA,NB,K,MW,MATRIX,LDISCR,LOGDEC,INVST(3),IT,
     2 MATPOS,KREYD
      COMMON /UNITS/ LIN,LUT,MTA,MTB,MTMABC,MTARA,MTBRA
C     ALL INPUT OF DATA IS DONE IN CALLING PROGRAM
      WRITE (LUT,300)
      ENA=NA
      ENB=NB
      NDF=NA+NB-2
      WRITE (LUT,200) NA,NB,NDF
      DO 13 I=1,M
   13 WRITE (LUT,272) I,SA(I),SB(I),SC(I)
 4448 CONTINUE
      CALL GETMAT (C,3)
      CALL MATINV(C,M,Z,0,DETERM,MW)
      DO 17 I=1,M
      Z(I)=0.
      DO 17 J=1,M
   17 Z(I) = Z(I) + C(I,J)*SC(J)
      WRITE (LUT,309)
      WRITE (LUT,222) (Z(I),I=1,M)
      DO 1064 I=1,M
 1064 SB(I) = Z(I)
```

```
      DSQ=0.
      DO 18 I=1,M
   18 DSQ = DSQ + SC(I)*Z(I)
      DSQ=ABS(DSQ)
      WRITE (LUT,310)
      EM=M
      DS=SQRT(DSQ)
      WRITE (LUT,223) DS,DSQ
      WRITE (LUT,312)
      DSQ=(ENA*ENB/(ENA+ENB))*DSQ
      WRITE (LUT,313) DSQ
      DSQ=DSQ*((ENA+ENB-EM-1.)/(EM*(ENA+ENB-2.)))
      WRITE (LUT,314) DSQ
      NP = NA+NB-M-1
      WRITE (LUT,315) M,NP
      CALL JZUP2
      RETURN
C
C
  200 FORMAT(13H            N1 =,I3,12H           N2 =I3,12H           DF =,I3/
     151H  VAR     MEAN VECTOR 1  MEAN VECTOR 2  DIFF.VECTOR   )
  222 FORMAT(2X,12F10.5)
  223 FORMAT (2X,F10.5,11X,F10.5)
  272 FORMAT(I4,F14.3,2F15.3)
  300 FORMAT (32H1              STATISTICAL DISTANCE ,//1H0)
  309 FORMAT(28H0 DISCRIMINATOR COEFFICIENTS)
  310 FORMAT(33H0 RESULTS FOR MAHANALOBIS DSQUARE /33H                 D
     1         D SQUARE )
  312 FORMAT(40H0 SIGNIFICANCE FOR D SQUARE AND T SQUARE/)
  313 FORMAT(11H0T SQUARE =F13.3)
  314 FORMAT( 4H0F =F13.2)
  315 FORMAT(6H0DF1 =  I3,7H  DF2 = I3  )
      END

      SUBROUTINE MATINV(A,N,B,M,DETERM,JW)
      DIMENSION IPIVOT(45),A(JW,JW),B(JW,1),INDEX(45,2),PIVOT(45)
      EQUIVALENCE(IROW,JROW),(ICOLUM,JCOLUM),(AMAX,T,SWAP)
      DETERM=1.
      DO 20 J=1,N
   20 IPIVOT(J)=0
      DO 550 I=1,N
      AMAX=0.
      DO 105 J=1,N
      IF(IPIVOT(J)-1)60,105,60
   60 DO 100 K=1,N
      IF(IPIVOT(K)-1)80,100,740
   80 IF(ABS(AMAX)-ABS(A(J,K)))85,100,100
   85 IROW=J
      ICOLUM=K
      AMAX=A(J,K)
  100 CONTINUE
  105 CONTINUE
      IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1
C     INTERCHANGE ROWS TO PUT PIVOT ELEMNT ON DIAGONAL
      IF(IROW-ICOLUM)140,260,140
  140 DETERM=-DETERM
      DO 200 L=1,N
      SWAP=A(IROW,L)
      A(IROW,L)=A(ICOLUM,L)
  200 A(ICOLUM,L)=SWAP
```

```
      IF(M)260,260,210
  210 DO 250 L=1,N
      SWAP=B(IROW,L)
      B(IROW,L)=B(ICOLUM,L)
  250 B(ICOLUM,L)=SWAP
  260 INDEX(I,1)=IROW
      INDEX(I,2)=ICOLUM
      PIVOT(I)=A(ICOLUM,ICOLUM)
      DETERM=DETERM*PIVOT(I)
C     DIVIDE PIVOT ROW BY PIVCT ELEMENT
      A(ICOLUM,ICOLUM)=1.
      DO 350 L=1,N
  350 A(ICOLUM,L)=A(ICOLUM,L)/PIVOT(I)
      IF(M)380,380,360
  360 DO 370 L=1,M
  370 B(ICOLUM,L)=B(ICOLUM,L)/PIVOT(I)
  380 DO 550 L1=1,N
      IF(L1-ICOLUM)400,550,400
  400 T=A(L1,ICOLUM)
      A(L1,ICOLUM)=0.
      DO 450 L=1,N
  450 A(L1,L)=A(L1,L)-A(ICOLUM,L)*T
      IF(M)550,550,460
  460 DO 500 L=1,M
  500 B(L1,L)=B(L1,L)-B(ICOLUM,L)*T
  550 CONTINUE
      DO 710 I=1,N
      L=N+1-I
      IF(INDEX(L,1)-INDEX(L,2))630,710,630
  630 JROW=INDEX(L,1)
      JCOLUM=INDEX(L,2)
      DO 705 K=1,N
      SWAP=A(K,JROW)
      A(K,JROW)=A(K,JCOLUM)
      A(K,JCOLUM)=SWAP
  705 CONTINUE
  710 CONTINUE
  740 RETURN
      END

      SUBROUTINE PLTHS1(XX,III,START,ENDX,NUM)
C     PLOTS A FREQUENCE DISTRIBUTION IN NUM INCREMENTS FOR THE VALUES INPLTH
C     THE ARRAY XX BETWEEN START AND ENDX IN VALUE. THE FIRST III       PLTH
C     VALUES IN THE XXX ARRAY ARE CONSIDERED.                           PLTH
C                                                                       PLTH
C     ----------------------------------------------------------------- PLTH
      DIMENSION XX(100),KCNT(100),LINE(101)                             PLTHSI02
      DATA (LB=1H ),(LAST=1H*),(LDASH=1H=),(LI=1HI)
C=F IV- DATA LB,LAST,LDASH,LI /1H ,1H*,1H-,1HI/
C     ------------------------------------------------------------------PLTH
C                                                                       PLTH
      COMMON /UNITS/ LIN,LUT,MTA,MTB,MTMABC,MTARA,MTBRA
      DO 1 I=1,NUM                                                      PLTHSI04
    1 KONT(I) = 0                                                       PLTHSI05
C     SET NUM COUNTERS TO ZERO.                                         PLTH
      XNUM = NUM                                                        PLTHSI06
      XINT = (ENDX-START)/XNUM                                          PLTHSI07
C     THE INTERVAL SIZE FOR THE NUM INTERVALS IS XINT.                  PLTH
      NUM2 = 100/NUM                                                    PLTHSI08
      NUM3 = NUM2 - 1                                                   PLTHSI09
```

```
      NUM4 = NUM2*NUM+1
      DO 5 I=1,101                                                       PLTHST11
    5 LINE(I) = LB                                                       PLTHST12
C     LINE IS SET EQUAL TO BLANKS.                                       PLTH
      LINE(NUM4) = LI
C     THE LAST PLACE TO BE USED IN LINE IS SET EQUAL TO AN I.            PLTH
      DO 10 I=1,III                                                      PLTHST14
C     LOOK AT III PINTS IN THE XX ARRAY.                                 PLTH
      IF(XX(I).LT.START.OR.XX(I).GT.ENDX) GO TO 10                       PLTHST15
C     IF XX(I) IS OUTSIDE THE RANGE IGNORE IT.
      KNT = (XX(I)-START)/XINT                                           PLTHST16
      KNT = KNT + 1                                                      PLTHST17
C     KNT IS THE INTERVAL THAT XX(I) FALLS INTO.                         PLTH
      IF(KNT.GT.NUM) KNT = NUM                                           PLTHST18
C     IF XX(I) = ENDX KNT COULD BE GREATER THAN NUM.                     PLTH
      KONT(KNT) = KONT(KNT) + 1                                          PLTHST19
C     INCREASE THE KNT COUNTER BY 1.                                     PLTH
   10 CONTINUE                                                           PLTHST20
C                                                                        PLTH
      KMAX = KONT(1)                                                     PLTHST21
      DO 20 I=2,NUM                                                      PLTHST22
      IF(KONT(I).GT.KMAX) KMAX = KONT(I)                                 PLTHST23
C     FIND THE VALUE OF MAXIMUM COUNTER.                                 PLTH
   20 CONTINUE                                                           PLTHTS24
C
      KSCALE = KMAX/50                                                   PLTHST25
C     KSCALE IS THE VERTICAL SCALE FACTOR.                               PLTH
      IF(KSCALE*50.NE.KMAX) KSCALE = KSCALE + 1                          PLTHTS2
      WRITE (LUT,100) XINT,START,ENDX,NUM
      NTOP = KMAX + KSCALE                                               PLTHTS28
      DO 21 I=1,NUM
      IF(KONT(I).EQ.0) KONT(I) = -KSCALE                                 PLTHTS30
   21 CONTINUE                                                           PLTHTS31
   30 NTOP =NTOP-KSCALE                                                  PLTHTS32
      IF(NTOP.LE.(-KSCALE)) GO TO 40                                     PLTHTS33
      IF(NTOP.LT.0) NTOP = 0                                             PLTHST34
      ICONT = 0
      DO 32 I=1,NUM                                                      PLTHST36
      IF(KONT(I).LT.NTOP) GO TO 32                                       PLTHST37
C     IF THE COUNTER FOR THE INTERVAL IS GREATER THAN OR EQUAL TO NTOP   PLTH
C     PUT ASTERISKS INTO THE PART OF LINE ALOTTED TO THIS INTERVAL.      PLTH
      DO 33 II=1,NUM2                                                    PLTHST38
C     THERE ARE NUM2 SPACES IN LINE FOR EACH INTERVAL.                   PLTH
      INX = II + ICONT                                                   PLTHST39
   33 LINE(INX) = LAST                                                   PLTHST40
   32 ICONT = ICONT + NUM2                                               PLTHST41
      IF(NTOP.EQ.0) GO TO 40
      WRITE (LUT,101) NTOP,LINE
      GO TO 30                                                           PLTHST43
C
   40 IF(NUM3.EQ.0) GO TO 50                                             PLTHST44
C     THIS SEGMENT FROM HERE TO RETURN PRINTS THE BASE LINE WITH I AT    PLTH
C     THE END OF EACH INTERVAL AND MINUSES BETWEEN.                      PLTH
      ICONT = 0                                                          PLTHST45
      DO 41 I=1,NUM                                                      PLTHST46
      DO 42 II=1,NUM3                                                    PLTHST47
      INX = II + ICONT                                                   PLTHST48
   42 LINE(INX) = LDASH                                                  PLTHST49
      LINE(INX+1) = LI                                                   PLTHST50
   41 ICONT = ICONT + NUM2                                               PLTHST51
```

```
      WRITE (LUT,102) LINE
      RETURN                                                            PLTHSI53
C                                                                       PLTH
   50 DO 51 I=1,NUM                                                     PLTHSI54
   51 LINE(I) = LI                                                      PLTHSI55
      WRITE (LUT,102) LINE
      RETURN                                                            PLTHSI57
C
C
  100 FORMAT(34H1       HISTOGRAM FOR DISCRIMINANTS /14H0  INTERVAL = ,
     1F10.4,16H DATA STARTS AT ,F10.4/11H0  ENDS AT ,F10.4,19H CLASS INT
     2ERVALS = ,I4)
  101 FORMAT(I5,2X,1HI,101A1)                                           PLTHSI61
  102 FORMAT(7X,1HI,101A1)                                              PLTHSI62
      END

      SUBROUTINE DISTRB
C  THIS SUBROUTINE TESTS THE INDIVIDUAL VARIABLES FOR DEVIATIONS
C  FROM THE UNIVARIATE NORMAL DISTRIBUTION BY CALCULATING THE
C   COEFFICIENTS OF SKEWNESS AND KURTOSIS AND THEIR STANDARD ERRORS
      DIMENSION STDEVA(45),STDEVB(45),R(45,4)
      COMMON A(45,45),B(45,45),C(45,45),SA(45),SB(45),SC(45),DIFF(45),
     1 T(45),X(45),Z(45),M,NA,NE,K,MW,MATRIX,LDISCR,LOGDEC,INVST(3),IT,
     2 MATPOS,KREYD
      COMMON /UNITS/ LIN,LUT,MTA,MTB,MTMAEC,MTARA,MTBRA
      DO 5 I=1,M
      STDEVA(I)=SQRT(A(I,I))
    5 STDEVB(I) = SQRT(B(I,I))
      WRITE (LUT,6)
      INDKS = 1
      WRITE (LUT,10) INDKS
      DO 14 J=1,M
   14 WRITE (LUT,15) SA(J),STDEVA(J)
      INDKS = 2
      WRITE (LUT,10) INDKS
      DO 19 J=1,M
   19 WRITE (LUT,15) SB(J),STDEVB(J)
      KTOT = NA
      KNT=0
C------- SAMPLE A FIRST
      LUN=MTA
      DO 24 I=1,M
      DO 24 J=1,4
   24 R(I,J)=0.
  251 REWIND LUN
   25 CONTINUE
      READ (LUN) (X(I),I=1,M)
      TOT=KTOT
      ERROR1=SQRT((6.*TOT*(TOT-1.))/((TOT-2.)*(TOT+1.)*(TOT+3.)))
      ERROR2=SQRT((24.*TOT*(TOT-1.)**2)/((TOT-3.)*(TOT-2.)*(TOT+3.)*(TOT
     1+5.)))
      KNT = KNT+1
      DO 30 I=1,M
      PRODX=1.
      DO 30 J=1,4
      PRODX=PRODX*X(I)
   30 R(I,J) = R(I,J) + PRODX
      IF(KNT-KTOT) 25,35,35
   35 DO 40 J=1,4
      DO 40 I=1,M
```

```
   40 R(I,J) = R(I,J)/FLOAT(KTOT)
      WRITE (LUT,45)
      DO 100 I=1,M
      XM3=R(I,3)-3.*R(I,1)*R(I,2)+2.*R(I,1)**3
      XG1=XM3/(STDEVA(I)**3)
      XM4=R(I,4)-4.*R(I,1)*R(I,3)+6.*(R(I,1)**2)*R(I,2)-3.*R(I,1)**4
      XG2=XM4/(STDEVA(I)**4)-3.
      XXG1 = XG1
      XXG2 = XG2
      XXG1 = XXG1/ERROR1
      XXG2 = XXG2/ERROR2
      WRITE (LUT,46) I,XG1,XXG1,XG2,XXG2
  100 CONTINUE
      WRITE (LUT,52) ERROR1,ERROR2
      DO 55 I=1,M
      DO 55 J=1,4
   55 R(I,J)=0.
      DO 54 I=1,M
   54 STDEVA(I) = STDEVB(I)
      IF(KTOT.EQ.NA) GO TO 56
      IF(KTOT.EQ.NB) GO TO 200
   56 WRITE (LUT,60)
      KNT = 0
C------- SAMPLE B THE SECOND TIME
      KTOT=NB
      LUN=MTB
      GO TO 201
  200 WRITE (LUT,6)
      RETURN
C
C
    6 FORMAT(44H1  DISTRIBUTION DETAILS FOR REFERENCE SAMPLE        )
   10 FORMAT(33H0                    MEANS FOR SAMPLE I1,46H                ST
     1ANDARD DEVIATIONS FOR SAMPLE I1)
   15 FORMAT(15X,F18.5,12X,F18.5)
   45 FORMAT (52H0  VARIABLE  SKEWNESS   T(SKEW)  KURTOSIS   T(KURT)    )
   46 FORMAT (6X,I2,4X,4F10.4)
   52 FORMAT(25H0  ST.DEV.FOR SKEWNESS = ,F14.7/25H0  ST.DEV.FOR KURTOSI
     1S = ,F14.7)
   60 FORMAT(45H1  DISTRIBUTION DETAILS FOR COMPARISON SAMPLE)
      END


      SUBROUTINE JZUP2
      DIMENSION ODATA(1000),BIG(3),SMALL(3)
      COMMON A(45,45),B(45,45),C(45,45),SA(45),SB(45),SC(45),DIFF(45),
     1 T(45),X(45),Z(45),M,NA,NE,K,MW,MATRIX,LDISCR,LOGDEC,INVST(3),IT,
     2 MATPOS,KREYD
      COMMON /UNITS/ LIN,LUT,MTA,MTB,MTMABC,MTARA,MTBRA
      NTOTAL=NA+NB                                                   SPCL1708
      DO 5 I=1,NTOTAL
    5 ODATA(I) = 0.
      DO 45 IREP = 1,2
      IF (IREP .EQ. 2) GO TO 30
C------- SAMPLE A
      LUN=MTA
      JA=1
      NEJ=NA
      GO TO 31
C------- SAMPLE B
```

```
   30 LUN=MTB
      JA=NA+1
      NEJ=NTOTAL
   31 REWIND LUN
      DO 10 II=JA,NEJ
      READ (LUN) (X(I),I=1,M)
      DO 9 IV=1,M
    9 ODATA(II) = SB(IV)*X(IV) + ODATA(II)
   10 CONTINUE
   45 CONTINUE
      NLIM = NTOTAL
      LOWER=1                                                           SPCL1722
      DO 29 IX=1,3                                                      SPCL1724
      GO TO  (62,60,61),IX
   60 IF (INVST(2) .EQ. INVST(1)) GO TO 29
      GO TO 62
   61 IF (INVST(3) .EQ. INVST(1) .OR. INVST(3) .EQ. INVST(2)) GO TO 29
   62 BIG(IX)=ODATA(LOWER)
      SMALL(IX)=ODATA(LOWER)                                            SPCL1730
      DO 22 IV = LOWER,NLIM
      IF(ODATA(IV).GT.BIG(IX)) BIG(IX)=ODATA(IV)                        SPCL1734
      IF(ODATA(IV).LT.SMALL(IX)) SMALL(IX)=ODATA(IV)                    SPCL1736
   22 CONTINUE                                                          SPCL1738
      GO TO 20                                                          SPCL1740
   20 CALL PLTHS1(ODATA(LOWER),NTOTAL,SMALL(IX),BIG(IX),INVST(IX))
   29 CONTINUE
   55 RETURN
      END                                                               SPCL1760

      SUBROUTINE MATOUT (STMAT)
C------- MATOUT PRINTS OUT THE M*M MATRIX STMAT ON THE PRINTER UNIT LUT
      DIMENSION STMAT(45,45)
      COMMON A(45,45),B(45,45),C(45,45),SA(45),SB(45),SC(45),DIFF(45),
     1 T(45),X(45),Z(45),M,NA,NE,K,MW,MATRIX,LDISCR,LOGDEC,INVST(3),IT,
     2 MATPOS,KREYD
      COMMON /UNITS/ LIN,LUT,MTA,MTB,MTMABC,MTARA,MTBRA
C------- MATOUT PRINTS OUT THE MATRIX STMAT ON LOGICAL UNIT LUT.
      MEND=0
    1 MBEG=MEND+1
      MEND=MBEG+9
C------- PRINT MAXIMUM 10 COLUMNS EACH TIME
      IF (MEND .GT. M) MEND=M
      WRITE (LUT,101) (J,J=MBEG,MEND)
      DO 2 I=1,M
    2 WRITE (LUT,102) I,(STMAT(I,J),J=MBEG,MEND)
C------- IF ALL M COLUMNS HAVE BEEN PRINTED THEN RETURN
      IF (MEND .EQ. M) RETURN
C--------OR ELSE PRINT NEXT 10 COLUMNS
      GO TO 1
  101 FORMAT (1H0,10I12)
  102 FORMAT (2X,I2,2X,10F12.5)
      END
```

APPENDIX B. –Example of Output from Program.

ANALYSIS OF HOMOGENEITY OF COVARIANCE MATRICES
TEST OF NEW ORNTDIST FOR 45*45 MATRICES, GOTLAND.

DATA CARDS READ WITH VARIABLE FORMAT (F9.2,3F10.2)

REFERENCE SAMPLE SIZE BASED ON POPULATION OF 128

COMPARISON SAMPLE SIZE IS 95

FIRST DATA INPUT SET

735.0000000	720.0000000	615.0000000	435.0000000
715.0000000	660.0000000	580.0000000	430.0000000
710.0000000	700.0000000	625.0000000	360.0000000
770.0000000	755.0000000	655.0000000	405.0000000
770.0000000	730.0000000	640.0000000	450.0000000
720.0000000	660.0000000	565.0000000	395.0000000
765.0000000	715.0000000	650.0000000	460.0000000
760.0000000	735.0000000	650.0000000	455.0000000
740.0000000	725.0000000	655.0000000	440.0000000
760.0000000	740.0000000	675.0000000	410.0000000
775.0000000	730.0000000	610.0000000	360.0000000
710.0000000	700.0000000	610.0000000	340.0000000
705.0000000	675.0000000	600.0000000	435.0000000
790.0000000	740.0000000	660.0000000	420.0000000
740.0000000	710.0000000	575.0000000	460.0000000
715.0000000	670.0000000	590.0000000	385.0000000
765.0000000	705.0000000	635.0000000	470.0000000
760.0000000	740.0000000	610.0000000	415.0000000
745.0000000	725.0000000	655.0000000	425.0000000
730.0000000	700.0000000	635.0000000	410.0000000
805.0000000	745.0000000	655.0000000	465.0000000
780.0000000	725.0000000	630.0000000	420.0000000
775.0000000	740.0000000	650.0000000	735.0000000
760.0000000	720.0000000	650.0000000	410.0000000
755.0000000	705.0000000	610.0000000	385.0000000
805.0000000	720.0000000	650.0000000	460.0000000
740.0000000	690.0000000	570.0000000	400.0000000
730.0000000	695.0000000	620.0000000	405.0000000
770.0000000	735.0000000	645.0000000	400.0000000
755.0000000	755.0000000	600.0000000	450.0000000
765.0000000	770.0000000	625.0000000	410.0000000
755.0000000	710.0000000	630.0000000	400.0000000
760.0000000	720.0000000	645.0000000	435.0000000
780.0000000	750.0000000	620.0000000	410.0000000
780.0000000	735.0000000	675.0000000	370.0000000
800.0000000	730.0000000	595.0000000	400.0000000
750.0000000	685.0000000	585.0000000	425.0000000
750.0000000	740.0000000	610.0000000	400.0000000
750.0000000	730.0000000	620.0000000	395.0000000
720.0000000	690.0000000	550.0000000	395.0000000
735.0000000	750.0000000	640.0000000	415.0000000
735.0000000	685.0000000	560.0000000	380.0000000
760.0000000	680.0000000	605.0000000	440.0000000
800.0000000	720.0000000	685.0000000	420.0000000
795.0000000	750.0000000	650.0000000	435.0000000
730.0000000	715.0000000	630.0000000	410.0000000

775,0000000	775,0000000	630,0000000	450.0000000
790,0000000	740,0000000	615,0000000	425.0000000
780,0000000	775,0000000	655,0000000	385.0000000
765,0000000	735,0000000	645,0000000	420.0000000
780,0000000	810,0000000	660,0000000	410.0000000
775,0000000	765,0000000	630,0000000	360.0000000
790,0000000	760,0000000	640,0000000	470.0000000
825.0000000	800,0000000	690,0000000	430.0000000
740,0000000	720.0000000	645,0000000	430.0000000
775.0000000	795.0000000	670,0000000	380.0000000
760,0000000	785,0000000	650,0000000	380.0000000
745,0000000	735,0000000	650,0000000	400,0000000
730,0000000	705,0000000	600,0000000	475,0000000
800,0000000	750,0000000	635,0000000	470.0000000
750,0000000	770,0000000	620,0000000	410.0000000
780,0000000	770.0000000	630,0000000	415.0000000
745,0000000	675.0000000	595,0000000	365,0000000
735,0000000	755.0000000	625,0000000	390,0000000
780,0000000	735,0000000	650,0000000	425.0000000
740,0000000	765,0000000	595,0000000	375.0000000
760,0000000	705,0000000	650,0000000	440.0000000
730,0000000	660,0000000	565,0000000	405.0000000
735.0000000	670,0000000	650,0000000	410.0000000
715,0000000	710,0000000	650,0000000	410.0000000
720,0000000	690,0000000	585,0000000	405.0000000
710,0000000	645,0000000	580,0000000	410,0000000
750,0000000	750,0000000	600,0000000	415.0000000
730,0000000	665,0000000	630,0000000	390.0000000
780,0000000	735,0000000	635,0000000	395.0000000
770,0000000	765,0000000	660,0000000	385.0000000
755,0000000	740.0000000	615,0000000	450,0000000
725,0000000	730.0000000	610,0000000	415.0000000
695,0000000	670,0000000	630,0000000	375.0000000
755,0000000	670.0000000	595,0000000	410.0000000
740,0000000	735.0000000	635,0000000	425.0000000
705,0000000	665.0000000	575,0000000	425.0000000
660,0000000	700,0000000	630,0000000	405.0000000
690,0000000	690.0000000	590,0000000	370.0000000
740,0000000	715,0000000	670,0000000	390.0000000
740,0000000	670,0000000	635,0000000	385.0000000
715.0000000	685.0000000	580,0000000	420.0000000
745,0000000	690,0000000	580,0000000	400.0000000
745,0000000	720,0000000	650,0000000	430.0000000
735,0000000	715,0000000	630,0000000	450.0000000
700,0000000	725.0000000	610,0000000	405.0000000
735,0000000	715,0000000	600,0000000	470.0000000
775,0000000	775.0000000	695,0000000	450.0000000
750,0000000	730,0000000	605,0000000	385.0000000
735,0000000	730.0000000	605,0000000	390.0000000
765,0000000	700,0000000	660,0000000	420.0000000
725,0000000	730,0000000	650,0000000	470.0000000
795,0000000	790,0000000	660,0000000	490.0000000
770,0000000	725,0000000	695,0000000	425.0000000
810,0000000	780,0000000	645,0000000	450.0000000
745,0000000	775,0000000	640,0000000	470.0000000
780,0000000	720,0000000	610,0000000	425.0000000
770,0000000	745.0000000	635,0000000	450.0000000
740,0000000	720,0000000	670,0000000	420.0000000
770,0000000	770.0000000	625,0000000	430.0000000
750,0000000	720.0000000	660,0000000	435.0000000

760,0000000	745.0000000	645,0000000	380.0000000
745,0000000	725.0000000	625,0000000	370.0000000
805,0000000	820.0000000	690,0000000	440.0000000
790,0000000	780,0000000	640,0000000	395.0000000
705,0000000	680.0000000	600,0000000	390.0000000
725,0000000	745.0000000	680,0000000	435.0000000
730,0000000	690,0000000	600,0000000	390.0000000
765,0000000	760.0000000	640,0000000	415.0000000
690,0000000	720,0000000	595,0000000	395.0000000
740,0000000	695.0000000	630,0000000	400.0000000
705,0000000	695.0000000	600,0000000	435.0000000
750,0000000	740,0000000	605,0000000	420.0000000
670,0000000	635,0000000	575,0000000	400.0000000
690,0000000	645,0000000	600,0000000	425.0000000
730,0000000	710,0000000	600,0000000	450.0000000
750,0000000	700,0000000	610,0000000	370.0000000
680,0000000	645,0000000	520,0000000	365.0000000
765,0000000	750.0000000	640,0000000	450.0000000
695,0000000	690,0000000	610,0000000	405.0000000
715,0000000	720,0000000	600,0000000	440.0000000
735,0000000	735,0000000	655,0000000	400.0000000
725,0000000	725,0000000	630,0000000	420.0000000

SECOND DATA INPUT SET

690,0000000	880.0000000	720,0000000	560.0000000
810,0000000	1025.0000000	870,0000000	535.0000000
720,0000000	890.0000000	820,0000000	580.0000000
830.0000000	1100.0000000	875,0000000	520.0000000
750,0000000	945.0000000	865,0000000	520.0000000
775,0000000	995,0000000	890,0000000	520.0000000
745,0000000	880,0000000	810,0000000	485.0000000
810.0000000	980.0000000	820,0000000	580.0000000
755.0000000	975.0000000	790,0000000	590.0000000
770,0000000	995,0000000	850,0000000	510.0000000
690,0000000	945.0000000	740,0000000	490.0000000
725.0000000	895.0000000	775,0000000	545.0000000
745,0000000	930,0000000	810,0000000	490.0000000
780,0000000	980.0000000	820,0000000	495.0000000
725,0000000	990.0000000	770,0000000	420.0000000
805,0000000	985.0000000	765,0000000	480.0000000
745.0000000	915.0000000	820,0000000	490.0000000
715,0000000	945,0000000	810,0000000	540.0000000
690.0000000	910.0000000	700,0000000	510.0000000
770.0000000	930,0000000	815,0000000	520.0000000
780,0000000	890.0000000	860,0000000	480.0000000
695,0000000	895.0000000	775,0000000	440.0000000
815,0000000	910.0000000	835,0000000	515.0000000
755,0000000	950.0000000	830,0000000	535.0000000
745.0000000	900.0000000	840,0000000	495.0000000
710,0000000	860.0000000	720,0000000	440.0000000
820,0000000	1020,0000000	845,0000000	525.0000000
745,0000000	920.0000000	810,0000000	470.0000000
800,0000000	1055.0000000	850,0000000	520.0000000
730,0000000	955.0000000	805,0000000	515.0000000
870,0000000	1105,0000000	940,0000000	500.0000000
770.0000000	995.0000000	825,0000000	540.0000000
835.0000000	1085.0000000	930,0000000	500.0000000
785.0000000	985.0000000	805,0000000	450.0000000
735.0000000	900.0000000	855,0000000	530.0000000
680,0000000	910.0000000	775,0000000	540.0000000

760,0000000	965.0000000	770,0000000	575.0000000
745.0000000	965.0000000	805,0000000	530.0000000
760,0000000	990.0000000	820,0000000	535.0000000
720,0000000	925.0000000	835,0000000	525.0000000
785,0000000	925.0000000	860,0000000	510.0000000
775,0000000	950.0000000	850,0000000	540.0000000
725.0000000	880.0000000	800,0000000	430.0000000
770,0000000	980.0000000	865,0000000	550.0000000
680,0000000	860.0000000	790,0000000	465.0000000
720,0000000	900,0000000	765,0000000	510.0000000
745,0000000	915.0000000	855,0000000	530.0000000
745,0000000	930.0000000	840,0000000	505.0000000
760,0000000	920,0000000	825,0000000	510.0000000
760,0000000	915.0000000	790,0000000	505.0000000
760,0000000	1015.0000000	805,0000000	470,0000000
750,0000000	995,0000000	840,0000000	515.0000000
755.0000000	920.0000000	885,0000000	590.0000000
750.0000000	910,0000000	860,0000000	485.0000000
760,0000000	925,0000000	765,0000000	540.0000000
770,0000000	940,0000000	865,0000000	510,0000000
765,0000000	900,0000000	815,0000000	560.0000000
715,0000000	870.0000000	820,0000000	530.0000000
710.0000000	920,0000000	770,0000000	490.0000000
790,0000000	950.0000000	865,0000000	515.0000000
770.0000000	870.0000000	950,0000000	680.0000000
780.0000000	1025.0000000	865,0000000	570.0000000
760,0000000	960,0000000	860,0000000	540.0000000
685,0000000	870,0000000	750,0000000	485.0000000
780,0000000	995,0000000	860,0000000	600.0000000
755.0000000	980.0000000	850,0000000	505.0000000
730,0000000	940,0000000	835,0000000	555.0000000
790,0000000	1030,0000000	920,0000000	525.0000000
740,0000000	955,0000000	870,0000000	505,0000000
760,0000000	990.0000000	860,0000000	550.0000000
745,0000000	955.0000000	815,0000000	545.0000000
750,0000000	955.0000000	805,0000000	540.0000000
745,0000000	985,0000000	810,0000000	610.0000000
735,0000000	945.0000000	810,0000000	480.0000000
815.0000000	1060,0000000	860,0000000	580.0000000
740,0000000	925,0000000	850,0000000	535.0000000
730.0000000	940.0000000	760,0000000	510.0000000
740,0000000	935,0000000	830,0000000	520.0000000
670,0000000	865,0000000	710,0000000	470.0000000
725.0000000	885,0000000	815,0000000	500.0000000
690.0000000	880.0000000	790,0000000	490.0000000
685,0000000	875,0000000	810,0000000	510.0000000
710,0000000	870.0000000	760,0000000	480.0000000
670,0000000	850,0000000	785,0000000	445.0000000
755.0000000	960.0000000	845,0000000	445.0000000
745.0000000	920,0000000	840,0000000	470.0000000
650.0000000	800,0000000	695,0000000	440,0000000
760,0000000	900.0000000	835,0000000	445.0000000
700,0000000	905,0000000	790,0000000	515.0000000
670,0000000	850,0000000	735,0000000	520.0000000
765.0000000	950,0000000	850,0000000	525.0000000
675.0000000	850,0000000	750,0000000	485.0000000
730,0000000	880,0000000	795,0000000	495.0000000
680,0000000	865,0000000	760,0000000	580.0000000
765.0000000	945,0000000	805,0000000	555.0000000

DISTRIBUTION DETAILS FOR REFERENCE SAMPLE

MEANS FOR SAMPLE 1	STANDARD DEVIATIONS FOR SAMPLE
748.47656	31,06736
722.92969	36,68083
625.85938	32,03166
417.85156	40,66709

MEANS FOR SAMPLE 2	STANDARD DEVIATIONS FOR SAMPLE
746.15789	41,29198
938.47368	57,92546
818.15789	49,60510
515.84211	43,00948

VARIABLE	SKEWNESS	T(SKEW)	KURTOSIS	T(KURT)
1	-0,2013	-0.9403	-0,1028	-0.2419
2	0,0007	0.0032	-0,1935	-0.4554
3	-0,2992	-1.3981	0,0725	0.1706
4	3,6562	17.0832	26,6019	62.6037

ST.DEV,FOR SKEWNESS = 0,2140261

ST.DEV.FOR KURTOSIS = 0,4249253

DISTRIBUTION DETAILS FOR COMPARISON SAMPLE

VARIABLE	SKEWNESS	T(SKEW)	KURTOSIS	T(KURT)
1	0,0878	0.3549	0,0946	0.1930
2	0,5615	2.2690	0,3348	0.6830
3	-0.0905	-0.3658	0,2707	0.5522
4	0,4673	1.8882	1,2869	2.6254

ST.DEV.FOR SKEWNESS = 0,2474635

ST.DEV,FOR KURTOSIS = 0.4901701

COVARIANCE MATRIX 1 (REFERENCE MATRIX)

ROW	1	2	3	4
1	965,18055	778.90779	551,71321	327,60673
2	778,90779	1345.48321	709,66720	278,98161
3	551,71321	709.66720	1026,02731	282,37266
4	327,60673	278.98161	282,37266	1653,81244

COVARIANCE MATRIX 2 (COMPARISON MATRIX)

ROW	1	2	3	4
1	1705,02800	1889.81803	1490,71949	454,33371
2	1889,81803	3355.35836	1623,48824	548,37346
3	1490,71949	1623.48824	2460,66630	706,88691
4	454,33371	548.37346	706,88691	1849,81523

ROW POOLED MATRIX

	1	2	3	4
1	1279,86679	1251.42165	951,10955	381,50871
2	1251,42165	2200,36223	1098,35126	393,56457
3	951,10955	1098.35126	1636,23575	462,93528
4	381,50871	393.56457	462,93528	1737,18014

*****

*****

EIGENVALUES FOR REFERENCE MATRIX
2740,99516 1423.34108 479,04893 347,11834

EIGENVECTORS FOR REFERENCE MATRIX

	1	2	3	4
1	-0,48781	0,17283	0,35113	0,78030
2	-0,59810	0.33807	0,37958	-0,61960
3	-0,47418	0.20227	-0,85575	0,04385
4	-0,42363	-0.90273	0,01763	-0,07282

*****

EIGENVALUES FOR COMPARISON MATRIX
6176.19985 1725.86087 1129,30396 339,50320

EIGENVECTORS FOR COMPARISON MATRIX

	1	2	3	4
1	-0,47962	-0,11076	0,08636	0,86616
2	-0,66687	-0.38264	-0,52438	-0,36591
3	-0,52583	0.21152	0,75090	-0,33898
4	-0,22081	0.89251	-0,39206	0,03095

*****

*****

EIGENVALUES FOR POOLED MATRIX
4175.13456 1552.62951 765,53415 360,34669

EIGENVECTORS FOR POOLED MATRIX

	1	2	3	4
1	-0,48626	0,12064	0,01965	0,86522
2	-0,64950	0,29287	0,56313	-0,41865
3	-0,51393	0.02954	-0,81219	-0,27450
4	-0,27853	-0.94805	0,15115	-0,02778

LOG DETERMINANTS

REFERENCE	COMPARISON	POOLED
27.198	29.039	28.212

B SQUARE = 51.0763999
BETA SQUARE = 0.2004864
DEGREES OF FREEDOM = 10

TRACE B = 9370.8678775

PERCENTAGES FOR B(I)

65.9085149	18.4173002	12.0512206	3.6229643

ORIENTATION OF ELLIPSOIDS

VECTOR	CHISQUARE	DF
1	13.901	3
2	11.784	3
3	34.449	3
4	36.211	3

TEST = 23.2100000

STATISTICAL DISTANCE FOR HETEROGENEOUS COVARIANCE MATRICES

NA =128   NB = 95

ITERATIONS FOR DISTANCE CALCULATION
SAMPLE DIFFERENCE MEAN VECTOR
-2.31867
215.54400
192.29852
97.99054

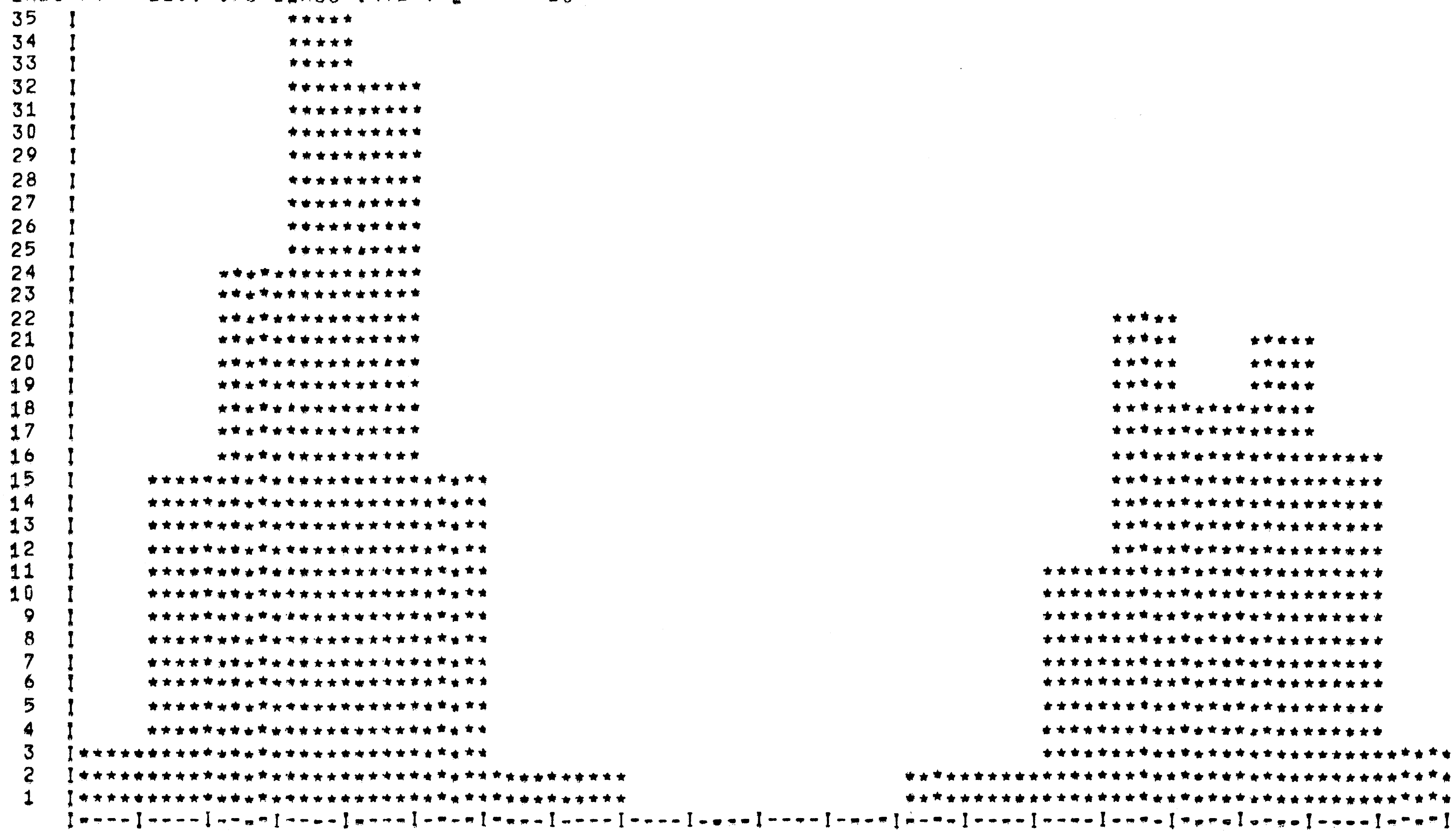
ITERATION	Z	D	DSQ	DIFF T
0	-3.61526	8.792	77.304	0.0000000
0	36.21169	8.745	76.470	0.0000000
0	15.32475	8.784	77.164	0.0000000
0	5.67311	8.792	77.292	0.0000000
4	0.98963	8.793	77.311	0.0625000
5	-1.32195	8.793	77.311	0.0312500
6	-0.16852	8.793	77.312	0.0156250
7	0.40995	8.793	77.312	0.0078125
8	0.12056	8.793	77.312	0.0039063
9	-0.02402	8.793	77.312	0.0019531
10	0.04826	8.793	77.312	0.0009766

ESTIMATE OF DISCRIMINANT VECTOR
-0.32796
0.19091
0.16544
0.03858

HISTOGRAM FOR DISCRIMINANTS

INTERVAL = 5.9571 DATA STARTS AT -9.1373

ENDS AT 110.0053 CLASS INTERVALS = 20



HETEROGENEOUS DISTANCE BY AVERAGING COVARIANCE MATRICES

ROW COVARIANCE MATRIX

ROW	1	2	3	4
1	1335.10427	1334.36291	1021.21635	390.97022
2	1334.36291	2350.42078	1166.57772	413.67753
3	1021.21635	1166.57772	1743.34680	494.62978
4	390.97022	413.67753	494.62978	1751.81383

DISTANCE, DISTANCE SQUARED
8.783 77.134

HETEROGENEOUS SIGNIFICANCES

MULTIPLICATION FACTOR

AN = 0.8615030

ITERATION	REYMENTS DSQ TEST
1	83.7096
2	68.4095
3	74.0644
4	81.8624
5	81.0875
6	85.7856
7	82.6404
8	81.9864
9	71.4774
10	87.0855
11	68.8185
12	97.1551
13	90.7586
14	79.9636
15	78.6520
16	76.5856
17	100.2027
18	97.7471
19	84.5251
20	98.4155
21	88.5189
22	97.8546
23	90.6790
24	81.1956
25	91.8051
26	80.7465
27	77.5857
28	92.5904
29	93.7228
30	131.5444

SIGNIFICANCE COMPUTATIONS

AVE.DSQ	TSQ
83.780	3979.543

VARIANCE RATIO IS = 963.13 FOR 94 AND 4 DEGREES OF FREEDOM

STATISTICAL DISTANCE

VAR	N1 =128 MEAN VECTOR 1	N2 = 95 MEAN VECTOR 2	DF =221 DIFF.VECTOR
1	748,477	-0,328	-2,319
2	722,930	0,191	215,544
3	625,859	0,165	192,299
4	417,852	0,039	97,991

DISCRIMINATOR COEFFICIENTS

-0,32631 0,19354 0.16597 0,03999

RESULTS FOR MAHANALOBIS DSQUARE

D	D SQUARE
8,84919	78,30818

SIGNIFICANCE FOR D SQUARE AND T SQUARE

T SQUARE = 4270,078

F = 1053.03

DF1 = 4 DF2 =218

HISTOGRAM FOR DISCRIMINANTS

INTERVAL = 6.0133 DATA STARTS AT -5.0137

ENDS AT 115.2513 CLASS INTERVALS = 20

KANSAS GEOLOGICAL SURVEY COMPUTER PROGRAM
THE UNIVERSITY OF KANSAS, LAWRENCE

PROGRAM ABSTRACT

Title (If subroutine state in title): FORTRAN IV program for generalized statistical distance and analysis of covariance matrices for the CDC 3600 computer.

Date: Final version in March 1969

Author, organization: R.A. Reyment and Hans-Ake Ramden (University of Uppsala), and Warren J. Wahlstedt (Kansas Geological Survey)

Direct inquiries to: Authors or

Name: D.F. Merriam   Address: Kansas Geological Survey
University of Kansas, Lawrence, Kansas

Purpose/description: For computing generalized distances and discriminant functions in the situations of homogeneous or heterogeneous covariance matrices and tests for homogeneity of covariance matrices.

Mathematical method: Generalized distances by the methods of Mahanalobis, Anderson and Bahadur, and Reyment.

Restrictions, range: The programs accept matrices up to 45 x 45

Computer manufacturer: CDC   Model: 3600

Programming language:

Memory required: 32 K   Approximate running time:

Special peripheral equipment required:

Remarks (special compilers or operating systems, required word lengths, number of successful runs, other machine versions, additional information useful for operation or modification of program) Subroutine REYDST requires access to a random access unit to be efficient. Running time of the example was about 15 minutes.

(continued from inside front cover)

9. FORTRAN IV program for mathematical simulation of marine sedimentation with IBM 7040 or 7094 computers, by J.W. Harbaugh and W.J. Wahlstedt, 1967 . . . . . $1.00
10. Three-dimensional response surface program in FORTRAN II for the IBM 1620 computer, by R.J. Sampson and J.C. Davis, 1967 . . . . $0.75
11. FORTRAN IV program for vector trend analyses of directional data, by W.T. Fox, 1967 . $1.00
12. Computer applications in the earth sciences: Colloquium on trend analysis, edited by D.F. Merriam and N.C. Cocke, 1967 . . . . $1.00
13. FORTRAN IV computer programs for Markov chain experiments in geology, by W.C. Krumbein, 1967 . . . . $1.00
14. FORTRAN IV programs to determine surface roughness in topography for the CDC 3400 computer, by R.D. Hobson, 1967 . . . . $1.00
15. FORTRAN II program for progressive linear fit of surfaces on a quadratic base using an IBM 1620 computer, by A.J. Cole, C. Jordan, and D.F. Merriam, 1967 . . . . . $1.00
16. FORTRAN IV program for the GE 625 to compute the power spectrum of geological surfaces, by J.E. Esler and F.W. Preston, 1967 . . . . $0.75
17. FORTRAN IV program for Q-mode cluster analysis of nonquantitative data using IBM 7090/7094 computers, by G.F. Bonham-Carter, 1967 . . . . $1.00
18. Computer applications in the earth sciences: Colloquium on time-series analysis, D.F. Merriam, editor, 1967 . . . . $1.00
19. FORTRAN II time-trend package for the IBM 1620 computer, by J.C. Davis and R.J. Sampson, 1967 . . . . $1.00
20. Computer programs for multivariate analysis in geology, D.F. Merriam, editor, 1968 . $1.00
21. FORTRAN IV program for computation and display of principal components, by W.J. Wahlstedt and J.C. Davis, 1968 . . . . $1.00
22. Computer applications in the earth sciences: Colloquium on simulation, D.F. Merriam and N.C. Cocke, editors, 1968 . . . . $1.00
23. Computer programs for automatic contouring, by D.B. McIntyre, D.D. Pollard, and R. Smith, 1968 . . . . $1.50
24. Mathematical model and FORTRAN IV program for computer simulation of deltaic sedimentation, by G.F. Bonham-Carter and A.J. Sutherland, 1968 . . . . . . $1.00
25. FORTRAN IV CDC 6400 computer program for analysis of subsurface fold geometry, by E.H.T. Whitten, 1968 . . . . $1.00
26. FORTRAN IV computer program for simulation of transgression and regression with continuous-time Markov models, by W.C. Krumbein, 1968 . . . . $1.00
27. Stepwise regression and nonpolynomial models in trend analysis, by A.T. Miesch and J.J. Connor, 1968 . . . . $1.00
28. KWIKR8 a FORTRAN IV program for multiple regression and geologic trend analysis, by J.E. Esler, P.F. Smith, and J.C. Davis, 1968 . . . . $1.00
29. FORTRAN IV program for harmonic trend analysis using double Fourier series and regularly gridded data for the GE 625 computer, by J.W. Harbaugh and M.J. Sackin, 1968 . $1.00
30. Sampling a geological population (workshop on experiment in sampling), by J.C. Griffiths and C.W. Ondrick, 1968 . . . . $1.00
31. Multivariate procedures and FORTRAN IV program for evaluation and improvement of classifications, by Ferruh Demirmen, 1969 . . . . $1.00
32. FORTRAN IV programs for canonical correlation and canonical trend-surface analysis, by P.J. Lee, 1969 . . . . $1.00
33. FORTRAN IV program for construction of Pi diagrams with the Univac 1108 computer, by Jeffrey Warner, 1969 . . . . $1.00
34. FORTRAN IV program for nonlinear estimation, by R.B. McCammon, 1969 . . . . $0.75
35. FORTRAN IV computer program for fitting observed count data to discrete distribution models of binomial, Poisson and negative binomial, by C.W. Ondrick and J.C. Griffiths, 1969 . . . . $0.75
36. GRAFPAC, graphic output subroutines for the GE 635 computer, by F.J. Rohlf, 1969 . $1.00
37. An iterative approach to the fitting of trend surfaces, by A.J. Cole $1.00
38. FORTRAN II programs for 8 methods of cluster analysis (CLUSTAN I), by David Wishart, 1969 . . . . $1.50
39. FORTRAN IV program for the generalized statistical distance and analysis of covariance matrices for the CDC 3600 computer, by R.A. Reyment, Hans-Ake Ramden, and Warren J. Wahlstedt $1.00