AD743108

Reproduced by
NATIONAL TECHNICAL
INFORMATION SERVICE
Springfield, Va. 22151

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) Univeristy of Georgia	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP Unclassified

3. REPORT TITLE

AN INTERACTIVE MULTIVARIATE DATA ANALYSIS PROGRAM

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

5. AUTHOR(S) (First name, middle initial, last name)

ANNE A. BALLENGEE and ROLF E. BARGMANN

6. REPORT DATE March 1972	7a. TOTAL NO. OF PAGES 150	7b. NO. OF REFS 5
8a. CONTRACT OR GRANT NO N00014-69-A-~~042-261~~ 0423 b. PROJECT NO NR 042-261 c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) Technical Report No. 80 Department of Statistics 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) THEMIS No. 18	

10. DISTRIBUTION STATEMENT

Reproduction in whole or in part is permitted for any purpose of the United States Government.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Office of Naval Research, Washington, D. C.

13. ABSTRACT

This report contains description and program documentation of a conversational unit for univariate and multivariate analysis of data in regular or irregular two-way classification designs. The main part of this report is Chapter IV which, on the basis of detailed illustrations, describes the question-and-answer frames on a Graphics (IBM 2250) console. The user is assumed to be a "layman" in the sense that he need not be familiar with statistical analysis in computer programming techniques. He receives instructions for description of his data from the graphically displayed questions. He has a choice to make several plots of his data (for cells, rows, or columns of the design, each response variable vs. another response variable). In the univariate analysis he obtains detailed reports on means, adjusted means, and the analysis of variance tables, for each response variable. After data have been edited the user may perform a multivariate analysis and will obtain discriminant functions, union-intersection and likelihood-ration test statistics, and correlations. He may repeat this for various subsets of response variables. Thus, this unit includes and supersedes the unit described in THEMIS report No. 13 (April, 1971). Program documentation is presented in Chapter V. The listing of the FORTRAN programs is contained in the appendix.

DD FORM 1 NOV 65 1473 (PAGE 1)

S/N 0101-807-6811

Unclassified

Security Classification

A-31408

14 KEY WORDS	LINK A ROLL	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
Statistics						
Analysis of Variance						
Multivariate Analysis						
Data Analysis						
Design of Experiments						
Discriminant Analysis						
Interactive Computation						
Conversational Computation						
Computer Graphics						

THEMIS REPORT NUMBER 18

TECHNICAL REPORT NUMBER 80

AN INTERACTIVE MULTIVARIATE DATA ANALYSIS PROGRAM

ANNE A. BALLENGEE AND ROLF E. BARGMANN

Reproduction in whole or in part is permitted for any purpose of the United States Government. This Research was supported, in part, by the Office of Naval Research, Contract Number N00014-69-A-0423, NR 042-261.

ROLF E. BARGMANN
PRINCIPAL INVESTIGATOR

D D C
RECEIVED
JUN 7 1972
B

THE UNIVERSITY OF GEORGIA
DEPARTMENT OF STATISTICS AND COMPUTER SCIENCES
ATHENS, GEORGIA 30601

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

MARCH 1972

TABLE OF CONTENTS

CHAPTER		PAGE
I	INTRODUCTION	1
II	DESCRIPTION OF THE CONVERSATIONAL UNIT	4
III	TOOLS FOR STATISTICAL ANALYSIS	7
	3.1 Program ANOT	7
	3.2 Program TEMAT	11
	3.3 Distribution Routines	13
IV	TOOLS FOR THE USER	15
	4.1 User's Guide	15
	4.2 Example	19
V	COMPUTER PROGRAMS	57
REFERENCES		81
APPENDIX		82

CHAPTER I

INTRODUCTION

The use of the computer in the statistical analysis of experimental data is usually available only to people, or teams, who have statistical and computational experience. In particular, the experimenter who collects the physical data can rarely do the analysis on the computer without help.

The ever-increasing use of terminals, both typewriter consoles and graphic display units, appears to become a promising link between the experimenter and the analysis of his data. Of course, these terminals need to be programmed by experts in computation and statistical analysis before they can serve the function of analyzing the experimenter's data. Too often, specialists in some field with a smattering knowledge of statistics and some elementary programming experience "write their own subroutines". There are literally thousands of "least squares" and "regression" programs embodied in terminal systems. In view of the utter simplicity of this problem, most would probably work all right, a few may be inefficient (especially where factorial designs are irregular and the inexperienced statistical layman would use a "general linear model" approach for their problem setting up, say, a 20 by 20 equation system where a statistician would obtain the same result from a 3 by 3 equation system), still others probably lack precision in any but the smallest studies.

Another problem in this universal practice of laymen writing their own ad hoc program, or using a poorly understood package, is that they apply inappropriate techniques. The more advanced tools of analysis which should be used on their data are forever inaccessible to them. Our approach to this problem has been to have experienced statisticians write conversational units addressed to the experimenter with supposedly incomplete knowledge of statistics and computation. On the basis of answers given to these questions, the computational program will perform the statistical analysis warranted by the data.

In the present thesis we deal with the general complex of "analysis of variance" in one-way and two-way (irregular) classification designs, involving either univariate or multivariate responses. There exists a variety of programs for this set of analysis of which most are for use in the batch mode. One major problem, however, especially in the multivariate pass (multivariate analysis of variance, discriminant analysis) is the disturbing influence of even very few faulty data, i.e., data which may have been transcribed incorrectly or contain keypunch errors, such as misplaced decimal points, or outliers or other types of mavericks. Hence, before any meaningful analysis can be made the investigator must be sure that he has good data.

Since it may be time-consuming to perform an initial analysis of variance, then in a new run to edit the data, and thereafter to perform more analyses, a conversational approach to the problem seems to be indicated. The conversational unit available to us is an IBM 2250 Graphics console which is connected with an IBM 360 Model 65 computer. The 2250 display unit has an alphameric keyboard and a program function

keyboard to serve as input devices, and a cathode-ray tube as an output device. Thus, a communication exists between the user and the computer.

This thesis presents a brief description of the classical approach of adjusted normal equations which is employed in the analysis of variance and its multivariate extension. The user is then presented with a description example of how the program can be used. Finally, the computer programs are documented.

Special notation which will be used:

μ incidicence matrix

n_{ij} cell size

$\underline{n}_{.j}$ vector of column sizes

$\underline{n}_{i.}$ vector of row sizes

n sample size

$\underline{R}$ vector of row totals

$\underline{c}$ vector of column totals

G grand total

$Dn_{i.}$ diagonal matrix with elements of $\underline{n}_{i.}$ in principle diagonal

See references [2] and [3] for further explanations.

CHAPTER II

DESCRIPTION OF THE CONVERSATIONAL UNIT

The user, at the console, is asked to supply some general information about his data (title, variable names, transformations, factor names, level codes, etc.). Second, he must enter his data either from the typewriter or by a special batch run if his data are voluminous.

On each response variable, a univariate analysis of variance is then performed (assuming, in any event, an irregular two-way classification model - see Section 3.1). In addition to the usual analysis of variance and the customary F statistics, the display contains a report of means in each cell, size (incidence) of each cell, standard deviations of observations within each cell, adjusted row means, adjusted column means, and various other quantities. Some hints are given in a special instructional display as to how the user should utilize these results in order to identify faulty data.

Next, the user is given an option to view selected graphical displays of his data (data in a cell, data in a row or column, all data; each response variable against each other response variable). This enables him to detect unusual behavior of some observations. He may then go back to a display of his original data, and make any changes which he desires on the basis of the study of tables of means and the graphs. He may then, if he chooses, perform another univariate pass for further inspection.

After thus editing his data, the user may now proceed to a multivariate analysis. He is asked, specifically, which variables he wishes to include in this, or any subsequent, multivariate pass [5]. This is an important aspect of our program in that, all too often, multivariate analyses are badly distorted by some variable which should not be included in the same study (e.g., when one response variable is a mathematical function of others; if it is linear, of course, the entire analysis would be false).

The output display of the multivariate pass contains the usual quantities of multivariate analysis, i.e., the Likelihood-Ratio test statistic for each effect, the Union-Intersection statistic ("canonical R^2 as goodness-of-fit"), the discriminant function, and most important, the correlation of each response variable versus the discriminant function [2]. In addition, the display contains correlation matrices between the selected response variables based on sums of squares and products for (a) error, (b) total, (c) rows plus error, (d) columns plus error.

After viewing the multivariate analysis displays the user may perform additional univariate and multivariate analyses; he may see his displays again and do additional editing. He may continue until he is satisfied that he has viewed his data from every conceivable angle.

In closing it is important to note that the CPU time needed for the analyses is quite short (a few seconds only even for the multivariate pass of extensive sets of data). Our console, as most other conversational terminals, operates in a time shared mode under a special monitor

[4]. Thus, the user has the ability to spend considerable time in front of the console, without requesting computer time any longer than he would have in batch mode. In fact, by employment of programs designed for speed [1] the computer time is of one or more orders of magnitude shorter than that required for other statistical packages.

A more detailed description of the flow of the analysis is presented in Chapter 4.

CHAPTER III

TOOLS FOR STATISTICAL ANALYSIS

In this chapter, we describe those subprograms which employ special statistical techniques of analysis. The other subprograms which deal with standard mathematical algorithms (inversion, eigenvectors, numerical integration) or data processing, are summarized in Chapter 5.

3.1 Program ANOT

Analysis of Irregular Two-Way Classification Designs

The model for an irregular two way design with interaction is

$$E(y_{ijk}) = \mu + \alpha_i + \beta_j + \delta_{ij}$$

where y_{ijk} is the kth observation in the ij^{th} cell,

$i = 1,\ldots,r$

$j = 1,\ldots,c$

$k = 1,\ldots,n_{ij}$, n_{ij} is the number of observation in the ij^{th} cell. If $n_{ij} = 0$, the ij^{th} cell is empty. For convenience in arithmetic, let $r \leq c$.

Let the incidence matrix be as follows:

$$N = \begin{bmatrix} n_{11} & n_{12} & \cdots & n_{1c} \\ n_{21} & n_{22} & \cdots & n_{2c} \\ \vdots & & & \\ n_{r1} & n_{r2} & \cdots & n_{rc} \\ n_{.1} & n_{.2} & \cdots & n_{.c} \end{bmatrix} \begin{matrix} n_{1.} \\ n_{2.} \\ \\ n_{r.} \\ n \end{matrix}$$

If the incidence matrix is proportional, i.e. $n_{ij} = (n_{i.} \cdot n_{.j})/n$, an abbreviated, unadjusted analysis can be performed which is well described in elementary texts.

As it now stands, the model is indeterminate, and no reasonable analysis can be made. Thus, it is necessary to make certain assumptions about the model or to impose certain conditions to make main effects "estimable". Let us impose the following r + c - 1 conditions:

$$\sum_i n_{ij}\delta_{ij} = 0 \quad \text{for all } j$$

$$\sum_j n_{ij}\delta_{ij} = 0 \quad \text{for all } i$$

These conditions minimize, in effect, the contribution of the interaction. With these conditions, the normal equations become as follows:

$$nm + \underline{n}_{i.}'\underline{a} + \underline{n}_{.j}'\underline{b} = G$$

$$\underline{n}_{i.}m + D_{n_{i.}}\underline{a} + N\underline{b} = \underline{R}$$

$$\underline{n}_{.j}m + N'\underline{a} + D_{n_{.j}}\underline{b} = \underline{C}$$

Now there are r + c + 1 equations in r + c + 1 unknowns.

By making the r + c - 1 conditions on our original model, we now have a new and more restricted model. Different conditions would have led to different models and hence different analyses.

To solve the equations, we can apply two constraints. Note that the constraints are applied to the estimates of the α and β effects and are only for arithmetic convenience in solving the set of singular equations. Different choices of a pair of constraints would have led to the same analysis.

After some elementary manipulations, we obtain the r "adjusted normal equations",

$$C\underline{a} = \underline{Q}.$$

The typical elements of C are as follows:

$$c_{ii} = n_{i.} - \sum_{j=1}^{c} n_{ij}^2/n_{.j}$$

and

$$c_{ik} = -\sum_{j=1}^{c} n_{ij}n_{kj}/n_{.j}\ , \quad i \neq k.$$

The typical element of $\underline{Q}$ is

$$Q_i = R_i - \sum_{j=1}^{c} n_{ij}C_j/n_{.j}$$

which is an adjusted row total.

C is symmetric and singular; there are r equations in r unknowns; and the rank of C is r-1. We still have one constraint which we can apply. Let $a_r = 0$. Hence, we now have r-1 equations in r-1 unknowns. Thus, we obtain the solutions $a_1, a_2, \ldots, a_{r-1}$, and $a_r = 0$.

The effects, as such, are not estimable but effect contrasts are. Since additions of a constant to a contrast does not change the contrast, we can add a constant k to our solutions a_i, i=1, ... , r, for mere convenience of representation of the same results.

Had the C matrix been "orthogonal", an estimate of the ith row effect could have been $R_i/n_{i.}$, i.e., the ith row mean, an easily interpretable quantity. This "effect estimate" would satisfy the constrains $\sum n_{i.} a_i^* = G$, the grand total. It is obtained from an a_i based upon another constraint, by addition of a constant k, $a_i^* = a_i + k$, where

$$k = (G - \sum_{i=1}^{r} n_{i.} a_i) / n.$$

Thus, in the non-orthogonal case, it seems convenient, for purposes of interpretation, to use $a_i^* = a_i + k$ where k is defined as above. These a_i^* will be called "adjusted row means". Note that, for the statistician, this would be somewhat inconvenient. He prefers sums or weighted sums of effects to add to zero, so that he may add these in the model equation without having to subtract constants (in our case, the grand mean). To the experimenter, however, a mean is much more useful than effect estimates which are preferable for mathematical convenience. To repeat, these are merely different representations of identical results.

Similarly, for column effects, we calculate, initially,

$$b_j = \frac{C_j - \sum_{i=1}^{r} n_{ij} a_i}{n_{.j}} - \frac{G}{n}$$

For the "adjusted column means", let $b_j^* = b_j + k$ where

$$k = \frac{G - \sum_{j=1}^{c} n_{.j} b_j}{n}$$

3.2 Program TEMAT

Multivariate Analysis of Variance

The likelihood ratio statistic is

$$-m \log(|E|/|H+E|)$$

where m = d.f. error + 1/2(d.f. hypothesis - number of responses - 1),

$|E|$ is the determinant of the matrix of sums of squares and products for error,

H is the matrix of "adjusted" sums of squares and products for each hypotheses (subtotals, interaction, rows, columns, in turn).

The first two terms of the Bernoulli expansion of the distribution, which would be in error by a term of order $1/m^4$, are employed to state at what level of significance the null hypothesis would be rejected. In the (hopefully frequent) case that this level is below 10^{-5}, the result is displayed as zero. Large values (.20 or larger) are desirable only in the interaction test. From time to time, the display for one effect is replaced by the message "MATRIX NOT GRAMIAN". This can have two reasons:

(a) There is linear dependence between the response variables, e.g., one may be the sum of the others. In this case, the multivariate analysis is obviously faulty. A variable should be excluded in the next run.

(b) The approach to the null-hypothesis situation is too close. This is the generalization of the case which produces negative variance-component estimates in univariate analysis. If this happens in an Interaction test, the result is quite desirable. If it happens in a main effect, it contributes nothing. If the "MATRIX NOT GRAMIAN" display occurs in a main effect it simply means that H_0 is to be accepted, and that no interpretation of discriminant functions or main effect differences should be attempted.

In the special case where the hypothesis has only one degree of freedom (2 levels in a factor) the likelihood-ratio test, just as the union-intersection test, reduces to an F statistic (more commonly known as "Hotelling's T^2 - with some minor variation in constants"). In these cases, the F statistic and the associated degrees of freedom are reported. The user would have to compare with F tables to decide on acceptance or rejection of the hypothesis. In our interactive system we have a "Calculator" mode which enables him to look up, i.e., actually compute, the probability levels.

The program also calculates the union-intersection statistic, i.e., the largest characteristic root of $E^{-1}H$ (same definitions as above). The right and left eigenvectors associated with this matrix are also computed and reported, the former being a representation of the discriminant function, the latter being used as a basis for calculating correlations of response variables versus the discriminant functions [2]. The parameters needed for entering the Roy-Heck charts are displayed. There is a practice to use the Beta statistic SSH/(SSH + SSE), usually called R^2, as a measure of goodness of fit of a univariate analysis of variance

model. To satisfy users who like this measure, the corresponding multivariate index ("canonical R^2") is reported. It is, in fact, the largest characteristic root of $(H + E)^{-1}H$.

3.3 Distribution Routines

FUNCTION GAMX(X,DF)

$$P(a,x) = GAMX(x,a) = \frac{1}{\Gamma(a)} \int_0^x e^{-y} y^{a-1} dy$$

If DF < 200, the probabilities are evaluated by a sum of Poisson terms

$$\frac{x^a e^{-x}}{\Gamma(a+1)} \qquad \text{where } a = DF,\ DF + 1,\ \ldots$$

If DF > 200, the evaluation is by cubic approximation

$$Gam(X,DF) = YORMX(Y) \qquad \text{where } Y = [\sqrt[3]{X/DF} - 1 + 1/9DF]\ \sqrt{9DF}$$

FUNCTION CHIX(X,DF)

$$P(\gamma,x) = CHIX(X,DF) = \frac{1}{2^{\gamma/2}\Gamma(\gamma/2)} \int_0^x y^{\gamma/2-1} e^{-y/2} dy$$

Since the chi-square distribution is a special case of the gamma distribution, the probabilities are evaluated by the use of the GAMX subroutine where

$$CHIX(X,DF) = GAMX(X/2.,DF/2.)$$

FUNCTION YORMX(X)

$$P(x) = \text{YORMX}(X) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-t^2/2} dt$$

If $|x| < 2.5$, the evaluation of the probabilities employs the Hastings form; otherwise, the Laplace continued fraction is used.

CHAPTER IV

TOOLS FOR THE USER

4.1 User's Guide

The program SPOOK is designed to provide the user with an analysis of irregular data matrices. A univariate analysis (i.e., every score or variable is analyzed separately) is performed, followed by, if desired, a multivariate analysis (where all variables are analyzed as a set).

The univariate analysis may be either a one-way or a two-way classification design. A one-way design is such that the individual experimental units (e.g., individuals, plots of land, animals, etc.) are classified by only one characteristic or factor. A two-way design is based upon the individual experimental units being classified by two characteristics or factors. Each experimental unit is then assigned to one, and only one, category for each factor. Examples of factors would be sex, age, weight, height, etc. Since the conversational unit is designed for two factors, a one-way classification requires special treatment. In this instance, the user should state, when asked, a dummy name (or no name at all) for the first factor and he should further declare that it has just one level. The second factor should then be his only factor.

Each factor must be such that levels may be designated. The designation of levels may be in the form of consecutive numbers such as

a to b or, if the data are continuous, records which fall below a_1 could be regarded as belonging to level 1, records falling between a_1 and a_2 as belonging to level 2, etc. Thus, the levels of a factor such as sex could be designated as 1,2 while the levels of a factor such as age could be designated as

5 - 9 years	level 1	("end code" here is 9)
10 - 14 years	level 2	("end code" here is 14)
15 - 19 years	level 3	("end code" here is 19)
20 - 25 years	level 4	("end code" here is 25).

There can be up to twelve levels for each factor.

For each experimental unit, there may be from one through ten responses to different variables. Examples of response variables would be a test score (or average) in English, a test score in arithmetic, and a test score in science, or the number of situps, number of pushups, time in 100-yard dash, and distance in broad jump. In these examples the experimental units would be individuals, and the factors may be sex, age, school type, etc.

A name or code (maximum of four characters) should be assigned to each factor and each response variable. The naming will be used for identification in the output. In addition to the names, a title might be desired.

Once having collected his data, the user could have a table such as Table 4.1 for quick reference. With such a table he will be able to give answers to the questions asked by the computer.

If the data are voluminous, and punched on cards, the user is advised to refer to the description of BUILD in Chapter 5 so that a

prior batch run can be made to store the data. Because preparation of the input may appear complicated to those with limited programming background, a user who finds the description difficult may wish to consult a somewhat more experienced programmer for the first run. Once the batch run has been made, the input portion of SPOOK is omitted, and the user may proceed as if he had entered his data from the console.

EFFECTIVENESS OF TEACHING

	MALE			FEMALE		
	ENG	ARIT	SCIE	ENG	ARIT	SCIE
PUBLIC	55.1	55.6	46.8	57	47.4	44.8
	56.1	53.2	41.7	55.6	46.3	39.9
	55.9	49.5	38.1	58	51.2	50
	54.4	58.9	39.6	55.9	50.3	39.4
				59.7	49.5	43.1
PRIVATE	52.8	59.4	47.1	55.6	54.8	42.7
	53.6	65.6	42.1	61.7	53	47.1
	53.9	59.2	42.3	56.1	60.8	45.2
	53.8	58.2	45.2	60.3	67.6	49.2
BOARDING	56.7	62.3	49.8	61.8	52.6	48.4
	58.2	64.6	47	55.6	69.9	49.3
	59.6	52.8	49.2	62.3	62.4	50.4
				59.9	54.3	46.2
				60.7	59.4	48.6

Table 4.1

4.2 Example

This section is intended to familiarize the user both with the IBM 2250 display unit and the program SPOOK.

The 2250 display unit is centered around a cathode-ray tube on which computer-programmed information is displayed. Thus, a visual communication exists between the user and the computer. The unit also has an alphameric keyboard similar to a typewriter keyboard. There are 44 keys and a space bar giving 62 standard characters. Special keys of interest to the user are the "ALT" and the "5" keys. When these two keys are depressed simultaneously, the end of a response is signalled. This combination will often be referred to as "end of block" or "EOB". The user should also note the "BACKSPACE" key. This moves the cursor back one space and as long as EOB has not been signalled the user can correct his typed input. To the left of the alphameric keyboard is a program function keyboard which consists of 32 lighted keys numbered 0 - 31. All keys, except key 0, are available to the user. At different points in the program, different keys will have significance, and thus be lighted to help the user remember them.

When the user first sits down in front of the console, he should depress any key (not key 0). Once the instructions appear he should type $LINK SPOOK and then signal EOB ("ALT" and "5" depressed simultaneously).

Even though each frame should be self-explanatory, a typical set of data was constructed and subjected to the program to illustrate the use of SPOOK. The data are shown in Table 4.1. Since one purpose of

this program is to provide a quick means of editing the data, the example will try to illustrate one way to hunt faulty ones.

The first display is shown in Figure 4.1. Several instructions which are of particular importance are as follows: key 30 restarts the program, key 31 will terminate the program, key 2 will allow the user to reenter data, and any remaining key (again, not key 0) will allow him to continue once he has seen the display of his data. The one instruction which needs all the emphasis it can receive is "CAUTION: DO NOT TRY TO SPEED UP THE PROGRAM BY ANSWERING QUESTIONS BEFORE THEY ARE ASKED. THIS WILL ONLY CREATE PROBLEMS." We know there are users who anticipate and answer the next question. They must contain themselves, since all keyboard responses are placed in a queue, and each question expects some response. Once the user gets "out of phase", the best solution is to press key 31 to terminate and then to reenter the program. Even the restart option (key 30) may not correct the situation.

Because the example is being created at the console, key 1 is depressed to begin the question and answer session. The first query is for a study title. The study title is entirely optional and is used to head the output. It may consist of up to 60 characters. The display following the entry of the title is shown in Figure 4.2. The next question is "HOW MANY RESPONSE VARIABLES DO YOU HAVE?" The number 3 is entered and the following message is displayed to the user: "YOU HAVE 3 RESPONSE VARIABLES." As always, to indicate approval, any one of the unused program function keys is depressed.

The next set of questions asks the user for a name (four letters are recognized) for each of the response variables. The purpose of naming each response variable is so that the data will be entered as the user wishes and also so that the user can identify each in the output. Once the names "ENG ARIT SCIE" are displayed and the user has indicated approval, he is asked for a transformation code for each response variable. The query and the user's reply is displayed to him as in Figure 4.3.

Next, the names of the two factors and the number of levels for each is required. Again the naming of the factors is for identification in the output, but if the first factor has more levels than the second factor, the two will be exchanged in the analysis and hence the naming is especially helpful. In this study, the first factor is SEX and has 2 levels, and the second is TYPE and has 3 levels. If there is only one factor (one-way classification analysis) the user should give a dummy name to the "first" factor (or just leave the name blank) and indicate the number of levels (below) as 1. The second factor is then his only one.

Figure 4.4 shows the next display. Since, in our example, all levels are discrete (rather than selected partitions on a continuous scale) and since they further have natural codes 1-2, 1-2-3, key 1 has been depressed. Had key 2 been depressed to indicate continuous levels, Figure 4.5 would have appeared asking for the "end codes" of the levels of each factor. The display in Figure 4.6 is the next one to appear in either case.

Figure 4.7 shows the instructions for entering dat . Note that key 2 is no longer used for reentry of data. Instead, the user types "BACK" and the number of entries which he wishes to go back. Each time this option is used the appropriate number of lines will be erased starting at the bottom of the display. Thus, one cannot change, at this stage, an entry without also reentering all of the following entries. Key 3 will be depressed when the user has entered all of the data. Once key 3 is depressed the user is forced into the univariate analysis. Hence, he should be sure that the data displayed on the console is what is desired at that point.

Depending upon the number of users concurrently on the central processor, waiting time for results may take several minutes even though the central processing time is only a matter of a second or two. Hence, displays similar to the one in Figure 4.8 keep the user posted on progress and allow him to stop the calculations if he wishes. Once key 2 is depressed the program will proceed to the output having performed univariate analyses only on the named variables.

As the display in Figure 4.9 indicates, the user has complete control over what portions of the output he views. Once the user reaches the instructions which follow each set of output, he is not bound to follow them at that point. He can still depress key 3 to decrement the page number and hence continue viewing the analyses more extensively.

Page 1L (Figure 4.10) gives some hints on how to begin looking at the preliminary output. Once the user has checked the F-ratios, adjusted means, and cell means and standard deviations of the univariate analyses

and if he finds nothing abnormal, he should still proceed to the plots of the response variable pairs and to the raw data. Plotting rows, columns, and cells may indicate faulty data. At the plotting segment he should check the upper and lower limits of each variable. This is another good way of checking on faulty data. Because of the many steps in bringing data to be analyzed, faulty data or outliers can easily occur: data can be transcribed incorrectly or have keypunch errors. Thus, the first analysis should never be the last!

Figures 4.11 - 4.16 show the output from this initial run. Let us look first at the variable ENG. The F-values are nonsignificant but a careful inspection does not stop with the ANOVAR table. Looking at the adjusted means, one sees that the means of the second level of each factor seems to be quite out of range. The next place that merits inspection is the table of means and standard deviations. "Deviation from a trend in rows or columns" is a mild way of reporting what is happening in the (2,2) cell. This clearly indicates that the quickest way to find out what is happening is to go to the plotting segment of the program. But, first look at the ARIT and SCIE variables. There is nothing to indicate faulty data in these variables. The data in the two variables should still be investigated by plots and by looking at the raw data.

Figure 4.17 shows the first display once the plotting segment is entered. To begin, the (2,2) cell is what we are interested in, and the best method is to plot ENG against each of the other variates. Figures 4.18 and 4.19 indicate that entry number 17 is the outlier.

Note the upper and lower limits on the ENG variable. Figures 4.20 - 4.22 show plots of other data. After other plots have been investigated for faulty data which did not show in the analysis, depression of key 5 passes control of the program to a display of the raw data. Now, changes can be made. Inspection of entry number 17 shows a misplaced decimal point (a quite common mistake). Depression of key 2 allows the correction to be made as shown in Figure 4.23. Once the correction is made, depression of key 6 will allow the user to see plots again.

Here, it should be emphasized that while the user is in the plotting segment key 5 will take him to the raw data and key 29 will allow him to see the previous univariate analyses. While he views the raw data, key 6 will allow him to see plots (these new plots will include any changes made) and key 29 will allow him to see the previous univariate results. Once the user has performed his first univariate analysis pass, he can always depress key 5 to see raw data or key 6 to see plots. This allows him to go back and forth as often as he wishes in order to make changes. Hence, if he made a change in the data and saw in the plots that he still did not like it, another change could easily be made.

In our example, continuation to a new univariate analysis is now indicated. Figures 4.24 and 4.25 show the new output for the variable ENG. Here, both factors are significant and outliers are no longer indicated. Thus, in the example, the faulty data point did produce obvious changes in the univariate analysis. On the other hand, even where faulty data make no obvious changes in the univariate analysis they could still influence the multivariate analysis in an unpredictable manner.

Once an edited analysis has successfully been made, the user may indicate a multivariate analysis by the depression of key 4. Once the multivariate option has been indicated, the user is given the opportunity to delete any of the response variables. After having studied the univariate results, the user may wish to exclude some of the variables from the multivariate pass. For example, a variable which shows an insignificant F-value even for subtotals serves no good purpose in the multivariate analysis and may dilute it. As each response variable name is displayed on the screen, the user should depress key 1 to include the variable in the multivariate analysis or key 2 to exclude it from the analysis. Once the user has made a decision on each name, he will be given the opportunity to revise his list if he should have made a mistake.

In the example we will include all the variables in the multivariate pass. This will give sample output as seen in Figures 4.27 - 4.30. The multivariate output includes likelihood ratio tests, union-intersection tests, weights of the discriminant functions, correlation between each variable and the discriminant function, E matrix of sums of squares and cross products for error, E^{-1}, and correlation matrices based on E and H + E for each H (see Section 3.2).

At the end of the multivariate output is a message instructing the user to depress key 31 if he wishes to terminate or key 4 for another multivariate analysis which might be indicated to investigate new combinations of response variables. Once again the user is still free to depress key 3 to go back to continue study of the output. Although the data should have been edited completely before the

multivariate analysis, the user may still depress key 6 to see plots or depress key 5 to see raw data.

OUTPUT AREA

THIS PROGRAM IS DESIGNED TO PERFORM AN ANALYSIS OF IRREGULAR DATA. YOU MAY HAVE A MAXIMUM OF 10 RESPONSE VARIABLES AND A MAXIMUM OF 2 FACTORS. YOUR DESIGN MAY BE QUITE UNBALANCED, AND WHOLE CELLS MAY BE MISSING. TRANSFORMATIONS CAN BE MADE. FOR EACH PAIR OF FACTORS, AN ANALYSIS OF VARIANCE IS PERFORMED FOR EACH RESPONSE VARIABLE, SEPARATELY. ONCE YOU SEE THESE UNIVARIATE ANALYSES YOU WILL BE GIVEN THE OPPORTUNITY TO SEE PLOTS OF YOUR DATA AND THE RAW DATA AGAIN. THUS, YOU WILL BE ABLE TO EDIT YOUR DATA AND RUN THE ANALYSES AGAIN. .

TO BEGIN YOU MUST ANSWER QUESTIONS BY USING THE TYPEWRITER KEYBOARD DIRECTLY IN FRONT OF YOU. TO SIGNAL YOUR COMPLETION OF QUESTIONS, FIRST DEPRESS THE "ALT" KEY, AND WHILE HOLDING IT DOWN, DEPRESS THE "5" KEY. THIS SEQUENCE WILL LATER BE REFERRED TO AS "EOB". ONCE YOU ANSWER QUESTIONS, THE ANSWERS WILL BE DISPLAYED BACK TO YOU. IF YOU ARE NOT SATISFIED, PRESS KEY 2 TO REENTER DATA; OTHERWISE, THE PROGRAM WILL CONTINUE BY YOUR PRESSING ANY KEY.

AT ANY TIME YOU MAY RESTART BY PRESSING KEY 30 OR TERMINATE BY PRESSING KEY 31.

CAUTION: DO NOT TRY TO SPEED UP THE PROGRAM BY ANSWERING QUESTIONS BEFORE THEY ARE ASKED. THIS WILL ONLY CREATE PROBLEMS.

IF YOU HAVE PREVIOUSLY USED THIS PROGRAM OR ENTERED YOUR DATA THROUGH BATCH MODE, PRESS KEY 2 TO SEE YOUR DATA.

PRESS KEY 1 TO PROCEED.

REPLY AREA

Figure 4.1

OUTPUT AREA

IF YOU DESIRE TO HAVE A TITLE TO HEAD YOUR OUTPUT, ENTER IT NOW.
IF YOU DO NOT DESIRE A TITLE, LEAVE THE RESPONSE AREA BLANK.

YOUR TITLE IS
EFFECTIVENESS OF TEACHING

REPLY AREA

Figure 4.2

OUTPUT AREA

FOR EACH RESPONSE VARIABLE ENTER A TRANSFORMATION CODE FROM THE FOLLOWING LIST:

CODE	TRANSFORMATION
0,1	NO TRANSFORMATION
2	LOGE(X) (X>0)
3	LOGE(1+X) (X>-1)
4	SQRT(X) (X>=0)
5	1/X (X>0)
6	ARCSIN(2X-1) (0<X<1)
7	VARIANCE-STABILIZING TRANSFORMATION FOR PROPORTIONS AVAILABLE (NOW: NO TRANSFORMATION)

SEPARATE ALL ANSWERS BY COMMAS.

ENG	ARIT	SCIE
1	1	1

REPLY AREA

Figure 4.3

OUTPUT AREA

```
YOUR FACTORS SHOULD BE IN DISCRETE LEVELS AS 1,2,...,12.  BUT
IF YOUR DATA IS CONTINUOUS, GROUPING CAN BE DONE FOR YOU.
IF YOUR LEVELS ARE ALREADY DISCRETELY DEFINED, PRESS KEY 1.
IF YOUR DATA IS CONTINUUUS, PRESS KEY 2.
```

REPLY AREA

Figure 4.4

OUTPUT AREA

```
TO GROUP THE DATA, ASSIGN EACH LEVEL OF THE FACTOR AN "END CODE"
OR AN IDENTIFIER;  EG, IF ONE OF THE FACTORS IS AGE RANGING FROM
5 YEARS TO 25 YEARS, WE MAY WISH TO ASSIGN AGES TO LEVELS AS
     5- 9 YEARS   LEVEL 1
    10-14 YEARS   LEVEL 2
    15-19 YEARS   LEVEL 3
    20-25 YEARS   LEVEL 4
THEN THE END CODE FOR LEVEL 1 WOULD BE 9, THE END CODE FOR
LEVEL 2 WOULD BE 14, ETC.
THIS DATA WOULD BE ENTERED AS   9,14,19,25

WHAT ARE THE "END CODES" FOR THE  2 LEVELS OF FACTOR SEX ?
WHAT ARE THE "END CODES" FOR THE  3 LEVELS OF FACTOR TYPE?
```

REPLY AREA

Figure 4.5

OUTPUT AREA

```
THE "END CODES" FOR THE  2 LEVELS OF FACTOR SEX
  1   2
THE "END CODES" FOR THE  3 LEVELS OF FACTOR TYPE
  1   2   3
```

REPLY AREA

Figure 4.6

OUTPUT AREA

```
NOW YOU MUST SUPPLY THE DATA.  YOU WILL NEED TO INDICATE THE
LEVELS OF THE FACTORS AND THE RESPONSE OF EACH VARIABLE.
     EXAMPLE:   2,7,412,34,5
     INDICATES FACTOR 1 IS AT LEVEL 2, FACTOR 2 IS AT LEVEL 7,
     THE RESPONSE FOR VARIABLE 1 IS 412, THE RESPONSE FOR
     VARIABLE 2 IS 34, AND THE RESPONSE FOR VARIABLE 3 IS 5.
IF YOU WANT TO OMIT DATA ON A PARTICULAR RANDOM VARIABLE,
REPLACE IT BY -999.
     EXAMPLE:   2,7,412,-999,5
     INDICATES THAT THE DATA FOR THE SECOND RESPONSE
     VARIABLE IS OMITTED.

WHEN YOU HAVE ENTERED ALL DATA, PRESS KEY 3.

SHOULD YOU REALIZE YOU HAVE A MISTAKE IN YOUR DATA, TYPE "BACK"
AND THE NUMBER OF ENTRIES YOU WISH TO GO BACK.

WHEN YOU ARE READY TO ENTER YOUR DATA, PRESS KEY 1.
```

REPLY AREA

Figure 4.7

OUTPUT AREA

THE ANALYSES ON THE FOLLOWING VARIABLES ARE NOW COMPLETE.
ENG ARIT

PRESS KEY 1 TO CONTINUE WITH CALCULATIONS OR KEY 2 TO STOP CALCULATIONS AT THIS STAGE.

REPLY AREA

Figure 4.8

OUTPUT AREA

```
KEYBOARD FUNCTIONS:
  1     ALTERNATE PORTION OF CURRENT PAGE.
  2     INCREMENT PAGE NUMBER.
  3     DECREMENT PAGE NUMBER.
 30     RESTART PROGRAM.
 31     TERMINATE PROGRAM.

PRESS KEY 1 TO PROCEED TO THE FIRST PAGE OF YOUR UNIVARIATE RESULTS.
```

REPLY AREA

Figure 4.9

OUTPUT AREA

PAGE 1L

AT THIS POINT YOU SHOULD INSPECT YOUR UNIVARIATE RESULTS VERY CAREFULLY.

IS THERE A LARGE INTERACTION VERSUS ERROR F-RATIO?
THIS MAY INDICATE THE PRESENCE OF OUTLIERS OR FAULTY DATA.

CHECK THE TABLE OF CELL TOTALS OF VARIABLES WITH LARGE F-RATIOS FOR LARGE STANDARD DEVIATIONS OR FOR A CELL MEAN WHICH DEVIATES IRREGULARLY FROM A TREND IN ROWS OR COLUMNS.

AFTER YOU HAVE CHECKED THESE RESULTS YOU CAN CHECK AND EDIT YOUR DATA.

REPLY AREA

Figure 4.10

```
                                OUTPUT AREA
                                                          PAGE   2L
                                     EFFECTIVENESS OF TEACHING
                                VARIABLE ( ENG  ) TABLE OF MEANS
ROWS =                                          C O L U M N S =   TYPE
 SEX          IN EACH BLOCK, ROW 1 DENOTES CELL MEANS, ROW 2 DENOTES
CODED                                             C O D E D  L E V E L S
LEV.  1          2          3

1 5.537E 01 5.352E 01 5.817E 01 5.546E 01
     4          4          3          11
  7.798E-01 5.000E-01 1.450E 00

2 5.724E 01 5.842E 01 6.006E 01 5.859E 01
     5          4          5          14
  1.671E 00 3.035E 00 2.664E 00

COL.
  5.641E 01 5.597E 01 5.935E 01 5.721E 01
     9          8          8          25

              ESTIMATES OF SEX   ADJUSTED MEANS, ORDERED
     2          1
  5.847E 01 5.561E 01

              ESTIMATES OF TYPE  ADJUSTED MEANS, ORDERED
     3          1          2
  5.916E 01 5.642E 01 5.615E 01

                ANALYSIS OF VARIANCE FOR VARIABLE ENG

   SOURCE OF VARIATION                D.F.       SUM OF SQUARES

       SEX     (ROWS)                   1        5.019498E 01
       TYPE    (COLS)                   2        4.473355E 01
    SEX *  TYPE(INTERACTION)            2        1.227552E 01

       SUBTOTALS                        5        1.170524D 02
         ERROR                         19        7.397502D 01
         TOTAL                         24        1.910274D 02
```

REPLY AREA

Figure 4.11

OUTPUT AREA

PAGE 2R

AND EFFECTS (SEX) VERSUS (TYPE)

CELL SIZE, AND ROW 3 DENOTES STANDARD DEVIATION.

MEAN SQUARE	F VS. ERROR	F VS. INTERACTION
1.175292E 04	1.207E 00	1.065E 00
1.127447E 04	1.158E 00	1.022E 00
1.103074E 04	1.133E 00	
1.078300E 04	1.107E 00	
9.739313E 03		
9.956746E 03		

ROOT MEAN SQUARE ERROR = 9.868796E 01
ROOT MEAN SQUARE INTERACTION = 1.050273E 02

REPLY AREA

Figure 4.12

```
                                   OUTPUT AREA
                                                                  PAGE   3L
                                        EFFECTIVENESS OF TEACHING
                                   VARIABLE ( ARIT ) TABLE OF MEANS
ROWS =                                                C O L U M N S =   TYPE
 SEX          IN EACH BLOCK, ROW 1 DENOTES CELL MEANS, ROW 2 DENOTES
CODED                                                 C O D E D  L E V E L S
LEV.  1          2          3

1 5.430E 01 6.060E 01 5.990E 01 5.812E 01
      4          4          3          11
  3.962E 00 3.374E 00 6.255E 00

2 4.894E 01 5.905E 01 5.972E 01 5.568E 01
      5          4          5          14
  2.038E 00 6.604E 00 6.910E 00

COL.
  5.132E 01 5.982E 01 5.979E 01 5.675E 01
      9          8          8          25

                ESTIMATES OF SEX   ADJUSTED MEANS, ORDERED
      1          2
  5.816E 01 5.564E 01

                ESTIMATES OF TYPE  ADJUSTED MEANS, ORDERED
      3          2          1
  5.995E 01 5.967E 01 5.131E 01

                    ANALYSIS OF VARIANCE FOR VARIABLE ARIT

    SOURCE OF VARIATION                D.F.        SUM OF SQUARES

        SEX      (ROWS)                  1         3.863693E 01
        TYPE     (COLS)                  2         4.165764E 02
      SEX *  TYPE(INTERACTION)           2         3.007245E 01

        SUBTOTALS                        5         4.833116D 02
          ERROR                         19         4.979627D 02
          TOTAL                         24         9.812743D 02
```

REPLY AREA

Figure 4.13

OUTPUT AREA

PAGE 3R

AND EFFECTS (SEX) VERSUS (TYPE)

CELL SIZE, AND ROW 3 DENOTES STANDARD DEVIATION.

MEAN SQUARE	F VS. ERROR	F VS. INTERACTION
3.863693E 01	1.474E 00	2.570E 00
2.082882E 02	7.947E 00	1.385E 01
1.503622E 01	5.737E-01	
9.666231E 01	3.688E 00	
2.620856E 01		
4.088643E 01		

ROOT MEAN SQUARE ERROR = 5.119430E 00
ROOT MEAN SQUARE INTERACTION = 3.877657E 00

REPLY AREA

Figure 4.14

```
                                   OUTPUT AREA
                                                                  PAGE   4L
                                          EFFECTIVENESS OF TEACHING
                                     VARIABLE ( SCIE ) TABLE OF MEANS
ROWS =                                                  C O L U M N S =   TYPE
 SEX          IN EACH BLOCK, ROW 1 DENOTES CELL MEANS, ROW 2 DENOTES
CODED                                                C O D E D  L E V E L S
LEV.  1          2          3

1 4.155E 01 4.417E 01 4.867E 01 4.445E 01
      4          4          3         11
  3.798E 00 2.410E 00 1.474E 00

2 4.344E 01 4.605E 01 4.858E 01 4.602E 01
      5          4          5         14
  4.297E 00 2.767E 00 1.543E 00

COL.
  4.260E 01 4.511E 01 4.861E 01 4.533E 01
      9          8          8         25
                ESTIMATES OF SEX   ADJUSTED MEANS, ORDERED
      2          1
  4.589E 01 4.461E 01
                ESTIMATES OF TYPE  ADJUSTED MEANS, ORDERED
      3          2          1
  4.853E 01 4.519E 01 4.261E 01
                     ANALYSIS OF VARIANCE FOR VARIABLE SCIE

    SOURCE OF VARIATION                D.F.        SUM OF SQUARES

        SEX     (ROWS)                   1          9.946443E 00
        TYPE    (COLS)                   2          1.482998E 02
      SEX *  TYPE(INTERACTION)           2          5.036934E 00

        SUBTOTALS                        5          1.686363D 02
          ERROR                         19          1.714142D 02
          TOTAL                         24          3.400505D 02
```

REPLY AREA

Figure 4.15

OUTPUT AREA

PAGE 4R

AND EFFECTS (SEX) VERSUS (TYPE)

CELL SIZE, AND ROW 3 DENOTES STANDARD DEVIATION.

MEAN SQUARE	F VS. ERROR	F VS. INTERACTION
9.946443E 00	1.102E 00	3.949E 00
7.414990E 01	8.219E 00	2.944E 01
2.518467E 00	2.792E-01	
3.372726E 01	3.738E 00	
9.021797E 00		
1.416877E 01		

ROOT MEAN SQUARE ERROR = 3.003631E 00
ROOT MEAN SQUARE INTERACTION = 1.586967E 00

REPLY AREA

Figure 4.16

OUTPUT AREA

```
AT THIS STAGE YOU MAY LOOK AT PLOTS OF YOUR DATA POINTS.
YOU HAVE THE CHOICE OF SEEING A CELL, A ROW, A COLUMN, OR
ALL YOUR DATA.  TO INDICATE YOUR CHOICE OF POINTS, TYPE IN
THE LEVELS OF BOTH FACTORS.  IF YOU WISH TO SEE ALL LEVELS
OF A FACTOR TYPE "0" FOR THE LEVEL.
EXAMPLE:      2,2      INDICATES BOTH LEVELS AT 2.
              0,4      INDICATES THE 4TH COLUMN.
              3,0      INDICATES THE 3RD ROW.
              0,0      INDICATES ALL POINTS.

WHICH SET OF POINTS DO YOU WISH TO SEE?

WHICH TWO RESPONSE VARIABLES DO YOU WISH TO SEE?
PLEASE ENTER THE FIRST, EOB, AND THE SECOND, EOB.
```

REPLY AREA

Figure 4.17

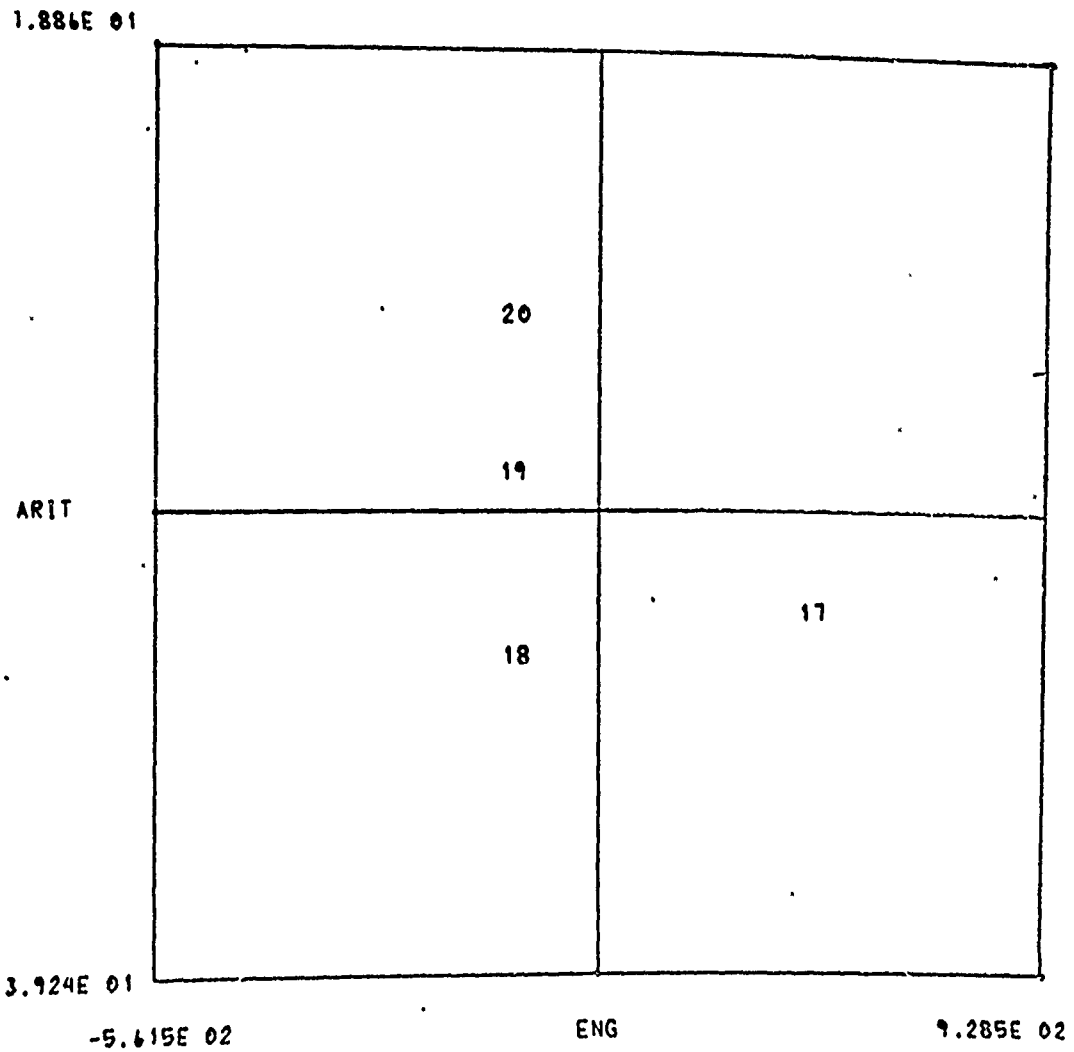


Figure 4.18

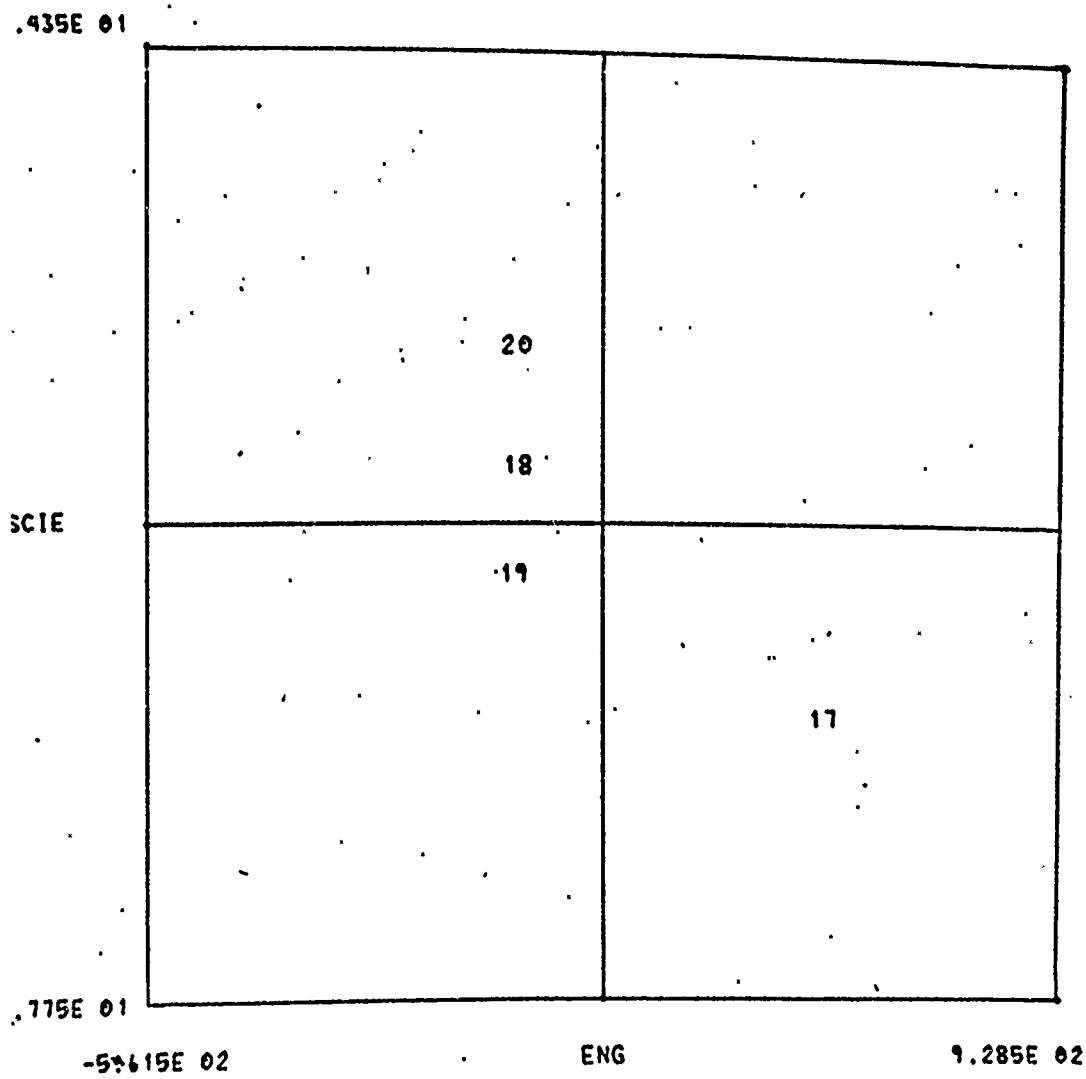


Figure 4.19

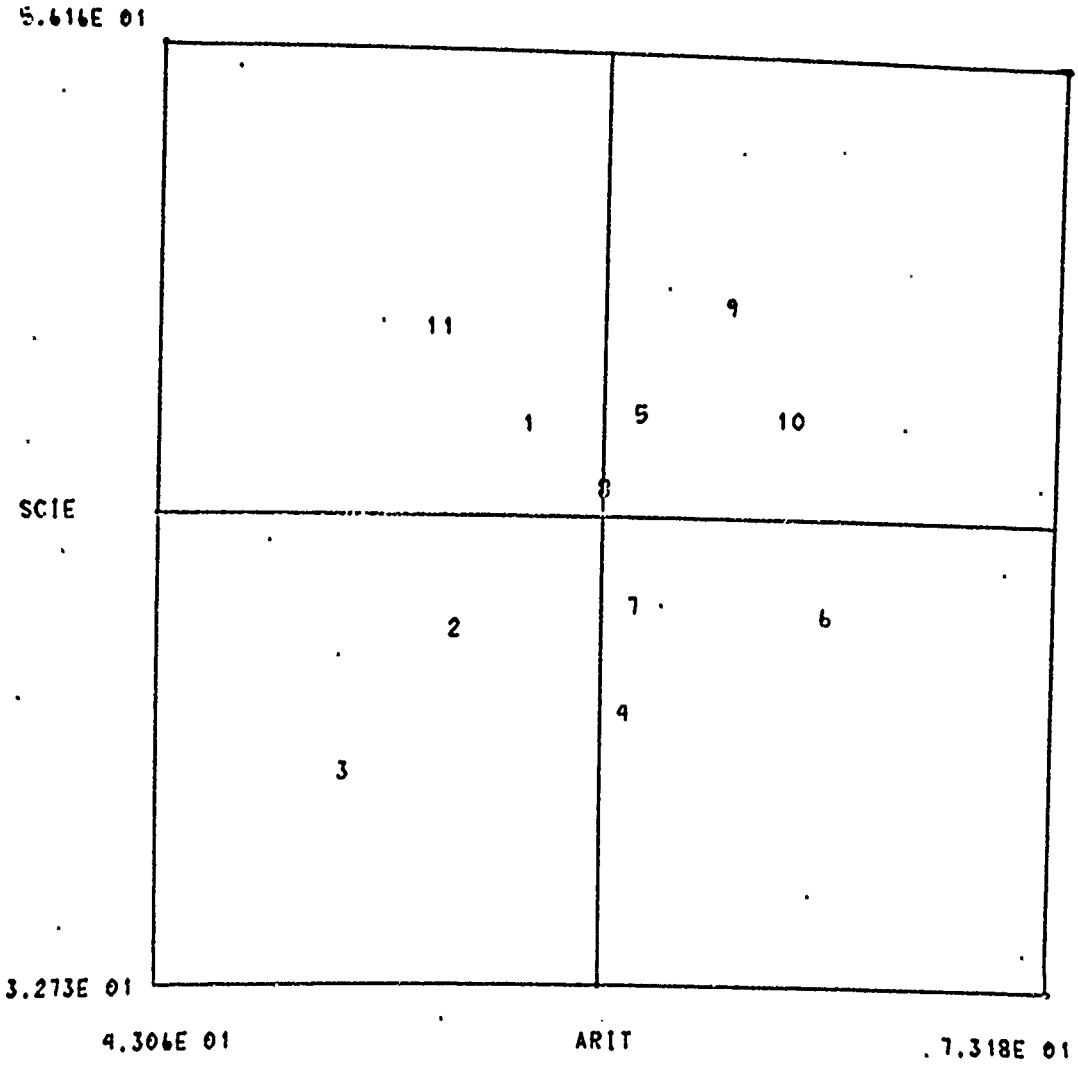


Figure 4.20

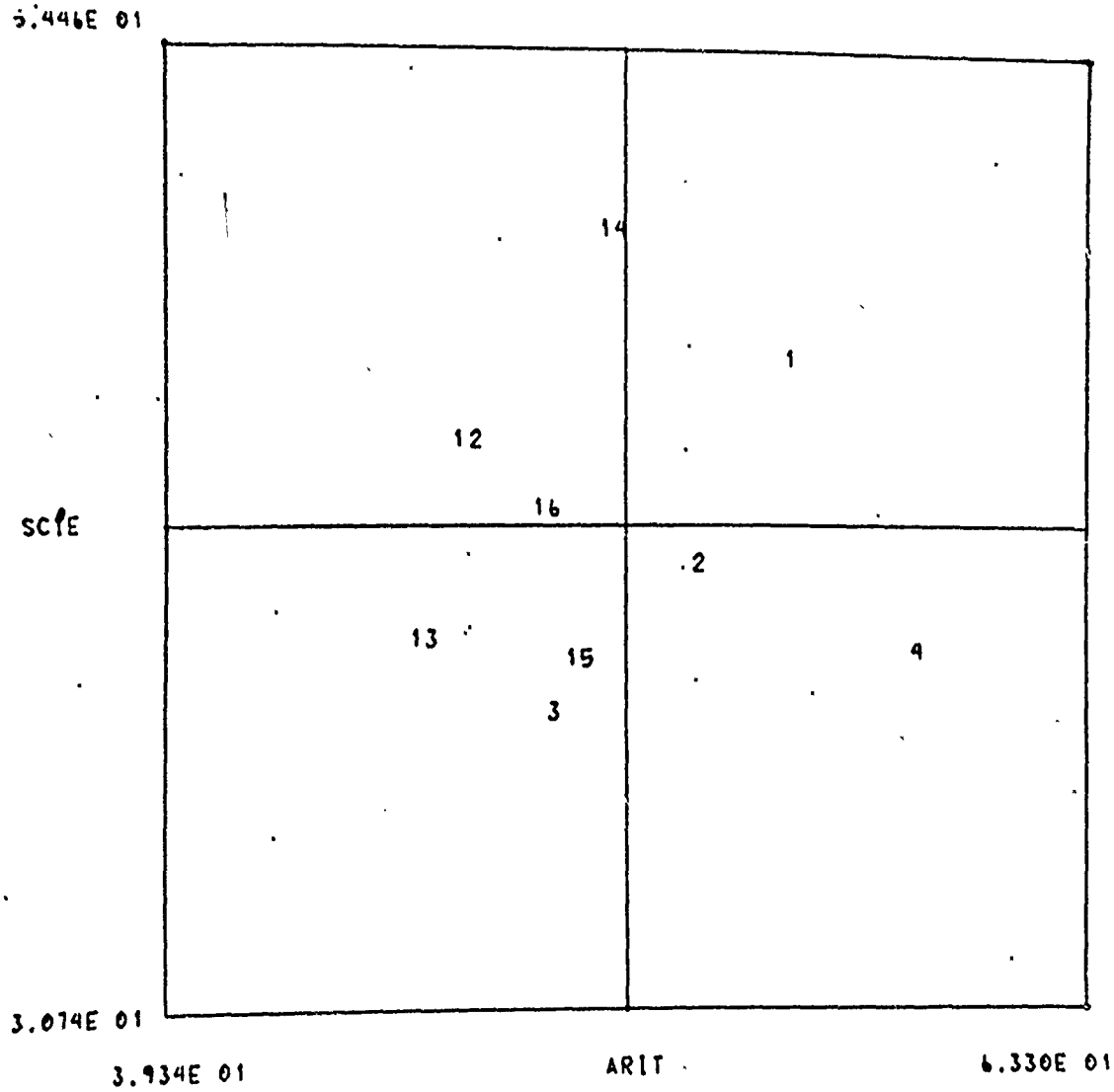

5.446E 01
14
1
12
16
SCYE
2
13
15
4
3
3.074E 01
3.934E 01
ARIT
6.330E 01

Figure 4.21

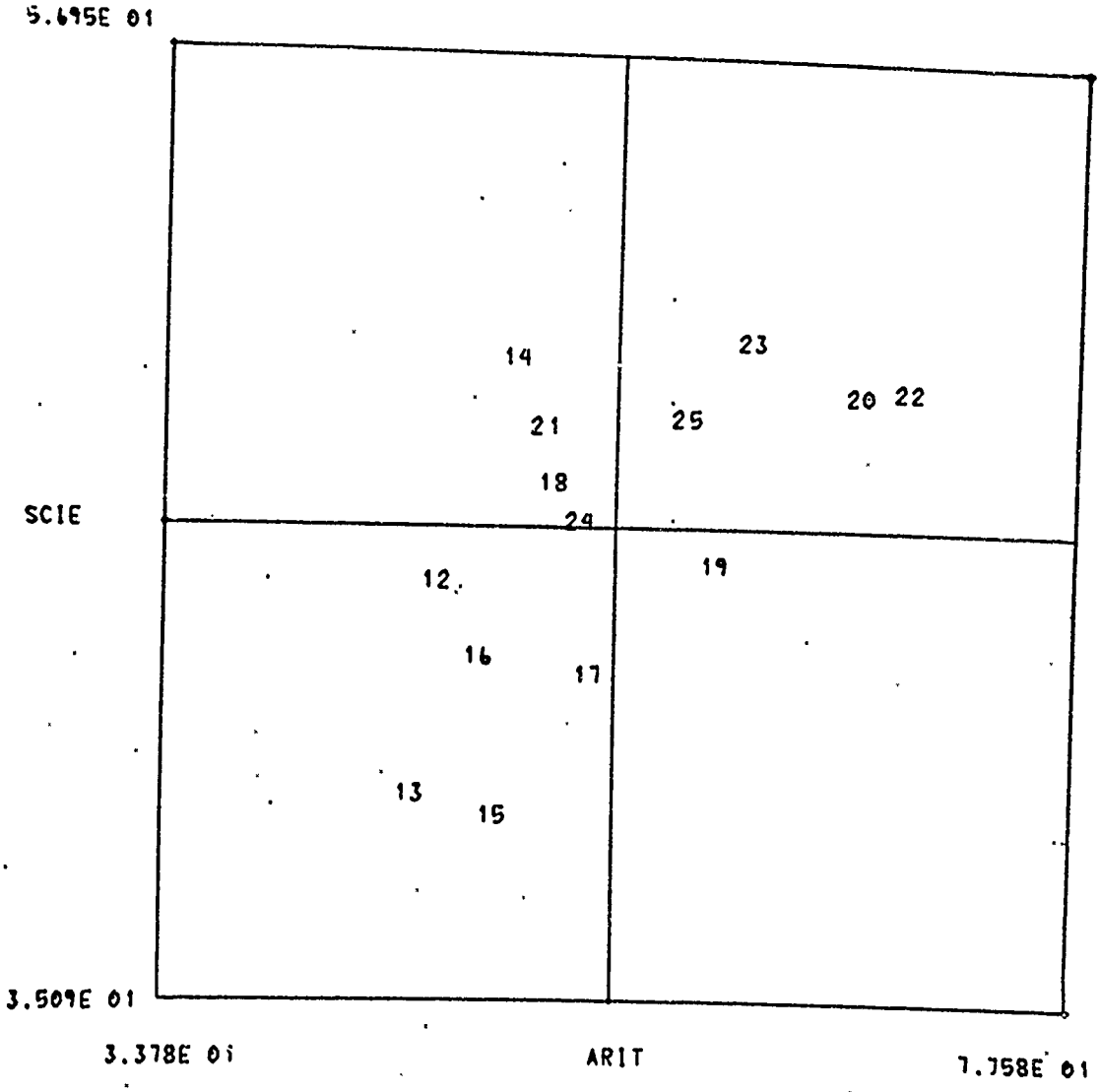

5.695E 01
SCIE
3.509E 01
3.378E 01
ARIT
7.758E 01
14
23
20 22
21
25
18
24
12
19
16
17
13
15

Figure 4.22

OUTPUT AREA

	SEX	TYPE	ENG	ARIT	SCIE
1	1	1	5.510E 01	5.560E 01	4.680E 01
2	1	1	5.610E 01	5.320E 01	4.170E 01
3	1	1	5.590E 01	4.950E 01	3.810E 01
4	1	1	5.440E 01	5.890E 01	3.960E 01
5	1	2	5.280E 01	5.940E 01	4.710E 01
6	1	2	5.360E 01	6.560E 01	4.210E 01
7	1	2	5.390E 01	5.920E 01	4.230E 01
8	1	2	5.380E 01	5.820E 01	4.520E 01
9	1	3	5.670E 01	6.230E 01	4.980E 01
10	1	3	5.820E 01	6.460E 01	4.700E 01
11	1	3	5.960E 01	5.280E 01	4.920E 01
12	2	1	5.700E 01	4.740E 01	4.480E 01
13	2	1	5.560E 01	4.630E 01	3.990E 01
14	2	1	5.800E 01	5.120E 01	5.000E 01
15	2	1	5.590E 01	5.030E 01	3.940E 01
16	2	1	5.970E 01	4.950E 01	4.310E 01
17	2	2	5.560E 02	5.480E 01	4.270E 01
18	2	2	6.170E 01	5.300E 01	4.710E 01
19	2	2	5.610E 01	6.080E 01	4.520E 01
20	2	2	6.030E 01	6.760E 01	4.920E 01
21	2	3	6.180E 01	5.260E 01	4.840E 01
22	2	3	5.560E 01	6.990E 01	4.930E 01
23	2	3	6.230E 01	6.240E 01	5.040E 01
24	2	3	5.990E 01	5.430E 01	4.620E 01
25	2	3	6.070E 01	5.940E 01	4.860E 01

REPLY AREA

Figure 4.23

```
                                OUTPUT AREA
                                                          PAGE   2L
                                      EFFECTIVENESS OF TEACHING
                                  VARIABLE ( ENG  ) TABLE OF MEANS
                                                  C O L U M N S =   TYPE
ROWS =
 SEX        IN EACH BLOCK, ROW 1 DENOTES CELL MEANS, ROW 2 DENOTES
CODED                                   C O D E D  L E V E L S
LEV.  1          2          3

1 5.537E 01 5.352E 01 5.817E 01 5.546E 01
    4           4          3          11
  7.798E-01 5.000E-01 1.450E 00

2 5.724E 01 1.835E 02 6.006E 01 9.433E 01
    5           4          5          14
  1.671E 00 2.483E 02 2.664E 00

COL.
  5.641E 01 1.185E 02 5.935E 01 7.723E 01
    9           8          8          25

              ESTIMATES OF SEX   ADJUSTED MEANS, ORDERED
    2          1
  9.655E 01 5.264E 01

              ESTIMATES OF TYPE  ADJUSTED MEANS, ORDERED
    2          1          3
  1.212E 02 5.661E 01 5.650E 01
                ANALYSIS OF VARIANCE FOR VARIABLE ENG
```

SOURCE OF VARIATION	D.F.	SUM OF SQUARES
SEX (ROWS)	1	1.175292E 04
TYPE (COLS)	2	2.254894E 04
SEX * TYPE (INTERACTION)	2	2.206148E 04
SUBTOTALS	5	5.391499D 04
ERROR	19	1.850470D 05
TOTAL	24	2.389620D 05

REPLY AREA

Figure 4.24

OUTPUT AREA

PAGE 2R

AND EFFECTS (SEX) VERSUS (TYPE)

CELL SIZE, AND ROW 3 DENOTES STANDARD DEVIATION.

MEAN SQUARE	F VS. ERROR	F VS. INTERACTION
5.019498E 01	1.289E 01	8.178E 00
2.236678E 01	5.745E 00	3.644E 00
6.137757E 00	1.576E 00	
2.341048E 01	6.013E 00	
3.893421E 00		
7.959476E 00		

ROOT MEAN SQUARE ERROR = 1.973175E 00
ROOT MEAN SQUARE INTERACTION = 2.477449E 00

REPLY AREA

Figure 4.25

OUTPUT AREA

YOUR RESPONSE VARIABLE NAMES WILL APPEAR BELOW ONE AT A TIME.
PRESS KEY 1 IF YOU WISH TO INCLUDE THE VARIABLE OR KEY 2 IF
YOU WISH TO DELETE THE VARIABLE.

ENG
ARIT
SCIE

THE ABOVE VARIABLES ARE THE ONES YOU DECIDED TO INCLUDE.
IF YOU AGREE, PRESS KEY 1; IF YOU WOULD LIKE TO TRY AGAIN,
PRESS KEY 2.

REPLY AREA

Figure 4.26

```
                                OUTPUT AREA
                                                          PAGE   4L
  EFFECTIVENESS OF TEACHING                                       MULTIVARIA
                                        MATRIX E AFTER ELIMINATION OF FA

            ENG          ARIT         SCIE
ENG   7.3975E 01 -6.2622E 01  3.1886E 01

            ENG          ARIT         SCIE
ARIT -6.2622E 01  4.9796E 02  6.7739E 01

            ENG          ARIT         SCIE
SCIE  3.1886E 01  6.7739E 01  1.7141E 02
                                        STEPWISE LOG DETERMINANTS

            ENG          ARIT         SCIE
      4.3037E 00  1.0402E 01  1.5325E 01
                                        CORRELATIONS BASED ON E

            ENG          ARIT         SCIE
ENG   1.0000E 00 -3.2628E-01  2.8316E-01

            ENG          ARIT         SCIE
ARIT -3.2628E-01  1.0000E 00  2.3186E-01

            ENG          ARIT         SCIE
SCIE  2.8316E-01  2.3186E-01  1.0000E 00
                                        SUBTOTALS (ALL EFFECTS) SEX  AND
LIKELIHOOD RATIO TEST STATISTIC, CHI-SQUARE =     0.3737E 02 WITH
                                SIGNIFICANT AT LEVEL     0.126E-02
                                        MATRIX H+E

            ENG          ARIT         SCIE
ENG   1.9106E 02 -3.9992E 01  1.3580E 02

            ENG          ARIT         SCIE
ARIT -3.9992E 01  9.8129E 02  2.4725E 02

            ENG          ARIT         SCIE
```

REPLY AREA

Figure 4.27

```
                                OUTPUT AREA
                                                          PAGE   5L
SCIE  1.3580E 02  2.4725E 02  3.4008E 02
                                        CORRELATIONS BASED ON H+E
          ENG          ARIT         SCIE
ENG   1.0000E 00 -9.2361E-02  5.3272E-01
          ENG          ARIT         SCIE
ARIT -9.2361E-02  1.0000E 00  4.2799E-01
          ENG          ARIT         SCIE
SCIE  5.3272E-01  4.2799E-01  1.0000E 00
STANDARDIZED ROY STATISTIC OR SQUARE OF A CANONICAL CORRELATION =
        READ HECK CHARTS WITH S =   3 M =      0.5 N =      7.5
                    WEIGHT OF DISCRIMINANT FUNCTION
          ENG          ARIT         SCIE
      4.6257E-02  1.2249E-02  8.4362E-03
     CORRELATIONS BETWEEN DISCRIMINANT FUNTION AND ORIGINAL VARIABLES
          ENG          ARIT         SCIE
      7.3635E-01  3.6639E-01  6.2056E-01
                                        INTERACTION SEX  * TYPE
LIKELIHOOD RATIO TEST STATISTIC, CHI-SQUARE.=     0.5889E 01 WITH
                              SIGNIFICANT AT LEVEL     0.437E 00
                                        MATRIX H+E
          ENG          ARIT         SCIE
ENG   8.6326E 01 -5.6571E 01  3.5484E 01
          ENG          ARIT         SCIE
ARIT -5.6571E 01  5.2809E 02  5.9087E 01
          ENG          ARIT         SCIE
SCIE  3.5484E 01  5.9087E 01  1.7649E 02
                                        CORRELATIONS BASED ON H+E
```

REPLY AREA

Figure 4.28

```
                                OUTPUT AREA
                                                                PAGE   6L

              ENG            ARIT           SCIE
ENG    1.0000E 00 -2.6495E-01  2.8748E-01

              ENG            ARIT           SCIE
ARIT -2.6495E-01  1.0000E 00  1.9354E-01

              ENG            ARIT           SCIE
SCIE  2.8748E-01  1.9354E-01  1.0000E 00

STANDARDIZED ROY STATISTIC OR SQUARE OF A CANONICAL CORRELATION =
          READ HECK CHARTS WITH S =    2 M =     0.0 N =     7.5

                    WEIGHT OF DISCRIMINANT FUNCTION

              ENG            ARIT           SCIE
          5.1753E-02  1.5122E-02 -1.6132E-02

       CORRELATIONS BETWEEN DISCRIMINANT FUNTION AND ORIGINAL VARIABLES

              ENG            ARIT           SCIE
          6.6301E-01  3.4509E-01 -1.6703E-02

                                                   SEX  EFFECTS

TEST STATISTIC F =      3.8465 WITH   3. AND    17. D.F.

                    WEIGHT OF DISCRIMINANT FUNCTION

              ENG            ARIT           SCIE
          9.4707E-02 -7.1749E-04  1.0651E-03

       CORRELATIONS BETWEEN DISCRIMINANT FUNTION AND ORIGINAL VARIABLES

              ENG            ARIT           SCIE
          9.9981E-01 -3.3809E-01  2.9237E-01

                                                   TYPE EFFECTS

LIKELIHOOD RATIO TEST STATISTIC, CHI-SQUARE =      0.2271E 02 WITH
                              SIGNIFICANT AT LEVEL      0.920E-03

                                   MATRIX H+E
```

REPLY AREA

Figure 4.29

OUTPUT AREA

PAGE 7L

	ENG	ARIT	SCIE
ENG	1.1871E 02	9.0370E-01	1.0189E 02

	ENG	ARIT	SCIE
ARIT	9.0370E-01	9.1454E 02	2.7959E 02

	ENG	ARIT	SCIE
SCIE	1.0189E 02	2.7959E 02	3.1971E 02

CORRELATIONS BASED ON H+E

	ENG	ARIT	SCIE
ENG	1.0000E 00	2.7427E-03	5.2303E-01

	ENG	ARIT	SCIE
ARIT	2.7427E-03	1.0000E 00	5.1707E-01

	ENG	ARIT	SCIE
SCIE	5.2303E-01	5.1707E-01	1.0000E 00

STANDARDIZED ROY STATISTIC OR SQUARE OF A CANONICAL CORRELATION =
READ HECK CHARTS WITH S = 2 M = 0.0 N = 7.5

WEIGHT OF DISCRIMINANT FUNCTION

ENG	ARIT	SCIE
3.4649E-02	1.5876E-02	1.3041E-02

CORRELATIONS BETWEEN DISCRIMINANT FUNTION AND ORIGINAL VARIABLES

ENG	ARIT	SCIE
4.7968E-01	6.1658E-01	7.0105E-01

REPLY AREA

Figure 4.30

CHAPTER V

COMPUTER PROGRAMS

The computer programs were written in FORTRAN IV for the IBM 360 Model 65 and operate under Graphics Monitor System (GMS). Extensive use is made of the COMFORT and COMPLOT graphics subroutine packages. The computational routines are from the MUDAID program, thus for more complete documentation, the reader is referred to A FORTRAN II PROGRAM FOR MUDAID: Multivariate, Univariate, and Discriminant Analysis of Irregular Data.

A load module under GMS is limited to 140K of core storage, and thus to reduce the size of the program an overlay structure was utilized. A main program or root was developed to control the flow of the program, and the remaining subroutines of the program were divided into segments which were not needed in core at the same time. Hence, several subroutines use the same core storage at different times. The overlay structure and listing of the job control cards are found at the end of this chapter.

This program uses five different data sets. Unit 11 (NW1) is a sequential file and is used to store the transformed data. Unit 12 (NW2) is also a sequential file and is used to store intermediate results and to pass them between overlays. Units 16, 17, and 38 are all direct access files. Units 16 (N1) and 17 (N2) each have 1980 records of 80 characters in length, are written under format control, and the associated variables are, respectively, IREC1 and IREC2. N1 and N2 are used to display output.

Unit 38 (N3) has 510 records of 80 characters in length, is written either with or without format control, and the associated variable is IREC3. N3 is used to store raw data for display and also to store selected common variables. N3 may be built within SPOOK or through the batch use of BUILD.

The main program serves as "traffic controller" to call the various subroutines. The order in which the subroutines is called depends largely on the variable N which is the number of the program function key depressed by the user. The main program also initially sets the variable MULT = 1 which forces the user to proceed through the univariate pass.

MAINA

This subroutine serves as control for the INPUT and PMATX subroutines. Once the multivariate option is indicated, MAINA only calls the subroutine PMATX.

INPUT

This subroutine is the main conversation unit of the entire program. By asking questions of the user, INPUT receives the raw data and provides the initial processing and storing of the data for future use.

In the general instructions, the user finds that key 30 can be depressed at any time to restart the program and that key 31 is the "panic button", i.e., any time it is depressed the program terminates. Each time the user answers a question, the answer will be displayed back to him and then he must either depress key 2 to reenter his response or any other key (not 30 or 31) to continue with the program. If an answer does not

fulfill the specific requirements, it will be displayed with an error message asking for the reentry of the response. Here, the program begins by asking the user if he has used the program previously or if he has entered his data by the use of the program BUILD. If the answer to this question is yes, the user is instructed to press program function key 2 to branch to statement 1001; otherwise, depression of key 1 will allow him to continue with the program and building of data.

The first query is for a study name header. This can be any combination of characters up to 60 characters in length. The title is optional and is used only for the heading of the output.

Beginning in statement 213 the user is asked for the number of response variables in his study. This number may range from one up to and including ten. If the number is not in this range, an error message is displayed asking for reentry of the response. In the 215 loop the user is asked to give each response variable a four character name or code so that he will be able to identify each variable in the output.

Each response variable can have a transformation assigned to it by the user selecting a code from the following list:

Code	Transformation
0, 1	no transformation
2	$\log_e(x)$ $(x > 0)$
3	$\log_e(1+x)$ $(x > -1)$
4	$\sqrt{x}$ $(x \geq 0)$
5	$1/x$ $(x > 0)$
6	$\arcsin(2x-1)$ $(0 < x < 1)$
7	available (now: no transformation)

A check is made to insure that the codes selected by the user are from the above list; if not, an error message appears. If the data assigned to a variable is "illegal" for the particular transformation, the data in question will then be set to -999 and thus later be ignored in the processing of data. An example of "illegal data" would be the square root or $\log_e$ of a negative number.

Because of limited space in the conversational mode, the user is limited to only two factors. As with the response variables, in loop 226 the user is asked to give each factor a four character name or code. Each factor may have a maximum of twelve levels. If the user has continuous data, the data will be encoded into levels by the user supplying "end codes". If one of the factors is age ranging from 5 to 25 years and is to grouped as follows:

5 - 9 years	level 1
10 - 14 years	level 2
15 - 19 years	level 3
20 - 25 years	level 4

the end codes would be 9, 14, 19, and 25. Loop 844 provides the encoding of the levels.

The last of the conversational segment is the entering of the data. The user is asked to indicate the level of each factor and the response of each variable. If the data on a particular variable is to be omitted, the user is instructed to enter -999 and thus the data will be later ignored.

Immediately after each set of data is entered, the raw data is displayed back and is also placed on direct access unit 38 for future

use in the 1001 segment of INPUT. Then the 106 loop places the factor levels into the coded levels, the 102 loop processes the variable responses through the appropriate transformations, and the data is stored on NW1 for future use.

Program function key 2 is no longer used to reenter data. Instead, the user types "BACK" and the number of entries he wants to go back. Then in the 380 loop the appropriate number of entries is removed from the screen, and NW1 and N3 are reset to the appropriate places. If only the work "BACK" is entered, only one entry is removed.

When all the data have been entered, the user depresses key 3 to indicate this. Control is then passed back to MAINA to begin the calculations.

The last segment of the INPUT routine begins at statement 1001 and is not accessible to the user until he has either seen the univariate results or has previously entered his data. The raw data are displayed from N3 and the segment allows the user to edit his data. To delete a record, the user types "DROP" and the subject number. The variable responses are then replaced by -999 which is always ignored in the calculations. To add data, the user gives the next subject number and the new record. The variable NSUBJ (number of data records) is then incremented. To change a record, the user types the record number and the new record. At this stage, changes are made only on the N3 unit. Once all the changes to the data are made, new transformations may be indicated. The data are then read from N3, reprocessed through the 106 and 102 loops, and stored on NW1 before the subroutine is left.

While in this segment of INPUT, the user may return to his previous univariate analysis output by depression of key 29 or to the plots of his data by depression of key 6.

A flow chart of this subroutine is included at the end of this chapter.

PMATX

This subroutine processes the data stored by INPUT on NW1. An initial decision is made depending on the value of the variable MULT. If MULT = 1, a univariate analysis is made; if MULT = 0, a multivariate analysis is made.

In the univariate pass (statements 511 - 600), the data are first checked for omitted data (-999). If missing data are found, the observation is ignored. PMATX stores on NW2 the following arrays: NN which consists of the incidence matrices, SUBT which is the array of cell totals, DSS which is the array of sums of squares in each cell, and DSQU which is the vector of the sum of squares of all observations for each variable, the latter being in double precision.

In the multivariate pass (statements 510 - 520), the number of variables to be used (NLUV) is first calculated by counting the nonzero elements in LUV. NLUV and NVBL are then exchanged. This exchange retains the value of NVBL so that the multivariate pass can be executed as many times as desired as if it were the first. Here, if missing data are found in any part of the record (signalled by ISET(13)), the entire record is eliminated. Also, if the LUV element for a variable is 0, the data for this variable are ignored. Once each record has been

processed through statement 505, the LUV vector is restored to its original values so that correct values will be obtained in all records.

For this multivariate pass, PMATX also stores, on NW2, DSSCP which is a matrix of sums of squares and cross products.

MAINC

For each response variable, the arrays DSSCP, NN, DSS, SUBT, and the vector DSQU are read from NW2. Because NN, DSS, and SUBT are read when needed, the storage areas are equivalenced thus reducing the size of the subroutine. An incidence matrix NN1 is constructed from NN; a matrix of sums of squares SS is constructed from DSS; a matrix of subtotals SUB is constructed from SUBT; and a variable DSQV which is the sum of squares of all observations is produced from the vector DSQU. MAINC then calls ANOT which performs the univariate analysis of each variable. After ANOT returns control, intermediate results once again are stored on NW2.

ANOT

This subroutine is called from MAINC once for each response variable. ANOT performs the univariate analysis by the use of adjusted normal equations.

If the number of rows is greater than the number of columns, exchanges are made in NR and NC, NX (incidence matrix), SQRX (cell sum of squares matrix), SUBX (cell totals matrix), and the variable IFLAG is set to 1. If any rows or columns are empty, a compacting routine then eliminates the rows or columns from NX, SQRX, and SUBX. Calculations of

the number of observations in each row (NIDOT), unadjusted row totals (R), number of observations in each column (NDOTJ), unadjusted column totals (C), grand total (G), total number of observations (NN), grand mean (GM), corrected sum of squares for total (SSTO), sum of squares for subtotals (SSB), sum of squares for error (SSE) are completed. In loop 60 the matrix CX of the "adjusted normal equations" and vector Q (vector of "adjusted row totals") are calculated. Loop 288 calculates the adjusted column totals (QB). The adjusted normal equations and adjusted row totals are reduced by one so the EQSYM inverts an (r-1)x(r-1) matrix. Estimates of row effects (RE) and column effects (CE) are both obtained by statement 216. Loop 92 obtains the cell standard deviations.

The analysis of variance begins with statement 215. Sums of squares, mean squares, F-ratios, and estimates of adjusted means (ordered) are all completed by statement 246.

The table for each variable which includes cell means, the number of observations in each cell, the standard deviation for each cell, unadjusted row and column means, estimates of row and column effects, and an analysis of variance table containing sums of squares, mean squares, degrees of freedom, F versus error, and F versus interaction is then placed on N1 and N2.

If the rows and columns were exchanged originally, they are exchanged again. ANOT then calls INTER before returning control to MAINC. ANOT returns to MAINC the estimates of row effects (RE), estimates of column effects (CE), the adjusted row totals (Q), and the adjusted column totals (QB). MAINC then stores these results on NW2.

A description of the algorithms employed is given in Section 3.1.

EQSYM

This subroutine solves the adjusted normal equations by the inversion of an (r-1)x(r-1) matrix where r is the number of levels of the factor with the least number of levels.

INTER

This subroutine is called from ANOT just before control is returned to MAINC. It displays a message to the user indicating the variables on which analysis has been completed. The user then has the choice of pressing program function key 1 to continue with calculations or key 2 to stop calculations. If key 2 is depressed, the variable NVBL (number of response variables) is reset to the variable IV (the number of the last response variable on which analysis has been completed). Control is then returned to ANOT.

OUTPUT

This subroutine is to display to the user the direct access units 16 and 17 on which all output was placed. Since a normal printout page is 131 characters wide and the maximum number of characters on the screen is 74 characters, unit 16 contains the left side of a page and statements 15 - 17 control the display while unit 17 contains the right side of a page and statements 25 - 27 control its display.

The user may see his output by using three program function keys: key 1 alternates the portion of the current page, key 2 increments the page number, and key 3 decrements the page number.

When the user finishes with his univariate results he determines where control is passed by the use of the program function keys: if he wants to continue to his multivariate results, he presses key 4 which passes control to subroutine DELETE; if he wants to see and edit his data, he presses key 5 which passes control to the 1001 statement of the INPUT subroutine; if he wants to see plots of his data, he presses key 6 which passes control to the subroutine PLOT.

When the user finishes with his multivariate results, he is asked to depress program function key 31 to terminate the program or key 4 to perform another multivariate analysis. Although the option is not mentioned on the display, the user may do exactly as he could when he completed the univariate passes; i.e., key 5 will pass control to the 1001 statement of INPUT and key 6 will pass control to PLOT.

PLOT

This subroutine allows the user to see his data of any two response variables plotted. In the general instructions, he is told to indicate combination of all levels of a factor by "0". Thus, when he gives the set of points which he wishes to see

2,2	indicates the (2,2) cell
0,4	indicates the 4th column
3,0	indicates the 3rd row
0,0	indicates all data

if the rows are factor 1. A check is made to insure that the user has requested a "legal" display; i.e., if the 4th column does not

exist for his data the user will be given an error message and asked to reenter his data. The user enters his two variable names, and once again a check is made on whether the names match those given in INPUT.

Loop 75 scans the raw data on NW1, picks out the appropriate points, and begins calculations of the means and standard deviations for the two variables. Immediately after the 75 loop, if the particular plot has one or fewer points, the user is given an error message and is asked to reenter a new set of levels. The upper and lower limits for each variable are calculated as the mean ± three sigma limits.

The plot displays the record number of the points. Thus, the user can quickly pick out which records he is seeing.

Once the user has seen the plot, control of the program depends upon his choice of the program function keys: key 1 will allow him to see additional plots, key 5 will allow him to return to the 1001 statement of INPUT to see and edit his data, and key 29 will allow him to return to the previous univariate analysis. As always, key 30 will allow the user to restart the program and key 31 will allow him to terminate the program.

DELETE

This subroutine is called from the root when the user has decided to continue with the multivariate option. Initially, the variable MULT is set to 0 to indicate the multivariate pass in future subroutines.

Each response variable name is displayed, and the user is asked to press program function key 1 to include the variable or key 2 to delete the variable. If key 1 is depressed, LUV(IK) = IK, MNLUV(IK) = IK,

and TNAM(IK) = VNAM(IK). If key 2 is depressed, LUV(IK) = 0, MNLUV(IK) = 0, TNAM(IK) = BLANK, and the name is erased from the screen. If the user is satisfied with the final list, beginning with statement 50 the TNAM array is then compressed so that it contains only the names of the response variables to be included in the multivariate analysis. Control is then returned to the root.

MAINB

This subroutine is not called until the multivariate option is entered. NW2 is backspaced one record, the incidence matrix NN1 is read, and the NW2 is rewound. DSSCP is read, and the NN, SUBT, DSS, and DSQU are all bypassed by dummy read statements. MAINB then uses the results from ANOT to build the input for GENHE which generates the H and E matrices (see Section 3.2). Once control is returned from GENHE, MAINB stores the results on NW2.

GENHE

This subroutine uses the adjusted row effect estimates, adjusted row totals, adjusted column effect estimates, adjusted column totals, the subtotal matrices, and the incidence matrix to compute the elements of the H matrices (sums of squares and products for hypotheses) and the E matrix (sums of squares and products for error).

MAINE

Here the data which will be needed by TEMAT are read from NW2. This subroutine also controls the intermediate output of results.

Before control is finally returned to the root, MAINE exchanges the values of NVBL and NLUV so that subsequent multivariate passes can have the original value of NVBL. This allows the user to use several different combinations of variables.

TEMAT

See Section 3.2.

TRI

This subroutine obtains the triangular matrix T such that TT' = A where A is a symmetric Gramian matrix. If A is singular, T will be rectangular such that the number of columns is equal to the rank of A.

FACE

This subroutine is called from TEMAT and it obtains the largest root and associated eigenvector of a symmetrix matrix.

WRIR

This subroutine is called from both MAINB and TEMAT. WRIR writes the symmetric matrices or vectors. The argument NN = 0 indicates vectors and NN = 1 indicates a symmetric matrix. Row and column numbers are replaced by names given to response variables (VNAM).

INSLD

This subroutine is called by EQSYM, MAINB, and TEMAT. INSLD obtains the inverse or conditional inverse of a Gramian matrix. It also obtains stepwise log-determinants, but if the matrix is singular these are just dummy values.

CHIX, GAMX, and YORMX

See Section 3.3.

ELGGM

This subroutine evaluates $\log_e \Gamma(x)$.

BUILD

This program is for the user of SPOOK who has voluminous data which are punched on cards. The user runs this batch program, and the needed data are stored on direct access unit 38 from where SPOOK obtains them.

Input cards must be prepared as follows:

Card 1: Title: up to 60 alphameric characters beginning in column 1

Card 2: Variable Designation and Transformations (see transformation codes under INPUT; blanks will be read as 0)

Col 1	blank
Col 2-3	number of response variables ($\leq$10)
Col 4-7	4-character name for variable 1

	Col 8	transformation code for variable 1
	Col 9-12	4-character name for variable 2
	Col 13	transformation code for variable 2
	⋮	
	Col 49-52	4-character name for variable 10
	Col 53	transformation code for variable 10
Card 3:	Col 5	number of variable FORMAT cards (1-5)

remaining columns may be blank or not; they will be ignored

Cards 4,5: One card for each of the two factors

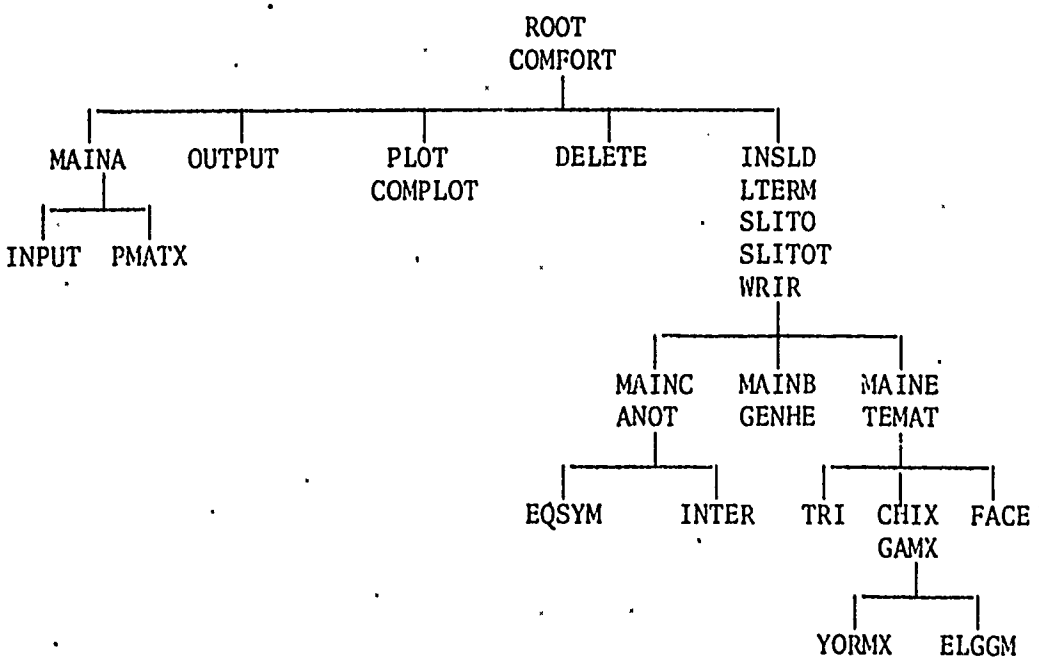
Col 1	blank
Col 2-5	4-character name for factor
Col 6-8	blank
Col 9-10	number of levels (≤ 12)
Col 11	ignored by program; may or may not be filled
Col 12-14	"end code" for level 1 (see explanation under INPUT)
Col 15	ignored
Col 16-18	"end code" for level 2
⋮	
Col 55	ignored
Col 56-58	"end code" for level 12

Cards 6-10: Variable FORMAT cards (number of cards must correspond to Col 5 of card 3); state the format for each record in usual FORTRAN FORMAT statement form; use all 80 columns of each card; start the first (or only) card with (, i.e., omit the word FORMAT; end the statement with); the factor levels are read first; hence, if the factor levels do not precede the variable responses, use T format.

Remaining cards are the data cards.

The output from this program is a list of the raw data as it will appear on the console when SPOOK is entered.

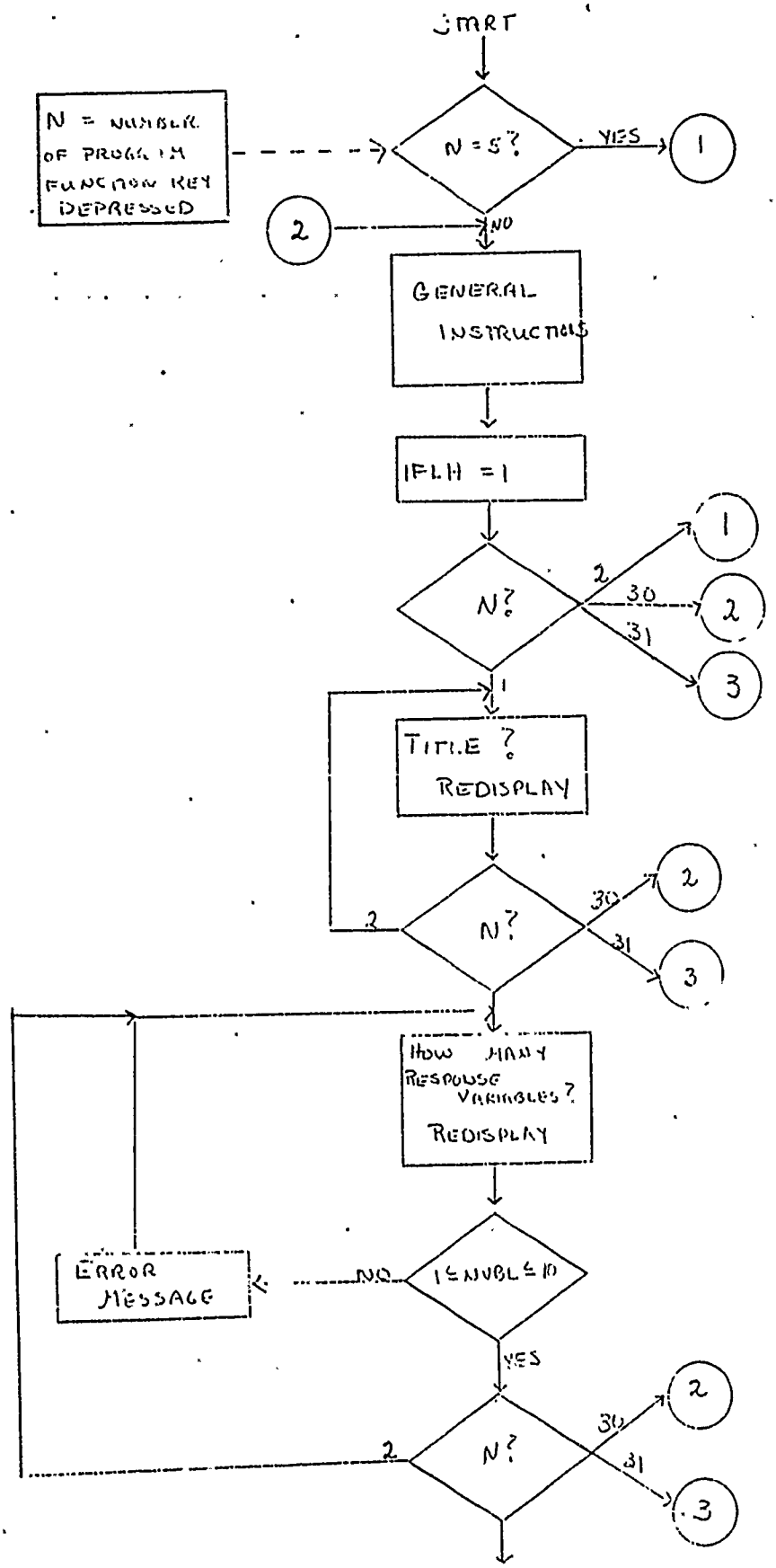
Overlay Structure

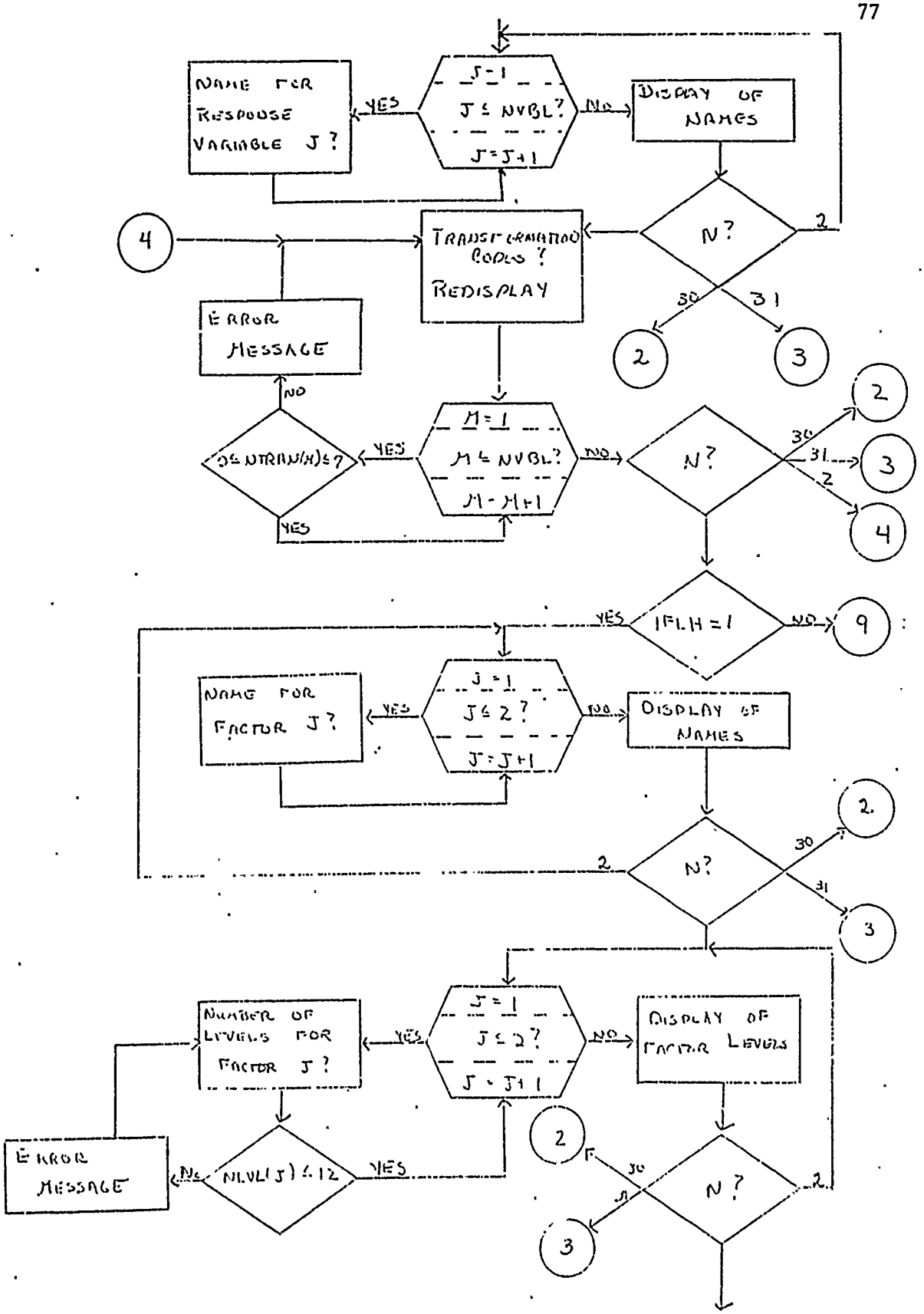


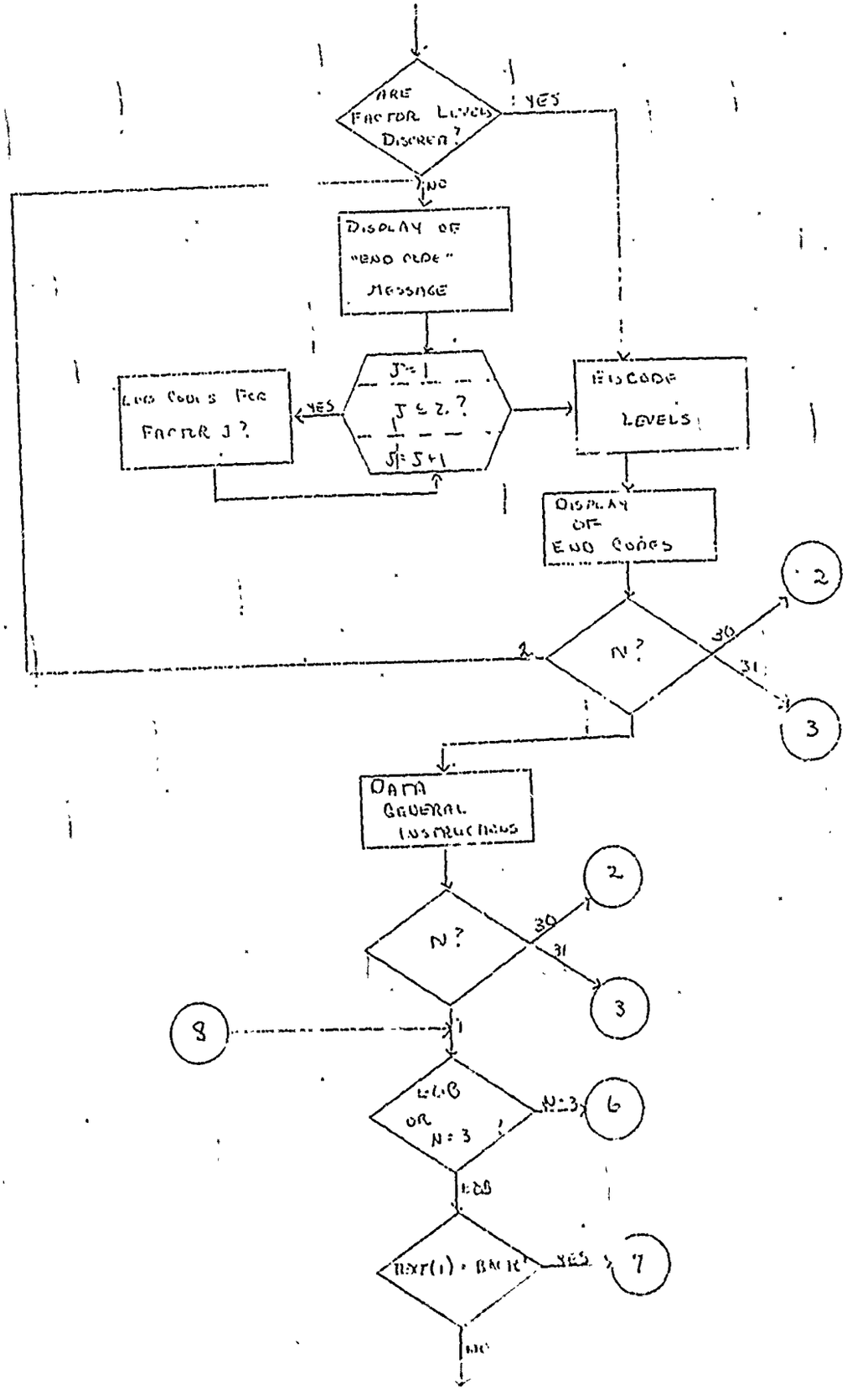
Job Control Cards for the Overlay Structure

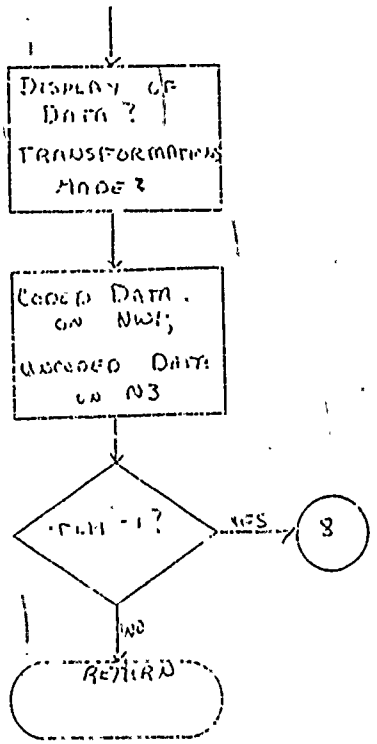
```
ENTRY MAIN
OVERLAY ONE
INSERT MAINA
OVERLAY TWO
INSERT INPUT
OVERLAY TWO
INSERT PMATX
OVERLAY ONE
INSERT OUTPUT
OVERLAY ONE
INSERT PLOT,COMPLOT
OVERLAY ONE
INSERT DELETE
OVERLAY ONE
INSERT INSLD,LTERM,SLITO,SLITOT,WRIR
OVERLAY TWO
INSERT MAINC,ANOT
OVERLAY THREE
INSERT EQSYM
OVERLAY THREE
INSERT INTER
OVERLAY TWO
INSERT MAINB,GENHE
OVERLAY TWO
```

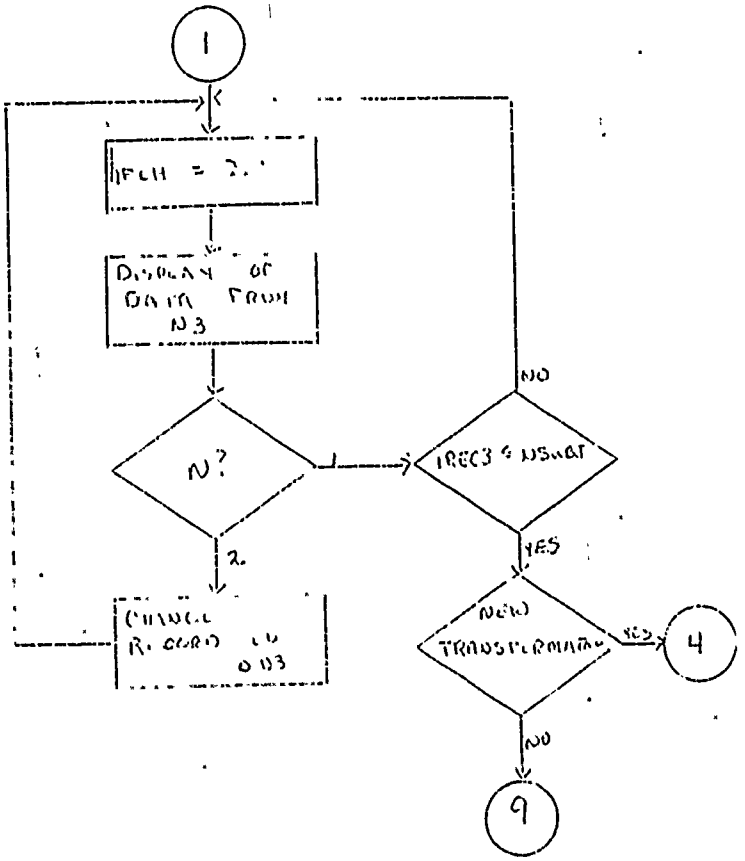
```
INSERT MAINE,TEMAT
OVERLAY THREE
INSERT TRI
OVERLAY THREE
INSERT FACE
OVERLAY THREE
INSERT CHIX,GAMX
OVERLAY FOUR
INSERT YORMX
OVERLAY FOUR
INSERT ELGGM
```

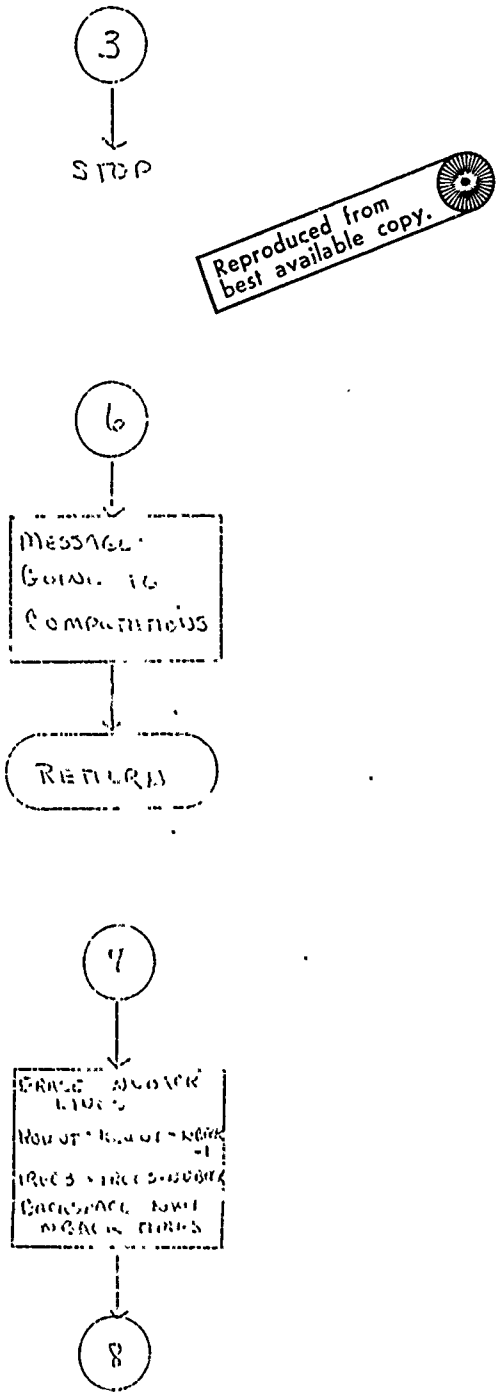

START
N = NUMBER OF PROGRAM FUNCTION KEY DEPRESSED
N = 5?
YES
1
2
NO
GENERAL INSTRUCTIONS
IFLH = 1
N?
2
30
31
1
2
3
1
TITLE ? REDISPLAY
N?
2
30
31
2
3
HOW MANY RESPONSE VARIABLES? REDISPLAY
ERROR MESSAGE
NO
1 ≤ NVBL ≤ 10
YES
N?
2
30
31
2
3


NAME FOR RESPONSE VARIABLE J ?
YES
J = 1
J ≤ NVBL ?
J = J+1
NO
DISPLAY OF NAMES
N ?
2
30
31
2
3
4
TRANSFORMATION CODES ?
REDISPLAY
ERROR MESSAGE
NO
0 ≤ NTRAN(M) ≤ 7
YES
M = 1
M ≤ NVBL ?
M = M+1
NO
N ?
30
31
2
2
3
4
YES
IFLH = 1
NO
9
NAME FOR FACTOR J ?
YES
J = 1
J ≤ 2 ?
J = J+1
NO
DISPLAY OF NAMES
N ?
2
30
31
2
3
NUMBER OF LEVELS FOR FACTOR J ?
YES
J = 1
J ≤ 2 ?
J = J+1
NO
DISPLAY OF FACTOR LEVELS
ERROR MESSAGE
NO
NLVL(J) ≤ 12
YES
2
30
N ?
2
3


ARE FACTOR LEVELS DISCRETE?
YES
NO
DISPLAY OF "END CODE" MESSAGE
J=1
J ≤ 2 ?
J=J+1
YES
END CODES FOR FACTOR J?
ENCODE LEVELS
DISPLAY OF END CODES
N?
30
31
2
3
DATA GENERAL INSTRUCTIONS
N?
30
31
2
3
8
EOB OR N=3
N=3
6
EOB
NEXT(1) = BACK
YES
7
NO


DISPLAY OF DATA ?
TRANSFORMATION MODE ?
CODED DATA ON NW1; UNCODED DATA ON N3
YES
8
NO
RETURN


1
DISPLAY OF DATA FROM N3
N ?
NO
YES
NEW TRANSFORMATION
YES
4
NO
9



Reproduced from best available copy.

REFERENCES

[1] Applebaum, Mark and Bargmann, Rolf E., A FORTRAN II PROGRAM FOR MUDAID: Multivariate, Univariate, and Discriminant Analysis of Irregular Data, University of Illinois, 1967.

[2] Bargmann, Rolf E., "Interpretation and Use of a Generalized Discriminant Function", Probability and Statistics, R. C. Bose, et. al. (eds), University of North Carolina Press, 1970, 35 - 60.

[3] Graybill, Franklin A., An Introduction to Linear Statistical Models, Vol. I, New York, McGraw-Hill Book Company, Inc., 1961.

[4] Penn, Lucius Wright, "An On-Line Statistical Computer System for Lay Usage", 2 Vol., Technical Report #68, University of Georgia, Department of Statistics, Athens, Georgia, May 1971.

[5] Swartz, Robert, "A Conversational Unit for Hierarchical Discriminant Analysis", Technical Report #66, University of Georgia, Department of Statistics, Athens, Georgia, April 1971

APPENDIX

COMPUTER LISTINGS

```
              C
              C
              C     THIS PROGRAM -- BUILD -- IS DESIGNED TO ALLOW THE USER
              C     OF  SPOOK  TO ENTER HIS DATA THRU BATCH MODE.
              C     THE DATA IS STORED ON DIRECT ACCESS UNIT 38
              C     FROM WHERE SPOOK OBTAINS IT.
              C
              C
0001                DIMENSION TTL(15),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),LEVEL(2,12),
                   1NTRAN(10),ISET(2),FMT(100),X(10)
0002                DEFINE FILE 38(510,80,L,IREC3)
0003                N3=38
0004                READ(5,25)(TTL(I),I=1,15)
0005                READ(5,26) NVBL,(VNAM(I),NTRAN(I),I=1,10)
0006            103 READ(5,27)NFMT
0007                DO 100 J=1,2
0008                READ(5,28)FNAM(J),NNN,(NEND(J,I),I=1,NNN)
0009            100 NLVL(J)=NNN
0010                DO 844 K=1,2
0011                JJ=NLVL(K)
0012                IF(NEND(K,JJ)) 999,999,844
0013            999 CONTINUE
0014                DO 998 LMN=1,JJ
0015                NEND(K,LMN)=LMN
0016            998 CONTINUE
0017            844 CONTINUE
0018                NFMT=NFMT*20
0019                READ(5,29)(FMT(I),I=1,NFMT)
0020                IREC3=502
0021                WRITE(N3'IREC3)TTL,VNAM,FNAM,NLVL,NEND,LEVEL,NTRAN,NVBL
0022                IREC3=1
0023                WRITE(6,583)(FNAM(I),I=1,2),(VNAM(I),I=1,5)
0024                WRITE(N3'IREC3,483)(FNAM(I),I=1,2),(VNAM(I),I=1,5)
0025                IF(NVBL.GT.5)WRITE(N3'IREC3,484)(VNAM(I),I=6,10)
0026                IF(NVBL.GT.5) WRITE(6,584)(VNAM(I),I=6,10)
0027                NSUBJ=0
0028            200 READ(5,FMT,END=201)(ISET(I),I=1,2),(X(K),K=1,NVBL)
0029                NSUBJ=NSUBJ+1
0030                WRITE(N3'IREC3,396)NSUBJ,ISET(1),ISET(2),(X(KQ),KQ=1,NVBL)
0031                WRITE(6,496)       NSUBJ,ISET(1),ISET(2),(X(KQ),KQ=1,NVBL)
0032                GO TO 200
0033            201 IREC3=510
0034                WRITE(N3'IREC3)NSUBJ
0035                STOP
0036             25 FORMAT(15A4)
0037             26 FORMAT(I3,10(A4,I1))
0038             27 FORMAT(4X,I1)
0039             28 FORMAT(1X,A4,2X,I3,12(1X,I3))
0040             29 FORMAT(20A4)
0041            396 FORMAT(3I3,1P5E12.3/9X,1P5E12.3)
0042            483 FORMAT(4X,2A4,5(4X,A4,4X))
0043            484 FORMAT(12X,5(4X,A4,4X))
0044            496 FORMAT(1H0,3I3,1P5E12.3/1H0,9X,1P5E12.3)
0045            583 FORMAT(1H1,4X,2A4,5(4X,A4,4X))
0046            584 FORMAT(1H0,12X,5(4X,A4,4X))
0047                END
```

```
0001          COMMON N,IDVLY,ITYPE,TTL(15),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),
             1LEVEL(2,12),NRA,NCA,NRB,NCB,JAB(8),N1,N2,NVBL,NPACT,NG12,MULT,
             2NSUBJ,NLUV ,NN1,NN2,NT,ISEG,NI2,NJ2,KOUNT,NDIAG,IREC1,IREC2,IREC3
             3,NTRAN(10),TRAN(10),LUV(10),NNLUV(10)
0002          DEFINE FILE 16(1980,80,E,IREC1),17(1980,80,E,IREC2),
             138(510,80,L,IREC3)
0003          MULT=1
0004        1 CALL MAINA
0005          IF(N.EQ.6)GO TO 5
0006          IF(N.EQ.24)GO TO 3
0007          CALL MAINC
0008          IF(N.EQ.30)GO TO 1
0009          IF(MULT.EQ.0)GO TO 7
0010        3 CALL OUTPUT
0011          IF(N.EQ.5.OR.N.EQ.30)GO TO 1
0012          IF(N.EQ.4)GO TO 4
0013          IF(N.EQ.6)GO TO 5
0014        5 CALL PLOT
0015          IF(N.EQ.29)GO TO 3
0016          IF(N.EQ.30.OR.N.EQ.5)GO TO 1
0017        4 CALL DELETE
0018          IF(N.EQ.30)MULT=1
0019          GO TO 1
0020        7 CALL MAINB
0021          CALL MAINE
0022          GO TO 3
0023        9 STOP
0024          END
```

```
0001          SUBROUTINE MAINA
0002          COMMON N,IOVLY,ITYPE,TT(615),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),
             1LEVEL(2,12),NRA,NCA,NRB,NCB,JAB(8),N1,N2,NVBL,NFACT,NG12,MULT,
             2NSUBJ,NLUV ,NW1,NW2,NY,ISEQ,NIZ,NJZ,KOUNT,NDIAG,IREC1,IREC2,IREC3
             3,NTRAN(10),TNAM(10),LUV(10),MNLUV(10)
0003          NI=1
0004          NJ=2
0005          IF(MULT.EQ.0.AND.N.NE.5)GO TO 902
0006          NW1 = 11
0007          NW2 = 12
0008      950 ISEQ = 0
0009          IFAC=0
0010          CALL INPUT
0011          IF(N.EQ.6.OR.N.EQ.29)GO TO 99
0012      902 CALL PMATX(NI,NJ)
0013      110 NIZ = NI
0014          NJZ = NJ
0015       99 RETURN
0016          END
```

```
0001          SUBROUTINE INPUT
0002          EXTERNAL PRINT,LGBINT
0003          COMMON N,ICVLY,ITYPE,TTL(15),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),
             1LEVEL(2,12),NRA,NCA,NRB,NCB,JAB(8),N1,N2,NVBL,NFACT,NG12,MULT,
             2NSUBJ,NLUV ,NW1,NW2,NT,ISEC,NIZ,NJZ,KOUNT,NDIAG,IREC1,IREC2,IREC3
             3,NTRAN(10),TNAP(10),LUV(10),NNLUV(10)
0004          DIMENSION SETUP(13),          ISET(13),TEXT(15),X(10),TEXT2(18),
             1TEXT3(20)
0005          EQUIVALENCE (SETUP(1),ISET(1)), (TEXT2(1),TEXT3(1))
0006          DOUBLE PRECISION DNP,DN(12)
0007          DATA CSIZE/'B'/
0008          DATA BLANK/'    '/
0009          DATA BACK/'BACK'/
0010          DATA GROP/'GROP'/
0011          CALL GRINIT(CSIZE)
0012          CALL GCECB(ECBIAT)
0013          NC2=72
0014          MASK1=1610612739
0015          MASK2=1073741827
0016          MASK3=268435459
0017          MASK4=1644167175
0018          MASK5=536870915
0019          N3=38
0020          NT=12
0021          NC=66
0022          NFAC1=2
0023          NG12=0
0024          NDIAG=14
0025          NOUR=4
0026          COTL=-999.
0027      371 FORMAT(18A4)
0028          IF(N.EQ.5)GO TO 1001
0029          CALL GCPEK(MASK1,PRINT)
0030          REWIND NW1
0031          REWIND NT
0032      211 FORMAT(15A4)
0033          IFLH=1
        C
        C
0034          NSTEP=1
0035      203 CALL GERAS(100)
0036          CALL GROPLY('                    ',20,6400)
0037          CALL GROPLY('                    ',20,6400)
0038          CALL GROPLY('THIS PROGRAM IS DESIGNED TO PERFORM AN ANALYSIS OF IR
             1REGULAR DATA.',66,6400)
0039          CALL GROPLY('YOU MAY HAVE A MAXIMUM OF 10 RESPONSE VARIABLES AND A
             2 MAXIMUM OF',64,6400)
0040          CALL GROPLY('2 FACTORS.  YOUR DESIGN MAY BE QUITE UNBALANCED, AND
             3WHOLE CELLS',64,6400)
0041          CALL GROPLY('MAY BE MISSING.  TRANSFORMATIONS CAN BE MADE.  FOR EA
             4CH PAIR OF',63,6400)
0042          CALL GROPLY('FACTORS, AN ANALYSIS OF VARIANCE IS PERFORMED FOR EAC
             5H RESPONSE',63,6400)
0043          CALL GROPLY('VARIABLE, SEPARATELY.  ONCE YOU SEE THESE UNIVARIATE
             1ANALYSES YOU',65,6400)
0044          CALL GROPLY('WILL BE GIVEN THE OPPORTUNITY TO SEE PLOTS OF YOUR DA
```

```
                1TA AND THE',63,&400)
0045             CALL GRDPLY('RAW DATA AGAIN.  THUS, YOU WILL BE ABLE TO EDIT YOUR
                1DATA AND RUN',65,&400)
0046             CALL GRDPLY('THE ANALYSES AGAIN.',19,&400)
0047             CALL GRDPLY('                    ',20,&400)
0048             CALL GRDPLY('TO BEGIN YOU MUST ANSWER QUESTIONS BY USING THE TYPEW
                7RITER',58,&400)
0049             CALL GRDPLY('KEYBOARD DIRECTLY.IN FRONT OF YOU.  TO SIGNAL YOUR CO
                8MPLETION',61,&400)
0050             CALL GRDPLY('OF QUESTIONS, FIRST DEPRESS THE "ALT" KEY, AND WHILE
                9HOLDING IT',63,&400)
0051             CALL GRDPLY('DOWN, DEPRESS THE "5" KEY.  THIS SEQUENCE WILL LEAD
                1PE',55,&400)
0052             CALL GRDPLY('REFERRED TO AS "EOB".  ONCE YOU ANSWER QUESTIONS, THE
                1 ANSWERS',61,&400)
0053             CALL GRDPLY('WILL BE DISPLAYED BACK TO YOU.  IF YOU ARE NOT SATISF
                1IED, PRESS',63,&400)
0054             CALL GRDPLY('KEY 2 TO REENTER DATA;  OTHERWISE, THE PROGRAM WILL C
                1ONTINUE ',61,&400)
0055             CALL GRDPLY('BY YOUR PRESSING ANY KEY.',25,&400)
0056             CALL GRDPLY(' ',1,&400)
0057             CALL GRDPLY('AT ANY TIME YOU MAY RESTART BY PRESSING KEY 3, OR TER
                1MINATE BY',62,&400)
0058             CALL GRDPLY('PRESSING KEY 31.',16,&400)
0059             CALL GRDPLY('                    ',20,&400)
0060             CALL GRDPLY(' ',1,&400)
0061             CALL GRDPLY('CAUTION:  DO NOT TRY TO SPEED UP THE PROGRAM BY ANSWE
                1RING',57,&400)
0062             CALL GRDPLY('QUESTIONS BEFORE THEY ARE ASKED.  THIS WILL ONLY CREA
                1TE PROBLEMS.',65,&400)
0063             CALL GRDPLY(' ',1,&400)
0064             CALL GRDPLY(' ',1,&400)
0065             CALL GRDPLY('IF YOU HAVE PREVIOUSLY USED THIS PROGRAM OR ENTERED Y
                1OUR DATA',61,&400)
0066             CALL GRDPLY('THROUGH BATCH MODE, PRESS KEY 2 TO SEE YOUR DATA.',49
                1,&400)
0067             CALL GRDPLY(' ',1,&400)
0068             CALL GRDPLY('PRESS KEY 1 TO PROCEED.',23,&400)
0069         205 CALL GWAIT
0070             IF(N .EQ. 30) GO TO 205
0071             IF (N.EQ.31) GO TO 210
0072             IF(N.EQ.2)GO TO 1001
0073             IF (N .EQ. 1) GO TO 206
0074             GO TO 205
          C
0075         206 CALL GERAS(100)
0076             NSTEP=2
0077             CALL GRDPLY('                    ',20,&400)
0078             CALL GRDPLY('                    ',20,&400)
0079             CALL GRDPLY('IF YOU DESIRE TO HAVE A TITLE TO HEAD YOUR OUTPUT, EN
                1TER IT NOW.',64,&400)
0080             CALL GRDPLY('IF YOU DO NOT DESIRE A TITLE, LEAVE THE RESPONSE AREA
                2 BLANK.',66,&400)
0081         811 CALL GWAIT
0082             IF(ITYPE.NE.3)GO TO 402
0083         212 CALL GRRPLY(TIL,NC)
```

```
0084          CALL GRDPLY('                    ',20,&400)
0085          CALL GRDPLY('YOUR TITLE IS',13,&400)
0086      234 CALL GRDPLY(TTL,NC,&400)
0087      214 CALL GWAIT
0088          IF(N.EQ.2) GO TO 212
0089          IF (N.EQ.30) GO TO 203
0090          IF (N.EQ.31) GO TO 210
        C
0091      213 CALL GERAS(100)
0092          NSTEP=3
0093          CALL GRDPLY('                    ',20,&400)
0094          CALL GRDPLY('                    ',20,&400)
0095      208 CALL GRDPLY('HOW MANY RESPONSE VARIABLES DO YOU HAVE?',40,&400)
0096      812 CALL GWAIT
0097          IF(ITYPE.NE.3)GO TO 402
0098          CALL XBLANK(TEXT,NC)
0099      297 CALL GRRPLY(TEXT,NC)
0100          INDEX=0
0101          CALL INX(TEXT,INDEX,NC,DNP,&401)
0102          NVBL=DNP+.01
0103          WRITE(NOUM,150)NVBL
0104      150 FORMAT('YOU HAVE',I3,'  RESPONSE VARIABLES.',28X)
0105          CALL FETCH(TEXT2,NCF,&400)
0106          CALL GRDPLY('                    ',20,&400)
0107          CALL GRDPLY(TEXT2,NCF,&400)
0108          IF ( 1 .LE. NVBL .AND. NVBL .LE.10) GO TO 204
0109          CALL GRDPLY('                    ',20,&400)
0110          CALL GRDPLY('THE NUMBER OF RESPONSE VARIABLES MUST BE AT LEAST 1 A
             1ND NO',58,&400)
0111          CALL GRDPLY('GREATER THAN 10.  PLEASE REENTER.',33,&400)
0112          GO TO 812
0113      204 CALL GWAIT
0114          IF(N.EQ.30) GO TO 203
0115          IF (N.EQ.31) GO TO 210
0116          IF (N.EQ.2) GO TO 213
        C
0117      201 CALL GERAS(100)
0118          NSTEP=4
0119          DO 215 J=1,NVBL
0120      217 WRITE(NT,216)J
0121      216 FORMAT('ENTER A FOUR LETTER NAME FOR RESPONSE VARIABLE',I3,11X)
0122          BACKSPACE NT
0123          READ(NT,211)TEXT
0124          CALL GRDPLY('                    ',20,&400)
0125          CALL GRDPLY(TEXT,NC,&400)
0126      803 CALL GWAIT
0127          IF(ITYPE.NE.3)GO TO 402
0128          CALL GRRPLY(TEXT,NC)
0129          CALL GERAS(2)
0130      215 VNAM(J) = TEXT(1)
0131          IF (NVBL.EQ.10) GO TO 157
0132          IA=NVBL+1
0133          DO 153 I=IA,10
0134      153 VNAM(I)=BLANK
0135      157 WRITE(NOUM,155)(VNAM(I),I=1,10)
0136      155 FORMAT(10(2X,A4))
```

```
0137          CALL FETCH(TEXT2,NCF,&400)
0138          CALL GRDPLY('                    ',20,&400)
0139          CALL GRDPLY('                    ',20,&400)
0140          CALL GRDPLY('YOUR NAMES ARE',14,&400)
0141          CALL GRDPLY(TEXT2,NCF,&400)
0142      160 CALL GWAIT
0143          IF (N.EQ.30) GO TO 203
0144          IF (N.EQ.31) GO TO 210
0145          IF (N.EQ.2) GO TO 201
        C
0146      224 CALL GERAS(100)
0147          NSTEP=5
0148          CALL GRDPLY('                    ',20,&400)
0149         ·CALL GRDPLY('FOR EACH RESPONSE VARIABLE ENTER A TRANSFORMATION COD
            1E FROM THE',63,&400)
0150          CALL GRDPLY('FOLLOWING LIST:',15,&400)
0151          CALL GRDPLY('                    ',20,&400)
0152          CALL GRDPLY('   CODE      TRANSFORMATION',25,&400)
0153          CALL GRDPLY('   0,1       NO TRANSFORMATION',29,&400)
0154          CALL GRDPLY('     2       LOGE(X)   (X>0)',27,&400)
0155          CALL GRDPLY('     3       LOGE(1+X)   (X>-1)',30,&400)
0156          CALL GRDPLY('     4       SQRT(X)   (X>=0)',28,&400)
0157          CALL GRDPLY('     5       1/X   (X>0)',23,&400)
0158          CALL GRDPLY('     6       ARCSIN(2X-1)   (0<X<1)',34,&400)
0159          CALL GRDPLY('             VARIANCE-STABILIZING TRANSFORMATION FOR P
            1ROPORTIONS',63,&400)
0160          CALL GRDPLY('     7       AVAILABLE (NOW: NO TRANSFORMATION)',46,
            1&400)
0161          CALL GRDPLY('                    ',20,&400)
0162          CALL GRDPLY('SEPARATE ALL ANSWERS BY COMMAS.',31,&400)
0163          CALL GRDPLY('                    ',20,&400)
0164      804 CALL GWAIT
0165          IF(ITYPE.NE.3)GO TO 402
0166      290 CALL GRRPLY(TEXT,NC)
0167         ·CALL INK(TEXT,NC,5,IER,DN(1),DN(2),DN(3),DN(4),DN(5))
0168          IF(IER.NE.0)GO TO 401
0169          IF(NVBL.LE.5)GO TO 219
0170          CALL INK(TEXT,NC,5,IER,DN(6),DN(7),DN(8),DN(9),DN(10))
0171          IF(IER.NE.0)GO TO 401
0172      219 CONTINUE
0173          DO 173 IJK=1,10
0174      173 NTRAN(IJK)=DN(IJK)
0175          WRITE(NDUM '55)(VNAM(I),I=1,10)
0176          CALL FETCH(TEXT2,NCF,&400)
0177          CALL GRDPLY(TEXT2,NCF,&400)
0178      174 WRITE(NDUM,175)(NTRAN(M),M=1,NVBL)
0179      175 FORMAT(10(I4,2X))
0180          CALL FETCH(TEXT2,NCF,&400)
0181          CALL GRDPLY(TEXT2,NCF,&400)
0182          DO 298 M=1,NVBL
0183          IF (NTRAN(M).GE.0 .AND. NTRAN(M).LE.7) GO TO 298
0184          WRITE(NT,299)NTRAN(M)
0185      299 FORMAT('A TRANSFORMATION OF',I3,' IS ILLEGAL.  REENTER ALL DATA.',
            17X)
0186          BACKSPACE NT
0187          READ(NT,211)TEXT
```

```
0188          CALL GRDPLY(TEXT,NC,&400)
0189          CALL GRDPLY('                    ',20,&400)
0190          GO TO 804
0191      298 CONTINUE
0192      169 CALL GWAIT
0193          IF (N.EQ.2) GO TO 224
0194          IF(N.EQ.30) GO TO 203
0195          IF (N.EQ.31) GO TO 210
0196      237 GO TO (225,660),IFLH
        C
0197      225 CALL GERAS(100)
0198          NSTEP=7
0199          DO 226 J=1,2
0200      229 WRITE(NOUN,227)J
0201      227 FORMAT('ENTER A FOUR LETTER NAME FOR FACTOR',I3,22X)
0202          CALL FETCH(TEXT2,NCF,&400)
0203          CALL GRDPLY('                    ',20,&400)
0204          CALL GRDPLY(TEXT2,NCF,&400)
0205      805 CALL GWAIT
0206          IF(ITYPE.NE.3)GO TO 402
0207          CALL GRRPLY(TEXT,NC)
0208          CALL GERAS(2)
0209      226 FNAM(J)=TEXT(1)
0210          WRITE(NOUN,326)FNAM(1),FNAM(2)
0211      326 FORMAT(2(2X,A4),40X)
0212          CALL FETCH(TEXT2,NCF,&400)
0213          CALL GRDPLY('                    ',20,&400)
0214          CALL GRDPLY('                    ',20,&400)
0215          CALL GRDPLY('YOUR NAMES ARE',14,&400)
0216          CALL GRDPLY(TEXT2,NCF,&400)
0217      327 CALL GWAIT
0218          IF (N.EQ.30) GO TO 203
0219          IF (N.EQ.31) GO TO 210
0220          IF (N.EQ.2) GO TO 225
        C
0221      228 CALL GERAS(100)
0222          NSTEP=8
0223          CALL GRDPLY('                    ',20,&400)
0224          CALL GRDPLY('                    ',20,&400)
0225          CALL GRDPLY('YOU MAY HAVE UP TO 12 LEVELS PER FACTOR.',40,&400)
0226          CALL GRDPLY('                    ',20,&400)
0227          DO 241 J=1,2
0228      243 WRITE(NOUN,240)FNAM(J)
0229      240 FORMAT('HOW MANY LEVELS DO YOU HAVE FOR FACTOR ',A4,'?',16X)
0230          CALL FETCH(TEXT2,NCF,&400)
0231          CALL GRDPLY(TEXT2,NCF,&400)
0232      806 CALL GWAIT
0233          IF(ITYPE.NE.3)GO TO 402
0234      291 CALL GRRPLY(TEXT,NC)
0235          CALL XBLANK(TEXT,NC)
0236          INDEX=0
0237          CALL INX(TEXT,INDEX,NC,DNP,&401)
0238          NLVL(J)=DNP+.01
0239          IF (NLVL(J).LE.12) GO TO 241
0240          CALL GRDPLY('NUMBER OF LEVELS CANNOT EXCEED 12.  TRY AGAIN.',46,
             1&400)
```

```
0241          GO TO 806
0242      241 CONTINUE
0243          WRITE(NDUR,330)FNAM(1),NLVL(1),FNAM(2),NLVL(2)
0244      330 FORMAT('FACTOR ',A4,' HAS',I3,' LEVELS, AND FACTOR ',A4,' HAS',I3,
             1' LEVELS.',3X)
0245          CALL FETCH(TEXT2,NCF,&400)
0246          CALL GRDPLY('                    ',20,&400)
0247          CALL GRDPLY(TEXT2,NCF,&400)
0248      242 CALL GWAIT
0249          IF (N.EQ.30) GO TO 203
0250          IF (N.EQ.31) GO TO 210
0251          IF(N.EQ.2) GO TO 228
        C
0252      331 CALL GFRAS(100)
0253          DO 340 K=1,2
0254          DO 340 I=1,12
0255          NEND(K,I)=0
0256      340 CONTINUE
0257          NSTEP=9
0258          CALL GRDPLY('                    ',20,&400)
0259          CALL GRDPLY('                    ',20,&400)
0260          CALL GRDPLY('YOUR FACTORS SHOULD BE IN DISCRETE LEVELS AS 1,2,...,
             112.  BUT',61,&400)
0261          CALL GRDPLY('IF YOUR DATA IS CONTINUOUS, GROUPING CAN BE DONE FOR
             1YOU.',57,&400)
0262          CALL GRDPLY('IF YOUR LEVELS ARE ALREADY DISCRETELY DEFINED, PRESS
             1KEY 1.',59,&400)
0263          CALL GRDPLY('IF YOUR DATA IS CONTINUOUS, PRESS KEY 2.',40,&400)
0264      245 CALL GWAIT
0265          IF (N.EQ.30) GO TO 203
0266          IF (N.EQ.31) GO TO 210
0267          IF (N.EQ.1) GO TO 247
0268          IF (N.EQ.2) GO TO 246
0269          GO TO 245
        C
0270      246 CALL GFRAS(100)
0271          NSTEP=10
0272          CALL GRDPLY('                    ',20,&400)
0273          CALL GRDPLY('                    ',20,&400)
0274          CALL GRDPLY('TO GROUP THE DATA, ASSIGN EACH LEVEL OF THE FACTOR AN
             1 "END CODE",64,&400)
0275          CALL GRDPLY('OR AN IDENTIFIER;  EG, IF ONE OF THE FACTORS IS AGE R
             1ANGING FROM',64,&400)
0276          CALL GRDPLY('5 YEARS TO 25 YEARS, WE MAY WISH TO ASSIGN AGES TO LE
             1VELS AS',60,&400)
0277          CALL GRDPLY('     5- 9 YEARS   LEVEL 1',25,&400)
0278          CALL GRDPLY('    10-14 YEARS   LEVEL 2',25,&400)
0279          CALL GRDPLY('    15-19 YEARS   LEVEL 3',25,&400)
0280          CALL GRDPLY('    20-25 YEARS   LEVEL 4',25,&400)
0281          CALL GRDPLY('THEN THE END CODE FOR LEVEL 1 WOULD BE 9, THE END COD
             1E FOR',58,&400)
0282          CALL GRDPLY('LEVEL 2 WOULD BE 14, ETC.',25,&400)
0283          CALL GRDPLY('THIS DATA WOULD BE ENTERED AS   9,14,19,25',42,&400)
0284          CALL GRDPLY('                    ',20,&400)
0285          DO 250 J=1,2
0286          NL=NLVL(J)
```

```
0287        256 WRITE(NOUM,251)NLVL(J),FNAM(J)
0288        251 FORMAT ('WHAT ARE THE "END CODES" FOR THE',I3,' LEVELS OF FACTOR '
               1,A4,'?',2X)
0289            CALL FETCH(TEXT2,NCF,&400)
0290            CALL GRDPLY(TEXT2,NCF,&400)
0291        807 CALL GWAIT
0292            IF(ITYPE.NE.3)GO TO 402
0293        293 CALL GRRPLY(TEXT,NC)
0294            CALL INK(TEXT,NC,8,IER,DN(1),DN(2),DN(3),DN(4),DN(5),DN(6),DN(7),
               1DN(8))
0295            IF(IER.NE.0)GO TO 401
0296            IF(NL.LE.8)GO TO 357
0297            CALL INK(TEXT,NC,4,IER,DN(9),DN(10),DN(11),DN(12))
0298            IF(IER.NE.0)GO TO 401
0299        357 DO 252 I=1,NL
0300        252 NEND(J,I)=DN(I)
0301        250 CONTINUE
          C
0302        247 CALL GERAS(100)
0303            DO 844 K=1,NFACT
0304            JJ = NLVL(K)
0305            IF(NEND(K,JJ))  999,999,844
0306        999 CONTINUE
0307            DO 998 LMN = 1,JJ
0308            NEND(K,LMN) = LMN
0309        998  CONTINUE
0310        844 CONTINUE
0311            CALL GRDPLY('                    ',20,&400)
0312            CALL GRDPLY('                    ',20,&400)
0313            DO 350 J=1,2
0314            NL=NLVL(J)
0315            WRITE(NOUM,345)NLVL(J),FNAM(J)
0316        345 FORMAT('THE "END CODES" FOR THE',I3,' LEVELS OF FACTOR ',A4,12X)
0317            CALL FETCH(TEXT2,NCF,&400)
0318            CALL GRDPLY(TEXT2,NCF,&400)
0319        352 WRITE(NOUM,356)    (NEND(J,I),I=1,NL)
0320        356 FORMAT(12(I3,1X),12X)
0321            CALL FETCH(TEXT2,NCF,&400)
0322        350 CALL GRDPLY(TEXT2,NCF,&400)
0323        360 CALL GWAIT
0324            IF(N.EQ.30) GO TO 203
0325            IF(N.EQ.31) GO TO 210
0326            IF(N.EQ.2) GO TO 246
          C
          C
0327        285 CALL GERAS(100)
0328            CALL GCPFK(MASK2,PFINT)
0329            NSTEP=11
0330            CALL GRDPLY('                    ',20,&400)
0331            CALL GRDPLY('                    ',20,&400)
0332            CALL GRDPLY('NOW YOU MUST SUPPLY THE DATA.  YOU WILL NEED TO INDIC
               1ATE THE',60,&400)
0333            CALL GRDPLY('LEVELS OF THE FACTORS AND THE RESPONSE OF EACH VARIAB
               1LE.',56,&400)
0334            CALL GRDPLY('     EXAMPLE:   2,7,412,34,5',28,&400)
0335            CALL GRDPLY('     INDICATES FACTOR 1 IS AT LEVEL 2, FACTOR 2 IS AT
```

```
                1 LEVEL 7,',62,&400)
0336             CALL GRDPLY('     THE RESPONSE FOR VARIABLE 1 IS 412, THE RESPONSE
                1 FOR',57,&400)
0337             CALL GRDPLY('     VARIABLE 2 IS 34, AND THE RESPONSE FOR VARIABLE
                13 IS 5.',60,&400)
0338             CALL GRDPLY('IF YOU WANT TO OMIT DATA ON A PARTICULAR RANDOM VARIA
                1BLE,',57,&400)
0339             CALL GRDPLY('REPLACE IT BY -999.',19,&400)
0340             CALL GRDPLY('     EXAMPLE:   2,7,412,-999,5',30,&400)
0341             CALL GRDPLY('     INDICATES THAT THE DATA FOR THE SECOND RESPONSE'
                1,52,&400)
0342             CALL GRDPLY('     VARIABLE IS OMITTED.',25,&400)
0343             CALL GRDPLY(' ',1,&400)
0344             CALL GRDPLY('WHEN YOU HAVE ENTERED ALL DATA, PRESS KEY 3.',44,&400
                1)
0345             CALL GRDPLY('                          ',20,&400)
0346             CALL GRDPLY('SHOULD YOU REALIZE YOU HAVE A MISTAKE IN YOUR DATA, T
                1YPE "BACK"',63,&400)
0347             CALL GRDPLY('AND THE NUMBER OF ENTRIES YOU WISH TO GO BACK.',46,
                1&400)
0348             CALL GRDPLY('                          ',20,&400)
0349             CALL GRDPLY('WHEN YOU ARE READY TO ENTER YOUR DATA, PRESS KEY 1.',
                151,&400)
0350             IREC3=512
0351             WRITE(N3'IREC3)TTL,VNAM,FNAM,NLVL,NEND,LEVEL,NTRAN,NVBL
0352             IREC3=1
0353             WRITE(N3'IREC3,483)BLANK,(FNAM(I),I=1,2),(VNAM(I),I=1,5)
0354         483 FORMAT(3A4,5(4X,A4,4X))
0355             KODE=0
0356             KOUNT=0
0357         269 CALL GWAIT
0358             IF (N.EQ.30) GO TO 203
0359             IF (N.EQ.31) GO TO 210
0360             IF (N.EQ.1) GO TO 482
0361             GO TO 269
           C
0362         482 CALL GERAS(100)
0363             CALL GRDPLY('                          ',20,&400)
0364             WRITE(NDUM,483)BLANK,(FNAM(I),I=1,2),(VNAM(I),I=1,5)
0365             CALL FETCH(TEXT2,NCF,&400)
0366             CALL GRDPLY(TEXT2,NCF,&400)
0367             IF(NVBL.LE.5) GO TO 485
0368             WRITE(NDUM,484)(VNAM(I),I=6,10)
0369             WRITE(N3'IREC3,484) (VNAM(I),I=6,10)
0370         484 FORMAT(12X,5(4X,A4,4X))
0371             CALL FETCH(TEXT2,NCF,&400)
0372             CALL GRDPLY(TEXT2,NCF,&400)
0373         485 CONTINUE
0374             CALL GRDPLY('                          ',20,&400)
0375             NSTEP=12
0376             CALL GCPFK(MASK3,PFINT)
0377             SETUP(13)=0.0
0378         294 KOUNT=KOUNT+1
0379         808 CALL GWAIT
0380             IF(ITYPE.EQ.1.AND.N.EQ.3)GO TO 390
0381             IF(ITYPE.NE.3)GO TO 402
```

```
0382          508 CALL GRRPLY(TEXT,NC)
0383              IF(TEXT(1).EQ.BACK) GO TO 380
0384              CALL INK(TEXT,NC,7,IER,DN(1),DN(2),DN(3),DN(4),DN(5),DN(6),DN(7))
0385              IF(IER.NE.0)GO TO 507
0386              IF(NVBL.LE.5)GO TO 260
0387              CALL INK(TEXT,NC,5,IER,DN(8),DN(9),DN(10),DN(11),DN(12))
0388              IF(IER.NE.0)GO TO 507
0389          260 ISET(1)=DN(1)
0390              ISET(2)=DN(2)
0391              DO 261 IZ=1,10
0392          261 X(IZ)=DN(IZ+2)
0393              IF(KCDE.NE.0) CALL GBKSP(KCDE)
0394              IF(NVBL.LT.5)WRITE(NDUN,370)KOUNT,(ISET(I),I=1,2),(X(I),I=1,NVBL)
0395              IF(NVBL.GE.5)WRITE(NDUN,370)KOUNT,(ISET(I),I=1,2),(X(I),I=1,5)
0396          370 FORMAT(I4,2(I3,1X),1P5E12.3)
0397              CALL FETCH(TEXT2,NCF,6400)
0398          374 CALL GRDPLY(TEXT2,NCF,6500)
0399              IF(NVBL.LE.5) GO TO 101
0400              WRITE(NDUN,375)(X(I),I=6,NVBL)
0401          375 FORMAT(12X,5(2X,1PE10.3))
0402              CALL FETCH(TEXT2,NCF,6400)
0403          378 CALL GRDPLY(TEXT2,NCF,6502)
0404          396 FORMAT(3I3,1P5E12.3/9X,1P5E12.3)
0405          101 KODE=0
0406              WRITE(N3'IREC3,396)KCUNT,(SET(1),ISET(2),(X(KQ),KQ=1,NVBL)
            C
0407          621 DO 106 K=1,NFACT
0408              JJ=NLVL(K)
0409          107 DO 108 J=1,JJ
0410              IF(ISET(K)-NENC(K,J))109,109,108
0411          109 ISET(K)=J
0412              GO TO 106
0413          108 CONTINUE
0414              ISET(K)=JJ
0415          106 CONTINUE
0416              DO 102 K=1,NVBL
0417              IF(X(K) - CUTL) 103,10,103
0418          103 KK=NTRAN(K)+1
0419              GO TO(1,1,2,3,4,5,6,7),KK
0420          1   SETUP(K+2)=X(K)
0421              GO TO 102
0422          2   IF(X(K))10,10,11
0423           10 SETUP(K+2)=OLTL
0424              SETUP(13)=1.0
0425              GO TO 102
0426          11  SETUP(K+2)=ALOG(X(K))
0427              GO TO 102
0428          3   IF(X(K)+1.0)10,10,12
0429          12  SETUP(K+2)=ALOG(X(K)+1.0)
0430              GO TO 102
0431          4   IF(X(K))10,13,13
0432          13  SETUP(K+2)=SQRT(X(K))
0433              GO TO 102
0434          5   IF(X(K))10,10,14
0435          14  SETUP(K+2)=1.0/X(K)
0436              GO TO 102
```

```
0437        6     IF(X(K)*(1.-X(K)))10,15,15
0438          15  SETUP(K+2)=ARSIN(2.*X(K)-1.)
0439              GO TO 102
0440        7     SETUP(K+2)=X(K)
            C     ******************DUMMY SUBROUTION*****************************************
0441        102   CONTINUE
0442              IREX=IREC3
0443              IREC3=510
0444              IF(IFLH.EQ.1)WRITE(N3'IREX)KOUNT
0445              IF(IFLH.EQ.2)WRITE(N3'IREX)NSUBJ
0446              IREC3=IREX
0447              WRITE (NW1) SETUP
0448              SETUP(13)=0.0
0449              GO TO (294,664),IFLH
            C
0450          390 CALL GERAS(1C0)
0451              NSUBJ=KOUNT-1
0452              CALL GCPFK(MASK2,PFINT)
0453              CALL GRDPLY(' ',1,&400)
0454              CALL GRDPLY(' ',1,&400)
0455              CALL GRDPLY('NOW THAT YOU HAVE ENTERED ALL OF YOUR DATA, YOU CAN P
                 1RESS KEY 1',63,&400)
0456              CALL GRDPLY('TO CONTINUE.  COMPUTATIONS MAY TAKE SEVERAL MINUTES.
                 1 BE PATIENT ',65,&400)
0457          391 CALL GWAIT
0458              IF (N.EQ.30) GO TO 203
0459              IF(N.EQ.1) GO TO 801
0460              IF(N.EQ.31) GO TO 210
0461              GO TO 391
0462          801 CONTINUE
0463          802 REWIND NW1
0464              CALL GERAS(1C0)
0465              CALL GRRLSE
0466              RETURN
            C
            C
0467          400 GO TO(203,206,213,201,224,237,225,228,241,246,285,210,604),NSTEP
0468          401 CALL GRDPLY('FORMAT ERROR.  REENTER DATA.',28,&400)
0469              IF(NSTEP.EQ.12)KODE=KODE+1
0470              GO TO(210,210,812,210,804,210,210,806,210,807,210,808,809),NSTEP
0471          402 IF(ITYPE.EQ.1.AND.N.EQ.30)GO TO 203
0472              IF(ITYPE.EQ.1.AND.N.EQ.31)GO TO 210
0473              CALL GRDPLY('YOU SHOULD BE IN A POSITION REQUIRING EOB SEQUENCE.',
                 151,&400)
0474              IF(NSTEP.EQ.12)KODE=KODE+1
0475              GO TO (210,811,812,803,804,210,805,806,210,807,210,808,809),NSTEP
0476          380 INDEX=4
0477              IF(TEXT(2).NE.BLANK) GO TO 384
0478              NBACK=1
0479              GO TO 382
0480          384 CALL INX(TEXT,INDEX,NC,DNP,&503)
0481              NBACK=DNP+.01
0482          382 NNBACK=NBACK
0483              IF(NVBL.LE.5) GO TO 385
0484              NNBACK=2*NNBACK
0485          385 CALL GBKSP(NNBACK)
```

```
0486              KOUNT=KOUNT-NBACK-1
0487              DO 383 I=1,NBACK
0488        383   BACKSPACE NW1
0489              IREC3=IREC3-NNBACK
0490              GO TO 294
0491        500   CALL GERAS(1)
0492              GO TO 374
0493        502   CALL GERAS(1)
0494              GO TO 378
0495        503   CALL GRDPLY('FORMAT ERROR.  PLEASE TRY AGAIN.',32,6504)
0496              GO TO 508
0497         504  CALL GERAS(1)
0498              GO TO 503
0499         507  CALL GRDPLY('FORMAT ERROR.  REENTER DATA.',28,6509)
0500              KODE=KODE+1
0501              GO TO 508
0502        509   CALL GERAS(1)
0503              GO TO 507
0504         210  CALL GERAS(100)
0505              CALL GRRLSE
0506              STOP
            C
            C
0507        1001  REWIND NW1
0508              CALL GCPFK(MASK4,PFINT)
0509              IFLH=2
0510              IREC3=502
0511              READ(N3'IREC3)TTL,VNAM,FNAM,NLVL,NEND,LEVEL,NTRAN,NVBL
0512              IREC3=510
0513              READ(N3'IREC3)NSUBJ
0514              IREC3=1
0515              INCR=1
0516              IF(NVBL.GT.5)INCR=2
0517         604  IQ=0
0518              NSTEP=13
0519              IRECX=IREC3
0520              CALL GERAS(100)
0521        5999  FORMAT(20A4)
0522         603  DO 610 IY=1,INCR
0523              READ(N3'IREC3,5999)TEXT3
0524              IQ=IQ+1
0525         610  CALL GRDPLY(TEXT2,NC2,6400)
0526              IF(IQ.LT.40.AND.IREC3.LE.INCR+NSUBJ+INCR)GO TO 603
0527              CALL GRDPLY(' ',1,6400)
0528              CALL GRDPLY('PRESS KEY 1 TO CONTINUE OR KEY 2 TO CHANGE DATA.',48,
                 16400)
0529              CALL GRDPLY('IF, AT ANY TIME, YOU WISH TO RETURN TO YOUR PREVIOUS'
                 ',52,6400)
0530              CALL GRDPLY('UNIVARIATE ANALYSIS OUTPUT, PRESS KEY 29.',41,6400)
0531              CALL GRDPLY('IF YOU WISH TO SEE THE PLOTS OF YOUR DATA, PRESS KEY
                 16.',55,6400)
0532         602  CALL GWAIT
0533              IF(N.LO.6)GO TO 660
0534              IF(N.EQ.29)GO TO 602
0535              IF(N.LO.30)GO TO 203
0536              IF(N.EQ.31)GO TO 210
```

```
0537          IF(N.EQ.1.AND.IREC3.GT.INCR*NSUBJ)GO TO 670
0538          IF(N.EQ.1) GO TO 604
0539          IF(N.EQ.2)GO TO 606
0540          GO TO 602
0541      606 CALL GBKSP(4)
0542          CALL GRDPLY('IF YOU WISH TO DELETE AN ENTIRE RECORD, TYPE "DROP" A
             1ND ITS',59,6400)
0543          CALL GRDPLY('RECORD NUMBER.',14,6400)
0544          CALL GRDPLY('IF YOU WISH TO MAKE CHANGES IN A RECORD, TYPE ITS REC
             .1ORD NUMBER',63,6400)
0545          CALL GRDPLY('AND THE NEW RECORD IN THE ORIGINAL FORMAT.  SEPARATE
             1ALL NUMBERS',64,6400)
0546          CALL GRDPLY('BY COMMAS.     YOU MAY NOW ENTER YOUR CHANGES.',46,
             16400)
0547      809 CALL GWAIT
0548          IF(N.EQ.29)GO TO 802
0549          IF(ITYPE.NE.3)GO TO 402
0550      625 CALL GRRPLY(TEXT,NC)
0551          CALL XBLANK(TEXT,NC)
0552          IF(TEXT(1).EQ.ORGP)GO TO 614
0553          INDEX=0
0554          NVX=NVBL+3
0555          DO 611 KS=1,NVX
0556          KSS=KS
0557          CALL INX(TEXT,INDEX,NC,DNP,6401)
0558          SETUP(KS)=DNP
0559      611 CONTINUE
0560          DO 616 KS=1,NVBL
0561      616 X(KS)=SETUP(KS+3)
0562      628 IRC=SETUP(1)
0563          IF(IRC.GT.NSUBJ)NSUBJ=NSUBJ+1
0564          ISET(1)=SETUP(2)
0565          ISET(2)=SETUP(3)
0566      629 IREC3=(IRC-1)*INCR+1+INCR
0567          WRITE(N3'IREC3,396)IRC,ISET(1),ISET(2),(X(KT),KT=1,NVBL)
0568      630 IREC3=IRECX
0569          GO TO 604
0570      614 INDEX=4
0571          CALL INX(TEXT,INDEX,NC,DNP,6401)
0572          IRC=DNP
0573          ISET(1)=0
0574          ISET(2)=0
0575          DO 626 JQ=1,NVBL
0576      626 X(JQ)=OUTL
0577          GO TO 629
0578      660 REWIND N3
0579          IREC3=INCR+1
0580          NVBX=NVBL+3
0581          KCCT=0
0582      664 IF(KCCT.GE.NSUBJ)GO TO 802
0583          KCCT=KCCT+1
0584          READ(N3'IREC3,396)(ISET(IR),IR=1,3),(SETUP(IV),IV=4,NVBX)
0585          ISET(1)=ISET(2)
0586          ISET(2)=ISET(3)
0587          DO 661 KB=1,NVBL
0588      661 X(KB)=SETUP(KB+3)
```

```
0589          GO TO 421
0590      670 CALL GERASE(100)
0591          CALL GRDPLY(' ',1,&400)
0592          CALL GRDPLY('IF YOU WOULD LIKE TO INDICATE NEW TRANSFORMATIONS FOR
             1',53,&400)
0593          CALL GRDPLY('ANY OF YOUR RESPONSE VARIABLES, PRESS KEY 2.',44,
             1&400)
0594          CALL GRDPLY('IF YOU WISH TO CONTINUE TO YOUR UNIVARIATE ANALYSIS,'
             1,52,&400)
0595          CALL GRDPLY('PRESS KEY 1.',12,&400)
0596      675 CALL GWAIT
0597          IF(N.EQ.30)GO TO 203
0598          IF(N.EQ.31)GO TO 210
0599          IF(N.EQ.1) GO TO 660
0600          IF(N.EQ.2)GO TO 224
0601          GO TO 675
0602          END
```

```
0001          SUBROUTINE PMATX(NI,NJ)
        C
        C     GENERATES INCIDENCE MATRICES AND SUBTOTALS MATRICES
        C     FOR A FACTOR PAIR, FOR ALL RESPONSE VARIABLES.
        C
        C     AN INITIAL DECISION IS MADE ON THE VALUE OF MULT
        C           IF MULT = 0        DO THE MULTIVARIATE ANALYSIS
        C           IF MULT = 1        DO UNIVARIATE ANALYSES
        C
        C
0002          COMMON N,IGVLY,ITYPE,TTL(15),VNAM(10),FNAM(2)-NLVL(2),NEND(2,12),
             1LEVEL(2,12),NRA,NCA,NRB,NCB,JAB(8),N1,N2,NVBL,NFACY,NG12,MULT,
             2NSUBJ,NLUV ,NW1,NW2,NT,ISEC,NIZ,NJZ,NOUNT,NDIAG,IREC1,IREC2,IREC3
             3,NTRAN(10),TRAN(10),LUV(10),NNLUV(10)
0003          DOUBLE PRECISION A,B,C,DSQU,DSSCP
0004          DIMENSION DSQU(10),DSSCP(10,10),SETUP(13),DSS(12,12,10),SUBT(12,12
             1,10),ISET(13),NN(12,12,10)
0005          EQUIVALENCE (SETUP(1),ISET(1))
0006          REWIND NW1
0007          IF(NSUBJ.GT.500) GO TO 99
0008          CUTL=-999.
0009          DO 500 K=1,10
0010          DSQU(K) = 0.0D0
0011          DO 500 I = 1,12
0012          DO 500 J = 1,12
0013          NN(I,J,K) = 0.0
0014          DSS(I,J,K) = 0.0
0015          SUBT(I,J,K) = 0.0
0016      500 CONTINUE
0017          DO 1500 II=1,10
0018          DO 1500 JJ=1,10
0019     1500 DSSCP(II,JJ) = 0.D0
0020          IF(MULT)511,510,511
        C
        C     UNIVARIATE PROCEDURE
0021      511 CONTINUE
0022          DO 600 I=1,NSUBJ
0023          READ (NW1) SETUP
0024          MI = ISET(NI)
0025          MJ = ISET(NJ)
0026          DO 501 K = 1,NVBL
0027          CK=SETUP(K+2)
0028          IF(CK-CUTL)503,501,503
0029      503 NN(MI,MJ,K) = NN(MI,MJ,K) + 1
0030          SUBT(MI,MJ,K) = SUBT(MI,MJ,K) + SETUP(K+2)
0031          DSS(MI,MJ,K) = DSS(MI,MJ,K) + SETUP(K+2)*SETUP(K+2)
0032          DSQU(K)=DSQU(K)+(DBLE(SETUP(K+2)))**2
0033      501 CONTINUE
0034      600 CONTINUE
0035          GO TO 529
        C
        C     MULTIVARIATE PROCEDURE
0036      510 CONTINUE
0037          NVBL1=NVBL-1
0038          NLUV=0
0039          DO 191 J=1,NVBL
```

```
0040        191 IF(LUV(J).NE.0) NLUV=NLUV+1
0041            IF(LUV(NVBL).EQ.0) LUV(NVBL)=99
0042            ITEMP=NVBL
0043            NVBL=NLUV
0044            NLUV=ITEMP
0045            DO 700 I=1,NSUBJ
0046            READ (NW1) SETUP
0047            IF(SETUP(13)-1.)506,700,700
0048        506 CONTINUE
0049            DO 300 J=1,NVBL1
0050            IF(LUV(J).NE.0) GO TO 300
0051        302 DO 301 K=J,NVBL1
0052            LUV(K)=LUV(K+1)
0053        301 SETUP(K+2)=SETUP(K+3)
0054            IF(LUV(J).EQ.0) GO TO 302
0055        300 CONTINUE
0056            MI = ISET(NI)
0057            MJ = ISET(NJ)
0058            DO 505 K = 1,NVBL
0059            NN(MI,MJ,K) = NN(MI,MJ,K) + 1
0060            SUBT(MI,MJ,K) = SUBT(MI,MJ,K) + SETUP(K+2)
0061            DSS(MI,MJ,K) = DSS(MI,MJ,K) + SETUP(K+2)*SETUP(K+2)
0062            DO 505 LM=1,NVBL
0063            A=DBLE(SETUP(K+2))
0064            B=DBLE(SETUP(LM+2))
0065            C=DSSCP(K,LM)
0066            C=C+A*B
0067            DSSCP(K,LM)=C
0068        505 CONTINUE
0069            DO 650 IAM=1,10
0070        650 LUV(IAM)=MALLUV(IAM)
0071        700 CONTINUE
0072        504 CONTINUE
0073        512 CONTINUE
0074            DO 520 K=1,NVBL
0075            DSQU(K)=DSSCP(K,K)
0076        520 CONTINUE
0077        529 REWIND NW1
0078            REWIND NW2
0079            WRITE(NW2) DSSCP
0080            WRITE(NW2) NN
0081            WRITE(NW2) DSS
0082            WRITE(NW2) SUBT
0083            WRITE(NW2) DSQU
0084            RETURN
0085         99 WRITE(6,98)NSUBJ
0086         98 FORMAT('   NSUBJ = ', I10)
0087            STOP
0088            END
```

```
0001          [illegible]
0002          [illegible] WNAK(10),FNAM(2),NLVL(2),NEND(2,12),
              [illegible] L(12,12),[illegible] N1,N2,NVBL,NFACT,NG12,KULT,
              [illegible] KOUNT,NDIAG,IREC1,IREC2,IREC3
              [illegible] NBLOV(10)
0003          [illegible]
0004          [illegible] DSCB(10),[illegible] G),RB(12),CF(12),Q(12),CS(12),
              [illegible] SS(12,12),
              [illegible]
0005          [illegible] NN(1,1,1))
0006          [illegible]
0007          [illegible]
0008          [illegible]
0009          [illegible]
0010          [illegible]
0011          [illegible] = NIV
0012          [illegible] = NIZ
0013          [illegible] NLVL(NI)
0014          [illegible] NLVL(N2)
0015          [illegible]
0016          [illegible] GO TO 604
0017          [illegible]
0018          [illegible]
0019          [illegible]
0020          [illegible]
0021          [illegible]
0022          [illegible]
0023          [illegible]
0024          [illegible]
0025          [illegible]
0026          [illegible]
0027          [illegible]
0028          [illegible]
0029          [illegible]
0030          [illegible]
0031          [illegible]
0032          [illegible]
0033          [illegible]
0034          [illegible]
0035          [illegible]
0036          [illegible]
0037          [illegible]
0038          [illegible]
0039          [illegible]
0040          [illegible]
0041          [illegible]
0042          [illegible]
0043          [illegible]
0044          [illegible]
0045          [illegible]
0046          [illegible]
0047          [illegible]
0048          [illegible]
0049          [illegible]
0050          [illegible]
0051          [illegible]
```

Reproduced from best available copy.

```
0052          READ(NW2) DSQU
0053          DSQV = DSQU(K)
0054          CALL ANOT(NN1,SUB,K,1 ,2 ,DSQV,SS,RE,CE,Q,QB)
0055      120 CONTINUE
0056          DUMB=0.
0057          KM=K-1
0058          IF(K.GT.1) GO TO 146
0059      148 WRITE(NW2) NI,NJ,K,SUB,RE,CE,Q,QB
0060          GO TO 907
0061      146 DO 145 IX=1,KM
0062      145 READ(NW2) DUMB,DUMB,DUMB,SS,DUM,DUM,DUM,DUM
0063          GO TO 148
0064      907  CONTINUE
0065          IF(MULT.EQ.0)GO TO 350
0066     1051 KKOUN=KOUNT-2
0067          WRITE(N1'IREC1,2050)KKOUN
0068          WRITE(N1'IREC1,2003)
0069          WRITE(N1'IREC1,2051)
0070          WRITE(N1'IREC1,2052)
0071          WRITE(N1'IREC1,2053)
0072          WRITE(N1'IREC1,2054)
0073          DO 309 I=1,43
0074      309 WRITE(N2'IREC2,2003)
0075          DO 310 I=1,57
0076      310 WRITE(N1'IREC1,2003)
0077      350 WRITE(NW2)NN1
0078          IF(MULT) 136,130,136
0079      136 ISEQ = 17
0080          GO TO 138
0081      130 ISEQ = 396
0082      138 RETURN
0083     2001 FORMAT(55X,'PAGE',I4,'L',16X)
0084     2003 FORMAT(80X)
0085     2041 FORMAT('AT THIS POINT YOU SHOULD INSPECT YOUR UNIVARIATE RESULTS V
            1ERY',19X)
0086     2042 FORMAT('CAREFULLY.',70X)
0087     2043 FORMAT('IS THERE A LARGE INTERACTION VERSUS ERROR F-RATIO?',30X)
0088     2044 FORMAT('THIS MAY INDICATE THE PRESENCE OF OUTLIERS OR FAULTY DATA.
            1',22X)
0089     2045 FORMAT('CHECK THE TABLE OF CELL TOTALS OF VARIABLES WITH LARGE F-R
            1ATIOS',17X)
0090     2046 FORMAT('FOR LARGE STANDARD DEVIATIONS OR FOR A CELL MEAN WHICH DEV
            1IATES',17X)
0091     2047 FORMAT('IRREGULARLY FROM A TREND IN ROWS OR COLUMNS.',36X)
0092     2048 FORMAT('AFTER YOU HAVE CHECKED THESE RESULTS YOU CAN CHECK AND EDI
            1T',21X)
0093     2049 FORMAT('YOUR DATA.',70X)
0094     2050 FORMAT('PAGE',I3,' WAS THE LAST PAGE OF YOUR UNIVARIATE RESULTS.',
            127X)
0095     2051 FORMAT('YOU NOW HAVE THE FOLLOWING OPTIONS:',45X)
0096     2052 FORMAT('PRESS  KEY 4 TO CONTINUE TO YOUR MULTIVARIATE RESULTS',
            126X)
0097     2053 FORMAT(12X,'5 TO SEE AND EDIT YOUR DATA',41X)
0098     2054 FORMAT(12X,'6 TO SEE PLOTS OF YOUR DATA',41X)
0099          END
```

```
0001          SUBROUTINE ANOT (NN1,SUB,IV,IA,IB,DSQU,SQR,RE,CE,Q,QB)
        C
        C     UNIVARIATE ANALYSIS OF IRREGULAR TWO-WAY FACTORIAL
        C
0002          COMMON N,ICVLY,ITYPE,TTL(15),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),
             1LEVCL(2,12),NRA,NCA,NRB,NCB,JAB(8),N1,N2,NVBL,NFACT,NG12,MULT,
             2NSUBJ,NLUV ,NW1,NW2,NT,ISEC,NIZ,NJZ,KOUNT,NDIAG,IREC1,IREC2,IREC3
             3,NTRAN(10),TNAM(10),LUV(10),NNLUV(10)
0003          DOUBLE PRECISION G,GN,SSTO,S,FN,DSQU,SSE,SSSB,SQ,DR,DC
0004          DIMENSION DR(12),DC(12)
0005          DIMENSION  SUB(12,12),R(12),C(12),Q(12),SUBX(12,12),CX(12,12),
             1    RE(12),CE(12),SQR(12,12),SQRX(12,12),QB(12),
             2NN1(12,12),NTCOT(12),NDOTJ(12),NX(12,12),JLVA(12),JLVB(12)
0006          DIMENSION TEXT2(60)
0007          DATA BLANK/'  '/
0008          NR=NLVL(IA)                                                    011
0009          NC=NLVL(IB)                                                    012
0010          NFA=IA                                                         013
0011          NFB=IB                                                         014
0012          IFLAG = 0
0013          ILINK = 12
0014          ITG=0                                                          015
0015          IF(NR-NC)11,11,1                                               016
0016      1   DO 2 I = 1,NC
0017          DO 2 J =1,NR
0018          NX(I,J)=NN1(J,I)
0019          SQRX(I,J)=SQR(J,I)                                             020
0020      2   SUBX(I,J)=SUB(J,I)                                             021
0021          ITMP=NR                                                        022
0022          NR=NC                                                          023
0023          NC=ITMP                                                        024
0024          NFA=IB                                                         025
0025          NFB=IA                                                         026
0026          IFLAG = 1
0027          GO TO 13                                                       027
0028     11   DO 12 I=1,NR                                                   028
0029          DO 12 J =1,NC                                                  029
0030          NX(I,J)=NN1(I,J)
0031          SQRX(I,J)=SQR(I,J)                                             031
0032     12   SUBX(I,J)=SUB(I,J)                                             032
0033     13   DO 24 I=1,NR
0034     24   JLVA(I)=NEND(NFA,I)                                            034
0035          I=1                                                            035
0036     14   NSUM=0                                                         036
0037          DO 16 J=1,NC                                                   037
0038     16   NSUM=NSUM+NX(I,J)                                              038
0039          IF(NSUM)17,17,18                                               039
0040     17   NR=NR-1                                                        040
0041          IF(I-NR)21,22,22                                               041
0042     21   DO 19 K=I,NR                                                   042
0043          DO 20 L = 1,NC                                                 043
0044          SQRX(K,L)=SQRX(K+1,L)                                          044
0045          SUBX(K,L)=SUBX(K+1,L)                                          045
0046     20   NX(K,L)=NX(K+1,L)                                              046
0047     19   JLVA(K)=JLVA(K+1)                                              047
0048          GO TO 23                                                       048
```

```
0049          10 I = I+1
0050          23 IF(I-NR)14,14,22
0051          22 DO 44 J=1,NC
0052          44 JLVB(J)=NEPO(NPO,J)
0053             J=1
0054          34 NSUM=0
0055             DO 36 I=1,NR
0056          36 NSUM=NSUM+NX(I,J)
0057             IF(NSUM)37,37,38
0058          37 NC=NC-1
0059             IF(J-NC)41,42,42
0060          41 DO39 K=J,NC
0061             DO 40 L=1,NR
0062             SUBX(L,K)=SUBX(L,K+1)
0063             SQRX(L,K)=SQRX(L,K+1)
0064          40 NX(L,K)=NX(L,K+1)
0065          39 JLVB(K)=JLVB(K+1)
0066             GO TO 43
0067          38 J=J+1
0068          43 IF(J-NC)34,34,42
0069          42 DO 50 I=1,NR
0070             NIDOT(I)=0
0071             R(I)=0.
0072             DR(I)=0.D0
0073             DO 50 J=1,NC
0074             NIDOT(I)=NIDOT(I)+NX(I,J)
0075             DR(I)=DR(I)+DBLE(SUBX(I,J))
0076          50 R(I)=R(I)+SUBX(I,J)
0077             DO 51 J=1,NC
0078             NDOTJ(J)=0
0079             C(J)=0.
0080             DC(J)=0.D0
0081             DO 51 I=1,NR
0082             NDOTJ(J)=NDOTJ(J)+NX(I,J)
0083             DC(J)=DC(J)+DBLE(SUBX(I,J))
0084          51 C(J)=C(J)+SUBX(I,J)
0085             G=0.D0
0086             GM = 0.D0
0087             SSTO = 0.D0
0088             NN=0
0089             DO 52 I=1,NR
0090             S = DR(I)
0091             G=S+G
0092          52 NN=NN+NIDOT(I)
0093             FN=0.D00
0094             FN=NN
0095             GM=G/FN
0096             SSTO=DSQU-G*GM
0097             NDFTO=NN-1
0098             NDFSB=-1
0099             SSSB=0.D00
0100             DO 55 I=1,NR
0101             DO 55 J=1,NC
0102             J=J
0103             IF(NX(I,J))55,55,57
0104          57 S=DBLE(SUBX(I,J))
```

```
0105          SSSB=SSSB+SNS/DFLOAT(NX(I,J))
0106          NDFSB = NDFSB + 1                                          100
0107     55   CONTINUE                                                   101
0108          SSSB=SSSB-G*GM
0109          SSE = 0.000
0110          SSE=SSTO-SSSB
0111          IF(SSE) 155,155,156                                        107
0112    155   SSE = 0.000
0113    156   NDFE = NDFTO - NDFSB                                       110
0114          DO 60 I=1,NR                                               132
0115          AI = FLOAT (NIDOT(I))
0116          Q(I) = DR(I)
0117          DO 62 K=I,NR                                               135
0118          SUM = 0.                                                   136
0119          DO 61 J=1,NC                                               137
0120          AKJ = FLOAT (NX(K,J))
0121          AIJ = FLOAT (NX(I,J))
0122     61   SUM = SUM - AIJ*AKJ/FLOAT (NDOTJ(J))
0123          CX(I,K) = SUM
0124          IF(I-K) 63,64,63                                           142
0125     64   CX(I,K) = CX(I,K) + AI                                     143
0126     63   CX(K,I) = CX(I,K)                                          144
0127     62   CONTINUE                                                   145
0128          DO 65 JU=1,NC                                              146
0129     65   Q(I) = DBLE(Q(I))-DFLOAT(NX(I,JU))*DC(JU)/DFLOAT(NDOTJ(JU))
0130     60   CONTINUE                                                   148
0131          DO 288 J=1,NC
0132          QB(J)=DC(J)
0133          DO 289 I=1,NR
0134    289   QB(J)= DBLE(QB(J))-DFLOAT(NX(I,J))*DR(I)/DFLOAT(NIDOT(I))
0135    288   CONTINUE
0136          NRM = NR - 1                                               158
0137          RE(NR) = 0.                                                159
0138          IF(NRM) 100,100,70                                         160
0139     70   DO 71 I=1,NRM                                              161
0140          Q(I) = Q(I) - Q(NR)                                        162
0141          DO 72 K=1,NRM                                              163
0142     72   CX(I,K) = CX(I,K) - CX(I,NR) - CX(NR,K) + CX(NR,NR)        164
0143     71   CONTINUE                                                   165
0144          CALL EQSYM(CX,Q,RE,NRM)                                    172
0145          DO 114 I =1,NRM
0146    114   Q(I) = Q(I) + Q(NR)
0147          DO 79 I=1,NRM
0148     79   RE(NR) = RE(NR) - RE(I)                                    174
0149    100   DO 82 J=1,NC                                               175
0150          CE(J)=DC(J)
0151          DO 83 I=1,NR                                               177
0152     83   CE(J)=CE(J)-RE(I)*FLOAT (NX(I,J))                          178
0153     82   CE(J) = CE(J)/FLOAT (NDOTJ(J)) - GM                        179
0154    216   CONTINUE
0155          NSTEP=1
0156          IREC=(IREC1-1
0157          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0158   1002   WRITE(N1'IREC1,2004) (YT(I),I=1,8)
0159          WRITE(N2'IREC2,2005) (YT(I),I=9,15)
0160          IF(MULT.EQ.1)WRITE(N1'IREC1,2006)VNAM(IV)
```

```
0161          IF(NBLF.EQ.0)WRITE(N1'IREC1,2006)TNAM(IV)
0162          WRITE(N2'IREC2,2007)FNAM(NFA),FNAM(NFB)
0163          WRITE(N1'IREC1,2008)FNAM(NFB)
0164          WRITE(N2'IREC2,2003)
0165          WRITE(N1'IREC1,2009)FNAM(NFA)
0166          WRITE(N2'IREC2,2010)
0167          WRITE(N1'IREC1,2011)
0168          WRITE(N2'IREC2,2003)
0169          IF(NC.LE.6) GO TO 6001
0170          IF(NC.GT.6) GO TO 6002
0171     6001 WRITE(N1'IREC1,2012)(JLVB(IW),IW=1,NC)
0172          WRITE(N2'IREC2,2003)
0173          GO TO 6003
0174     6002 WRITE (N1'IREC1,2012)(JLVB(IW),IW=1,6)
0175          WRITE(N2'IREC2,2013)(JLVB(IW),IW=7,NC)
0176     6003 CONTINUE
0177     96   DO 92 I=1,KK
0178          R(I) = R(I)/(FLOAT (NIDOT(I))                                    198
0179          DO 93 J=1,NC                                                     199
0180          GN = NX(I,J)                                                     200
0181          IF(GN-1.) 201,202,203                                            201
0182     203  SUBX(I,J) = SUBX(I,J)/GN                                         202
0183          SQRX(I,J)=SQRT(ABS((SQRX(I,J)-GN*SUBX(I,J)**2)/(GN-1.)))
0184          GO TO 93                                                         204
0185     202  SQRX(I,J) = 0.                                                   205
0186          GO TO 93                                                         206
0187     201  SUBX(I,J)=0.                                                     207
0188          SQRX(I,J)=0.                                                     208
0189     93   CONTINUE                                                         209
0190     92   CONTINUE                                                         210
0191          DO 266 J=1,NC                                                    211
0192     266  C(J) = C(J)/FLOAT (NDOTJ(J))                                     212
0193     97   CONTINUE
0194          DO 207 I=1,KK                                                    214
0195          NG = NIDOT(I)                                                    215
0196          RG = R(I)                                                        216
0197     209  JLVX = JLVA(I) - (JLVA(I)/100)
0198          NSTEP=2
0199          IREC=IREC1-1
0200          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0201     1003 WRITE(N1'IREC1,2003)
0202          WRITE(N2'IREC2,2003)
0203          NSTEP=3
0204          IREC=IREC1-1
0205          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0206          NC1=NC+1
0207          SUBX(I,NC1)=RG
0208     1004 IF(NC1.LE.6)GO TO 6004
0209          IF(NC1.GT.6)GO TO 6005
0210     6004 WRITE(N1'IREC1,2014)JLVX,(SUBX(I,J),J=1,NC1)
0211          WRITE(N2'IREC2,2003)
0212          GO TO 6006
0213     6005 WRITE(N1'IREC1,2014)JLVX,(SUBX(I,J),J=1,6)
0214          WRITE(N2'IREC2,2015)(SUBX(I,J),J=7,NC1)
0215     6006 CONTINUE
0216          NSTEP=4
```

Reproduced from best available copy.

```
0217          IREC=IREC1-1
0218          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0219          NX(1,NC1)=NG
0220     1005 IF(NC1.LE.6)GO TO 6007
0221          IF(NC1.GT.6) GO TO 6008
0222     6007 WRITE(N1'IREC1,2016)(NX(I,J),J=1,NC1)
0223          WRITE(N2'IREC2,2003)
0224          GO TO 6009
0225     6008 WRITE(N1'IREC1,2016)(NX(I,J),J=1,6)
0226          WRITE(N2'IREC2,2017)(NX(I,J),J=7,NC1)
0227     6009 CONTINUE
0228          NSTEP=5
0229          IREC=IREC1-1
0230          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0231     1006 IF(NC.LE.6)GO TO 6010
0232          IF(NC.GT.6)GO TO 6011
0233     6010 WRITE(N1'IREC1,2018)(SQRX(I,J),J=1,NC)
0234          WRITE(N2'IREC2,2003)
0235          GO TO 6012
0236     6011 WRITE(N1'IREC1,2018)(SQRX(I,J),J=1,6)
0237          WRITE(N2'IREC2,2019)(SQRX(I,J),J=7,NC)
0238     6012 CONTINUE
0239     207  CONTINUE                                                    226
0240          NG = NN                                                    227
0241          RG = GN                                                    228
0242          NSTEP=6
0243          IREC=IREC1-1
0244          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0245     1007 WRITE(N1'IREC1,2003)
0246          WRITE(N2'IREC2,2003)
0247          NSTEP=7
0248          IREC=IREC1-1
0249          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0250     1008 WRITE(N1'IREC1,2020)
0251          WRITE(N2'IREC2,2003)
0252          NSTEP=8
0253          IREC=IREC1-1
0254          C(NC1)=RG
0255          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0256     1009 IF(NC1.LE.6)GO TO 6013
0257          IF(NC1.GT.6)GO TO 6014
0258     6013 WRITE(N1'IREC1,2018)(C(J),J=1,NC1)
0259          WRITE(N2'IREC2,2003)
0260          GO TO 6015
0261     6014 WRITE(N1'IREC1,2018)(C(J),J=1,6)
0262          WRITE(N2'IREC2,2015)(C(J),J=7,NC1)
0263     6015 CONTINUE
0264          NSTEP=9
0265          NDOTJ(NC1)=NG
0266          IREC=IREC1-1
0267          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0268     1010 IF(NC1.LE.6)GO TO 6016
0269          IF(NC1.GT.6) GO TO 6017
0270     6016 WRITE(N1'IREC1,2016)(NDOTJ(J),J=1,NC1)
0271          WRITE(N2'IREC2,2003)
0272          GO TO 6018
```

```
0273        6017 WRITE(N1'IREC1,2016)(NCOTJ(J),J=1,6)
0274             WRITE(N2'IREC2,2017)(NCOTJ(J),J=7,NC1)
0275        6016 CONTINUE
           C                                                                239
           C     BEGIN ANALYSIS OF VARIANCE                                 240
0276         215 SSRAD = 0.                                                 241
0277             DO 221 I=1,NR                                              242
0278         221 SSRAD=SSRAD+RE(I)*O(I)                                     243
0279             VSTO = SSTO/FLOAT (NDFTO)
0280             NDFRO = NR-1                                               245
0281             IF(NDFRO) 222,222,223                                      246
0282         222 VSRO = 0.                                                  247
0283             GO TO 224                                                  248
0284         223 VSRO = SSRAD/FLOAT (NDFRO)                                 249
0285         224 S = 0.D0
0286             DO 225 J=1,NC                                              251
0287         225 S = S + CC(J)**2/DFLOAT(NCOTJ(J))
0288             S = S-G*GN
0289             NDFINT = NDFSB - NDFRO - NC + 1                            254
0290             IF(NDFINT) 226,226,227                                     255
0291         226 NDFINT = 0                                                 256
0292             VSINT = 0.                                                 257
0293             GO TO 228                                                  258
0294         227 SSINT = SSSB - DBLE(SSRAD) - S
0295             VSINT = SSINT/FLOAT (NDFINT)                               260
0296         228 SD = 0.D0
0297             DO 229 I=1,NR                                              262
0298         229 SD = SD + CR(I)**2/DFLOAT(NIROT(I))
0299             SD = SD - G*GN
0300             SSCAD = S-SD+DBLE(SSRAD)
0301             NDFCOL = NC-1                                              266
0302             IF(NDFCOL) 230,230,231                                     267
0303         230 VSCOL = 0.                                                 268
0304             GO TO 232                                                  269
0305         231 VSCOL = SSCAD/FLOAT (NDFCOL)                               270
0306         232 VSSB = SSSB/FLOAT (NDFSB)                                  271
0307             IF(NDFE) 233,233,234                                       272
0308         233 FRO = 0.                                                   273
0309             VSE = 0.
0310             FCOL = 0.                                                  274
0311             FINT = 0.                                                  275
0312             FSUB = 0.                                                  276
0313             GO TO 235                                                  277
0314         234 VSE = SSE/FLOAT (NDFE)                                     278
0315             FRO = VSRO/VSE                                             279
0316             FCOL = VSCOL/VSE                                           280
0317             FINT = VSINT/VSE                                           281
0318             FSUB = VSSB/VSE                                            282
0319         235 IF(VSINT) 237,237,238                                      283
0320         237 FROI = 0.                                                  284
0321             FCOLI = 0.                                                 285
0322             GO TO 239                                                  286
0323         238 FROI = VSRO/VSINT                                          287
0324             FCOLI = VSCOL/VSINT                                        288
0325         239 GAM=0.
0326             DO 601 I=1,NR
```

Reproduced from best available copy.

```
0327        601  GAN=GAN+RE(I)*FLOAT(NIDOT(I))/FLOAT(NN)
0328             DO 241 I=1,NR
0329        241  P(I) = RE(I) + SNGL(GM) - GAN
0330             IF(NRM) 244,244,245                                    291
0331        245  DO 240 I=1,NRR                                         292
0332             IP = I+1                                               293
0333             DO 240 J=IP,NR                                         294
0334             IF(R(I)-R(J)) 242,240,240                              295
0335        242  TEM = R(I)                                             296
0336             R(I) = R(J)                                            297
0337             R(J) = TEM                                             298
0338             ITEM = JLVA(I)                                         299
0339             JLVA(I) = JLVA(J)                                      300
0340             JLVA(J) = ITEM                                         301
0341        240  CONTINUE                                              302
0342        244  NCM = NC-1                                            303
0343             DO 249 J=1,NC                                          304
0344         249 C(J)=CE(J)+GAN+SNGL(GM)
0345             IF(NCM) 246,246,247                                    306
0346        247  DO 250 I=1,NCM                                         307
0347             IP = I+1
0348             DO 250 J=IP,NC                                         309
0349             IF(C(I)-C(J)) 252,250,250                              310
0350        252  TEM = C(I)                                             311
0351             C(I) = C(J)                                            312
0352             C(J) = TEM                                             313
0353             ITEM = JLVB(I)                                         314
0354             JLVB(I) = JLVB(J)                                      315
0355             JLVB(J) = ITEM                                         316
0356        250  CONTINUE                                              317
0357         246 CONTINUE
0358             NSTEP=11
0359             IREC=IREC1 1
0360             IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0361        1012 WRITE(N1'IREC1,2003)
0362             WRITE(N2'IREC2,2003)
0363             NSTEP=10
0364             IREC=IREC1-1
0365             IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0366        1011 WRITE(N1'IREC1,2023)FNAM(GEA)
0367             WRITE(N2'IREC2,2003)
0368             NSTEP=12
0369             IREC=IREC1-1
0370             IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0371        1013 IF(NR.LE.6)GO TO 6019
0372             IF(NR.GT.6)GO TO 6020
0373        6019 WRITE(N1'IREC1,2016)(JLVA(I),I=1,NR)
0374             WRITE(N2'IREC2,2003)
0375             GO TO 6021
0376        6020 WRITE(N1'IREC1,2015)(JLVA(I),I=1,6)
0377             WRITE(N2'IREC2,2017)(JLVA(I),I=7,NR)
0378        6021 CONTINUE
0379             NSTEP=13
0380             IREC=IREC1-1
0381             IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0382        1014 IF (NR.LE.6)GO TO 6022
```

Reproduced from best available copy.

```
0383            IF(NR.GT.6)GO TO 6023
0384      6022  WRITE(N1'IREC1,2018)(R(I),I=1,NR)
0385            WRITE(N2'IREC2,2003)
0386            GO TO 6024
0387      6023  WRITE(N1'IREC1,2018)(R(I),I=5,6)
0388            WRITE(N2'IREC2,2019)(R(I),I=7,NR)
0389      6024  CONTINUE
0390            NSTEP=33
0391            IREC=IREC1-1
0392            IF(MOD(IREC,45).EQ.0) GO TO 1001
0393      1060  WRITE(N1'IREC1,2003)
0394            WRITE(N2'IREC2,2003)
0395            NSTEP=14
0396            IREC=IREC1-1
0397            IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0398      1015  WRITE(N1'IREC1,2023)FNAM(NF8)
0399            WRITE(N2'IREC2,2003)
0400            NSTEP=15
0401            IREC=IREC1-1
0402            IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0403      1016  IF(NC.LE.6)GO TO 6025
0404            IF(NC.GT.6)GO TO 6026
0405      6025  WRITE(N1'IREC1,2016)(JLVB(I),I=1,NC)
0406            WRITE(N2'IREC2,2003)
0407            GO TO 6027
0408      6026  WRITE(N1'IREC1,2016)(JLVB(I),I=1,6)
0409            WRITE(N2'IREC2,2017)(JLVB(I),I=7,NC)
0410      6027  CONTINUE
0411            NSTEP=16
0412            IREC=IREC1-1
0413            IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0414      1017  IF(NC.LE.6) GO TO 6028
0415            IF(NC.GT.6) GO TO 6029
0416      6028  WRITE(N1'IREC1,2018)(C(I),I=1,NC)
0417            WRITE(N2'IREC2,2003)
0418            GO TO 6030
0419      6029  WRITE(N1'IREC1,2018)(C(I),I=1,6)
0420            WRITE(N2'IREC2,2019)(C(I),I=7,NC)
0421      6030  CONTINUE
0422      260   CONTINUE                                                    350
0423            NSTEP=17
0424            IREC=IREC1-1
0425            IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0426      1018  WRITE(N1'IREC1,2003)
0427            WRITE(N2'IREC2,2003)
0428            NSTEP=18
0429            IREC=IREC1-1
0430            IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0431      1019  IF(MULT.EQ.1)WRITE(N1'IREC1,2024)VNAM(IV)
0432            IF(MULT.EQ.0)WRITE(N1'IREC1,2024)TNAM(IV)
0433            WRITE(N2'IREC2,2003)
0434            NSTEP=19
0435            IREC=IREC1-1
0436            IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0437      1020  WRITE(N1'IREC1,2002)
0438            WRITE(N2'IREC2,2003)
```

Reproduced from best available copy.

```
0439          NSTEP=20
0440          IREC=IREC1-1
0441          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0442     1021 WRITE(N1'IREC1,2025)
0443          WRITE(N2'IREC2,2026)
0444          NSTEP=21
0445          IREC=IREC1-1
0446          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0447     1022 WRITE(N1'IREC1,2003)
0448          WRITE(N2'IREC2,2003)
0449          NSTEP=22
0450          IREC=IREC1-1
0451          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0452     1023 WRITE(N1'IREC1,2027)FNAM(NFA),NDFRO,SSRAD
0453          WRITE(N2'IREC2,2028)VSRO,FRO,FROI
0454          NSTEP=23
0455          IREC=IREC1-1
0456          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0457     1024 WRITE(N1'IREC1,2029)FNAM(NFB),NDFCOL,SSCAD
0458          WRITE(N2'IREC2,2028)VSCOL,FCOL,FCOLI
0459          NSTEP=24
0460          IREC=IREC1-1
0461          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0462     1025 WRITE(N1'IREC1,2030)FNAM(NFA),FNAM(NFB),NDFINT,SSINT
0463          WRITE(N2'IREC2,2031)VSINT,FINT
0464          NSTEP=25
0465          IREC=IREC1-1
0466          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0467     1026 WRITE(N1'IREC1,2003)
0468          WRITE(N2'IREC2,2003)
0469          NSTEP=26
0470          IREC=IREC1-1
0471          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0472     1027 WRITE(N1'IREC1,2032)NDFSB,SSSB
0473          WRITE(N2'IREC2,2031)VSSB,FSUB
0474          NSTEP=27
0475          IREC=IREC1-1
0476          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0477     1028 WRITE(N1'IREC1,2033)NDFE,SSE
0478          WRITE(N2'IREC2,2034) VSE
0479          NSTEP=28
0480          IREC=IREC1-1
0481          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0482     1029 WRITE(N1'IREC1,2035)NDFTU,SSTU
0483          WRITE(N2'IREC2,2036)VSTU
0484          IF(NDFE) 270,270,271                                    350
0485      271 RMSE = SQRT(VSE)                                        351
0486          NSTEP=29
0487          IREC=IREC1-1
0488          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0489     1030 WRITE(N1'IREC1,2003)
0490          WRITE(N2'IREC2,2003)
0491          NSTEP=30
0492          IREC=IREC1-1
0493          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0494     1031 WRITE(N1'IREC1,2003)
```

Reproduced from best available copy.

```
0495          WRITE(N2'IREC2,2036)RMSE
0496      270 IF(NDFINT)2730,2730,274
0497      274 RMSIN = SQRT(VS(N1))                                          355
0498          NSTEP=31
0499          IREC=IREC1-1
0500          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0501     1032 WRITE(N1'IREC1,2003)
0502          WRITE(N2'IREC2,2037)RMSIN
0503     2730 NSTEP=32
0504     1049 IREC=IREC1-1
0505          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0506     1050 WRITE(N1'IREC1,2003)
0507          WRITE(N2'IREC2,2003)
0508          GO TO 1049
0509      273 IF((FLAG) 293,293,294
0510      294 DO 295 I=1,ILINK
0511          R(I) = RE(I)
0512          C(I) = Q(I)
0513          RE(I) = CE(I)
0514          Q(I) = QB(I)
0515          CE(I) = R(I)
0516      295 QB(I) = C(I)
0517      293 CALL INTER(IV)
0518          RETURN
0519     2001 FORMAT(55X,'PAGE',I4,'L',16X/)
0520     2002 FORMAT(55X,'PAGE',I4,'R',16X/)
0521     2003 FORMAT(80X)
0522     2004 FORMAT(33X,8A4,15X/)
0523     2005 FORMAT(7A4,52X/)
0524     2006 FORMAT(35X,'VARIABLE ( ',A4,' ) TABLE OF MEANS',15X/)
0525     2007 FORMAT('AND EFFECTS ( ',A4,' ) VERSUS ( ',A4,' )',44X/)
0526     2008 FORMAT(' ROWS =',34X,'C O L U M N S = ',A6,17X)
0527     2009 FORMAT(A6,8X,'IN EACH BLOCK, ROW 1 DENOTES CELL MEANS, ROW 2 DENOT
             1ES',12X)
0528     2010 FORMAT(' CELL SIZE, AND ROW 3 DENOTES STANDARD DEVIATION.',31X)
0529     2011 FORMAT(' CODED',34X,'C O D E D  L E V E L S',18X)
0530     2012 FORMAT(' LEV.',I3,5I10,22X)
0531     2013 FORMAT(6I13,20X)
0532     2014 FORMAT(I2,1P6E10.3,16X)
0533     2015 FORMAT(1P6E10.3,20X)
0534     2016 FORMAT(5X,I3,5I10,22X)
0535     2017 FORMAT(7I10,10X)
0536     2018 FORMAT(2X,1P6E10.3,17X)
0537     2019 FORMAT(1P6E10.3,20X)
0538     2020 FORMAT(' COL.',7I5X)
0539     2023 FORMAT(35X,'ESTIMATES OF ',A4,'  ADJUSTED MEANS, ORDERED',23X)
0540     2024 FORMAT(20X,'ANALYSIS OF VARIANCE FOR VARIABLE ',A4,22X)
0541     2025 FORMAT(5X,'SOURCE OF VARIATION',14X,'D.F.',7X,'SUM OF SQUARES',17X
             1)
0542     2026 FORMAT(5X,'MEAN SQUARE',8X,'F VS. ERROR',3X,'F VS. INTERACTION',
             125X)
0543     2027 FORMAT(8X,A6,4X,'(ROWS)',I10,1PE20.6,18X)
0544     2028 FORMAT(1PE20.6,E17.3,26X)
0545     2029 FORMAT(8X,A6,4X,'(COLS)',I10,1PE20.6,18X)
0546     2030 FORMAT(5X,A6,'*',A6,'(INTERACTION)',I11,1PE20.6,18X)
0547     2031 FORMAT(1PE20.6,E17.3,43X)
```

```
0548      2032 FORMAT(10X,'SUBTOTALS',35X,I8,1PE20.6,18X)
0549      2033 FORMAT(12X,'ERROR',7X,I8,1PE20.6,18X)
0550      2034 FORMAT(1PE20.6,60X)
0551      2035 FORMAT(12X,'TOTAL',7X,I8,1PE20.6,18X)
0552      2036 FORMAT('ROOT MEAN SQUARE ERROR = ',1PE20.6,35X)
0553      2037 FORMAT(7X,'ROOT MEAN SQUARE INTERACTION = ',1PE16.6,26X)
0554      1001 WRITE(N1'IREC1,2001)KOUNT
0555           WRITE(N2'IREC2,2002)KOUNT
0556           KOUNT=KOUNT+1
0557           GO TO (1002,1003,1004,1005,1006,1007,1008,1009,1010,1011,1012,
              11013,1014,1015,1016,1017,1018,1019,1020,1021,1022,1023,1024,1025,
              21026,1027,1028,1029,1030,1031,1032,273,1060),NSTEP                359
0558           END
```

```
0001          SUBROUTINE EQSYM(A,Q,B,NR)
        C
        C     SOLVE A POSITIVE SEMIDEFINITE OR POSITIVE DEFINITE
        C     SYMMETRIC EQUATION
        C
0002          COMMON N,IOVLY,ITYPE,TTL(15),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),
             1LEVEL(2,12),NRA,NCA,NRB,NCB,JAB(8),N1,N2,NVBL,NFACT,NG12,MULT,
             2NSU3J,NLOV ,NW1,NW2,NT,ISEG,NI2,NJZ,KOUNT,NDIAG,IREC1,IREC2,IREC3
             3,NYRAN(10),TNAM(10),LUV(10),NNLOV(10)
0003          DIMENSION A(12,12),Q(12),B(12),AA(78),T(78),D(13)
0004          IJ = 0                                                          010
0005          DO 1 J=1,NR                                                     011
0006          DO 1 I=1,J                                                      012
0007          IJ = IJ+1                                                       013
0008      1   AA(IJ) = A(I,J)                                                 014
0009          NRA = NR                                                        015
0010          NRB = NR                                                        016
0011          NCA = NR                                                        017
0012          NCB = NR                                                        018
0013          CALL INSLD(AA,T,D)                                              019
0014          IX = 1                                                          020
0015          DO 2 I=1,NR                                                     021
0016          IX = IX+I-1                                                     022
0017          JX = 1                                                          023
0018          B(I) = 0.                                                       024
0019          DO 6 J=1,NR                                                     025
0020          IF(J-I) 3,3,4                                                   026
0021      3   JX = J                                                          027
0022          GO TO 5                                                         028
0023      4   JX = JX+J-1                                                     029
0024      5   IJ = IX + JX - 1                                                030
0025      6   B(I) = B(I) + T(IJ)*Q(J)                                        031
0026      2   CONTINUE                                                        032
0027          DO 10 I=1,NR                                                    033
0028          SUM = 0.                                                        034
0029          DO 12 J=1,NR                                                    035
0030     12   SUM = SUM + A(I,J)*B(J)                                         036
0031          IF(Q(I)) 13,14,13                                               037
0032     14   DEN = 1.                                                        038
0033          GO TO 15                                                        039
0034     13   DEN = ABS (Q(I))                                                040
0035     15   IF(ABS (SUM-Q(I))/DEN - 1.E-5) 10,10,16
0036     10   CONTINUE                                                        042
0037          GO TO 20                                                        043
0038     16   CONTINUE
0039     20   RETURN                                                          046
0040          END                                                             047
```

```
0001          SUBROUTINE INTER(IV)
0002          EXTERNAL PRINT
0003          COMMON N,YOVLY,TYPE,TTL(15),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),
             1LEVEL(2,12),NRA,NCA,NRP,NCB,JAC(5),N1,N2,NVBL,NFACT,NG12,MULT,
             2NSUBJ,NCOV ,NM1,NM2,NT,ISEC,NIZ,NJZ,KOUNT,NGIAG,IREC1,IREC2,IREC3
             3,NTRAN(10),TNAM(10),LOV(10),RNLOV(10)
0004          DIMENSION TEX(15)
0005          DATA CSIZE/'8'/
0006          CALL GRINIT(CSIZE)
0007          MASK=16106127 39
0008          CALL GCPFX(MASK,PRINT)
0009          CALL GERAS(100)
0010          CALL GRDPLY(' ',1,6400)
0011          CALL GRDPLY('THE ANALYSES ON THE FOLLOWING VARIABLES ARE NOW COMPL
             1ETE.',57,6400)
0012          IF(MULT.EQ.1)WRITE(4,25)(VNAM(IJ),IJ=1,IV)
0013          IF(MULT.EQ.0)WRITE(4,25)(TNAM(IJ),IJ=1,IV)
0014       25 FORMAT(1J(2X,A4))
0015          CALL PUTCH(TEXT,NCF,6400)
0016          CALL GRDPLY(TEXT,NCF,6400)
0017          CALL GRDPLY(' ',1,6400)
0018          CALL GRDPLY('PRESS KEY 1 TO CONTINUE WITH CALCULATIONS OR KEY 2 TO
             1 STOP',50,6400)
0019          CALL GRDPLY('CALCULATIONS AT THIS STAGE.',27,6400)
0020       10 CALL GWAIT
0021          IF(N.EQ.30.OR.N.EQ.1)GO TO 203
0022          IF(N.EQ.31)GO TO 210
0023          IF(N.EQ.2)GO TO 75
0024          GO TO 10
0025       75 NVBL=IV
0026      203 CALL GERAS(100)
0027          CALL GRRLSE
0028          RETURN
0029      210 CALL GERAS(100)
0030          CALL GRRLSE
0031          STOP
0032      400 GO TO 210
0033          END
```

Reproduced from best available copy.

```
0001          SUBROUTINE OUTPUT
0002          EXTERNAL PFINT
0003          COMMON N,IGVLY,ITYPE,TTLF(5),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),
             1LEVEL(2,12),NRA,NCA,NRB,NCB,JAB(8),N1,N2,NVBL,NFACT,NGLC,NULT,
             2NSUBJ,NLUV ,NM1,NM2,NT,ISEQ,NIZ,NJZ,KOUNT,NDIAG,IREC1,IREC2,IREC3
             3,NTRAN(10),TNAP(10),LUV(10),MNLUV(10)
0004          DIMENSION TEXT(18),TEXT2(20)
0005          EQUIVALENCE (TEXT(1),TEXT2(1))
0006          DATA CSIZE/'8'/
0007          CALL GRINIT(CSIZE)
0008          N=-1
0009          MASK1=2013265923
0010          MASK=2143929219
0011          NC=72
0012          IF(MULT.EQ.0)CALL GCPFK(MASK1,PFINT)
0013          IF(MULT.EQ.1) CALL GCPFK(MASK,PFINT)
0014          IREC1=1
0015          IREC2=1
0016        3 FORMAT(20A4)
0017          CALL GRDPLY(' ',1,&400)
0018          CALL GRDPLY(' ',1,&400)
0019          CALL GRDPLY('     KEYBOARD FUNCTIONS:',24,&400)
0020          CALL GRDPLY('       1     ALTERNATE PORTION OF CURRENT PAGE.',46,
             1&400)
0021          CALL GRDPLY('       2     INCREMENT PAGE NUMBER.',34,&400)
0022          CALL GRDPLY('       3     DECREMENT PAGE NUMBER.',34,&400)
0023          CALL GRDPLY('      30     RESTART PROGRAM.',28,&400)
0024          CALL GRDPLY('      31     TERMINATE PROGRAM.',30,&400)
0025          CALL GRDPLY(' ',1,&400)
0026          IF(MULT.EQ.1 ) CALL GRDPLY('PRESS KEY 1 TO PROCEED TO THE FIRST PA
             1GE OF YOUR UNIVARIATE RESULTS.',68,&400)
0027          IF(MULT.EQ.0 ) CALL GRDPLY('PRESS KEY 1 TO PROCEED TO THE FIRST PA
             1GE OF YOUR MULTIVARIATE RESULTS.',70,&400)
0028        5 CALL GWAIT
0029          IF(N.EQ.1) GO TO 15
0030          IF(N.EQ.30) GO TO 75
0031          IF(N.EQ.31) GO TO 50
0032          GO TO 5
0033       15 CALL GERAS(100)
0034          DO 16 I=1,45
0035          READ(N1'IREC1)TEXT2
0036       16 CALL GRDPLY(TEXT,NC,&400)
0037       17 CALL GWAIT
0038          IF(N.EQ.1) GO TO 30
0039          IF (N.EQ.2) GO TO 15
0040          IF(N.EQ.3) GO TO 32
0041          IF(N.EQ.4.OR.N.EQ.5.OR.N.EQ.6.OR.N.EQ.30)GO TO 75
0042          IF(N.EQ.31) GO TO 50
0043          GO TO 17
0044       25 CALL GERAS(100)
0045          DO 26 I=1,45
0046          READ(N2'IREC2)TEXT2
0047       26 CALL GRDPLY(TEXT,NC,&400)
0048       27 CALL GWAIT
0049          IF(N.EQ.1) GO TO 34
0050          IF(N.EQ.2) GO TO 25
```

```
0051          IF(N.EQ.3) GO TO 36
0052          IF(N.EQ.4.OR.N.EQ.5.OR.N.EQ.6.OR.N.EQ.30)GO TO 75
0053          IF(N.EQ.31) GO TO 50
0054          GO TO 27
0055       30 IREC2=IREC1-45
0056          GO TO 25
0057       32 IF(IREC1.EQ.46) GO TO 17
0058          IREC1=IREC1-90
0059          GO TO 15
0060       75 CALL GERAS(100)
0061          CALL GRRLSE
0062          RETURN
0063       34 IREC1=IREC2-45
0064          GO TO 15
0065       36 IF(IREC2.EQ.46) GO TO 27
0066          IREC2=IREC2-90
0067          GO TO 25
0068       50 CALL GERAS(100)
0069          CALL GRRLSE
0070          STOP
0071      400 CALL GERAS(100)
0072          WRITE(6,402)
0073      402 FORMAT(1H1,'ERROR')
0074          GO TO 50
0075          END
```

```
0001          SUBROUTINE PLOT
0002          EXTERNAL PFINT,ECBINT
0003          INTEGER XNVBL,YNVBL
0004          COMMON N,IOVLY,ITYPE,TTL(15),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),
             1LEVEL(2,12),NRA,NCA,NRB,NCB,JAB(6),N1,N2,NVBL,NFACT,NG12,MULT,
             2NSUBJ,NLUV ,NW1,NW2,NT,ISEQ,NIZ,NJZ,KOUNT,NDIAG,IREC1,IREC2,IREC3
             3,NTRAN(10),TNAM(10),LUV(10),MNLUV(10)
0005          DIMENSION X(100),Y(100),NCOUNT(100),NTEN(100),NHUND(100),
             1MIXCH(3),MAXCH(3),MIYCH(3),MAYCH(3),TEXT(15),ISET(13),SETUP(15)
0006          EQUIVALENCE (ISET(1),SETUP(1))
0007          DOUBLE PRECISION DNP
0008          DATA CSIZE/'B'/
0009       10 REWIND NW1
0010          IMPY=-16777216
0011          NR=NLVL(1)
0012          NCC=NLVL(2)
0013          MASK1=1140850695
0014          NSTEP=1
0015          NC=60.
0016          XSUM=0.0
0017          YSUM=0.0
0018          XSUM2=0.0
0019          YSUM2=0.0
0020          KOUNT=0
0021          CALL GRINIT(CSIZE)
0022          CALL GCECB(ECBINT)
0023          CALL GCPFK(MASK1,PFINT)
0024          CALL GERAS(100)
0025          CALL GRDPLY(' ',1,6400)
0026          CALL GRDPLY('AT THIS STAGE YOU MAY LOOK AT PLOTS OF YOUR DATA POIN
             1TS.',50,6400)
0027          CALL GRDPLY('YOU HAVE THE CHOICE OF SEEING A CELL, A ROW, A COLUMN
             1, OR',57,6400)
0028          CALL GRDPLY('ALL YOUR DATA.  TO INDICATE YOUR CHOICE OF POINTS, TY
             1PE IN',58,6400)
0029          CALL GRDPLY('THE LEVELS OF BOTH FACTORS.  IF YOU WISH TO SEE ALL L
             1EVELS',58,6400)
0030          CALL GRDPLY('OF A FACTOR TYPE "0" FOR THE LEVEL.',35,6400)
0031          CALL GRDPLY('EXAMPLE:  SUPPOSE FACTOR 1 IS DESIGNATED AS THE ROWS
             1IN THE',59,6400)
0032          CALL GRDPLY('          ANOVAR TABLE',22,6400)
0033          CALL GRDPLY('              2,2      INDICATES BOTH LEVELS AT 2.',43,
             16400)
0034          CALL GRDPLY('              0,4      INDICATES THE 4TH COLUMN.',46,
             16400)
0035          CALL GRDPLY('              3,0      INDICATES THE 3RD ROW.',43,6400)
0036          CALL GRDPLY('              0,0      INDICATES ALL POINTS.',42,6400)
0037          CALL GRDPLY(' ',1,6400)
0038      185 DO 25 I=1,2
0039          WRITE(4,26)FNAM(I)
0040       26 FORMAT('WHICH LEVEL DO YOU WISH TO SEE OF ',A4,'?')
0041          CALL FETCH(TEXT,NCF,6400)
0042          CALL GRDPLY(TEXT,NCF,6400)
0043          CALL GWAIT
0044          IF(ITYPE.NE.3)GO TO 402
0045          CALL GRDPLY(TEXT,NC)
```

```
0046          CALL XBLANK(TEXT,NC)
0047          INDEX=0
0048          CALL INX(TEXT,INDEX,NC,DNP,&401)
0049          IF(I.EQ.1)NFAC1=DNP
0050          IF(I.EQ.2)NFAC2=DNP
0051       25 CONTINUE
0052      175 IF(NFAC1.GT.NN.OR.NFAC2.GT.NCC)GO TO 180
0053          CALL GRDPLY(' ',1,&400)
0054      191 CALL GRDPLY('WHICH VARIABLE DO YOU WISH TO ASSIGN TO THE X-AXIS?',
             .151,&400)
0055          NSTEP=2
0056      190 CALL GWAIT
0057          IF(ITYPE.NE.3)GO TO 402
0058          CALL GRDPLY(TEXT,NC)
0059          CALL XBLANK(TEXT,NC)
0060          VNAM1=TEXT(1)
0061          CALL GRDPLY('WHICH VARIABLE DO YOU WISH TO ASSIGN TO THE Y-AXIS?',
             151,&400)
0062          NSTEP=3
0063      195 CALL GWAIT
0064          IF(ITYPE.NE.3)GO TO 402
0065          CALL GRDPLY(TEXT,NC)
0066          CALL XBLANK(TEXT,NC)
0067          VNAM2=TEXT(1)
0068          XNVBL=-1
0069          YNVBL=-1
0070          DO 50 I=1,NVBL
0071          IF(VNAM1.EQ.VNAM(I)) XNVBL=I
0072       50 IF(VNAM2.EQ.VNAM(I)) YNVBL=I
0073          IF(XNVBL.EQ.-1.OR.YNVBL.EQ.-1)GO TO 450
0074          CALL GRDPLY(' ',1,&400)
0075          CALL GRDPLY(' ',1,&400)
0076          CALL GRDPLY('IF, AT ANY TIME, YOU WISH TO RETURN TO YOUR PREVIOUS'
             1,52,&400)
0077          CALL GRDPLY('UNIVARIATE ANALYSIS OUTPUT, PRESS KEY 29.',41,&400)
0078          CALL GRDPLY(' ',1,&400)
0079          CALL GRDPLY('TO RETURN TO THE BEGINNING OF THIS SEGMENT TO SEE',49
             1,&400)
0080          CALL  GRDPLY('ADDITIONAL PLOTS, PRESS KEY 1.',30,&400)
0081          CALL GRDPLY(' ',1,&400)
0082          CALL GRDPLY('TO SEE YOUR DATA AGAIN, PRESS KEY 5.',36,&400)
0083          I=0
0084          DO 75 N=1,NSLOJ
0085          READ(NW1) SETUP
0086          IF(NFAC1.EQ.0.AND.NFAC2.EQ.0)GO TO 76
0087          IF(NFAC1.EQ.0.AND.ISET(2).EQ.NFAC2)GO TO 76
0088          IF(NFAC1.EQ.ISET(1).AND.NFAC2.EQ.0)GO TO 76
0089          IF(ISET(1).EQ.NFAC1.AND.ISET(2).EQ.NFAC2)GO TO 76
0090          GO TO 75
0091       76 IF(SETUP(2+XNVBL).EQ.-999..OR.SETUP(2+YNVBL).EQ.-999.)GO TO 75
0092          I=I+1
0093          NCOUNT(I)=N
0094          NTEN(I)=MOD(NCOUNT(I),10)
0095          NHUN=(NCOUNT(I)-NTEN(I))/10
0096          NHUND(I)=MOD(NHUN,10)
0097          X(I)=SETUP(2+XNVBL)
```

```
0098          Y(I)=SETUP(2+YNVBL)
0099          XSUM=XSUM+X(I)
0100          YSUM=YSUM+Y(I)
0101          XSUM2=XSUM2+X(I)*X(I)
0102          YSUM2=YSUM2+Y(I)*Y(I)
0103          KOUNT=KOUNT+1
0104       75 CONTINUE
0105          IF(KOUNT.LE.1)GO TO 425
0106          XMEAN=XSUM/FLOAT(KOUNT)
0107          YMEAN=YSUM/FLOAT(KOUNT)
0108          XSD=((XSUM2-XSUM*XSUM/FLOAT(KOUNT))/FLOAT(KOUNT-1))**.5
0109          YSD=((YSUM2-YSUM*YSUM/FLOAT(KOUNT))/FLOAT(KOUNT-1))**.5
0110          XMID=3*XSD
0111          YMID=3*YSD
0112          XMIN=XMEAN-XMID
0113          XMAX=XMEAN+XMID
0114          YMIN=YMEAN-YMID
0115          YMAX=YMEAN+YMID
0116          XDIS=2.*XMID/56.
0117          WRITE(NT,101)XMIN,XMAX
0118      101 FORMAT(2(1PE10.3,2X))
0119          BACKSPACE NT
0120          READ(NT,102)MIXCH,MAXCH
0121      102 FORMAT(6A4)
0122          CALL GERAS(100)
0123          CALL GRRLSE
0124          NSTEP=4
0125          CALL INITP
0126          CALL PFKP(MASK1,PFINT)
0127          CALL FCBP(EOBINT)
0128          CALL GRAXES(0.,4092.,0.,4092.,0.,4092,0,4092)
0129          CALL GRCHAR('BP',MIXCH,10,376.,400.,KODEB)
0130          CALL GRCHAR('BP',VNAM(XNVBL),4,2250.,400.,KODEB)
0131          CALL GRCHAR('BP',MAXCH,10,3580.,400.,KODEB)
0132          WRITE(NT,101)YMIN,YMAX
0133          BACKSPACE NT
0134          READ(NT,102)MIYCH,MAYCH
0135          CALL GRCHAR('BP',MIYCH,10,0.,600.,KODEC)
0136          CALL GRCHAR('BP',VNAM(YNVBL),4,100.,2320.,KODEB)
0137          CALL GRCHAR('BP',MAYCH,10,0.,4092.,KODEB)
0138          CALL GRAXES(XMIN,XMAX,YMIN,YMAX,600,4012,600,4012)
0139          CALL GRGRID(XMID,YMID,'A',KODEE)
0140          DO 90 II=1,KCUNT
0141          CALL UV00(2)
0142          DO 91 KK=1,2
0143       91 CALL PUTUV(X(II),Y(II),KODEF)
0144          ICT=IMPY*(16-NTEN(II))
0145          CALL GRPLOT('BP','B','B',1 ,2,ICT,'O',KODEG)
0146          CALL UV99
0147       90 CONTINUE
0148          DO 95 II=1,KCUNT
0149          IF(NHOUND(II).EQ.0.AND.NCOUNT(II).LT.10)GO TO 95
0150          X(II)=X(II)-XDIS
0151          CALL UV00(2)
0152          DO 96 KK=1,2
0153       96 CALL PUTUV(X(II),Y(II),KODEF)
```

```
0154          ICT=(RDY*(16-NROUND(II)))
0155          CALL GRPLOT('BP','B','B',1 ,2,ICT,'O',KODEG)
0156           CALL UV99
0157       95 CONTINUE
0158      505 FORMAT(1H ,I10)
0159          CALL GWAIT
0160       99 CALL REFILL
0161          CALL RLSEP
0162          IF(N.EQ.1)GO TO 10
0163          IF(N.EQ.31)GO TO 299
0164          RETURN
0165      400 CALL GERAS(100)
0166          GO TO 10
0167      401 CALL GRDPLY('FORMAT ERROR.  PLEASE REENTER DATA.',35,&400)
0168          GO TO (185,259,299),NSTEP
0169      402 IF(ITYPE.EQ.1.AND.N.EQ.29)GO TO 99
0170          IF(ITYPE.EQ.1.AND.N.EQ.30)GO TO 99
0171          IF(ITYPE.EQ.1.AND.N.EQ.31)GO TO 299
0172          CALL GRDPLY('YOU SHOULD BE IN A POSITION REQUIRING EOB.',42,&400)
0173          GO TO (185,190,195),NSTEP
0174      180 CALL GRDPLY(' ',1,&400)
0175          CALL GRDPLY('ONE OF YOUR LEVEL NUMBERS EXCEEDS THE NUMBER OF LEVEL
             1S FOR THE FACTOR.',70,&400)
0176          CALL GRDPLY('PLEASE REENTER YOUR SET OF POINTS.',34,&400)
0177          GO TO 185
0178      425 CALL GRDPLY(' ',1,&400)
0179          CALL GRDPLY(' ',1,&400)
0180          CALL GRDPLY('DOES IT MAKE SENSE TO PLOT ONLY ONE POINT?',42,&400)
0181          CALL GRDPLY('PRESS KEY 1 TO REENTER A NEW SET OF POINTS.',43,&400)
0182      426 CALL GWAIT
0183          IF(N.EQ.1)GO TO 10
0184          IF(N.EQ.31)GO TO 299
0185          GO TO 426
0186      450 CALL GRDPLY(' ',1,&400)
0187          CALL GRDPLY('PLEASE REENTER YOUR NAMES.',26,&400)
0188          GO TO 191
0189      299 IF(NSTEP.EQ.4)GO TO 300
0190          CALL GERAS(100)
0191          CALL GRRLSE
0192           STOP
0193      300 CALL REFILL
0194          CALL RLSEP
0195          CALL UV99
0196          STOP
0197          END
```

```
0001          SUBROUTINE DELETE
0002          EXTERNAL PFINT,COBINT
0003          COMMON N,IGVLY,ITYPE,TTL(15),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),
             1LEVEL(2,12),NRA,NCA,NRB,NCB,JAB(8),N1,N2,NVBL,NFACT,NG12,MULT,
             2NSUB2,NLUV ,NN1,NN2,NT,ISEC,NI2,NJ2,KOUNT,NOTAG,IREC1,IREC2,IREC
             3,NTRAN(10),TNAM(10),LUV(10),MNLUV(10)
0004          DATA CSIZE/'8'/
0005          DATA BLANK/'    '/
0006          MULT=0
0007          CALL GRINIT(CSIZE)
0008          CALL GCEGB(COBINT)
0009          MASK=:610612739
0010          CALL GCPFK(MASK,PFINT)
0011       20 CALL GERAS(100)
0012          CALL GRDPLY(' ',1,6400)
0013          CALL GRDPLY(' ',1,6400)
0014          CALL GRDPLY('YOUR RESPONSE VARIABLE NAMES WILL APPEAR BELOW ONE AT
             1 A TIME.',61,6400)
0015          CALL GRDPLY('PRESS KEY 1 IF YOU WISH TO INCLUDE THE VARIABLE OR KE
             1Y 2 IF',59,6400)
0016          CALL GRDPLY('YOU WISH TO DELETE THE VARIABLE.',32,6400)
0017          CALL GRDPLY(' ',1,6400)
0018          IK=0
0019       25 IK=IK+1
0020          TEST=VNAM(IK)
0021          CALL GRDPLY(TEST,4,6400)
0022       26 CALL GWAIT
0023          IF(N.EQ.30)GO TO 203
0024          IF(N.EQ.31)GO TO 210
0025          IF(N.EQ.1)GO TO 29
0026          IF(N.EQ.2)GO TO 35
0027          GO TO 26
0028       29 LUV(IK)=IK
0029          MNLUV(IK)=IK
0030          TNAM(IK)=VNAM(IK)
0031          IF(IK.LT.NVBL)GO TO 25
0032          IF(IK.EQ.NVBL)GO TO 30
0033       35 CALL GBKSP(1)
0034          LUV(IK)=0
0035          MNLUV(IK)=0
0036          TNAM(IK)=BLANK
0037          IF(IK.EQ.NVBL)GO TO 30
0038          IF(IK.LT.NVBL)GO TO 25
0039       30 CALL GRDPLY(' ',1,6400)
0040          CALL GRDPLY(' ',1,6400)
0041          CALL GRDPLY('THE ABOVE VARIABLES ARE THE ONES YOU DECIDED TO INCLU
             1DE.',56,6400)
0042          CALL GRDPLY('IF YOU AGREE, PRESS KEY 1;  IF YOU WOULD LIKE TO TRY
             1AGAIN,',59,6400)
0043          CALL GRDPLY('PRESS KEY 2.',12,6400)
0044       32 CALL GWAIT
0045          IF(N.EQ.1)GO TO 50
0046          IF(N.EQ.2)GO TO 20
0047          IF(N.EQ.30)GO TO 203
0048          IF(N.EQ.31)GO TO 210
0049          GO TO 32
```

Reproduced from best available copy.

```
0050          50 IF(LUV(NVBL).EQ.0)TNAM(NVBL)=VNAM(NVBL)
0051             NVBL1=NVBL-1
0052             DO 150 IQ=1,NVBL1
0053             IF(TNAM(IQ).NE.BLANK)GO TO 150
0054         153 DO 152 JA=IQ,NVBL1
0055         152 TNAM(JA)=TNAM(JA+1)
0056             IF(TNAM(IQ).EQ.BLANK)GO TO 153
0057         150 CONTINUE
0058         203 CALL GERAS(100)
0059             CALL GRRLSE
0060             RETURN
0061         400 GO TO 203
0062         210 CALL GERAS(100)
0063             CALL GRRLSE
0064             STOP
0065             END
```

```
0001          SUBROUTINE MAINB
0002          COMMON N,ICVLY,ITYPE,TTL(15),VNAM(10),FNAM(2),NLVL(2),NENG(2,12),
             1LEVEL(2,12),NRA,NCA,NRB,NCB,JAD(8),N1,N2,NVBL,NFACT,NC12,MULT,
             2NSUBJ,NLUV ,NW1,NW2,NT,ISEC,NF2,NJ2,NCOUNT,NC1AG,IREC1,IREC2,IREC3
             3,NTRAN(10),TRAP(10),LUV(10),MALUV(10)
0003          COUBLE PRECISION DSQV,DSQU,DSSCP,D
0004          DIMENSION SUB(12,12),RE(12),CE(12),C(12),CB(12),AA1(12,12),RE1(12)
             1,RE2(12),CE1(12),CE2(12),CA1(12),CA2(12),CB1(12),CB2(12),SUB1(12,1
             22),SUB2(12,12),F(55),E(55),CRAP(11),EI(55),HA(55),HB(55),HAB(55),
             3HSB(55),HF(55),DSSCP(10,10),COMC(10)
0005          N1=NI2
0006          NJ=NJ2
0007     4000 BACKSPACE NW2
0008          READ(NW2)AA1
0009          REWIND NW2
0010          READ(NW2) DSSCP
0011          DO 101 IX=1,3
0012      101 READ(NW2) SUB,SUB,SUB,SUB,SUB,SUB,SUB,SUB,SUB,SUB
0013          READ(NW2) COMC,COMC
0014          KG = NVBL
0015          DO 3011 NV1=1,KC
0016          DO 299 ITX=1,NV1
0017      299 READ (NW2) N1,NJ,K,SUB,RE,CE,C,CB
0018          BACKSPACE NW2
0019          DO 302 JG = 1,12
0020          RE1(JG)=RE(JG)
0021          CE1(JG) = CE(JG)
0022          CA1(JG) = C(JG)
0023          CB1(JG) = CB(JG)
0024          DO 302 JH=1,12
0025      302 SUB1(JG,JH) = SUB(JG,JH)
0026          DO 301 NV2 = NV1,KG
0027          READ (NW2) N1,NJ,K,SUB,RE,CE,C,CB
0028          DO 303 JG = 1,12
0029          RE2(JG) = RE(JG)
0030          CE2(JG) = CE(JG)
0031          CA2(JG) = C(JG)
0032          CB2(JG) = CB(JG)
0033          DO 303 JH = 1,12
0034      303 SUB2(JG,JH) = SUB(JG,JH)
0035          D = DSSCP(NV1,NV2)
0036          CALL GENHED(D,AA1,SUB1,SUB2,RE1,RE2,CE1,CE2,CA1,CA2,CB1,CB2,NV1,
             1NV2,N1,NJ,E,FA,HB,HAB,HSE,ABESB,NDFE)
0037      301 CONTINUE
0038          REWIND NW2
0039          READ(NW2)COMC,COMC,COMC,COMC,COMC,COMC,COMC,COMC,COMC,COMC,COMC,
             1COMC,COMC,COMC,COMC,COMC,COMC,COMC,COMC,COMC
0040          DO 108 IZ=1,3
0041      108 READ(NW2) SUB,SUB,SUB,SUB,SUB,SUB,SUB,SUB,SUB,SUB
0042          READ(NW2) COMC,COMC
0043     3011 CONTINUE
0044          NRA = NVBL
0045          NCA = NVBL
0046          NRB = NVBL
0047          NCB=NVBL
0048          CALL INSLCE(E,EI,CRAP)
```

```
0049              ALDE = CRAP(NRA)
0050              NSTEP=1
0051              IREC=IREC1-1
0052              IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0053        501   WRITE(N1'IREC1,2005)(TTL(IJ),IJ=1,15)
0054              WRITE(N2'IREC2,2006)
0055              NSTEP=2
0056              IREC=IREC1-1
0057              IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0058        502   WRITE(N1'IREC1,2003)
0059              WRITE(N2'IREC2,2003)
0060              NSTEP=3
0061              IREC=IREC1-1
0062              IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0063        503   WRITE(N1'IREC1,2007)
0064              WRITE(N2'IREC2,2008)FNAM(NI),FNAM(NJ)
0065              CALL WRIR(E,NRA,1)
0066              NSTEP=4
0067              IREC=IREC1-1
0068              IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0069        504   WRITE(N1'IREC1,2003)
0070              WRITE(N2'IREC2,2003)
0071              NSTEP=5
0072              IREC=IREC1-1
0073              IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0074        505   WRITE(N1'IREC1,2009)
0075              WRITE(N2'IREC2,2003)
0076              CALL WRIR(CRAP,NRA,0)
0077              DO 312 I=1,NRA
0078              II = LTERM(I,1)
0079              IF(E(II)) 321,321,325
0080        325   CRAP(I) = SQRT(E(II))
0081        312   CONTINUE
0082              DO 313 I = 1,NRA
0083              DO 313 J=I,NRA
0084              IJ = LTERM(I,J)
0085        313   RR(IJ) = E(IJ)/(CRAP(I)*CRAP(J))
0086              NSTEP=6
0087              IREC=IREC1-1
0088              IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0089        506   WRITE(N1'IREC1,2003)
0090              WRITE(N2'IREC2,2003)
0091              NSTEP=7
0092              IREC=IREC1-1
0093              IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0094        507   WRITE(N1'IREC1,2010)
0095              WRITE(N2'IREC2,2003)
0096              CALL WRIR(RR,NRA,1)
0097        321   CONTINUE
0098              NSTEP=8
0099              IREC=IREC1-1
0100              IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0101        508   WRITE(N1'IREC1,2003)
0102              WRITE(N2'IREC2,2003)
0103              NSTEP=9
0104              IREC=IREC1-1
```

```
0105          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0106      509 WRITE(N1'IREC1,2011)FNAM(NI)
0107          WRITE(N2'IREC2,2012)FNAM(NJ)
0108          REWIND NW2
0109          WRITE(NW2) HAB,KA,HB,E,CRAP,EI,HSB,NDFE,NDFI,ALDE,NI,NJ,NDFSB
0110          REWIND NW2
0111          RETURN
0112     1001 WRITE(N1'IREC1,2001)KOUNT
0113          WRITE(N2'IREC2,2002)KOUNT
0114          KOUNT=KOUNT+1
0115          GO TO(501,502,503,504,505,506,507,508,509),NSTEP
0116     2001 FORMAT(55X,'PAGE',I4,'L',16X/)
0117     2002 FORMAT(55X,'PAGE',I4,'R',16X/)
0118     2003 FORMAT(80X)
0119     2005 FORMAT(15A4,'  MULTIVARIA',8X)
0120     2006 FORMAT('TE ANALYSIS',69X)
0121     2007 FORMAT(40X,'MATRIX E AFTER ELIMINATION OF FA',8X)
0122     2008 FORMAT('CTORS(',A4,') AND (',A4,')',58X)
0123     2009 FORMAT(40X,'STEPWISE LOG DETERMINANTS',15X)
0124     2010 FORMAT(40X,'CORRELATIONS BASED ON E',17X)
0125     2011 FORMAT(40X,'SUBTOTALS (ALL EFFECTS) ',A4,' AND',8X)
0126     2012 FORMAT(A4,76X)
0127          END
```

```
0001          SUBROUTINE GENHE(D,NN1,SUB1,SUB2,RE1,RE2,CE1,CE2,QA1,QA2,QB1,QB2,
             1NV1,NV2,NI,NJ,E,HA,HB,HAB,HSB,NDFSB,NDFE)
        C
        C     GENERATES THE ELEMENTS OF SUM OF SQUARES AND PRODUCTS
        C     MATRICES FOR ERROR (E) AND EFFECTS OR HYPOTHESES (H)
        C
0002          DOUBLE PRECISION SSUB,SPR,S,T,DN,ET,D
0003          COMMON N,ICVLV,ITYPE,TTL(15),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),
             1LEVEL(2,12),NRA,NCA,NRB,NCB,JAB(8),N1,N2,NVBL,NFACT,NG12,MULT,
             2NSUBJ,NLUV ,NW1,NW2,NT,ISEQ,NIZ,NJZ,KOUNT,NDIAG,IREC1,IREC2,IREC3
             3,N(RAN(10),TRAN(10),LUV(10),MNLUV(10)
0004          DIMENSION  SUB1(12,12),SUB2(12,12),RE1(1),RE2(1),CE1(1),CE2(1),
             1    QA1(1),QA2(1),QB1(1),CB2(1),E(1),HA(1),HB(1),HAB(1),HSB(1),
             2    R(12),RB(12),NN1(12,12),NIDOT(12)
0005          NG = 0
0006          SSUB = 0.D0
0007          SUMA=0.
0008          SUMB=0.
0009          NR = NLVL(NI)
0010          NC = NLVL(NJ)
0011          NDFSB = -1
0012          SPR = 0.D0
0013          DO 1 I = 1,NR
0014          DO 1 J = 1,NC
0015          S = 0.D0
0016          T = 0.D0
0017          S = SUB1(I,J)
0018          T = SUB2(I,J)
0019          DN = 0.D0
0020          DN=NN1(I,J)
0021          IF(NN1(I,J))1,1,2
0022        2 SUMA = SUMA +SUB1(I,J)
0023          SUMB = SUMB + SUB2(I,J)
0024          NG=NG+NN1(I,J)
0025          SPR=SPR+S*T/DN
0026          NDFSB = NDFSB + 1
0027        1 CONTINUE
0028          ET = 0.D0
0029          ET=D-SPR
0030          S = 0.D0
0031          DN = 0.D0
0032          S = SUMA
0033          T = 0.D0
0034          T = SUMB
0035          DN = NG
0036          SSUB=SPR-S*T/DN
0037          DO 6 I = 1,NR
0038          NIDOT(I) = 0
0039          R(I) = 0.
0040          RB(I) = 0.
0041          DO 6 J = 1,NC
0042          NIDOT(I)=NIDOT(I)+NN1(I,J)
0043          R(I) = R(I) + SUB1(I,J)
0044        6 RB(I) = RB(I) + SUB2(I,J)
0045          SRD = 0.
0046          DO 7 I = 1,NR
```

```
0047          AN = NIDOT(I)
0048      7   SRU = SRU + R(I)*RB(I)/AN
0049          SRA = 0.
0050          DO 8 I = 1,NR
0051      8   SRA = SRA + RE1(I)*QA2(I)
0052          SCA = 0.
0053          DO 9 J = 1,NC
0054      9   SCA = SCA + CE1(J) * QB2(J)
0055          SINT = SPR - SPU - SCA
0056          IJ = LTERM(NV1,NV2)
0057          E(IJ) = ET
0058          HA(IJ) = SRA
0059          HB(IJ) = SCA
0060          HAB(IJ) = SINT
0061          HSB(IJ) = SSUB
0062          NDFE = NG - NDFSB - 1
0063          RETURN
0064          END
```

```
0001          SUBROUTINE MAINE
0002          COMMON N,IOVLY,ITYPE,TTL(15),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),
             1LEVEL(2,12),NRA,NCA,NRB,NCB,JAB(8),N1,N2,NVBL,NFACT,NG12,MULT,
             2NSUBJ,NLUV ,NW1,NW2,NT,ISEQ,NIZ,NJZ,KOUNT,NDIAG,IREC1,IREC2,IREC3
             3,NTRAN(10),TNAM(10),LUV(10),NNLUV(10)
0003          DIMENSION HAB(55),HA(55),HB(55),CRAP(11),EI(55),E(55),HSB(55)
0004          READ(NW2)  HAB,HA,HB,E,CRAP,EI,HSB,NDFE,NDFI,ALDE,NI,NJ,NDFSB
0005          REWIND NW2
0006          CALL TEMAT(HSB,E,CRAP,EI,NDFE,NDFSB,ALDE,NI,NJ)
0007          NDFI = NDFSB - NLVL(NI) - NLVL(NJ) + 2
0008          IF(NDFI) 501,501,502
0009      502 NSTEP=1
0010          IREC=IREC1-1
0011          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0012      50  WRITE(N1'IREC1,2003)
0013          WRITE(N2'IREC2,2003)
0014          NSTEP=2
0015          IREC=IREC1-1
0016          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0017      51  WRITE(N1'IREC1,25)FNAM(NI),FNAM(NJ)
0018          WRITE(N2'IREC2,2003)
0019          CALL TEMAT(HAB,E,CRAP,EI,NDFE,NDFI,ALDE,NI,NJ)
0020      501 NDFI = NLVL(NI) - 1
0021          IF(NDFI) 503,503,504
0022      504 NSTEP=3
0023          IREC=IREC1-1
0024          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0025      52  WRITE(N1'IREC1,2003)
0026          WRITE(N2'IREC2,2003)
0027          NSTEP=4
0028          IREC=IREC1-1
0029          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0030      53  WRITE(N1'IREC1,27)FNAM(NI)
0031          WRITE(N2'IREC2,2003)
0032          CALL TEMAT(HA,E,CRAP,EI,NDFE,NDFI,ALDE,NI,NJ)
0033      503 NDFI = NLVL(NJ) - 1
0034          IF(NDFI) 505,505,506
0035      506 NSTEP=5
0036          IREC=IREC1-1
0037          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0038      54  WRITE(N1'IREC1,2003)
0039          WRITE(N2'IREC2,2003)
0040          NSTEP=6
0041          IREC=IREC1-1
0042          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0043      55  WRITE(N1'IREC1,27)FNAM(NJ)
0044          WRITE(N2'IREC2,2003)
0045          CALL TEMAT(HB,E,CRAP,EI,NDFE,NDFI,ALDE,NI,NJ)
0046      505 ISEQ = 17
0047          WRITE(NW2) NSUBJ
0048          REWIND NW2
0049          NSTEP=7
0050      605 IREC=IREC1-1
0051          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0052          WRITE(N1'IREC1,2003)
0053          WRITE(N2'IREC2,2003)
```

```
0054          GO TO 605
0055       56 KKOUN=KOUNT-2
0056          WRITE(N1'IREC1,2004)KNOUN'
0057          WRITE(N1'IREC1,2003)
0058          WRITE(N1'IREC1,2005)
0059          WRITE(N1'IREC1,2006)
0060          WRITE(N1'IREC1,2007)
0061          DO 601 IJ=1,38
0062      601 WRITE(N1'IREC1,2003)
0063          DO 602 IJ=1,43
0064      602 WRITE(N2'IREC2,2003)
0065          ITEMP=NVBL
0066          NVBL=NLUV
0067          NLUV=ITEMP
0068          RETURN
0069     1001 WRITE(N1'IREC1,2001)KOUNT
0070          WRITE(N2'IREC2,2002)KOUNT
0071          KOUNT=KOUNT+1
0072          GO TO (50,51,52,53,54,55,56),NSTEP
0073     2001 FORMAT(55X,'PAGE',I4,'L',16X/)
0074     2002 FORMAT(55X,'PAGE',I4,'R',16X/)
0075     2003 FORMAT(60X)
0076     25   FORMAT(40X,'INTERACTION ',A4,' * ',A4,17X)
0077     27   FORMAT(49X,A4,' EFFECTS',19X)
0078     2004 FORMAT('PAGE',I3,' WAS THE LAST PAGE OF YOUR MULTIVARIATE RESULTS.
             1',25X)
0079     2005 FORMAT( 'YOU NOW HAVE THE FOLLOWING OPTIONS:',45X)
0080     2006 FORMAT('PRESS   KEY  4 TO PREFORM ANOTHER MULTIVARIATE ANALYSIS',
             125X)
0081     2007 FORMAT('        KEY 31 TO TERMINATE',53X)
0082          END
```

```
0001          SUBROUTINE TEMAT(H,E,CRAP,EI,NDFE,NDFH,ALDE,NI,NJ)
        C
        C     CONTROLS THE MULTIVARIATE ANALYSIS (LIKELIHOOD-RATIO,
        C     UNION-INTERSECTION, AND DISCRIMINANT ANALYSIS)
        C
0002          COMMON N,IGVLY,ITYPE,TTL(15),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),
             1LEVEL(2,12),NRA,NCA,NRB,NCB,JAB(8),N1,N2,NVBL,AFACT,NG12,MULT,
             2NSUBJ,NLUV ,NM1,NM2,NT,ISEQ,NIZ,NJZ,KOUNT,NDIAG,IREC1,IREC2,IREC3
             3,NTRAN(10),TNAM(10),LUV(10),NNLUV(10)
0003          DIMENSION H(1),E(1),CRAP(1),EI(1),T(100),U(10,10),SYM(55),PE(30),
             1HX(55),DIS(50),ROIS(30)
0004          NRA = NVBL
0005          NCA = NDFH
0006          NRB = NRA
0007          NCB = NRA
0008          NRD = NLVL(NI)
0009          NCGL = NLVL(NJ)
0010          IFG=1
0011          B = FNAM(NJ)
0012          IHX = (NRA*(NRA+1))/2
0013          DO 44 I = 1,IHX
0014          I = I
0015          IF (H(I))43,44,43
0016     44   CONTINUE
0017          GO TO 45
0018     43   CALL TRI(H,T,CRAP)
0019          IF(NCB-1)40,40,41
0020     41   NCT = NCB
0021          NCB = NRA
0022          DO 1 I=1,IHX
0023      1   H(I) = H(I)+E(I)
0024          CALL INSLD(H,HX,CRAP)
0025          ALDH = CRAP(NRA)
0026          AM = NDFE
0027          AM = AM   + 0.5*FLOAT (NDFH-NVBL-1)
0028          DF = NDFH*NRA
0029          XST = -(ALDE-ALDH)*AM
0030          IFG=1
0031          NSTEP=1
0032          IREC=IREC1-1
0033          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0034     3001 WRITE(N1'IREC1,2003)
0035          WRITE(N2'IREC2,2003)
0036          NSTEP=2
0037          IREC=IREC1-1
0038          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0039     3002 WRITE(N1'IREC1,2010)XST
0040          WRITE(N2'IREC2,2011)DF
0041          PR = CHIX(XST,DF)
0042          FCX = (FLOAT (NDFH))**2 + FLOAT (NRA**2) - 5.
0043          DEL = DF*FCX
0044          DEL = DEL/(48.*AM**2)
0045          DG = DF + 4.
0046          PS = CHIX(XST,DG)
0047          PZ = PR +(PS-PR)*DEL
0048          IF(PZ - 1.0) 310,310,311
```

```
0049        310  PR = PZ
0050        311  IF(PR - 1.0) 312,313,313
0051        313  PR = 1.0
0052        312  PR = 1.-PR
0053             NSTEP=3
0054             IREC=IREC1-1
0055             IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0056       3003  WRITE(N1'IREC1,2012)PR
0057             WRITE(N2'IREC2,2003)
0058             NSTEP=4
0059             IREC=IREC1-1
0060             IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0061       3004  WRITE(N1'IREC1,2003)
0062             WRITE(N2'IREC2,2003)
0063             NSTEP=5
0064             IREC=IREC1-1
0065             IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0066       3005  WRITE(N1'IREC1,2013)
0067             WRITE(N2'IREC2,2003)
0068             CALL WRIR(H,NRA,1)
0069        212  DO 161 I=1,NVBL
0070             II = LTERM(I,I)
0071             IF(H(II)) 171,171,315
0072        315  DIS(I) = SQRT(H(II))
0073        161  CONTINUE
0074             DO 162 I=1,NVBL
0075             DO 162 J=1,NVBL
0076             IJ = LTERM(I,J)
0077        162  SYM(IJ) = H(IJ)/(DIS(I)*DIS(J))
0078             NSTEP=6
0079             IREC=IREC1-1
0080             IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0081       3006  WRITE(N1'IREC1,2003)
0082             WRITE(N2'IREC2,2003)
0083             NSTEP=7
0084             IREC=IREC1-1
0085             IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0086       3007  WRITE(N1'IREC1,2014)
0087             WRITE(N2'IREC2,2003)
0088             CALL WRIR(SYM,NVBL,1)
0089        171  CONTINUE
0090             GO TO (213,95),IFG
0091        213  DO 3 J = 1,NCT
0092             DO 3 I = 1,NRA
0093             U(I,J) = 0.
0094             DO 3 K = 1,NRA
0095             IK = LTERM(I,K)
0096             KJ = (J-1)*NRA + K
0097          3  U(I,J) = U(I,J) + HX(IK)*T(KJ)
0098             DO 4 J = 1,NCT
0099             DO 4 I = 1,J
0100             IJ = LTERM(I,J)
0101             SYM(IJ) = 0.
0102             DO 4 K=1,NRA
0103             KI = (I-1)*NRA + K
0104          4  SYM(IJ) = SYM(IJ) + T(KI)*U(K,J)
```

```
0105          DO 60 J=1,NCT
0106          DO 60 I=1,J
0107          IJ = LTERM(I,J)
0108          HX(IJ) = 0.
0109          DO 80 K=1,NCT
0110          IK = LTERM(I,K)
0111          KJ = LTERM(K,J)
0112     80   HX(IJ) = HX(IJ)+SYM(IK)*SYM(KJ)
0113          CALL FACE(HX,PE,NCT)
0114          RCAN = 0.
0115          DO 10 I = 1,NCT
0116     10   RCAN = RCAN + PE(I)**2
0117          DO 11 I = 1,NCT
0118     11   PE(I) = PE(I)/SQRT (RCAN)
0119          RCAN = SQRT(RCAN)
0120          IF(NRA-NDFH)12,13,13
0121     12   NS = NRA
0122          GO TO 14
0123     13   NS = NDFH
0124     14   AM = NDFH - NRA
0125          AM = (ABS (AM)-1.)/2.
0126          AN = NDFE-NRA-1
0127          AN = AN/2.
0128          NSTEP=8
0129          IREC=IREC1-1
0130          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0131     3008 WRITE(N1'IREC1,2003)
0132          WRITE(N2'IREC2,2003)
0133          NSTEP=9
0134          IREC=IREC1-1
0135          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0136     3009 WRITE(N1'IREC1,2015)
0137          WRITE(N2'IREC2,2016)RCAN
0138          NSTEP=10
0139          IREC=IREC1-1
0140          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0141     3010 WRITE(N1'IREC1,2017)NS,AM,AN
0142          WRITE(N2'IREC2,2003)
0143          DO 16 I = 1,NRA
0144          DIS(I) = 0.
0145          DO 16 J = 1,NCT
0146     16   DIS(I) = DIS(I) + U(I,J)*PE(J)
0147     52   NSTEP=11
0148          IREC=IREC1-1
0149          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0150     3011 WRITE(N1'IREC1,2003)
0151          WRITE(N2'IREC2,2003)
0152          NSTEP=12
0153          IREC=IREC1-1
0154          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0155     3012 WRITE(N1'IREC1,2018)
0156          WRITE(N2'IREC2,2003)
0157          CALL WRIR(DIS,NRA,0)
0158          DO 18 I = 1,NRA
0159          PDIS(I) = 0.
0160          DO 18 J = 1,NRA
```

```
0161          IJ = LTERM(I,J)
0162     18   RDIS(I) = E(IJ)*DIS(J)+RDIS(I)
0163          CON = 0.
0164          DO 19 J = 1,NRA
0165     19   CON = CON + RDIS(J)*DIS(J)
0166          DO 20 I = 1,NRA
0167          II = LTERM(I,I)
0168     20   RDIS(I) = RDIS(I)/SQRT (CON*E(II))
0169          NSTEP=13
0170          IREC=IREC1-1
0171          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0172     3013 WRITE(N1'IREC1,2003)
0173          WRITE(N2'IREC2,2003)
0174          NSTEP=14
0175          IREC=IREC1-1
0176          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0177     3014 WRITE(N1'IREC1,2019)
0178          WRITE(N2'IREC2,2003)
0179          CALL WRIR(RDIS,NRA,0)
0180          GO TO 99
0181     45   NSTEP=15
0182          IREC=IREC1-1
0183          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0184     3015 WRITE(N1'IREC1,2003)
0185          WRITE(N2'IREC2,2003)
0186          NSTEP=16
0187          IREC=IREC1-1
0188          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0189     3016 WRITE(N1'IREC1,2020)
0190          WRITE(N2'IREC2,2003)
0191          GO TO (99,210),IFG
0192          GO TO 99
0193     40   DO 50 I = 1,NRA
0194          DIS(I) = 0.
0195          DO 50 J = 1,NRA
0196          IJ = LTERM(I,J)
0197     50   DIS(I) = E(IJ)*T(J)+DIS(I)
0198          F = 0.
0199          DO 51 I = 1,NRA
0200     51   F = F+DIS(I)*T(I)
0201          DFN = NRA
0202          DFN = NDFE-NRA+1
0203          F = F*DFN/DFM
0204          IFG=2
0205          NSTEP=17
0206          IREC=IREC1-1
0207          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0208     3017 WRITE(N1'IREC1,2003)
0209          WRITE(N2'IREC2,2003)
0210          NSTEP=18
0211          IREC=IREC1-1
0212          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0213     3018 WRITE(N1'IREC1,2021)   F,DFN,DFN
0214          WRITE(N2'IREC2,2003)
0215          GO TO 52
0216     210  DO 211 I=1,IMX
```

```
0217          211 H(I)=H(I)+E(I)
0218              GO TO 212
0219           99 RETURN
0220         1001 WRITE(N1'IREC1,2001)KOUNT
0221              WRITE(N2'IREC2,2002)KOUNT
0222              KOUNT=KOUNT+1
0223              GO TO(3001,3002,3003,3004,3005,3006,3007,3008,3009,3010,3011,3012,
                 13013,3014,3015,3016,3017,3018),NSTEP
0224         2001 FORMAT(55X,'PAGE',I4,'L',16X/)
0225         2002 FORMAT(55X,'PAGE',I4,'R',16X/)
0226         2003 FORMAT(80X)
0227         2010 FORMAT('LIKELIHOOD RATIO TEST STATISTIC, CHI-SQUARE = ',E14.4,' WI
                 1TH',15X)
0228         2011 FORMAT(F10.0,' D.F.',65X)
0229         2012 FORMAT(30X,'SIGNIFICANT AT LEVEL ',E13.3,16X)
0230         2013 FORMAT(40X,'MATRIX H+E',30X)
0231         2014 FORMAT(40X,'CORRELATIONS BASED ON H+E',15X)
0232         2015 FORMAT('STANDARDIZED ROY STATISTIC OR SQUARE OF A CANONICAL CORREL
                 1ATION = ',13X)
0233         2016 FORMAT(E16.6,64X)
0234         2017 FORMAT(10X,'READ HECK CHARTS WITH S = ',I3,' M = ',F8.1,' N = ',F6
                 1.1,15X)
0235         2018 FORMAT(20X,'WEIGHT OF DISCRIMINANT FUNCTION',28X)
0236         2019 FORMAT(7X,'CORRELATIONS BETWEEN DISCRIMINANT FUNTION AND ORIGINAL
                 1VARIABLES',8X)
0237         2020 FORMAT(20X,'DEGENERATE CASE ALL H-ELEMENTS ZERO',24X)
0238         2021 FORMAT('TEST STATISTIC F = ',F10.4,' WITH ',F4.0,' AND ',F6.0,' D.
                 1F.',25X)
0239              END
```

```
0001          SUBROUTINE  TRI(A,T,D)
        C
        C       TRI ROUTINE . THE UPPER HALF OF A SYMMETRIC GRAMIAN MATRIX
        C       IS STORED AS A ONE DIMENSIONAL VECTOR IN PACKED FORM A(1,1),
        C       A(1,2),A(2,2),A(1,3),A(2,3),A(3,3), ETC. NRA AND NCA (FIRST
        C       TWO COMMONS) MUST BE SET EQUAL TO THE ORDER OF THE MATRIX.
        C       D IS A WORKING FIELD, AND SHOULD BE DIMENSIONED ONE LARGER
        C       THAN NRA AT THE END,  NRB=NRA, BUT NCB IS REPLACED BY THE
        C       RANK OF THE MATRIX A . THE MATRIX T , PACKED IN ORDER T(1,1),
        C       T(2,1),T(3,1),....,T(NRA,1),T(1,2),....,T(NRA,2),....,T(NRA,NRB),
        C       IS A TRIANGULAR MATRIX T SUCH THAT T*T' = A . IF THE CALLING
        C       PROGRAM WISHES TO USE T AS A DOUBLE SUBSCRIPTED FIELD, THE
        C       DIMENSION OF THE FIRST ARGUMENT MUST E Q U A L NRA . IF THIS
        C       IS INCONVENIENT, SINGLE-DIMENSION T IN THE SAME WAY AS A, THEN
        C       UNPACK T INTO A DOUBLE SUBSCRIPTED TU BY CALCULATING IJ =
        C       (J-1)*NRA + I . THEN TU(I,J) = T(IJ)
        C
0002          COMMON N,IOVLY,ITYPE,TTL(15),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),
             1LEVEL(2,12),NRA,NCA,NRB,NCB,JAB(8),N1,N2,NVBL,NFACT,NG12,MULT.
             2NS,BJ,NLUV ,NW1,NW2,NT,ISEC,NIZ,NJZ,KOUNT,NDIAG,IREC1,IREC2,IREC3
             3,NRAN(10),TNAM(10),LUV(10),MNLUV(10)
0003          DIMENSION   A(1),T(1),D(1)
0004          NRB = NRA
0005          NCB = NRA
0006          DO 577 I=1,NRA
0007          I = I
0008          CALL SLITO(0)
0009          DO 577 J=I,NRA
0010          J=J
0011          IJA = (J*(J-1))/2 + I
0012          SUM = A(IJA)
0013          IM = I-1
0014     400  IF(IM) 566,566,575
0015     575  DO 574 K=1,IM
0016          K=K
0017          KIA = (K-1)*NRB + I
0018          KJA = KIA + J - I
0019     574  SUM = SUM - T(KIA)*T(KJA)
0020     566  IF(J-I) 585,585,573
0021     585  IF(A(IJA)) 572,572,576
0022     576  D(I) = SUM
0023          QQ = A(IJA)
0024     401  IF(SUM/QQ + 5.E-4) 572,570,570
0025     570  IF(SUM/QQ - 5.E-5) 590,590,661
0026     661  IF(I-NCA) 573,573,590
0027     572  WRITE(N1'IREC1,650)
0028          WRITE(N2'IREC2,2003)
0029     650  FORMAT(40X,'MATRIX IS NOT GRAMIAN',19X)
0030    2003  FORMAT(80X)
0031          GO TO 607
0032     590  CALL SLITO(1)
0033          D(I) = 0.
0034     573  IJ = (I-1)*NRB + J
0035          CALL SLITOT(1,JLL)
0036          IF(JLL) 569,569,591
0037     569  T(IJ) = SUM/SQRT (D(I))
```

```
0038          GO TO 577
0039      591 CALL SLITO(1)
0040      402 IF(ABS (SUM/CO) - 5.E-5) 592,592,662
0041      662 IF(I-NCA) 572,572,592
0042      592 T(IJ) = 0.
0043      577 CONTINUE
0044          IF(NRB-1) 601,601,630
0045      630 DO 600 I=2,NRB
0046          I=I
0047          DO 600 J=2,I
0048          J=J
0049          IJC = (I-1)*NRB + J - 1
0050      600 T(IJC)=0.
0051      601 CONTINUE
0052          DO 602 I=1,NCB
0053          I=I
0054          DO 603 J=1,NRB
0055          J=J
0056          IJD = (I-1)*NRB + J
0057          IF(T(IJD)) 602,604,602
0058      604 IF(J-NRB) 603,605,605
0059      603 CONTINUE
0060      605 NCB = NCB-1
0061      403 IF(I-NCB-1) 606,610,610
0062      606 DO 608 K=I,NCB
0063          DO 608 L=1,NRB
0064          IJE = K*NRB + L
0065          IJF = IJE - NRB
0066      608 T(IJF) = T(IJE)
0067          GO TO 601
0068      602 CONTINUE
0069      610 GO TO 607
0070      607 RETURN
0071          END
```

```
0001          FUNCTION CHIX(X,DF)
        C
        C        CHI-SQUARE P FROM X
        C
0002          CHIX=GAMX(X/2.,DF/2.)
0003          RETURN
0004          END
```

```
0001          FUNCTION GAMX(X,DE)
        G
        C     INCOMPLETE GAMMA DISTRIBUTION, FORWARD, P FROM X
        C
0002          Y=X
0003          DF=DE
0004          SUM=0.
0005          IF(DF-200.)20,21,21
0006     20   IF(X)1,1,2
0007      1   GAMX=0.
0008          GO TO 99
0009      2   IF(DF)3,3,4
0010      3   GAMX = 1.
0011          GO TO 99
0012      4   AI = DF
0013          F = AI * ALOG(Y) -Y -ELGGM(AI+1.)
0014     16   IF(F+80.)10,10,11
0015     11   FG=EXP (F)
0016          FH = FG
0017     12   SUM = SUM + FG
0018          FG=FG*Y/(AI+1.)
0019          AI = AI + 1.
0020          IF(AI-200.)28,28,25
0021     28   IF(FG-FH)13,14,14
0022     14   IF(AI-200.)12,12,25
0023     13   FH = FG
0024          IF(FG/SUM - 1.E-8)15,15,12
0025     10   AI = AI+1.
0026          IF(AI-200.)27,27,25
0027     27   F=F+ALOG(Y/AI)
0028          GO TO 16
0029     21   DH=9.*DF
0030     26   YN = ((Y/DF)**0.33333333-1.+1./DH)*SQRT (DH)
0031          UNMIX=YORHX(YN)
0032          GAMX=UNMIX+SUM
0033          GO TO 99
0034     15   GAMX = SUM
0035          GO TO 99
0036     25   DH = 9.*AI
0037          DF = AI
0038          GO TO 26
0039     99   RETURN
0040          END
```

```
0001          FUNCTION YORMX(X)
        C
        C        NORMAL DISTRIBUTION, ARGUMENT X,RESULT P
        C
0002          Y=X
0003          IF(X+13.)1,1,2
0004     1    YORMX=0
0005          GO TO 99
0006     2    IF(X-7.)3,3,4
0007     4    YORMX = 1.
0008          GO TO 99
0009     3    G=1.12837917*EXP (-Y*Y/2.)
0010          Z=G/2.82842712
0011          XA = ABS (Y)
0012          IF(XA-2.5)5,6,6
0013     6    U = 1./(XA+1./(XA+2./(XA+3./(XA+4./(XA+5./(XA+6./(XA+7./(XA+8./(XA
             1+9./(XA+10./(XA+11./(XA+12./XA)))))))))))
0014     10   IF(Y)7,8,8
0015     7    YORMX = U*Z
0016          GO TO 99
0017     8    YORMX=1.-U*Z
0018          GO TO 99
0019     5    ET = 1.41421356/(1.41421356+0.3275911*XA )
0020          U = G*((((0.94064607*ET-1.28782245)*ET+1.25969513)*ET-0.25212
             1866)*ET+0.225836846)*ET * 0.5
0021          Z=1.
0022          GO TO 10
0023     99   RETURN
0024          END
```

```
0001          FUNCTION ELGGM(X)
        C
        C       LOG OF GAMMA FUNCTION FOR ARGUMENTS GREATER THAN ZERO
        C
0002          Y=X
0003          TERM=1.0
0004          IF(X)1,1,2
0005      1   ELGGM = 0.
0006          GO TO 99
0007      2   IF(Y-10.)3,3,4
0008      3    TERM=TERM*Y
0009          Y=Y+1.
0010          GO TO 2
0011      4   ELGGM=(Y-0.5)*ALOG(Y)-Y+1./(12.*Y)-1./(360.*Y**3)+1./(1260.
             1*Y**5)-1./(1680.*Y**7)+0.91893853-ALOG(TERM)
0012      99  RETURN
0013          END
```

```
0001          SUBROUTINE FACE(R,FE,NRR)
        C
        C     OBTAINS THE LARGEST ROOT AND ASSOCIATED EIGENVECTOR OF
        C     A SYMETRIC MATRIX
        C
0002          DIMENSION R(1275),F(50),FL(50),NTAG(50),FE(50),RTR(50)
0003          DOUBLE PRECISION OND,SUMA,SUMB,FJ,FEJ,RIJD,FEI
0004          NREF=0
0005          ITEC = 600/NRR
0006       80 DO 99 I=1,NRR
0007       99 NTAG(I)=0
0008          NCOU=0
0009          DO 100 J=1,NRR
0010          F(J)=0.0
0011          DO 101 I=1,NRR
0012          IJA=LTERM(I,J)
0013          IF(I-J)105,101,105
0014      105 F(J)=F(J) + R(IJA)
0015      101 CONTINUE
0016      100 F(J)= -F(J)/2.0
0017      109 DO 110 I=1,NRR
0018          IF(NTAG(I)) 111,112,111
0019      111 FL(I)=-F(I)
0020          GO TO 110
0021      112 FL(I)=F(I)
0022      110 CONTINUE
0023          DO 120 I=1,NRR
0024          IF(FL(I))120,120,121
0025      120 CONTINUE
0026          GO TO 199
0027      121 IF(NCOU-2*NRR)122,122,199
0028      122 JA=1
0029          DO 123 J=2,NRR
0030          IF(FL(I)-FL(J))124,123,123
0031      124 FL(I)=FL(J)
0032          JA=J
0033      123 CONTINUE
0034          IF(NTAG(JA))125,126,125
0035      126 NTAG(JA)=1
0036          GO TO 130
0037      125 NTAG(JA)=0
0038      130 NCOU=NCOU+1
0039          DO 131 I=1,NRR
0040          IF(I-JA)132,131,132
0041      132 IJB=LTERM(I,JA)
0042          F(I)=F(I)+R(IJB)
0043      131 CONTINUE
0044          GO TO 109
0045      199 DO 200 I=1,NRR
0046      200 F(I)=-2.*F(I)
0047          DO 211 I=1,NRR
0048          II = LTERM(I,I)
0049      211 FE(I) = F(I) + SIGN (R(II),F(I))
0050          SUM=0.0
0051          DO 222 J=1,NRR
0052      222 SUM=SUM+ABS (FE(J))
```

```
0053          DO 223 K=1,NRR
0054      223 FE(K)=FE(K)/SQRT(SUM)
0055          NITE=0.
0056          ONC = 1.00
0057      303 DO 304 I=1,NRR
0058          SUMA=0.000
0059          SUMB=0.000
0060          DO 305 J=1,NRR
0061      306  FJ=0.000
0062          FJ = FE(J)
0063           FEJ=0.000
0064          FEJ=FE(J)
0065          IJD=LTERM(I,J)
0066          RIJD=0.000
0067          RIJD=R(IJD)
0068          SUMA=SUMA+RIJD*FJ
0069          SUMB=SUMB+FEJ*FJ
0070      305 CONTINUE
0071          FL(I)=FE(I)
0072          FEI=0.000
0073          FEI=SUMA/SUMB
0074      304 FE(I)=FEI
0075          NITE=NITE+1
0076          IF(NITE - ITEC) 320,320,399
0077      320 DO 321 I=1,NRR
0078          IF(ABS (FE(I)-FL(I))-1.E-6)321,321,303
0079      321 CONTINUE
0080      399 RETURN
0081          END
```

```
0001          SUBROUTINE WRIR(R,NV,NN)
        C
        C  SUBROUTINE FOR WRITING CORRELATIONS, MEANS, AND FACTOR LOADINGS
        C   NN IS ZERO FOR FACTOR LOADINGS AND MEANS, ONE OTHERWISE
        C
0002          COMMON N,IOVLY,ITYPE,TTL(15),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),
             1LEVEL(2,12),NRA,NCA,NRB,NCB,JAB(8),N1,N2,NVBL,NFACT,NG12,MULT,
             2NSUBJ,NLUV ,NW1,NW2,NT,ISEQ,NIZ,NJZ,KOUNT,NDIAG,IREC1,IREC2,IREC3
             3,NTRAN(10),TNAM(10),LUV(10),MNLUV(10)
0003          DIMENSION R(1275),JWR(7),RWR(7),UWR(7)
0004          DO 1101 I=1,NV
0005          NRE=(NV-1)/7+1
0006          DO 1102 J=1,NRE
0007          IF(7*J-NV) 1103,1103,1104
0008     1103 K=7
0009          GO TO 1105
0010     1104 K=7-7*J+NV
0011     1105 DO 1107 L=1,K
0012          JWR(L)=7*(J-1)+L
0013          JWRL = JWR(L)
0014     1107 UWR(L)=TNAM(JWRL)
0015          NSTEP=1
0016          IREC=IREC1-1
0017          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0018     401  WRITE(N1'IREC1,2003)
0019          WRITE(N2'IREC2,2003)
0020          NSTEP=2
0021          IREC=IREC1-1
0022          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0023     402  IF(K.LE.5)GO TO 301
0024          IF(K.GT.5)GO TO 302
0025     301  WRITE(N1'IREC1,501)(UWR(M),M=1,K)
0026          WRITE(N2'IREC2,2003)
0027          GO TO 303
0028     302  WRITE(N1'IREC1,501)(UWR(M),M=1,5)
0029          WRITE(N2'IREC2,502)(UWR(M),M=6,K)
0030     303  CONTINUE
0031          IF(NN)1129,1130,1129
0032     1130 DO 1131 M=1,K
0033          MF = JWR(M)
0034     1131 RWR(M) = R(MF)
0035          NSTEP=3
0036          IREC=IREC1-1
0037          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0038     403  IF(K.LE.5)GO TO 304
0039          IF(K.GT.5)GO TO 305
0040     304  WRITE(N1'IREC1,503)(RWR(N),N=1,K)
0041          WRITE(N2'IREC2,2003)
0042          GO TO 306
0043     305  WRITE(N1'IREC1,503)(RWR(N),N=1,5)
0044          WRITE(N2'IREC2,504)(RWR(N),N=6,K)
0045     306  CONTINUE
0046          GO TO 1102
0047     1129 DO 1120 M=1,K
0048          MR= LTERM(I,JWR(M))
0049     1120 RWR(M)=R(MR)
```

```
0050          VX=TNAM(I)
0051          NSTEP=4
0052          IREC=IREC1-1
0053          IF(MOD(IREC,45) .EQ. 0) GO TO 1001
0054      404 IF(K.LE.5)GO TO 307
0055          IF(K.GT.5)GO TO 308
0056      307 WRITE(N1'IREC1,505)VX,(RWR(N),N=1,K)
0057          WRITE(N2'IREC2,2003)
0058          GO TO 309
0059      308 WRITE(N1'IREC1,505)VX,(RWR(N),N=1,5)
0060          WRITE(N2'IREC2,504)(RWR(N),N=6,K)
0061      309 CONTINUE
0062     1102 CONTINUE
0063          IF(NN)1127,1128,1127
0064     1127 CONTINUE
0065     1101 CONTINUE
0066     1128 RETURN
0067     1001 WRITE(N1'IREC1,2001)KCUNT
0068          WRITE(N2'IREC2,2002)KCUNT
0069          KOUNT=KOUNT+1
0070          GO TO(401,402,403,404),NSTEP
0071     2001 FORMAT(55X,'PAGE',I4,'L',16X/)
0072     2002 FORMAT(55X,'PAGE',I4,'R',16X/)
0073     2003 FORMAT(80X)
0074      501 FORMAT(12X,A4,40X,A4),16X)
0075      502 FORMAT(2(8X,A4),56X)
0076      503 FORMAT(7X,1P5E12.4,13X)
0077      504 FORMAT(1P2E12.4,56X)
0078      505 FORMAT(3X,A4,1P5E12.4,13X)
0079          END
```

```
0001          SUBROUTINE INSLD (A,T,D)
        C
        C     OBTAINS INVERSE OR CONDITIONAL INVERSE OF GRAMIAN
        C     MATRICES; ALSO OBTAINS STEP-WISE LOG-DETERMINANTS
        C
0002          COMMON N,IOVLY,ITYPE,TTL(15),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),
             1LEVEL(2,12),NRA,NCA,NRB,NCB,JAB(8),N1,N2,NVBL,NFACT,NGIZ,MULT,
             2NSUBJ,NLUV ,NW1,NW2,NF,ISEG,NIZ,NJZ,KCLNT,NDIAG,IREC1,IREC2,IREC3
             3,NTRAN(10),TNAM(10),LUV(10),MNLUV(10)
0003          DIMENSION A(1),T(1),D(1)
0004          CO = 0.
0005          DO 577 I=1,NRA
0006          CALL SLITO(0)
0007          DO 577 J=I,NRA
0008          IJA= ((J-1)*J)/2+ I
0009          SUM = A(IJA)
0010          IM= I-1
0011      400 IF(IM)566,566,575
0012      575 DO 574 K=1,IM
0013          KIA=(I*(I-1))/2 + K
0014          KJA = (J*(J-1))/2 + K
0015          IF(D(K)) 1584,574,1584
0016     1584 SUM=SUM-T(KIA)*T(KJA)/D(K)
0017      574 CONTINUE
0018      566 IF(J-I)585,585,573
0019      585 IF(A(IJA))576,1100,576
0020     1100 D(I)= SUM
0021          DO 1101 M=1,NRA
0022          MX= (M*(M+1))/2
0023          IF(A(MX))1102,1101,1102
0024     1102 CO= ABS (A(MX))
0025          GO TO 1103
0026     1101 CONTINUE
0027     1103 IF(CO)572,1104,401
0028     1104 LLP= (NRA*(NRA+1))/2
0029          DO 1105 M=1,LLP
0030          IF(A(M))1106,1105,1106
0031     1106 CO=ABS (A(M))
0032          GO TO 1107
0033     1105 CONTINUE
0034     1107 IF(CO)572,572,401
0035      576 D(I)=SUM
0036          CO=A(IJA)
0037      401 IF(ABS (SUM/CO)-5.E-5)590,590,573
0038      572 WRITE(N1'IREC1,1572)
0039          WRITE(N2'IREC2,2003)
0040     1572 FORMAT(17X,'MATRIX NOT POS DEFINITE OR POS SEMIDEFINITE',20X)
0041     2003 FORMAT(80X)
0042     1573 RETURN
0043      590 CALL SLITO(1)
0044          D(I)=0.0
0045      573 IJ= (J*(J-1))/2 + I
0046          CALL SLITOT(1,JLL)
0047          IF(JLL)569,569,591
0048      569 T(IJ)=SUM
0049          GO TO 577
```

```
0050          591 CALL SLITC(1)
0051          402 IF(ABS (SUM/CO)-5.E-5)592,592,569
0052          592 T(IJ)=0.0
0053          577 CONTINUE
0054              DO 300 I=1,NRB
0055              IA=((I+1)*I)/2
0056              IF(T(IA))301,300,301
0057          301 T(IA)=1.0/T(IA)
0058          300 CONTINUE
0059              IF(NRB-1)701,701,702
0060          702 NM = NRB-1
0061              DO 303 I=1,NM
0062              IPX=I+1
0063              DO 303 J=IPX,NRA
0064              IIN=(J*(J+1))/2
0065              IAC=(J*(J-1))/2+I
0066              SUMI=0.0
0067              JMI=J-1
0068              DO 304 K=I,JMI
0069              IAB=(J*(J-1))/2+K
0070              IBC=(K*(K-1))/2+I
0071              IF(T(IIN))304,305,304
0072          304 SUMI=SUMI-T(IAB)*T(IBC)
0073              T(IAC)=SUMI*T(IIN)
0074              GO TO 303
0075          305 T(IAC)=0.0
0076          303 CONTINUE
0077              DO 101 I=1,NRA
0078              DO 101 J=I,NRA
0079              SUMX=0.0
0080              DO 901 K=J,NRA
0081              KIX=(K*(K-1))/2 +I
0082              KJX=KIX+ J- I
0083          901 SUMX=SUMX+T(KIX)*T(KJX)*C(K)
0084              JIX=(J*(J-1))/2+I
0085          101 T(JIX)=SUMX
0086          701  CONTINUE
0087              IF(D(1))114,114,115
0088          115 D(1)=ALOG(C(1))
0089              GO TO 118
0090          114 C(1)=0.-.99999999E38
0091          118 IF(NRA-1) 1573,1573,1575
0092         1575 DO 122 I=2,NRA
0093              IM=I-1
0094              IF(D(I))119,119,120
0095          120 IF(D(IM)+.99999999E+38)119,119,123
0096          123 D(I)=D(IM)+ALOG(C(I))
0097              GO TO 122
0098          119 C(I)=0.-.99999999E38
0099          122  CONTINUE
0100              RETURN
0101              END
```

```
0001          FUNCTION LTERM(II,JJ)
        C
        C     LOCATES TERMS OF SYMMETRIC MATRICES STORED IN PACKED
        C     (UPPER TRIANGULAR) FORM
        C
0002          IF(II-JJ) 1,2,2
0003        1 LTERM = (JJ * (JJ-1))/2 + II
0004          GO TO 3
0005        2 LTERM = (II * (II-1))/2 + JJ
0006        3 RETURN
0007          END
```

```
0001          SUBROUTINE SLITO (IDS)
0002          COMMON N,IOVLY,ITYPE,TTL(15),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),
             1LEVEL(2,12),NRA,NCA,NRB,NCB,JAB(8),N1,N2,NVBL,NFACT,NG12,MULT,
             2NSUBJ,NDATA,NW1,NW2,NT,ISEC,NIZ,NJZ,KCUNT,NDIAG,IREC1,IREC2,IREC3
             3,NTRAN(10)
0003          IF(IDS) 1,1,2
0004     2    JAB(IDS) = 1
0005          RETURN
0006     1    DO 3 K=1,8
0007     3    JAB(K) = 0
0008          RETURN
0009          END
```

```
0001          SUBROUTINE SLITOT(IDO,JDO)
0002          COMMON N,IOVLY,ITYPE,TTL(15),VNAM(10),FNAM(2),NLVL(2),NEND(2,12),
             1LEVEL(2,12),NRA,NCA,NRB,NCB,JAB(8),N1,N2,NVBL,NFACT,NG12,MULT,
             2ASUBJ,NDATA,NW1,NW2,NT,ISEC,NIZ,NJZ,KOUNT,NDIAG,IREC1,IBEC2,IBEC3
             3,NTRAN(10)
0003          IF(JAB(IDO)) 1,1,2
0004     1    JDO = 0
0005          GO TO 3
0006     2    JDO = 1
0007     3    JAB(IDO) = 0
0008          RETURN
0009          END
```