AN IMPROVED MULTIPLE LINEAR REGRESSION AND DATA ANALYSIS COMPUTER PROGRAM PACKAGE

by Steven M. Sidik

Lewis Research Center
Cleveland, Ohio 44135

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION - WASHINGTON, D. C. - APRIL 1972
AN IMPROVED MULTIPLE LINEAR REGRESSION AND DATA ANALYSIS COMPUTER PROGRAM PACKAGE

Steven M. Sidik

Lewis Research Center
National Aeronautics and Space Administration
Cleveland, Ohio 44135

NEWRAP, an improved version of a previous multiple linear regression program called RAPIER, CREDUC, and CRSPLT, allows for a complete regression analysis including cross plots of the independent and dependent variables, correlation coefficients, regression coefficients, analysis of variance tables, t-statistics and their probability levels, rejection of independent variables, plots of residuals against the independent and dependent variables, and a canonical reduction of quadratic response functions useful in optimum seeking experimentation. A major improvement over RAPIER is that all regression calculations are done in double precision arithmetic.
## CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUMMARY</td>
<td>1</td>
</tr>
<tr>
<td>INTRODUCTION</td>
<td>2</td>
</tr>
<tr>
<td>SYMBOLS</td>
<td>3</td>
</tr>
<tr>
<td>ESTIMATION OF BASIC LINEAR MODEL</td>
<td>5</td>
</tr>
<tr>
<td>BASIC LINEAR MODEL</td>
<td>5</td>
</tr>
<tr>
<td>ESTIMATING $b$</td>
<td>8</td>
</tr>
<tr>
<td>CORRELATION MATRIX</td>
<td>11</td>
</tr>
<tr>
<td>ESTIMATING $\sigma^2$</td>
<td>11</td>
</tr>
<tr>
<td>HYPOTHESIS TESTING</td>
<td>13</td>
</tr>
<tr>
<td>NORMALITY OF $\epsilon$</td>
<td>13</td>
</tr>
<tr>
<td>ANALYSIS OF VARIANCE TABLE</td>
<td>14</td>
</tr>
<tr>
<td>CHOICE OF ESTIMATOR FOR $\sigma^2$</td>
<td>15</td>
</tr>
<tr>
<td>TEST OF OVERALL REGRESSION</td>
<td>16</td>
</tr>
<tr>
<td>$t$-TESTS</td>
<td>17</td>
</tr>
<tr>
<td>PREDICTING VALUES FROM ESTIMATED REGRESSION EQUATION</td>
<td>18</td>
</tr>
<tr>
<td>NEWRAP PROGRAM</td>
<td>19</td>
</tr>
<tr>
<td>USERS GUIDE TO NEWRAP INPUT</td>
<td>19</td>
</tr>
<tr>
<td>Illustrative Regression Problem Requiring No Transformations</td>
<td>19</td>
</tr>
<tr>
<td>Detailed Description of Input Cards</td>
<td>21</td>
</tr>
<tr>
<td>Illustrative Problem Requiring Transformations</td>
<td>31</td>
</tr>
<tr>
<td>NEWRAP DOCUMENTATION AND LISTINGS</td>
<td>40</td>
</tr>
<tr>
<td>General Mathematical and Logical Flow of Program</td>
<td>40</td>
</tr>
<tr>
<td>Routines and Their Major Functions</td>
<td>42</td>
</tr>
<tr>
<td>Call Structure of Program</td>
<td>42</td>
</tr>
<tr>
<td>Dictionary of Program</td>
<td>43</td>
</tr>
<tr>
<td>Program Listing</td>
<td>46</td>
</tr>
<tr>
<td>CRSPLT PROGRAM</td>
<td>76</td>
</tr>
<tr>
<td>CREDUC PROGRAM</td>
<td>85</td>
</tr>
<tr>
<td>APPENDIX - BORROWED ROUTINES</td>
<td>91</td>
</tr>
<tr>
<td>REFERENCES</td>
<td>94</td>
</tr>
</tbody>
</table>
AN IMPROVED MULTIPLE LINEAR REGRESSION AND DATA ANALYSIS COMPUTER PROGRAM PACKAGE
by Steven M. Sidik
Lewis Research Center

SUMMARY

NEWRAP is a digital computer program which can be used with ease to perform extensive regression analyses or a simple least-squares curve fit. The program is written in FORTRAN IV, version 13, for the IBM 7094/7044 DCS. The major value of the program is the comprehensiveness of its calculations and options.

NEWRAP computes the variance-covariance matrix of the independent variables, regression coefficients, t-statistics for individual tests, and analysis of variance tables for overall testing of regression. There is a provision for a choice of three strategies for the variance estimate to be used in computing t-statistics.

Also, more than one set of responses of dependent variables can be analyzed for the same set of independent variables.

A backward rejection option method based on the first dependent variable may be used to delete nonsignificant terms from the model. In this case, a critical significance level is supplied as input. The least significant independent variable is deleted and the regression recomputed. This process is repeated until all remaining variables have significantly nonzero coefficients.

The NEWRAP program uses the triangular form of symmetric matrices throughout. It also allows for the use of weighted regression, computation of predicted values at any combination of independent variables, a table of residuals, and plots of residuals.

By use of CRSPLT, a preregression analysis may be performed which may aid in the choice of model to use in NEWRAP. This program accepts the same raw data in the same format and computes the variance-covariance matrix and correlation matrix of all the variables and an eigenvector decomposition of the variance-covariance matrix corresponding to the independent variables. Microfilm plots are then printed of specified pairs of variables. Punched output of residuals and predicted values from NEWRAP can also be used for more complicated residual plots than the direct use of the plotting option NEWRAP permits.

When a quadratic response function has been estimated (as for example in optimum-seeking experimentation) CREDEC may be used to obtain all information necessary for a canonical analysis of the function.

The three programs together provide a useful data analysis package that can be applied to a large variety of common research and development situations.
INTRODUCTION

RAPIER (ref. 1) is a very flexible multiple linear regression analysis computer program which has been in frequent use at the NASA Lewis Research Center. It was tested with the data presented in Wampler (ref. 2) and performed quite poorly. This alone was not very disturbing since real data are seldom even nearly as ill-conditioned as that set of data. A second factor, however, is that Wampler's data leads to a 5 by 5 matrix to be inverted whereas RAPIER is designed to handle matrices of up to 60 by 60. With real data it is not uncommon for the matrix to become more ill-conditioned as the dimension increases. Often the user increases the size of the model by adding terms which are functions of the original independent variables (as for example in polynomial models) and this often leads to increased correlations and ill-conditioning. For this reason, RAPIER was modified primarily by rearranging the storage of variables in COMMON blocks and performing all the regression calculations in double precision. This was done without losing any of the capabilities of the original program (in fact adding new options). The resulting version is called NEWRAP.

It may be of interest to some RAPIER users that in a number of sample calculations the major numerical inaccuracies arising in the regression calculations were not involved in the actual inversion of the $X'X$ matrix but in the calculation of the inner products which give

$$\hat{b} = (X'X)^{-1}(X'y)$$

Thus a major improvement might be made by computing inner products in double precision arithmetic and truncating to single precision answers without going to complete double precision arithmetic although the latter alternative would further increase the accuracy. As a matter of fact, the double precision inner product calculation is used in a different least-squares method proposed by Golub (ref. 3) which is reference 19 of Wampler's paper.

It should be pointed out that in estimation problems an alternative to the obvious step of more accurate routines is provided by Hoerl and Kennard (ref. 4). They present a technique called "ridge regression" which uses the method of minimum mean squared error estimation in place of minimum variance unbiased estimation. The ridge regression technique should have some definite appeal to statisticians, because it recognizes the fact that existence of ill-conditioned data indicates a problem which should be accounted for statistically as well as computationally. They do consider the problem of rejecting terms but their methods are not amenable to incorporation in NEWRAP in its present form.

A second reason for modifying the program was the desire to provide plots of the residuals as was strongly recommended in chapter 3 of Draper and Smith (ref. 5). With
the microfilm plotting capabilities provided by CINEMATIC (ref. 6) available at the Lewis Research Center computer facility, this feature was also added to NEWRAP without significantly increasing printed output. CINEMATIC is a very specialized set of routines for the 7094/7044 DCS and 360/67 systems. If microfilm plotting is not available at other computer installations, the subroutines used in plotting may readily be changed to routines which produce line printer plots or CALCOMP plots however.

The RAPIER program used an algorithm for the coefficient calculations that inverted the correlation matrix and then converted this to the \((X'X)^{-1}\) matrix to calculate \(\hat{b}\). After inspection of several test cases, it seemed that this method did not improve the accuracy of the calculation of \((X'X)^{-1}\). Thus it was dropped and NEWRAP inverts \(X'X\) directly.

As with the RAPIER report, only the statistics and mathematics necessary to explain the program capabilities will be presented along with illustrative input and output listings and listings of the programs.

**SYMBOLS**

- **B** matrix
- **b** vector (column)
- \(b_i\) true regression coefficient
- \(\hat{b}_i\) estimated regression coefficient
- \(b_0\) constant term
- \(b_1, \ldots, b_J\) unknown parameters
- **C** correlation matrix
- \(C_{ij}\) elements of **C**
- **D** indicator variable, equal to 0 if no \(b_0\) coefficient is estimated and equal to 1 if \(b_0\) is estimated
- **E(x)** expected value of \(x\) (i.e., mean of \(x\) over all possible values of \(x\))
- **e** vector of observed values minus predicted values
- **F_{a,d}** statistic distributed as variance ratio with \(a\) and \(d\) degrees of freedom
- \(f_j(z_1, \ldots, z_K)\) term of regression equation
- **H_0** statistical hypothesis to be tested
$H_1$ alternate hypothesis to be accepted if $H_0$ is judged to be false

$J$ number of coefficients estimated, excluding $b_0$

$K$ number of independent variables observed

$k$ number of segments or cells in range of possible studentized residuals

$LOF$ lack of fit

$M$ total number of independent and dependent variables

$MS(source)$ mean square due to source, where source is REG, RES, etc.

$N$ number of observations

$NPDEG$ pooled degrees of freedom for replication error

$N(\mu, \sigma^2)$ normal distribution with mean $\mu$ and variance $\sigma^2$

$R$ number of sets of replicates

$REG$ regression

$REP$ replication

$RES$ residual

$r_i$ number of replicates in set $i$

$S$ diagonal matrix

$S_c$ sum of squares correction if $D = 1$, and 0 if $D = 0$

$SSQ(source)$ sum of squares due to source, where source is REG, RES, etc.

$s_j$ elements of diagonal matrix

$TOT$ total

$t_n$ statistic distributed as Student's $t$ with $n$ degrees of freedom

$V(x)$ variance of $x$, expected value of $(x - E(x))^2$

$W,X$ matrices

$w,x$ vectors (column)

$X_s$ stationary point of estimated quadratic surface

$x(J)$ $x_J$

$x,j \quad \frac{1}{N} \sum_{i=1}^{N} x_{ij}$

$y$ vector (column)

$Z_i$ studentized residual
In multiple linear regression, a dependent or response variable \( Y \) (such as temperature or pressure) measured on an object or experiment is assumed to be correlated with a function of one or more other variables \((z_1, \ldots, z_K)\) measured on the same object or experiment. This function includes a number of unknown parameters \((b_1, \ldots, b_J)\) and can be represented as

\[
y = h(b_1, \ldots, b_J, z_1, \ldots, z_K) + \epsilon
\]  

The only restriction imposed on this function is that it be linear in the parameters; that is, the function is of the form

\[
y = \sum_{j=1}^{J} b_j f_j(z_1, \ldots, z_K) + \epsilon
\]  

where \( f_j(z_1, \ldots, z_K) \) is a TERM of the regression equation. (A TERM is a quantity which may be a variable or a function of a variable, e.g., \( T \) is a TERM and \( Z \), after it is defined as \( Z = \log T \), is also a TERM.)

Suppose that there are \( N \) observations of the dependent variable. Let the subscript \( i \) indicate that the values are associated with the \( i^{th} \) observation; in particular, the value of the response variable \( y_i \) would depend on the observed values of the variables \((z_{i1}, \ldots, z_{iK})\). Also, let the subscript \( j \) denote the \( j^{th} \) term in the regression
model so that $x_{ij} = f_j(z_{i1}, \ldots, z_{iK})$ describes the transformations of the $z_{i1}, \ldots, z_{iK}$ to produce the value of $x_{ij}$ for the $j^{th}$ term at the $i^{th}$ observation.

The regression model can now be rewritten as

$$y_i = b_1 x_{i1} + b_2 x_{i2} + \ldots + b_J x_{iJ} + \epsilon_i \quad i = 1, \ldots, N$$  \hspace{1cm} (3)

where $\epsilon_i$ denotes the difference between the observed value and the expected value of $y_i$. For the $N$ observations, it is convenient to write this regression model in matrix notation as $y = Xb + \epsilon$ where

$$y = \begin{pmatrix} y_1 \\ \vdots \\ y_N \end{pmatrix}, \quad X = \begin{pmatrix} x_{11} & \cdots & x_{1J} \\ \vdots & \ddots & \vdots \\ x_{N1} & \cdots & x_{NJ} \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ \vdots \\ b_J \end{pmatrix}, \quad \epsilon = \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_N \end{pmatrix}$$  \hspace{1cm} (4)
More often than not, the analyst feels the following model is more appropriate:

\[ y_i = b_0 + b_1 x_{i1} + \ldots + b_J x_{iJ} + \epsilon_i \quad i = 1, \ldots, N \]  

(5)

Let \( a_0 = b_0 + b_1 \bar{x}_{1.1} + \ldots + b_J \bar{x}_{J.1} \). Then, the result of adding this equation to and subtracting it from equation (5) and rearranging the terms is

\[ y_i = (b_0 + b_1 \bar{x}_{1.1} + \ldots + b_J \bar{x}_{J.1}) \]

\[ + b_1(x_{i1} - \bar{x}_{1.1}) + \ldots + b_J(x_{iJ} - \bar{x}_{J.1}) + \epsilon_i \quad i = 1, \ldots, N \]  

(6)

If, then, a dummy variable \( x_0 \) is introduced such that, for all values of \( i \), \( x_{i0} = 1.0 \), equation (6) may be written as

\[ y_i = a_0 x_{i0} + b_1(x_{i1} - \bar{x}_{1.1}) + \ldots + b_J(x_{iJ} - \bar{x}_{J.1}) + \epsilon_i \quad i = 1, \ldots, N \]  

(6a)

Equation (6a) now resembles equation (3) and may be written in matrix notation, similar to equation (4), as \( y = Xb + \epsilon \) where now

\[ y = \begin{pmatrix} y_1 \\ \vdots \\ y_N \end{pmatrix} \]

\[ x = \begin{pmatrix} 1.0 & x_{11} - \bar{x}_{1.1} & \ldots & x_{1J} - \bar{x}_{J.1} \\ 1.0 & x_{21} - \bar{x}_{1.1} & \ldots & x_{2J} - \bar{x}_{J.1} \\ \vdots & \vdots & \ddots & \vdots \\ 1.0 & x_{N1} - \bar{x}_{1.1} & \ldots & x_{NJ} - \bar{x}_{J.1} \end{pmatrix} \]

\[ b = \begin{pmatrix} a_0 \\ b_1 \\ \vdots \\ b_J \end{pmatrix} \]

\[ \epsilon = \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_N \end{pmatrix} \]  

(7)
ESTIMATING $b$

Equations (4) and (7) are similar in form and for $N > J$ are an overdetermined set of linear equations. There will be some vector $\hat{b}$ which is a "best" vector to use. If the vector $\epsilon$ is composed of random variables $\epsilon_i$ such that $E(\epsilon_i) = 0$, $V(\epsilon_i) = \sigma^2 < +\infty$, and the $\epsilon_i$ are uncorrelated, then as is well known, the method of least squares gives the linear minimum variance unbiased estimators $\hat{b}$ for $b$. And $\hat{b}$ is given by

$$\hat{b} = (X'X)^{-1}X'y$$

(8)

The matrix $X'X$ divided by $N - 1$ is called the moment matrix of the experiment. The variance-covariance matrix of $\hat{b}$ is given by

$$V(\hat{b}) = \sigma^2(X'X)^{-1}$$

(9)

It is important to note that when the form of equation (7) is used, $X'X$ is given by

$$X'X = \begin{pmatrix}
N & 0 & \ldots & 0 \\
0 & \sum_{1}^{N} (x_{i1} - \bar{x}_1)^2 & \ldots & \sum_{1}^{N} (x_{i1} - \bar{x}_1)(x_{iJ} - \bar{x}_J) \\
0 & \ldots & \ldots & \ldots \\
0 & \sum_{1}^{N} (x_{i1} - \bar{x}_1)(x_{iJ} - \bar{x}_J) & \ldots & \sum_{1}^{N} (x_{iJ} - \bar{x}_J)^2
\end{pmatrix}$$

(10)

This is seen to be symmetric and of the form

$$X'X = \begin{pmatrix}
A & 0 \\
0 & B
\end{pmatrix}$$

Hence,

$$(X'X)^{-1} = \begin{pmatrix}
A^{-1} & 0 \\
0 & B^{-1}
\end{pmatrix}$$
NEWRAP uses this relation to advantage by storing only the lower triangular part of B and computing only the coefficients $b_1$, \ldots, $b_J$ by matrix manipulations. Then $b_0$ is given by the simple equation

$$b_0 = \bar{y} - \hat{b}_1 \bar{x}_1 - \hat{b}_2 \bar{x}_2 - \cdots - \hat{b}_J \bar{x}_J$$

where $\bar{y} = \sum y_i / N = \hat{a}_0$. It can also be shown that

$$V(\hat{b}_0) = V(\bar{y}) + V(\hat{b}'\bar{x}) = \left[ \frac{1}{N} + \bar{x}'(X'X)^{-1} \bar{x} \right] \sigma^2$$

$$\text{COV}(\hat{b}_0, \hat{b}) = -(X'X)^{-1} \bar{x} \sigma^2$$

When there is no $b_0$ term in the regression model,

$$X'X = \begin{bmatrix}
\sum_{i=1}^{N} x_{i1}^2 & \sum_{i=1}^{N} x_{i1}x_{i2} & \cdots & \sum_{i=1}^{N} x_{i1}x_{iJ} \\
\sum_{i=1}^{N} x_{i2}x_{i1} & \sum_{i=1}^{N} x_{i2}x_{i2} & \cdots & \sum_{i=1}^{N} x_{i2}x_{iJ} \\
\vdots & \vdots & \ddots & \vdots \\
\sum_{i=1}^{N} x_{iJ}x_{i1} & \sum_{i=1}^{N} x_{iJ}x_{i2} & \cdots & \sum_{i=1}^{N} x_{iJ}x_{iJ}
\end{bmatrix}$$

Comparing this to equation (10) shows this form of $X'X$ to be similar to the lower right submatrix in equation (10). This similarity is used to simplify notation by assuming that $X'X$ represents either the form of equation (12) or the lower right portion of equation (10) and considering the calculation of $b_0$ as a special case. Thus, further reference to $b$ implies

$$b = \begin{pmatrix}
b_1 \\
\vdots \\
b_J
\end{pmatrix}$$

There are two different methods of computing the regression coefficients which may
be used in NEWRAP. The first method uses bordering (ref. 7) on the full $X'X$ matrix. If $X'X$ is a nearly singular matrix, there may be problems with accuracy resulting in overflows or underflows causing execution to terminate without any results being printed. The second method uses a method of bordering which enters one term at a time into the model equation. After each term is entered, a full regression analysis is printed. Typically, if $X'X$ is nearly singular, a number of terms will have been added to the model before the results become unreliable or cause execution to be terminated. Thus, at least a partial analysis of the full model is available to aid in selection of further models to submit. After all the terms have been entered, the program then switches to the procedure which inverts the appropriate full $X'X$ matrix at each stage for further analyses.

The use of the bordering method leads to a large volume of printed output and is not recommended as a standard procedure. Through use of CRSPLT as a preregression analysis program it may be easier to determine if bordering should be used. CRSPLT can also help indicate the order of arrangement of the terms of the model so that those thought to be most important can be entered into the model first.

Also note that the individual observations may be weighted to perform a weighted regression analysis. NEWRAP permits the use of weights (ref. 5). In this case, the $X'X$ and $X'y$ matrices take the following form:

$$X'X = \begin{pmatrix}
\sum_{i=1}^{N} [(x_{i1} - \bar{x}_1)^2 w_i] & \ldots & \sum_{i=1}^{N} [w_i (x_{i1} - \bar{x}_1) (x_{iJ} - \bar{x}_J)] \\
\ldots & \ldots & \ldots \\
\sum_{i=1}^{N} [(x_{iJ} - \bar{x}_J) (x_{i1} - \bar{x}_1) w_i] & \ldots & \sum_{i=1}^{N} [(x_{iJ} - \bar{x}_J)^2 w_i]
\end{pmatrix}$$

$$X'y = \begin{pmatrix}
\sum_{i=1}^{N} x_{i1} y_i w_i \\
\ldots \\
\sum_{i=1}^{N} x_{iJ} y_i w_i
\end{pmatrix}$$

(12a)
CORRELATION MATRIX

Another matrix of interest both computationally and statistically in the correlation matrix $C$. The elements of $C$, which are denoted $C_{ij}$, are the sample correlation coefficients between the terms $X_i$ and $X_j$. These are

$$
C_{ij} = \sum_{l=1}^{N} \frac{(x_{l1} - \bar{x}_1)(x_{lj} - \bar{x}_j)}{\sqrt{\sum_{k=1}^{N} (x_{ki} - \bar{x}_i)^2 \sum_{k=1}^{N} (x_{kj} - \bar{x}_j)^2}}
$$

and all these numbers are between 1.0 and -1.0.

The calculation of $C$ can be expressed in matrix notation conveniently by defining a diagonal matrix $S = \text{diag}(s_1, s_2, \ldots, s_J)$ with elements

$$
s_j = \frac{1.0}{\sqrt{(X'X)_{jj}}} \quad j = 1, \ldots, J
$$

Then

$$
C = S(X'X)S
$$

It may also be that the independent variables are random variables. Then $X'X$ divided by $N - 1$ represents the sample variance-covariance matrix and $C$ the sample correlation matrix. If the independent variables are considered to be from a multivariate distribution, it is useful in some cases to consider the eigenvalues and eigenvectors of $X'X$. For these reasons, NEWRAP includes options to compute and print these quantities. These may also be computed and printed through use of the CRSPLT program.

ESTIMATING $\sigma^2$

For any regression model $y = Xb + \epsilon$, there are possibly two methods of estimating $\sigma^2$. First, if the assumed regression model is, in reality, the true model, it is well known that an unbiased estimator is given by

$$
\hat{\sigma}^2 = \frac{1}{N} \sum_{i=1}^{N} (y_i - X_i b)^2
$$
Second, where there are replicated data points, another estimator of $\sigma^2$, depending only on $V(\epsilon_i) = \sigma^2$ for all $i$ and not on the validity of the assumed model, is the pooled mean squares computed from the replicated data points.

Assume the observations are grouped into replicate sets in sequence. Let $R$ be the number of sets of replicates and $r_i$ be the number of replicates in the $i^{th}$ replicate set. Let

$$SSQ(i) = \sum_{k=r^*+1}^{r^*+r_i} (y_{ik} - \bar{y}_i)^2$$

(17)

where

$$r^* = \sum_{j=1}^{i-1} r_j$$

It is assumed $y_n$ is from the $i^{th}$ replicate set and $\bar{y}_i$ is calculated only from those $y_n$ in the $i^{th}$ replicate set. Then define the pooled sum of squares due to replication as

$$SSQ(REP) = \sum_{i=1}^{R} SSQ(i)$$

and the pooled degrees of freedom as $NPDEG = \sum_{i=1}^{R} (r_i - 1)$. The second estimator of $\sigma^2$ becomes

$$\sigma^2_{REP} = \frac{SSQ(REP)}{NPDEG}$$

(18)

$$= MS(REP)$$

It can be shown (ref. 5, p. 26) that the sums of squares due to residuals can be partitioned into a component due to replication and a component due to lack of fit; that is,

$$SSQ(RES) = SSQ(LOF) + SSQ(REP)$$

(19)
HYPOTHESIS TESTING

NORMALITY OF $\epsilon$

As stated before, the only assumption necessary for $\hat{b}$ to be a linear minimum variance unbiased estimator is that $E(\epsilon_i) = 0.0$, $V(\epsilon_i) = \sigma^2 < +\infty$, and $\epsilon_i$ be uncorrelated. If it can further be assumed that $\epsilon_i \sim N(0, \sigma^2)$, a number of standard tests become available. NEWRAP computes a chi-squared statistic which can be used as an approximate test.

Under the hypothesis $\epsilon_i \sim N(0, \sigma^2)$, the studentized residuals defined by

$$Z_i = \frac{y_i - \hat{y}_i}{\hat{\sigma}} = \frac{e_i}{\hat{\sigma}}$$

will be distributed as Student's $t$ with the degrees of freedom associated with the estimate $\sigma$. If the degree of freedom is 30 or more, the $t$ distribution is very close to the normal.

The range of possible studentized residuals is $(-\infty, +\infty)$ and may be divided into $k$ segments or cells each with probability $p_i$, so that each segment will have $Np_i$ as the expected number of observations falling into it. Let $n_i$ denote the number of studentized residuals in the $i^{th}$ cell. Then a chi-squared goodness-of-fit statistic may be calculated as

$$\chi^2_{k-1} = \sum_{i=1}^{k} \frac{(n_i - Np_i)^2}{Np_i}$$

NEWRAP computes this statistic by using an even number of cells greater than or equal to four and less than or equal to 20, such that the expected numbers of observations per cell is five or more. This statistic is not computed when there are less than 20 observations. The bounding values for the $i^{th}$ cell are $Z_{i-1}, Z_i$ where $F(Z_i) = (i \cdot k)/N$ and $F(Z)$ is the cumulative normal distribution function. Then each cell has the same expected number of observations, say $f = N/k$. Then
There is a point to be made concerning the chi-squared calculations. The validity of the use of the chi-squared statistic in a test depends upon the residuals forming a sample of independent and identically distributed random variables. This is not usually the case for regression residuals. Although the tail probabilities of the chi-squared tests might be in error, they should still be able to tell the statistician whether one intended normalizing transformation was more successful than another.

### ANALYSIS OF VARIANCE TABLE

For most hypothesis testing of the regression model, it is convenient to summarize the available information in an Analysis of Variance (ANOVA) table, as follows:

<table>
<thead>
<tr>
<th>Source</th>
<th>Sums of squares</th>
<th>Degrees of freedom</th>
<th>Mean squares</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regression (REG)</td>
<td>SSQ(REG) = ( b'X'y - S_c^a )</td>
<td>J</td>
<td>MS(REG) = SSQ(REG) J</td>
</tr>
<tr>
<td>Residual (RES)</td>
<td>SSQ(RES) = ( y'y - b'X'y )</td>
<td>N - J - D^b</td>
<td>MS(RES) = SSQ(RES) (N - J - D)</td>
</tr>
<tr>
<td>Total</td>
<td>SSQ(TOT) = ( y'y - S_c )</td>
<td>N - D</td>
<td></td>
</tr>
</tbody>
</table>

\[
S_c = \begin{cases} 0 & \text{if no } b_0 \text{ coefficient is estimated.} \\
Ny^2 & \text{if a } b_0 \text{ coefficient is estimated.}
\end{cases}
\]

\[
b_D = \begin{cases} 0 & \text{if no } b_0 \text{ coefficient is estimated.} \\
1 & \text{if } b_0 \text{ is estimated.}
\end{cases}
\]

If there are replicated data points, another ANOVA table can be constructed to show the separation of the residual sums of squares into components from lack of fit and replication, as in the following table:
As mentioned previously, there are two possible methods of estimating \( \sigma^2 \) depending on whether there are replicated data points. This is true for any given model equation. When the backward rejection option of NEWRAP is used, there is no longer one hypothetical model but a series of models. Thus, there is the choice of estimator for \( \sigma^2 \) to be made after each rejection of a term in the previous model.

As an example, consider the model

\[
y = b_0 + b_1 x_1 + b_2 x_2 + b_3 x_3 + \epsilon
\]

with replicated data points. The first step is to estimate \( b_0, b_1, b_2, \) and \( b_3 \). There will then be the estimators \( \hat{\sigma}^2_{\text{RES}(J)} \) and \( \hat{\sigma}^2_{\text{REP}} \). If the model in equation (20) has not left out any important terms, \( \hat{\sigma}^2_{\text{RES}(J)} \) as well as \( \hat{\sigma}^2_{\text{REP}} \) is a valid estimator.

The ratio \( F = \frac{\text{MS(LOF)}}{\text{MS(REP)}} \) can be used to test the hypothesis that there is no lack of fit, where \( F \sim F_a,d \) with \( a = N - J - D - \text{NPDEG} \) and \( d = \text{NPDEG} \) degrees of freedom. If the test accepts the hypothesis of no lack of fit, \( \text{MS(RES)} \) is a pooled estimate of \( \sigma^2 \) with more degrees of freedom. But there is the possibility that the hypothesis was accepted as a result of random fluctuation when there really is some lack of fit; that is, there is the possibility that \( \hat{\sigma}^2_{\text{RES}(J)} \) is a biased estimator. If lack of fit is not concluded to be significant, the decision to pool or not is usually made on the basis of the number of degrees of freedom for replication. If this is "large" (no definition of large is given herein), \( \hat{\sigma}^2_{\text{REP}} \) is used. If "small," the pooled estimate \( \hat{\sigma}^2_{\text{RES}(J)} \) is used.

In testing equation (20), should it be decided that \( b_3 \) is not significantly different from zero (see section \( t \)-TESTS), the coefficients of the following model would be estimated:

\[
y = b_0 + b_1 x_1 + b_2 x_2 + \epsilon
\]
From this model there is an estimate $\hat{\sigma}^2_{\text{RES}(J-1)}$. This estimate could also be biased since $b_3$ may be small but nonzero and the decision of $b_3 = 0$ may have been due to the low power of the test.

At the first step, the lack of fit can be considered a random sample of an infinite possibility of biases. But the biases due to pooling mean squares after rejecting terms can be considered to be systematic biasing. In such a case the use of Cochran’s test for "the largest of a set of estimated variances as a fraction of their total" might be appropriate.

NEWRAP provides three strategies of pooling estimates for use in the decision procedure:

1. Never pool. This is usable only when there are replicated data points. The estimator used in all t-tests is $\hat{\sigma}^2_{\text{REP}}$.
2. Pool initial residual. This will pool the lack of fit and replication (if any) from the first model only. Additional mean squares due to rejected terms will be ignored.
3. Always pool. This strategy will always use $\hat{\sigma}^2_{\text{RES}(J-i)}$ for the model with $i$ rejected terms.

Wherever a $\hat{\sigma}$ or $\hat{\sigma}^2$ is indicated, NEWRAP always uses the value calculated according to the strategy chosen by the user.

**TEST OF OVERALL REGRESSION**

One of the first tests usually applied to a regression model is the test of the overall significance of the model. In the notation of hypothesis testing this is stated $H_0: b = 0$; $H_1: b \neq 0$ where

\[
\begin{pmatrix}
  b_1 \\
  . \\
  . \\
  . \\
  b_J
\end{pmatrix}
\]

The statistic for this test is $F = \text{MS(REG)}/\hat{\sigma}^2$. The $F \sim F_{a, d}$ with $a = J - D$, and $d$ equals the degrees of freedom associated with $\hat{\sigma}^2$.

Another useful statistic for judging the significance of overall regression is $R^2 = \frac{\text{SSQ(REG)}}{\text{SSQ(TOT)}}$. The sampling distribution of $R$ does not lend itself to very simple tests except in the case of $H_0: b = 0$. The main value of $R^2$ is that it is a
number in the range 0 to 1 and $100 \, R^2$ is a measure of the percentage of variation in the $y$ values that is accounted for by the regression model.

\section*{t-TESTS}

In many cases, the regression model contains terms whose estimated coefficients are "small." This may be an indication that the term does not have a real effect on the dependent variable and that the estimate is nonzero due to random sampling variation. If this is true, it is desirable to delete the term from the regression model. A test statistic for deciding this is

\begin{equation}
    t = \frac{\hat{b}_i}{\sqrt{\widehat{\sigma}^2 (X'X)^{-1}}}_{ii}
\end{equation}

where $(X'X)^{-1}_{ii}$ denotes the $i^{th}$ diagonal element of the $(X'X)^{-1}$ matrix. The statistic $t \sim t_{N-J-D}$. An equivalent test statistic is

\begin{equation}
    F = \frac{b_i^2}{\widehat{\sigma}^2 (X'X)^{-1}}_{ii}
\end{equation}

where $F \sim F_{1, \, N-J-D}$. This is often referred to (ref. 5) as the partial F-test. The quantity $\frac{\hat{b}_i^2}{\left[(X'X)_{ii}^{-1}\right]}$ is called the additional sum of squares due to $b_i$, if $x_i$ were last to enter the equation. NEWRAP computes and prints the $t$-statistics, the probability associated with the interval $(-t, t)$, and the additional sums of squares for each term.

This particular test is the basis for the rejection option of NEWRAP. The analyst initially chooses which $\widehat{\sigma}^2$ estimator to use by the choice of strategy. Then the analyst may choose a confidence level which all coefficients must meet. For example, suppose a confidence level of 0.900 is chosen. The $t$-statistic is then computed for each coefficient, and the coefficient with minimum $|t|$ is identified. If $\min |t| > t_{N-J-D}, 0.950$, all terms are concluded to be significant at the 0.1 level (or 90.0 percent level of confidence). If $\min |t| < t_{N-J-D}, 0.950$, the term corresponding to the minimum $|t|$ is dropped from the hypothetical model, and the regression is recomputed. This process is repeated until all remaining coefficients are significant at the specified level of probability. This procedure can be overridden by an option which allows certain specified terms of the model to be retained regardless of the significance of the coefficient. Ken-
nedy and Bancroft (ref. 8) present a study indicating the backward deletion method is slightly more efficient than forward selection in special situations.

**PREDICTING VALUES FROM ESTIMATED REGRESSION EQUATION**

Regression equations are often used to predict an estimated response at some condition of the independent variables. Useful estimates of parameters to know are the variance of the regression equation and the variance of a single further observation at the desired combination of the independent variables.

Let \( x' = (x_1, \ldots, x_J) \) denote the vector of independent variables at which a prediction is desired. Let \( x^* = x - \bar{x} \). Let \( \hat{\sigma}^2_{\mu \cdot x} \) denote the estimated variance of the regression equation at \( x \). Let \( \hat{\sigma}^2_{y \cdot x} \) denote the estimated variance of a single further observation at \( x \). Then,

\[
\begin{align*}
\hat{\sigma}^2_{\mu \cdot x} &= \hat{\sigma}^2 \left[ \frac{D}{N} + x^*(X'X)^{-1} x^* \right] \\
\hat{\sigma}^2_{y \cdot x} &= \hat{\sigma}^2 \left[ 1.0 + \frac{D}{N} + x^*(X'X)^{-1} x^* \right]
\end{align*}
\]

where, as before, \( D = 1 \) if a \( b_0 \) coefficient is estimated and \( D = 0 \) if a \( b_0 \) coefficient is not estimated. The quantity \( s = \hat{\sigma}_{\text{RES}(J)} \) is called the standard error of estimate and often is used as a simple approximation to \( \hat{\sigma}_{y \cdot x} \). This approximation is close if \( N \) is very large and \( x = \bar{x} \), in which case,

\[
\hat{\sigma}^2_{y \cdot \bar{x}} = s^2 \left( 1.0 + \frac{D}{N} \right) \approx s^2
\]

When \( x \neq \bar{x} \), this may be a poor approximation. NEWRAP accepts input vectors \( x \) and computes \( \hat{y} = \hat{b}_0 + \hat{b}_1 x_1 + \ldots + \hat{b}_J x_J \), as well as \( \hat{\sigma}^2_{\mu \cdot x}, \hat{\sigma}^2_{\mu \cdot x'}, \hat{\sigma}^2_{y \cdot x}, \hat{\sigma}^2_{y \cdot x'} \) and the standard error of estimate.
The illustrative example is described in chapter 7 of reference 5. The data is reproduced in table I. Figure 1 presents this data in a sample input form.

The basic model to be fitted is

\[ y = b_0 + b_1 x_1 + b_2 x_2 + b_3 x_3 + b_4 x_4 \]  (25)

**TABLE I. - OBSERVED VALUES FOR EXAMPLE PROBLEM**

<table>
<thead>
<tr>
<th>Unit number</th>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
<th>( x_4 )</th>
<th>( y )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-75</td>
<td>0</td>
<td>0</td>
<td>-65</td>
<td>1.4</td>
</tr>
<tr>
<td>2</td>
<td>175</td>
<td>0</td>
<td>0</td>
<td>150</td>
<td>26.3</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>0</td>
<td>-65</td>
<td></td>
<td>29.4</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>0</td>
<td>165</td>
<td>-65</td>
<td>9.7</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>150</td>
<td>32.9</td>
</tr>
<tr>
<td>10</td>
<td>-75</td>
<td>-75</td>
<td>0</td>
<td>150</td>
<td>26.4</td>
</tr>
<tr>
<td>11</td>
<td>175</td>
<td>175</td>
<td>0</td>
<td>-65</td>
<td>8.4</td>
</tr>
<tr>
<td>14</td>
<td>-75</td>
<td>-75</td>
<td>-65</td>
<td>150</td>
<td>28.4</td>
</tr>
<tr>
<td>15</td>
<td>175</td>
<td>175</td>
<td>165</td>
<td>-65</td>
<td>11.5</td>
</tr>
<tr>
<td>18</td>
<td>0</td>
<td>0</td>
<td>-65</td>
<td>-65</td>
<td>1.3</td>
</tr>
<tr>
<td>19</td>
<td>0</td>
<td>0</td>
<td>165</td>
<td>150</td>
<td>21.4</td>
</tr>
<tr>
<td>20</td>
<td>0</td>
<td>-75</td>
<td>-65</td>
<td>-65</td>
<td>.4</td>
</tr>
<tr>
<td>21</td>
<td>0</td>
<td>175</td>
<td>165</td>
<td>150</td>
<td>22.9</td>
</tr>
<tr>
<td>24</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-65</td>
<td>3.7</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>-75</td>
<td>0</td>
<td>150</td>
<td>28.5</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>-75</td>
<td>0</td>
<td>150</td>
<td>23.4</td>
</tr>
<tr>
<td>16</td>
<td>0</td>
<td>-75</td>
<td>0</td>
<td>150</td>
<td>26.5</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>175</td>
<td>0</td>
<td>-65</td>
<td>5.8</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>175</td>
<td>0</td>
<td>-65</td>
<td>7.4</td>
</tr>
<tr>
<td>17</td>
<td>0</td>
<td>175</td>
<td>0</td>
<td>-65</td>
<td>5.8</td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>-75</td>
<td>-65</td>
<td>150</td>
<td>28.8</td>
</tr>
<tr>
<td>22</td>
<td>0</td>
<td>-75</td>
<td>-65</td>
<td>150</td>
<td>26.4</td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>175</td>
<td>165</td>
<td>-65</td>
<td>11.8</td>
</tr>
<tr>
<td>23</td>
<td>0</td>
<td>175</td>
<td>165</td>
<td>-65</td>
<td>11.4</td>
</tr>
</tbody>
</table>
### SAMPLE INPUT

| DATA IS FROM DRAPER AND SMITH, APPLIED REGRESSION ANALYSIS (REFERENCE 4 OF WRIGHT REPORT) |
| SAMPLE NEWRAP PROBLEM 2 |
| INITIAL MODEL EQUATION IS |
| $Y = B_0 + B_1 X_1 + B_2 X_2 + B_3 X_3 + B_4 X_4 + E$ |
| $Y =$ CHAMBER PRESSURE |
| $X_1 =$ TEMPERATURE OF CYCLE |
| $X_2 =$ VIBRATION LEVEL |
| $X_3 =$ DROP (SHOCK) |
| $X_4 =$ STATIC FIRE |
| RESIDUALS ARE BEING REQUESTED TO BE PUNCHED (FOR CRISPIT PROGRAM). |
| FOR RESIDUAL PLOTTING ANALYSES |

### TABLE 1

| SAMPLE INPUT (Concluded) |
| FORTRAN STATEMENT |

<table>
<thead>
<tr>
<th>UNIT NO.</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>75</td>
</tr>
<tr>
<td>11</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>75</td>
</tr>
<tr>
<td>12</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>75</td>
</tr>
<tr>
<td>13</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>75</td>
</tr>
<tr>
<td>14</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>75</td>
</tr>
<tr>
<td>15</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>75</td>
</tr>
<tr>
<td>16</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>75</td>
</tr>
<tr>
<td>17</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>75</td>
</tr>
<tr>
<td>18</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>75</td>
</tr>
</tbody>
</table>

### Figure 1

- Sample input form.

---

20
The preceding model requires no transformations of the tabulated data for the dependent or independent variables. Suitable input statements are also given in figure 1.

A subsequent example will illustrate the requirements on the input cards when transformations are involved.

**Detailed Description of Input Cards**

This section of the report describes the input cards as classified into nine sets according to table II.

<table>
<thead>
<tr>
<th>Set number</th>
<th>Name of set</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IDENTIFICATION</td>
<td>Identify and describe problem</td>
</tr>
<tr>
<td>2</td>
<td>PROBLEM SIZE</td>
<td>Define problem size</td>
</tr>
<tr>
<td>3</td>
<td>LOGIC</td>
<td>Specify general logical controls</td>
</tr>
<tr>
<td>4</td>
<td>MODEL</td>
<td>Define model equation</td>
</tr>
<tr>
<td></td>
<td>(a) MODEL SIZE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(b) TERMS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(c) TRANSFORMATIONS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(d) CONSTANTS</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>REJECTION</td>
<td>Backward rejection controls</td>
</tr>
<tr>
<td>6</td>
<td>REPLICATES</td>
<td>Identify replicated data</td>
</tr>
<tr>
<td>7</td>
<td>FORMAT</td>
<td>Give data format</td>
</tr>
<tr>
<td>8</td>
<td>DATA</td>
<td>Input observed data</td>
</tr>
<tr>
<td>9</td>
<td>PREDICTIONS</td>
<td>Predicted values data</td>
</tr>
</tbody>
</table>

The model equation is defined by set 4 of table II. An example illustrating the use of one blank card for input set 4 which can be used for simple linear regressions is presented by figure 1 and table I. A second example illustrating the use of the set of MODEL cards in the presence of prior constants and transformations will be given at the end of this section of the report. A pictorial representation of an input deck is given by figure 2.
Figure 2. Sample input deck. (Asterisk denotes the card is optional and its use depends upon data input on previous cards.)
Figure 2. - Concluded.
Details of the input cards are as follows:

1) **IDENTIFICATION** (I, IDENT)(I2, 13A6): IDENT is Hollerith data used to identify the problem. I indicates the number of additional cards to read for further identification or description (columns 1 to 78).

2) **PROBLEM SIZE** (NOVAR, NODEP, NOTERM, NOOB, NTKEEP)(3I4, I5, I4)

   - **NOVAR**: Number of input independent variables (number of $z$'s in eq. (2))
   - **NODEP**: Number of input dependent variables
   - **NOTERM**: Number of terms in model equation (number of $x$'s in eq. (3)). Note that $b_0$ is not counted as a term.
   - **NOOB**: Number of observations
   - **NTKEEP**: First NTKEEP independent terms of model equation will be retained in model regardless of significance level

3) **LOGIC**: One card with nine one-column fields

   - **BZERO**: $b_0$ term appears in model equation (T or F)
   - **IFTT**: t-statistics and their descriptive confidence levels are to be computed (T or F)
   - **IFWT**: Weight of 1.0 is applied to all observations (T or F). If this is F, see input sets 7 and 8 for further information.
   - **IFCHI**: Compute and plot residuals (T or F)
   - **STORXX**: Calculate eigenvalues and eigenvectors of $X'X$ (T or F)
   - **IFSSR**: Model shall be increased by one term at a time using bordering method for matrix inversion (T or F)
   - **ECONMY**: Use economy version of output (T or F). NEWRAP does not print $X'y$, $(X'X)^{-1}$, or $C$ when set to F.
   - **ISTRAT**: Pooling strategy is 1, 2, or 3:
     1. Never pool. Use replication error as estimate of error. If 1 is selected and no replication is found, strategy 3 is used.
     2. Pool initial residual only.
     3. Pool all residuals.
PUNCH      Punch residuals and predicted values (T or F). If T, observation number is punched and then residuals and predicted values are punched in (I6, 4E16.8/(6X, 4E16.8)) format in pairs (observation number, \( e_1, \hat{y}_1, e_2, \hat{y}_2 \), etc.).

(4) MODEL: The MODEL cards are used to manipulate the observed input data, supplied by input set 8, into the form of the desired model equation. There are four subsets of this input set 4, namely, MODEL SIZE, TERMS, TRANSFORMATIONS, and CONSTANTS, of which the latter three are used only in the development of complex models.

If a simple linear model is being analyzed, the MODEL SIZE card is left blank, indicating that the number of transformations is zero and the number of constants to be read in is zero. In this case, the TERMS, TRANSFORMATIONS, and CONSTANTS cards of this input set are not expected by the program, and the program assumes the independent and dependent variables are arranged on the input cards of input set 8 as

\[
x_1, x_2, \ldots, x_J, y_1, \ldots, y_{\text{NODEP}}
\]

where NODEP is the number of dependent variables.

If a weighting factor other than 1.0 is to be used (i.e., if item 3 of the LOGIC card contains an F), the value of the weighting factor for each observation must appear as the last item in the list, so that in this case the data for each observation is entered on the cards as

\[
x_1, x_2, \ldots, x_J, y_1, \ldots, y_{\text{NODEP}}, WT
\]

If the weighting factor is identically 1.0, NEWRAP reads a total of \( M \) numerical values for each observation, where \( M \) is the sum of the number of independent and dependent variables. The variables are stored consecutively in an array called \( X \), beginning with location 01 and ending with location \( M \). If the weighting factor is not identically 1.0, then \( M + 1 \) numerical values are read for each observation, but the last value, being the weighting factor, is treated and stored separately. The data in \( X \) are used with their appropriate weighting factors to cumulatively create \( X'X \) and \( X'y \) as shown in equations (12a).

The remaining discussion of this set explains the use of transformations and/or constants to build more complex models. Therefore, the reader who does not immediately need a complex model may skip this material and proceed directly to the description of input set 5.

As mentioned previously, there are up to four subsets of the MODEL cards. Their purpose is to give the structure of the model equation and thereby specify the initial
operations to be performed on the input data. As used here, CONSTANTS means any numerical value specified to be in the model equation in advance of parameter estimation. These numerical values are read from the CONSTANTS cards.

Also, the word TRANSFORMATIONS is to be interpreted as the operations performed on the input data (read from data cards) to compute the \( f_j \) values (eq. (2)) of the model equation. The structure of these functions (and of any transformations of the dependent variables) is read from the TRANSFORMATION cards. Finally, the word TERMS is to be interpreted as the computed results of the operations specified by the transformation (including any operations that leave the input data unchanged). The results of the TRANSFORMATIONS are stored in an array CON, and the TERMS cards designate the order of the relative locations in CON where the final values for the terms of the model equations are to be found.

The four subsets, MODEL SIZE, TERMS, TRANSFORMATIONS, and CONSTANTS, will be described in detail now. Also, at the end of the description of this input set, a summary of these cards, with the formats used, is given for convenience.

The MODEL SIZE card specifies NTRANS and KONNO(214) where

- **NTRANS**: Number of transformations that will be performed
- **KONNO**: Number of constants that will be read in which are required to specify model equation

If the number of transformations is zero, and therefore, the number of constants is zero, the TERMS, TRANSFORMATIONS, and CONSTANTS cards are not expected by the program. This being a simple linear model case, only the MODEL SIZE card, which can be blank, is necessary in this subset, but the values for the observations which are provided in input set 8 must conform to the order as specified in the first three paragraphs describing this input set.

When, however, a more complex model is desired, information must be supplied instructing the program as to (1) where to find the values for the TERMS of the equation, (2) how to create the terms from the variables and the constants, and (3) what the values of the constants are. This information is supplied on the TERMS, TRANSFORMATIONS, and CONSTANTS cards.

The numerical values to be used in the transformations are stored in two arrays called X and CON. The transformations always require that an operator (some value from CON) performs an operation (see table III) on an operand (some specified value from X) to produce a result which will be stored in CON. Thus CON serves two purposes. First, if the number of constants (KONNO) specified on the MODEL SIZE card is nonzero, that many constants will be read from the CONSTANTS cards and stored in CON beginning with location 01. If the number of constants is zero, a CONSTANTS card is not expected by the program. Second, all intermediate and final results of
### TABLE III. - OPERATIONS AND CODE NUMBERS

[X indicates a value from X and C a value from CON.]

<table>
<thead>
<tr>
<th>Operation code (OP)</th>
<th>Resulting operation</th>
<th>Operation code (OP)</th>
<th>Resulting operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>No operation</td>
<td>01</td>
<td>N + C</td>
</tr>
<tr>
<td>01</td>
<td>N + C</td>
<td>02</td>
<td>X*C</td>
</tr>
<tr>
<td>02</td>
<td>X*C</td>
<td>03</td>
<td>C/X</td>
</tr>
<tr>
<td>03</td>
<td>C/X</td>
<td>04</td>
<td>EXP(X)</td>
</tr>
<tr>
<td>04</td>
<td>EXP(X)</td>
<td>05</td>
<td>X**C</td>
</tr>
<tr>
<td>05</td>
<td>X**C</td>
<td>06</td>
<td>ALOG(X)</td>
</tr>
<tr>
<td>06</td>
<td>ALOG(X)</td>
<td>07</td>
<td>ALOG10(X)</td>
</tr>
<tr>
<td>07</td>
<td>ALOG10(X)</td>
<td>08</td>
<td>SIN(X)</td>
</tr>
<tr>
<td>08</td>
<td>SIN(X)</td>
<td>09</td>
<td>COS(X)</td>
</tr>
<tr>
<td>09</td>
<td>COS(X)</td>
<td>10</td>
<td>SIN(πX)</td>
</tr>
<tr>
<td>10</td>
<td>SIN(πX)</td>
<td>11</td>
<td>COS(πX)</td>
</tr>
<tr>
<td>11</td>
<td>COS(πX)</td>
<td>12</td>
<td>1.0/X</td>
</tr>
<tr>
<td>12</td>
<td>1.0/X</td>
<td>13</td>
<td>EXP(C/X)</td>
</tr>
<tr>
<td>13</td>
<td>EXP(C/X)</td>
<td>14</td>
<td>EXP(C,X**2)</td>
</tr>
<tr>
<td>14</td>
<td>EXP(C,X**2)</td>
<td>15</td>
<td>SQRT(X)</td>
</tr>
<tr>
<td>15</td>
<td>SQRT(X)</td>
<td>16</td>
<td>1.0, SQRT(X)</td>
</tr>
<tr>
<td>16</td>
<td>1.0, SQRT(X)</td>
<td>17</td>
<td>C**X</td>
</tr>
<tr>
<td>17</td>
<td>C**X</td>
<td>18</td>
<td>10.0**X</td>
</tr>
<tr>
<td>18</td>
<td>10.0**X</td>
<td>19</td>
<td>SINH(X)</td>
</tr>
<tr>
<td>19</td>
<td>SINH(X)</td>
<td>20</td>
<td>COSH(X)</td>
</tr>
<tr>
<td>20</td>
<td>COSH(X)</td>
<td>21</td>
<td>(1.0 - COS(X)), 2.0</td>
</tr>
<tr>
<td>21</td>
<td>(1.0 - COS(X)), 2.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>ATAN(X)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>ATAN2(X, C)</td>
<td>24</td>
<td>X**2</td>
</tr>
<tr>
<td>24</td>
<td>X**2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>X**3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>ARCSIN(SQRT(X))</td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>2.0*πX</td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>1.0, (2.0*πX)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>ERF(X)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>GAMMA(X)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>X*C</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

transformation are also stored in CON as specified on the TRANSFORMATION cards. The TERMS card then specifies which of the locations in CON finally contain the values needed to construct the X'X and X'y matrices. After all the transformations have been performed on an observation, the contents of the relative locations of the CON array specified on the TERMS card are moved back to X in consecutive locations beginning with location 01.

Note especially that CONSTANTS data are stored in CON from location 01 through KONNO. Thus, if a transformation specifies that a result is to be placed in any of these locations, the result will replace the constant, so that further operations on subsequent transformations would use the new value stored instead of the constant value to which it was initialized. Care should be taken, therefore, that the results of the transformations be stored in relative locations greater than KONNO.

Each transformation code is made up of four subfields of two card columns each, with the following interpretation:
Thus, subfield 1 always references the X array, and subfields 3 and 4 reference the CON array. The result of every transformation is a term which is stored in the designated location of the CON array, with the added feature that, if the term is stored in relative location 61 or beyond, it is also stored in the parallel location in the X array. This is illustrated by the arrows in figure 3. This feature allows successive transformations to be performed more easily.

The OP (operation codes) are tabulated in table III. The transformation with \( \text{OP} = 00 \) is simply an identity transformation. This transfers data from X to CON so that when terms are selected there is a value available in CON that can be moved back to X.

When there are no transformations, NEWRAP assumes the first \text{NOTERM} values on a data card are the independent variables and the last \text{NODEP} values are the depend-

<table>
<thead>
<tr>
<th>Subfield</th>
<th>Interpretation</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Operand</td>
<td>Relative location in X</td>
</tr>
<tr>
<td>2</td>
<td>Operation (OP)</td>
<td>Arithmetic operation</td>
</tr>
<tr>
<td>3</td>
<td>Operator</td>
<td>Relative location in CON</td>
</tr>
<tr>
<td>4</td>
<td>Result</td>
<td>Relative location in CON</td>
</tr>
</tbody>
</table>

Figure 3. Map of X and CON arrays. Data transferred into any location of CON array beyond location 60 are immediately duplicated in same relative location in X array.
ent variables. When transformations are used, this convention need not hold for the raw input data but instead holds for the terms on the TERMS card. Thus, the first NOTERM values input on the TERMS card indicate the locations of the CON array which correspond to the independent variables and the last NODEP values indicate the locations of the CON array which correspond to the dependent variables. If the analyst desires to force certain terms to remain in the model regardless of their significance, these terms must be the first terms of the model. Then if the input for NTKEEP of the PROBLEM SIZE card is not zero, the first NTKEEP terms of the model will be retained.

The complete sequence of MODEL cards and the formats used are summarized as follows:

(a) **MODEL SIZE** (NTRANS, KONNO)(2I4): NTRANS specifies the number of transformations required and KONNO the number of CONSTANTS involved. NTRANS may not be greater than 100 and KONNO not greater than 60. If NTRANS = 0, the following three sets are skipped and the program goes directly to input set 5.

(b) **TERMS** (4O12): One or more cards as necessary, using two-column fields to denote the relative locations of the CON array containing the final values for the terms to be used and the order in which they enter into the model equation. The number of terms used is specified on the PROBLEM SIZE card.

(c) **TRANSFORMATIONS** (4O12): As many cards as necessary containing up to 10 transformation instructions per card. Each transformation instruction is composed of four two-column fields. See table III for the list of available transformations.

(d) **CONSTANTS** (5E15.7): As many cards as necessary containing the number of CONSTANTS as specified by KONNO. Up to 60 CONSTANTS may be specified. If KONNO = 0, these cards are not expected by the program.

(5) **REJECTION** (DELETE, P)(L1, F3.3): If DELETE is set to T, the backward rejection option is used and the desired level of confidence is given by P. The P value is written without a decimal point so that a 95 percent confidence level is indicated by a 950, a 99.9 percent level as 999, and so forth.

(6) **REPLICATION** (REPS)(L1): If REPS is F, the program skips to set 7 and assumes there is no replicated data. If REPS is set to T, then more cards are read in 2014 format specifying:

IREP in the first field of the first card indicates the number of replicate sets.

NARAY in the second field of the first card and the remaining fields of this and succeeding cards consists of an array containing the number of observations in each of the replicate sets.

Note that it is not safe for the program to assume that all the data points for an experiment with the same levels of the independent variables are true replicates. Thus, the user must explicitly specify the truly replicated sets. NEWRAP does check that all
independent terms within a replicate set are the same. If not, the program stops. A nonreplicated data point is considered to be a group of size 1. Note that the data in table I are grouped to clearly indicate the replicated data points.

(7) FORMAT (INPUT, FMT)(I2, 13A6): INPUT specifies the unit number on which the input data is stored; and FMT supplies the format for reading it.

Note that, if a weighting factor other than 1.0 is to be used, its value will be read with each data point, and the format must allow for this.

The example from Draper and Smith (ref. 5) uses a weighting value of 1.0 for all data. The format is (12X, 5F6.0) since there are four independent and one dependent variable to be read. If a weighting value other than 1.0 is used, it must appear with every data point as the last value on the card. In such a case, the format could, for example, be (12X, 5F6.0, F10.3).

(8) DATA: Each observation consisting of the given z's and y's read by the execution of one READ statement. Thus, there will be at least one card for each observation. As mentioned previously, if the transformation option is not used, the program expects the first variables to be the independent variables, in the order in which they enter the model, followed by the dependent variables and then the weighting value if IFWT = . F. Otherwise, if transformations are used, the independent and dependent variables may be entered in any convenient order, because the TERMS card(s) will be needed to specify the order in which the values will enter the model equation. However, if IFWT = . F., the weighting value is still the last value supplied with each observation.

(9) PREDICTIONS (PREDCT)(L1): (DATA) If, with the program LOGIC (input set 3) card, a computation of residuals is requested by a T in card column 4, then predicted values of the dependent variables are computed for all the input values of the independent variables. In addition to these predicted values, predictions at other values of the independent variables might be desired. In PREDICTIONS input, one card with one column is used to indicate if these other predictions are desired (T or F). If this is F, a new case is started and the new case should start with input set 1 cards. If it is T, the following cards are read: One card with one four-column field specifying the number of predictions desired. This is followed by cards with the values of the independent variables at which predictions are desired. Only the final regression model is used, but the number of independent and dependent variables originally supplied on the PROBLEM SIZE data cards are read. All transformations indicated on the MODEL cards are performed. Then the proper terms are chosen by the program to correspond to the final model. Since the dependent variables are not needed in this part of the program, the numerical values for the dependent variables are dummy values and should be in the appropriate range so that when subroutines required for the transformations (e.g., ALOG, SQRT) use these values, abnormal exits will not occur.
**Illustrative Problem Requiring Transformations**

As an example of the MODEL cards usage consider the following. Suppose the model we are required to construct is

\[ \log_{10}(y + 273.15) = b_0 + b_1z_1 + b_2z_2 + b_3z_3 + b_4z_1z_2 + b_5z_1z_3 + b_6z_2z_3 + b_7z_1^2 + b_8z_2^2 + b_9z_3^2 \]

Thus, in terms of equation (2) we have

\[
\begin{align*}
x_1 &= z_1 \\
x_2 &= z_2 \\
x_3 &= z_3 \\
x_4 &= z_1z_2 \\
x_5 &= z_1z_3 \\
x_6 &= z_2z_3 \\
x_7 &= z_1^2 \\
x_8 &= z_2^2 \\
x_9 &= z_3^2 \\
y &= \log_{10}(y + 273.15)
\end{align*}
\]

Table IV shows a sequence of transformations which could be used to construct this model equation. Figure 4 shows how the MODEL cards describing this equation would appear on a FORTRAN data sheet. Figure 5 shows the X and CON array contents both before and after the transformations are performed upon one observation and the X array after the appropriate terms have been selected according to the TERMS card data.
<table>
<thead>
<tr>
<th>Transformation number</th>
<th>Operand</th>
<th>Operation</th>
<th>Operator</th>
<th>Result</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>01</td>
<td>00</td>
<td>00</td>
<td>11</td>
<td>$x_1 \rightarrow \text{CON}(11)$</td>
</tr>
<tr>
<td>2</td>
<td>01</td>
<td>00</td>
<td>00</td>
<td>61</td>
<td>$x_1 \rightarrow X(61), \text{CON}(61)$</td>
</tr>
<tr>
<td>3</td>
<td>02</td>
<td>00</td>
<td>00</td>
<td>12</td>
<td>$x_2 \rightarrow \text{CON}(12)$</td>
</tr>
<tr>
<td>4</td>
<td>02</td>
<td>00</td>
<td>00</td>
<td>62</td>
<td>$x_2 \rightarrow X(62), \text{CON}(62)$</td>
</tr>
<tr>
<td>5</td>
<td>03</td>
<td>00</td>
<td>00</td>
<td>13</td>
<td>$x_3 \rightarrow \text{CON}(13)$</td>
</tr>
<tr>
<td>6</td>
<td>03</td>
<td>00</td>
<td>00</td>
<td>63</td>
<td>$x_3 \rightarrow X(63), \text{CON}(63)$</td>
</tr>
<tr>
<td>7</td>
<td>61</td>
<td>02</td>
<td>61</td>
<td>17</td>
<td>$x_1^2 \rightarrow \text{CON}(17)$</td>
</tr>
<tr>
<td>8</td>
<td>62</td>
<td>02</td>
<td>62</td>
<td>18</td>
<td>$x_2^2 \rightarrow \text{CON}(18)$</td>
</tr>
<tr>
<td>9</td>
<td>63</td>
<td>02</td>
<td>63</td>
<td>19</td>
<td>$x_3^2 \rightarrow \text{CON}(19)$</td>
</tr>
<tr>
<td>10</td>
<td>61</td>
<td>02</td>
<td>62</td>
<td>14</td>
<td>$x_1x_2 \rightarrow \text{CON}(14)$</td>
</tr>
<tr>
<td>11</td>
<td>61</td>
<td>02</td>
<td>63</td>
<td>15</td>
<td>$x_1x_3 \rightarrow \text{CON}(15)$</td>
</tr>
<tr>
<td>12</td>
<td>62</td>
<td>02</td>
<td>63</td>
<td>16</td>
<td>$x_2x_3 \rightarrow \text{CON}(16)$</td>
</tr>
<tr>
<td>13</td>
<td>04</td>
<td>01</td>
<td>01</td>
<td>98</td>
<td>$y + 273.15 \rightarrow X(98), \text{CON}(98)$</td>
</tr>
<tr>
<td>14</td>
<td>98</td>
<td>07</td>
<td>00</td>
<td>20</td>
<td>$\log_{10}(y + 273.15) \rightarrow \text{CON}(20)$</td>
</tr>
</tbody>
</table>

Figure 4. - An example of MODEL cards.
Figure 5. - Arrays X and CON before and after transformations and terms selection for the example.
SAMPLE NEWRAP REPORT
DATA IS FROM GRAPPE AND SMITH
APPLIED REGRESSION ANALYSIS (REFERENCE 4 OF NEWRAP REPORT)
CHAPTER 7
INITIAL MODEL EQUATION IS
Y = CHAMBER PRESSURE
X1 = TEMPERATURE OF CYCLE
X2 = VIBRATION LEVEL
X3 = DROP HUCK
Y = 80 + 8X1 + 82X2 + 83X3 + 84X4 + ERR
RESIDUALS ARE BEING REQUESTED TO BE PUNCHLED (FOR CASPLT PROGRAM)
FOR RESIDUAL PLOTTING ANALYSES.

There is 80 to estimate
There are 10 replicate sets

1 1 1 1 1 1 1 1 1 1 3 3 2 2

12x.5fa.01
SAMPLE NEWRAP PROBLEM
TERMS OF THE EQUATION, OBSERVATION = 1
-75.00000 0 0 -65.00000 1.400000
** replicate set 1
---------------------------------------------
TERMS OF THE EQUATION, OBSERVATION = 2
-75.00000 0 0 150.0000 26.30000
** replicate set 2
---------------------------------------------
TERMS OF THE EQUATION, OBSERVATION = 3
0 0 -65.00000 150.0000 29.40000
** replicate set 3
---------------------------------------------
TERMS OF THE EQUATION, OBSERVATION = 4
0 0 0 0 165.0000 -65.00000 9.70000
** replicate set 4
---------------------------------------------
TERMS OF THE EQUATION, OBSERVATION = 5
0 0 0 0 150.0000 32.90000
** replicate set 5
---------------------------------------------
TERMS OF THE EQUATION, OBSERVATION = 6
-75.00000 -75.00000 0 150.0000 26.40000
** replicate set 6
---------------------------------------------
TERMS OF THE EQUATION, OBSERVATION = 7
-75.00000 -75.00000 0 -65.00000 8.400000
** replicate set 7
---------------------------------------------
TERMS OF THE EQUATION, OBSERVATION = 8
-75.00000 -75.00000 0 150.0000 28.40000
** replicate set 8
---------------------------------------------
TERMS OF THE EQUATION, OBSERVATION = 9
175.0000 175.0000 0 -65.00000 11.50000
** replicate set 9
---------------------------------------------
TERMS OF THE EQUATION, OBSERVATION = 10
0 0 -65.00000 -65.00000 1.300000
** replicate set 10
---------------------------------------------
TERMS OF THE EQUATION, OBSERVATION = 11
0 0 165.0000 150.0000 21.40000
** replicate set 11
---------------------------------------------
TERMS OF THE EQUATION, OBSERVATION = 12
0 0 -75.00000 -65.00000 -65.00000 0.400000
** replicate set 12
---------------------------------------------
TERMS OF THE EQUATION, OBSERVATION = 13
0 175.0000 165.0000 150.0000 22.90000
** replicate set 13
---------------------------------------------
TERMS OF THE EQUATION, OBSERVATION = 14
0 0 -65.00000 3.700000

There are 18 replicate sets
**REPLICATE SET 16 **************************************************************************

ITEMS OF THE EQUATION, OBSERVATION = 15
0 0 -75.0000 20 150.0000 26.50000
0 -75.0000 0 150.0000 23.40000
0 -75.0000 0 150.0000 26.50000

**REPLICATE SET 15 **************************************************************************

ITEMS OF THE EQUATION, OBSERVATION = 16
0 0 -75.0000 0 150.0000 26.50000
0 -75.0000 0 150.0000 23.40000
0 -75.0000 0 150.0000 26.50000

**REPLICATE SET 16 **************************************************************************

ITEMS OF THE EQUATION, OBSERVATION = 17
0 0 -75.0000 0 150.0000 26.50000
0 -75.0000 0 150.0000 23.40000
0 -75.0000 0 150.0000 26.50000

**REPLICATE SET 17 **************************************************************************

ITEMS OF THE EQUATION, OBSERVATION = 18
0 0 -75.0000 0 150.0000 26.50000
0 -75.0000 0 150.0000 23.40000
0 -75.0000 0 150.0000 26.50000

**REPLICATE SET 18 **************************************************************************

ITEMS OF THE EQUATION, OBSERVATION = 19
0 0 -75.0000 0 150.0000 26.50000
0 -75.0000 0 150.0000 23.40000
0 -75.0000 0 150.0000 26.50000

**REPLICATE SET 19 **************************************************************************

ITEMS OF THE EQUATION, OBSERVATION = 20
0 0 -75.0000 0 150.0000 26.50000
0 -75.0000 0 150.0000 23.40000
0 -75.0000 0 150.0000 26.50000

**REPLICATE SET 20 **************************************************************************

ITEMS OF THE EQUATION, OBSERVATION = 21
0 0 -75.0000 0 150.0000 26.50000
0 -75.0000 0 150.0000 23.40000
0 -75.0000 0 150.0000 26.50000

**REPLICATE SET 21 **************************************************************************

ITEMS OF THE EQUATION, OBSERVATION = 22
0 0 -75.0000 0 150.0000 26.50000
0 -75.0000 0 150.0000 23.40000
0 -75.0000 0 150.0000 26.50000

**REPLICATE SET 22 **************************************************************************

ITEMS OF THE EQUATION, OBSERVATION = 23
0 0 -75.0000 0 150.0000 26.50000
0 -75.0000 0 150.0000 23.40000
0 -75.0000 0 150.0000 26.50000

**REPLICATE SET 23 **************************************************************************

ITEMS OF THE EQUATION, OBSERVATION = 24
0 0 -75.0000 0 150.0000 26.50000
0 -75.0000 0 150.0000 23.40000
0 -75.0000 0 150.0000 26.50000

MEANS OF INDEP AND DEP VARIABLES
12.500000 33.333333 25.000000 42.500000 16.579167

X TRANSPOSE X MATRIX WHERE X IS DEVIATION FROM MEAN
ROW 1 105000.0
ROW 2 62500.00 263333.3
ROW 3 26250.00 115000.0 175000.0
ROW 4 -26875.00 -161250.0 -105000.0 277350.0

X TRANSPOSE Y MATRIX WHERE X AND Y ARE DEVIATIONS FROM MEAN
ROW 1 -1103.750
ROW 2 -12394.33
ROW 3 -2767.500
ROW 4 -25875.25

CORRELATION COEFFICIENTS
ROW 1 1.000000
ROW 2 0.375865 1.000000
ROW 3 0.194372 0.537706 1.000000
ROW 4 -0.157485 -0.525631 -0.225295 1.000000

THE FOLLOWING ARE EIGENVALUES OF X TRANSPOSE X MATRIX
489039.32 169438.49 99358.56 81547.348

THIS IS THE MODAL MATRIX OR MATRIX OF EIGENVECTORS. EIGENVECTORS ARE WRITTEN IN COLUMNS LEFT TO RIGHT IN SAME ORDER AS EIGENVALUES
ROW 1 0.117713 0.231943 0.504097
ROW 2 0.674835 0.220435 -0.671570
ROW 3 0.358164 0.666753 0.420793
ROW 4 -0.430269 0.682691 0.177159 -0.343240
SAMPLE \text{VAP PHBND}

EACH COLUMN CONTAINS THE COEFFICIENTS FOR ONE DEPENDENT TERM

\text{GENETIC Teil}

1 2 3 4

\text{REGRESSION COEFFICIENTS (B1,...,Bk)}

1 0.73117E+02 0.11135E-01 0.43458E-02 0.101374

\text{ANCOVA OF REGRESSION ON DEPENDENT VARIABLE}

\begin{tabular}{lrrr}
\hline
\text{SOURCE} & \text{SUMS OF SQUARES} & \text{DEGREES OF FREEDOM} & \text{MEAN SQUARES} \\
\hline
\text{REGRESSION} & 2483.66246 & 4 & 625.91615 \\
\text{EFFECTS} & 244.61668 & 8 & 30.57710 \\
\text{TOTAL} & 2728.27914 & 12 & 227.35659 \\
\hline
\end{tabular}

\text{R2 = 0.907179}

\text{ANCOVA OF LACK OF FIT}

\begin{tabular}{lrrr}
\hline
\text{SOURCE} & \text{SUMS OF SQUARES} & \text{DEGREES OF FREEDOM} & \text{MEAN SQUARES} \\
\hline
\text{LACK OF FIT} & 37.63749 & 13 & 2.90445 \\
\text{ACOMPARISON} & 1117.3571 & 6 & 186.22585 \\
\text{RESIDUAL} & 244.61668 & 19 & 12.87199 \\
\text{TOTAL} & 2728.27914 & 38 & 71.63941 \\
\hline
\end{tabular}

\text{R2 = 0.907179}

\text{SUMS OF SQUARES DUE TO EACH VARIABLE IF IT WERE LAST TO ENTER REGRESSION}

1 5.42731
2 14.52751
3 2.21554
4 1783.635

\text{STANDARD DEVIATION OF REGRESSION COEFFICIENTS (DERIVED FROM DIAGONAL ELEMENTS OF $(X^T X)^{-1}$ MATRIX)}

0 0.354589
1 0.55431E-02
2 0.40437E-02
3 0.29452E-02
4 0.32408E-02

\text{E(X TRANSPOSE X) INVERSE MATRIX}

\begin{tabular}{l}
\text{SSENTIAL MRTICES} \\
\hline
1 0.11135E-01 0.20220E-01 0.39529E-01 0.41825E-01 0.57615E-05 \\
2 -0.36204E-06 -0.42695E-06 -0.49849E-06 0.35720E-05 \\
3 0.22021E-06 0.41364E-06 0.48642E-06 0.57615E-05 \\
4 0.22021E-06 0.41364E-06 0.48642E-06 0.57615E-05 \\
\hline
\end{tabular}

\text{SAMPLE \text{VAP PHBND}}

\text{CALCULATE T STATISTICS}

\text{THE T STATISTICS CAN BE USED TO TEST THE REGRESSION COEFFICIENTS B(i).}

\text{UNDER NULL HYPOTHESIS THE INTERVAL (-T,T) WHERE T IS GIVEN ABOVE, HAS APPROX PROBABILITY LISTED BELOW.}

\begin{tabular}{lrrr}
\hline
\text{MINUS} & \text{T(i)} & \text{MINUS} & \text{T(i)} \\
\hline
1 3.856 & 3.968 & 3.685 & 3.999 \\
2 2.78262 & 2.999 & 2.851 & 2.999 \\
3 1.679 & 1.804 & 1.734 & 1.832 \\
4 1.101 & 1.187 & 1.107 & 1.187 \\
\hline
\end{tabular}

\text{THE DESIRABLE VALUE OF PROBABILITY IS 95.54 PERCENT}

\text{THE TERM AT 33 IS BLIND DELETED}
SAMPLE DATA 

EACH COLUMN CONTAINS THE COEFFICIENTS FOR ONE DEPENDENT TERM 
CONSTANT TERM (1.0) 

1 1.074939 
2 0.341725 
3 -0.791326 
4 0.341725 

ANOVA OF REGRESSION ON DEPENDENT VARIABLE 

<table>
<thead>
<tr>
<th>SOURCE</th>
<th>SUMS OF SQUARES</th>
<th>DEGREES OF FREEDOM</th>
<th>MEAN SQUARES</th>
</tr>
</thead>
<tbody>
<tr>
<td>REGRESSION</td>
<td>2460.34143</td>
<td>3</td>
<td>820.113808</td>
</tr>
<tr>
<td>RESIDUAL</td>
<td>251.258142</td>
<td>20</td>
<td>12.5629079</td>
</tr>
<tr>
<td>TOTAL</td>
<td>2711.60154</td>
<td>23</td>
<td></td>
</tr>
</tbody>
</table>

R SQUARED = SS(REGRESSION / SS(TOTAL)) = 0.907339 
R = 0.952544 

STANDARD ERROR OF ESTIMATE: 3.344419 

USING HOULING STRATEGY 1 THE ERROR MEAN SQUARE = 1,845,5995 WITH DEGREES OF FREEDOM = 6 

F = SS(REGRESSION/ SS(RESIDUAL)) = 444.37 COMPARE TO F (3, 6) 

ANOVA OF LACK OF FIT

<table>
<thead>
<tr>
<th>SOURCE</th>
<th>SUMS OF SQUARES</th>
<th>DEGREES OF FREEDOM</th>
<th>MEAN SQUARES</th>
</tr>
</thead>
<tbody>
<tr>
<td>LACK OF FIT</td>
<td>240.184813</td>
<td>14</td>
<td>17.1560971</td>
</tr>
<tr>
<td>REPLICAION</td>
<td>11.378571</td>
<td>6</td>
<td>1.8959991</td>
</tr>
<tr>
<td>RESIDUAL</td>
<td>251.258142</td>
<td>20</td>
<td>12.5629079</td>
</tr>
</tbody>
</table>

SOMES OF SQUARES DUE TO EACH VARIABLE IF IT WERE LAST TO ENTER REGRESSION 

| 1 | 5.053823 |
| 2 | 27.08253 |
| 4 | 1.894.079 |

STANDARD DEVIATION OF REGRESSION COEFFICIENTS (DERIVED FROM DIAGONAL ELEMENTS OF (X TRANSPOSE X) INVERSE MATRIX ) 

| 0 | 0.353984 |
| 1 | 0.479324 |
| 2 | 0.353984 |
| 4 | 0.353984 |

(X TRANSPOSE X)^-1 

| M | 1 | 0.111808E-04 |
| 2 | -0.34902E-05 | 0.675106E-05 |
| 3 | -0.714402E-05 | 0.342357E-05 | 0.564434E-05 |

SAMPLE N. AP PROBLEM 

CALCULATED T STATISTICS 

THE T STATISTICS CAN BE USED TO TEST THE REGRESSION COEFFICIENTS 

| 1.654794 |
| 3.430724 |
| 3.156764 |

UNDER NULL HYPOTHESIS, THE INTERVAL [-T,T] WHERE T IS GIVEN ABOVE, HAS APPROXIMATE PROBABILITY LISTED BELOW. 

MINUS 5% INDICATES PROB EXCEEDS .04999 

| 3.391 |
| 2 | 3.991 |
| 4 | 3.999 |

THE RESIDUAL VALUE OF PROBABILITY IS .053 PERCENT 

THE TERM AT IT IS BEING WANTED
SAMPLE 4 A RAP PROBLEM
EACH COLUMN CONTAINS THE COEFFICIENTS FOR ONE DEPENDENT TERM
CONSTANT TERM [1]
11.7471
REGRESSION COEFFICIENTS [B1,...,Bk]
2 0.1354936 = 0.1
4 0.103394

<table>
<thead>
<tr>
<th>Source</th>
<th>Sum of Squares</th>
<th>Degrees of Freedom</th>
<th>Mean Squares</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regression</td>
<td>2495.26760</td>
<td>2</td>
<td>1247.6381</td>
</tr>
<tr>
<td>Residual</td>
<td>256.311962</td>
<td>21</td>
<td>12.2053314</td>
</tr>
<tr>
<td>Total</td>
<td>2711.57958</td>
<td>23</td>
<td></td>
</tr>
</tbody>
</table>

R squared = SS(REGR) / SS(TOT) = 0.905676  R = 0.951565

Using pooling strategy 1 the error mean square = 1.8455595 with degrees of freedom = 6
F = MS(REGR) / MS(REPS) = 665.19

<table>
<thead>
<tr>
<th>Source</th>
<th>Sum of Squares</th>
<th>Degrees of Freedom</th>
<th>Mean Squares</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lack of Fit</td>
<td>245.238604</td>
<td>15</td>
<td>16.392401</td>
</tr>
<tr>
<td>Replication</td>
<td>11.073591</td>
<td>6</td>
<td>1.8455595</td>
</tr>
<tr>
<td>Residual</td>
<td>256.311962</td>
<td>21</td>
<td>12.2053314</td>
</tr>
<tr>
<td>Total</td>
<td>2711.57958</td>
<td>23</td>
<td></td>
</tr>
</tbody>
</table>

F = MS(Lack of Fit) / MS(REPS) = 8.859

SUMS OF SQUARES DUE TO EACH VARIABLE IF IT WERE LAST TO ENTER REGRESSION
2 1.26720
4 1.20156

STANDARD DEVIATION OF REGRESSION COEFFICIENTS (DERIVED FROM DIAGONAL ELEMENTS OF (X TRANSPOSE X)INVERSE MATRIX)
0 0.394975
2 0.3249383E-02
4 0.361448E-02

[X TRANSPOSE X] INVERSE MATRIX
ROW 1 0.509096E-05
ROW 2 0.3249383E-02
SAMPLE MEAN PROBLEM
CALCULATED T STATISTICS
THE T STATISTICS CAN BE USED TO TEST THE REGRESSION COEFFICIENTS B1(X).
4.12864
31.84463
UNDER NULL HYPOTHESIS THE INTERVAL [-T,T] WHERE T IS GIVEN ABOVE, HAS APPROX. PROBABILITY LISTED BELOW.
MINUS SIGN INDICATES P (T > |T|) EXCEEDS .999.
2 0.997
4 0.999
THE DESIRED VALUE OF PROBABILITY IS 95.0 PERCENT
SAMPLE MEAN PROBLEM
FOR EACH DEPENDENT TERM AND OBSERVATION IS PRINTED
OBSERVED (Y OBSERVED)
CALCULATED RESPONSE (Y CALC)
RESIDUAL (Y CALC - Y IDS)
STUDENTIZED RESIDUAL (T Z)

<table>
<thead>
<tr>
<th>Y OBSERVED</th>
<th>Y CALC</th>
<th>Y DIFF</th>
<th>STUDENTIZED</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.4000</td>
<td>4.5050</td>
<td>-3.1050</td>
<td>-2.9909</td>
</tr>
<tr>
<td>26.300</td>
<td>27.063</td>
<td>-0.763</td>
<td>-0.5014</td>
</tr>
<tr>
<td>29.400</td>
<td>27.063</td>
<td>2.3367</td>
<td>1.7200</td>
</tr>
</tbody>
</table>

38
| Y OBSERVED | 9.7000 |
| Y CALC   | 5.2950 |
| Y DIF    | 4.4050 |
| STUDENTIZ'D | 3.4191 |
| Y OBSERVED | 32.900 |
| Y CALC   | 27.493 |
| Y DIF    | 5.8067 |
| STUDENTIZ'D | 4.2954 |
| Y OBSERVED | 26.400 |
| Y CALC   | 25.293 |
| Y DIF    | 0.5286 |
| STUDENTIZ'D | 0.3729 |
| Y OBSERVED | 8.4000 |
| Y CALC   | 7.7650 |
| Y DIF    | 0.6350 |
| STUDENTIZ'D | 0.4527 |
| Y OBSERVED | 28.400 |
| Y CALC   | 25.893 |
| Y DIF    | 2.5066 |
| STUDENTIZ'D | 1.8831 |
| Y OBSERVED | 11.500 |
| Y CALC   | 10.7650 |
| Y DIF    | 3.7150 |
| STUDENTIZ'D | 2.7346 |
| Y OBSERVED | 1.3600 |
| Y CALC   | 5.6560 |
| Y DIF    | -3.7250 |
| STUDENTIZ'D | -2.7641 |

**SAMPLE NEWRAP PROBLEM**

| Y OBSERVED | 21.400 |
| Y CALC   | 27.083 |
| Y DIF    | -5.6633 |
| STUDENTIZ'D | -4.1668 |
| Y OBSERVED | 0.4500 |
| Y CALC   | 3.8851 |
| Y DIF    | -3.4351 |
| STUDENTIZ'D | -2.5654 |
| Y OBSERVED | 22.900 |
| Y CALC   | 29.793 |
| Y DIF    | -6.8932 |
| STUDENTIZ'D | -5.0761 |
| Y OBSERVED | 3.7000 |
| Y CALC   | 5.0550 |
| Y DIF    | -1.3550 |
| STUDENTIZ'D | -0.9974 |
| Y OBSERVED | 26.500 |
| Y CALC   | 25.093 |
| Y DIF    | 1.4066 |
| STUDENTIZ'D | 1.4466 |
| Y OBSERVED | 23.400 |
| Y CALC   | 25.893 |
| Y DIF    | -2.4934 |
| STUDENTIZ'D | -1.8354 |
| Y OBSERVED | 26.200 |
| Y CALC   | 29.793 |
| Y DIF    | -3.5932 |
| STUDENTIZ'D | -2.5066 |
| Y OBSERVED | 5.8000 |
| Y CALC   | 7.7930 |
| Y DIF    | -1.9930 |
| STUDENTIZ'D | -1.4461 |
| Y OBSERVED | 7.4000 |
| Y CALC   | 7.7930 |
| Y DIF    | -0.3930 |
| STUDENTIZ'D | -0.2034 |
| Y OBSERVED | 5.8000 |
| Y CALC   | 7.7930 |
| Y DIF    | -1.9930 |
| STUDENTIZ'D | -1.4461 |

**SAMPLE VWRAP PROBLEM**

| Y OBSERVED | 28.300 |
| Y CALC   | 25.693 |
| Y DIF    | 2.6066 |
| STUDENTIZ'D | 2.1346 |
NEWRAP DOCUMENTATION AND LISTINGS

The contents of this section include a flow chart of the program, a listing of the routines used in NEWRAP and their major functions, the call structure of the program, a dictionary of the program, and the listing.

General Mathematical and Logical Flow of Program

The flow of operation in NEWRAP is illustrated in figure 6.
Figure 6. - Flow chart for NEWRAP.
Routines and Their Major Functions

<table>
<thead>
<tr>
<th>FORTRAN name</th>
<th>Function of routine</th>
</tr>
</thead>
<tbody>
<tr>
<td>BORD</td>
<td>Inverts symmetric matrix of order ( n ) by adding bordering column to already inverted matrix of order ( n - 1 )</td>
</tr>
<tr>
<td>EIGEN</td>
<td>Computes eigenvalues and eigenvectors of input symmetric matrix</td>
</tr>
<tr>
<td>HIST</td>
<td>Prints histogram of residuals</td>
</tr>
<tr>
<td>INVXTX</td>
<td>Inverts symmetric matrix</td>
</tr>
<tr>
<td>LOC</td>
<td>When given row and column indices of symmetric matrix element, it computes location this element would have if only lower triangular part were stored as vector.</td>
</tr>
<tr>
<td>MATINV</td>
<td>Controls inversion process; computes regression coefficients; computes eigenvalues and eigenvectors of ( X'X ) if requested</td>
</tr>
<tr>
<td>MFIX</td>
<td>Prints ( X'X ) and computes and prints ( C )</td>
</tr>
<tr>
<td>NEWRAP</td>
<td>Executes overall problem control; computes replication error; controls deletion of variables when given results of ( t )-test</td>
</tr>
<tr>
<td>OUTPLT</td>
<td>Computes residuals at observed points and plots them. Compute chi-squared statistic</td>
</tr>
<tr>
<td>PREDCT</td>
<td>Computes predicted values, variances, and standard deviations of regression line and further observations at specified points</td>
</tr>
<tr>
<td>RECT</td>
<td>Writes rectangular matrix</td>
</tr>
<tr>
<td>RSTATS</td>
<td>Computes regression statistics and writes regression and lack-of-fit analysis of variance tables</td>
</tr>
<tr>
<td>SUMUPS</td>
<td>Constructs ( X'X ) and ( X'y ) matrices one observation at a time, in double precision</td>
</tr>
<tr>
<td>TRAN</td>
<td>Performs transformations</td>
</tr>
<tr>
<td>TRIANG</td>
<td>Writes lower triangular part of symmetric matrix</td>
</tr>
<tr>
<td>TTEST</td>
<td>Computes ( t )-statistics and their significance levels; determines which variable should be deleted</td>
</tr>
</tbody>
</table>

Call Structure of Program

The call structure of the program is illustrated in figure 7.
**Dictionary of Program**

<table>
<thead>
<tr>
<th>FORTRAN name</th>
<th>Mathematical symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACTDEV</td>
<td>$e_i$</td>
<td>Error in observation $i$; difference between observed and predicted response</td>
</tr>
<tr>
<td>B</td>
<td>$b$</td>
<td>Regression coefficients other than the constant</td>
</tr>
<tr>
<td>BO</td>
<td>$b_0$</td>
<td>Constant regression coefficient</td>
</tr>
<tr>
<td>BZERO</td>
<td></td>
<td>Logical variable set to T if constant $b_0$ coefficient should be in regression model</td>
</tr>
<tr>
<td>CHISQ</td>
<td>$\chi^2$</td>
<td>Chi-squared statistic</td>
</tr>
<tr>
<td>CON</td>
<td></td>
<td>Constants used in transformations, and results of transformations</td>
</tr>
<tr>
<td>DELETE</td>
<td></td>
<td>Logical variable set to T when deletion of terms is desired</td>
</tr>
<tr>
<td>DUMMY</td>
<td></td>
<td>Extra array used for plotting data</td>
</tr>
<tr>
<td>ECONMY</td>
<td></td>
<td>Logical variable indicating suppress printout of $X'X$, $X'X$ deviations, and C if T</td>
</tr>
<tr>
<td>ERRMS</td>
<td>$\sigma^2$</td>
<td>Estimate of $\sigma^2$ used in hypothesis tests</td>
</tr>
<tr>
<td>FMT</td>
<td></td>
<td>Variable input format</td>
</tr>
</tbody>
</table>
FMTTRI  Format for printing matrix
IDENT    First identification printed at top of each page
IDOUT    Original sequence number of each term relating reduced models to original model
IFCHI    Logical variable set to T if residual computations and plots are desired
IFSSR    Logical variable set to T if sequential regressions are desired
IFTT     Logical variable set to T if t-statistics are desired
IFWT     Logical variable set to T if all weights of observations are 1.0
INPUT    Input logical tape unit number for data
INPUT5   Set equal to 5 to denote input device is card reader
INTER    Tape unit where input data is stored for use in OUTPLT
IOUT     Sequence number of term among those remaining which is to be deleted
JCOL     Total number of independent and dependent terms in regression model
KONNO    Number of constants originally supplied for transformations
LENGTH   Number of locations in matrix storage area currently needed
LIST     Set equal to 6 to denote output device is printer
NARAY    Number of replications per replicate set
NCON     Array containing addresses in CON array for use in transformations
NERROR   Degrees of freedom for error mean square estimate
NLOF     \( N - J - NPDEG - D \) Degrees of freedom for estimating variance due to lack of fit
NODEP    Number of dependent variables
NOOB     Number of observations
<table>
<thead>
<tr>
<th>Variable</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOTERM</td>
<td>Number of terms in current regression model</td>
</tr>
<tr>
<td>NOVAR</td>
<td>Number of independent variables to be read</td>
</tr>
<tr>
<td>NPDEG</td>
<td>Pooled degrees of freedom for replication error</td>
</tr>
<tr>
<td>NREG</td>
<td>Degrees of freedom for determining variance due to regression</td>
</tr>
<tr>
<td>NRES</td>
<td>Degrees of freedom for estimation of residual variance</td>
</tr>
<tr>
<td>NTERM</td>
<td>Array containing locations of terms in CON array that should be in regression model</td>
</tr>
<tr>
<td>NTOT</td>
<td>Total degrees of freedom</td>
</tr>
<tr>
<td>NTRAN</td>
<td>Array containing transformation codes for use in performing transformations</td>
</tr>
<tr>
<td>NTRANS</td>
<td>Number of transformations to perform</td>
</tr>
<tr>
<td>NWHERE</td>
<td>Location in X array of first dependent variable; used in prediction routine to adjust for deleted terms</td>
</tr>
<tr>
<td>NXCOD</td>
<td>Array containing addresses of variables (or terms with address &gt;60) for use in transformations</td>
</tr>
<tr>
<td>P</td>
<td>Probability that interval (-t, t) must have before a term is considered to be significant</td>
</tr>
<tr>
<td>PNCH</td>
<td>Logical variable set to T if residuals are to be punched</td>
</tr>
<tr>
<td>POOLED</td>
<td>Array containing pooled sums of squares from replications for each dependent term</td>
</tr>
<tr>
<td>PREDCT</td>
<td>Logical variable set to T if prediction option is desired</td>
</tr>
<tr>
<td>REPS</td>
<td>Logical variable set to T if there are replicate sets in data</td>
</tr>
<tr>
<td>REPVAR</td>
<td>Array containing replication variance of each dependent term</td>
</tr>
<tr>
<td>RESMS</td>
<td>Array containing residual mean square or variance of each dependent term</td>
</tr>
<tr>
<td>RNLOF</td>
<td>Reciprocal of degrees of freedom for lack of fit</td>
</tr>
<tr>
<td>RNREG</td>
<td>Reciprocal of degrees of freedom for regression</td>
</tr>
</tbody>
</table>
RNRES  Reciprocal of degrees of freedom for residual
RWT  Reciprocal of total weight
SATRTD  Logical variable indicating that there are no degrees of freedom for residual if T
STORYX  Logical variable set to T if eigenvectors and eigenvalues of X'X are to be computed and printed
SUMX  \( \sum x, \sum y \)  Array containing sums of independent and dependent terms
SUMX2  \( \sum x^2, \sum y^2 \)  Array containing sum of squared independent and dependent terms
SUMXX  X'X  Sums of squares and crossproducts matrix, and variance-covariance matrix of independent terms
SUMXXI  \((X'X)^{-1}\)  Inverse of variance-covariance matrix of independent variables
SUMXY  X'y  Array containing sums of crossproducts of independent terms with dependent terms
TOTWT  \( w_i \)  Sum of weight of observations
X  Before transformations are performed, this contains the variables as read in. After transformations are performed, appropriate data from CON array are placed here according to information on TERMS cards.
XCHK  Array used in checking if all values of independent terms are the same within a replicate set
ZEAN  E(X), \( \mu \)  Expected or mean value (or X)

Program Listing

SIBFTC BLDV

BLOCK DATA
COMMON /FRMTS/ FMT(13),FMTTRI(14)
COMMON/BIG/B(60,9),SUMXY(60,9),SUMXX(1830),SUMXXI(1830)
X,DUMMY(2300)
DOUBLE PRECISION B,SUMXY,SUMXX,SUMXXI
COMMON/MED/BO(9),SUMX(69),SUMX2(69),SUMY(9),LEAN(69),
X CON(99),ERRMS(9),IDENT(13),IDOUT(60),NCON(200),NTERM(69),

46
COMMON /SMALL/ BYPASS, BZERO, DELETE, FIRST, IFCHI, IFSSR
X IFTT, IFWT, INPUT, INPUT5, INTER
X ISTRAT, JCOL, KONNO, LENGTH, LIST
X NERROR, NODEP, NOOB, NOTERM
X NOVAR, NPDEG, NKES, NTRANS, NWHERE
X P, PREDCT, REPS, RTW
X STORYI, STORYC, STORYX, TOTWT, WEIGHT
X ERRFXD, ECONMY, IOUT, ICOL

LOGICAL ECONMY
DOUBLE PRECISION RWT, TOTWT, WEIGHT
DATA INTER/3/, INPUT5/5/, LIST/6/
DATA (FMTRI(1), I=1,4)/6H5H, 6H6, (8G1, 6H5.6)/
COMMON/MAX/Maxplt

MAXPLT SHOULD BE THE NUMBER OF SINGLE LENGTH WORDS IN COMMON/BIG/
BEGINNING AT THE FIRST LOCATION OF SUMXY
DATA MAXPLT/10700/
END

$IBFTC NEWRAP

C
C THIS IS NEWRAP, MAIN PROGRAM FOR REGRESSION ANALYSIS PROVIDING
C INTERNAL EVALUATION OF RESULTS.
C******************************************************************************
C
C COMMON /FRMTS/ FMT(13), FMTRI(14)
COMMON /BIG/B60,9), SUMXY(60,9), SUMXX(1830), SUMXXI(1830)
DOUBLE PRECISION B, SUMXY, SUMXX, SUMXXI
COMMON /MED/B60,9), SUMX(69), SUMX2(69), SUMXX2(69), ZEAN(69),
X CON(99), ERRMS(9), IDENT(13), IOUT(60), NCON(1200), NTERM(69),
X NTRAN(100), NXCOD(100), POOLED(9), REPVAR(9), RESMS(9), X(99)
DOUBLE PRECISION B, SUMXY, SUMXX, SUMXXI
COMMON /SMALL/ BYPASS, BZERO, DELETE, FIRST, IFCHI, IFSSR
X IFTT, IFWT, INPUT, INPUT5, INTER
X ISTRAT, JCOL, KONNO, LENGTH, LIST
X NERROR, NODEP, NOOB, NOTERM
X NOVAR, NPDEG, NKES, NTRANS, NWHERE
X P, PREDCT, REPS, RTW
X STORYI, STORYC, STORYX, TOTWT, WEIGHT
X ERRFXD, ECONMY, IOUT, ICOL

LOGICAL ECONMY
DOUBLE PRECISION RWT, TOTWT, WEIGHT
LOGICAL BYPASS, BZERO, DELETE, IFCHI
X IFSSR, IFTT, IFWT, REPS, PREDCT
X STOR YI, STORY, STORYX, FIRST, ERRFXD
LOGICAL XSAVE
LOGICAL PNCHE
DIMENSION XCHK(60)

C******************************************************************************
C
C EQUIVALENCE (NARAY, SUMXXI), (S, BO), (SSQ, REPVAR)
DIMENSION NARAY(1830), S(9), SSQ(9)

C ZERO OUT ALL DATA ARRAYS EACH NEW DATA SET
100 DO 101 J=1,4740
READ(IDENTIFICATION CARD AND OPTIONS CARD)

READ(INPUT5,110) I,IDENT
WRITE(LIST,111) IDENT
FIRST=.TRUE.
ERRFJX=.FALSE.
IF(I) 120,120,115
READ(INPUT5,300) FMT
WRITE(LIST,301) FMT
I=I-1
GO TO 113
120 READ(INPUT5,1282) NOVAR,NODEP,NOTERM,NOOB,NTKEEP
WRITE(LIST,1283) NOVAR,NODEP,NOTERM,NOOB
IF(NTKEEP.NE.0) WRITE(LIST,1307) NTKEEP
READ(INPUT5,117) BZERO, IFTT, IFWT, IFCHI, STORYX, IFSSR,
X ECONMY, ISTRAT, PNCX
WRITE(LIST,118) BZERO, IFTT, IFWT, IFCHI, STORYX, IFSSR, ECONMY,
X ISTRAT, PNCX

THESE ARE INITIALIZATIONS MADE BEFORE EACH SET OF DATA
ICOL DETERMINES THE NUMBER OF VARIABLES READ PER OBSERVATION
JCOL IS THE NUMBER OF TERMS IN THE TOTAL REGRESSION EQUATION
LENGTH IS THE NUMBER OF STORES NEEDED FOR THE MATRICES
LENGTH= NOTERM*(NOTERM+1)/2
ICOL=NOVAR + NODEP
JCOL = NOTERM + NODEP
NWHERE= NOTERM
REWIND INTER
DO 140 J=1,60
IDOUT(J) = J
140 NTTERM(J)=J
DO 145 J=1,100
NXCOD(J)=J
NTRAN(J)=O
145 NCON(2*J)=J

IF(BZERO) WRITE(LIST,190)
IF(NET,BZERO) WRITE(LIST,170)

READ(INPUT5,282) NTRANS,KONNO
IF(NTRANS.EQ.0) GO TO 255
220 READ (INPUT5,230) (NTERM(K),K=1,JCOL)
WRITE(LIST,235) (NTERM(K),K=1,JCOL)
READ (INPUT5,230) (NXCOD(I),NTRAN(I),NCON(2*I-1),NCON(2*I),I=1,NTR
AANS )
WRITE(LIST,240) (NXCOD(I),NTRAN(I),NCON(2*I-1),NCON(2*I),I=1,
X NTRANS)
IF(KONNO) 255,255,250
250 READ (INPUT5,260) (CON(I),I=1,KONNO)
WRITE(LIST,262) (CON(I),I=1,KONNO)

255 READ(INPUT5,257) DELETE,P
IF(DELETE) IFTT=.TRUE.
C IF THERE ARE REPLICATED POINTS READ IN THE NUMBER OF POINTS AND
C THE NUMBER OF REPLICATIONS. SINGLE DATA POINTS ARE DATA POINTS
C REPLICATED ONCE. IMPLIED HERE.

READ(INPUT5,257) REPS  
XSAVE=.FALSE.

265 IF(.NOT.REPS) GO TO 290
READ(INPUT5,282) IREP,(NARAY(I),I=1,IREP)
WRITE(LIST,284) IREP
WRITE(LIST,283) (NARAY(I),I=1,IREP)
NPDCG=0
IREP=1
IC=NARAY(I)
XSAVE=.TRUE.
DO 315 I=1,NODEP
POOLED(I)=0.0
S(I)=0.0
315 SSQ(I)=0.0

C READ VARIABLE FORMAT FOR DATA

290 READ(INPUT5,110) INPUT,FMT
WRITE(LIST,111) FMT
310 TOTWT=0.0
WEIGHT=1.0
WRITE(LIST,301) IDENT

C READ IN INPUT VARIABLES

DO 490 J=1,NOOR
330 IF(.NOT.IFHT) GO TO 350
340 READ (INPUT,FMT) (X(I),I=1,ICOL)
GO TO 360
350 READ (INPUT,FMT) (X(I),I=1,ICOL), WEIGHT
360 CONTINUE
IF(EC3NY) WRITE(LIST,381) J,(X(I),I=1,ICOL)
381 FORMAT(1H14,9G14.6/(5X,9G14.6))
IF(TRANSM)EQ.0) GO TO 450
IF(EC3NY) GO TO 390
WRITE(LIST,370) WEIGHT,J
WRITE(LIST,380) (X(I),I=1,ICOL)
390 CALL TRANS
420 DO 330 K=1,JCOL
I=TERM(K)
X(K) = CON(I)
430 CONTINUE
450 CONTINUE
IF(EC3NY) GO TO 4609
WRITE(LIST,460) J
461 WRITE (LIST,380) (X(I),I=1,JCOL)
4609 CONTINUE
IF(.IFHT) WRITE(INTER) (X(I),I=1,69),WEIGHT
IF(.NOT.XSAVE) GO TO 4611
DO 4610 K=1,NOTERM
4610 XCHK(K)=X(K)
XSAVE=.FALSE.
4611 CONTINUE

C
C COMPTE THE ERROR VARIANCE FROM REPLICATED DATA

IF(*N3IoREPS) GO TO 480
IGOTO =1
IF(NA?Y(IREP), GT.1) IGOTO=2
IF(J, GE, IC) WRITE(6,462) IREP
DO 475 I=1, NODEP
IF(I-1) 4629,4629,464
4629 DO 463 K=1, NOTERM
IF(X(K).NE., XCHK(K)) GO TO 2001
463 CONTINUE
464 CONTINUE

KBAR=NOTERM+1
S(I)=S(I)+X(KBAR)
SSQ(I)=SSQ(I)+X(KBAR)**2
IF(J-IC) 475,465,465
465 GO TO (468,466), IGOTO
466 ZEAN(I)=S(I)/FLOAT(NARY(IREP))
SSQ(I) = SSQ(I) - ZEAN(I)*S(I)
POOLE(I)=POOLE(I)+SSQ(I)
WRITE(LIST,467) I,SSQ(I),S(I),ZEAN(I)
468 IF(I. LT. NODEP) GO TO 469
NPDEG=NPDEG+NARAY(IREP)-1
IREP=IREP+1
IC = IC + NARY(IREP)
WRITE(LIST,4671)
469 S(I)=0.0
SSQ(I)=0.0
XSAVE=.TRUE.
475 CONTINUE

C******************************************************************************
C CALCULATE SUMS, SUMS OF SQUARES AND SUMS OF CROSS PRODUCTS.
C******************************************************************************
480 CALL SUMUP
490 CONTINUE

C 490 CONTINUE IS THE END OF THE LOOP FOR READING DATA CARDS
IF(*N3ToREPS) GO TO 496
DO 493 I=1, NODEP
REPVAR(I)=POOLED(I)/FLOAT(NPDEG)
493 CONTINUE
496 CONTINUE

C******************************************************************************
C ALL DATA HAS BEEN READ IN AND THE XTRANSPOSEX AND XTRANSPOSEY
C MATRIX HAVE BEEN CALCULATED.
C NOW WRITE THE MATRICES
CALL MFIX
REWIN3 INTER
GO TO 640

C******************************************************************************
C THIS CODING DELETES THE DATA FROM THE SUMXX MATRIX
C CORRESPONDING TO THE INDEPENDENT TERM DELETED
6500 CONTINUE
IR=IOJT-1
IC = NTERM - IOUT
IF ( IC .EQ. 0) GO TO 6700
INOC= IOUT*IR/2
INEw = INOC
IOLI= INEW + IOUT
IRC=0
IBC=0
ITC=0

50
DO 6630 I=IOLD,LENGTH
  INEW = INEW+1
  IOLD=IOLD + 1
  IF(ITC.GT.0) GO TO 6540
  IRC=IRC + 1
  IF(IRC.GT.IR) GO TO 6530
  SUMXX(INEW)=SUMXX(IOLD)
  GO TO 6600
6530  IBC=IBC + 1
  ITC = IBC
  IOLD = IOLD+1
  IRC= J
6540  ITC = ITC -1
  SUMXX(INEW)=SUMXX(IOLD)
6600  CONTINUE
6700  LENGTH = LENGTH-NOTERM
  NOTERM= NOTERM -1
  JCOL= VOTERM+NODEP
C******************************************************************************
C INVERT THE SUMXX MATRIX AND COMPUTE REGRESSION COEFS
C AND SJYS OF SQUARES DUE TO REGRESSION IN THE MATRIX INVERSION
C ROUTINE
640  CONTINUE
CALL MATINV
FIRST= .FALSE.
C******************************************************************************
C WRITE(XTX INVERSE, THIS MATRIX TIMES ERROR MEAN SQUARE (ERRMS)
C IS THE VARIANCE-COVARIANCE MATRIX OF REGRESSION COEFFICIENTS.
C IF(ECONMY) GO TO 970
WRITE(LIST,700)
   CALL TRIANG(X, SUMXI, NOTERM, 8, FMTTRI, 2)
C******************************************************************************
C IF A VARIABLE HAS BEEN DELETED ADJUST COUNTERS AND RECOMPUTE THE
C REGRESSION. IF NO VARIABLE HAS BEEN DELETED CONTROL WILL PASS
C FROM THE TTEST ROUTINE TO THE CHI-SQUARE OPTION.
970  CONTINUE
   IF(.NOT.IFFT) GO TO 1020
980  WRITE (LIST,301)IDENT
   CALL TTEST($1020, NTKEEP)
   IF(NODEP-1) 985,990,985
985  WRITE(LIST,986) NODEP
   NODEP=1
990  J=JCOL-1
   DO 995 K=IOUT,J
      NTERM(K)=NTERM(K+1)
      ZEAN(K)= ZEAN(K+1)
      SUMX(K) = SUMX(K+1)
      SUMX2(K) = SUMX2(K+1)
      IDOUT(K) = IDOUT(K+1)
      SUMXY(K,1)= SUMXY(K+1,1)
995  CONTINUE
   IF(NOTERM.EQ.1) GO TO 1000
   GO TO 6500
1000 WRITE(LIST,1005)
   NOTERM=0
   GO TO 1035
C******************************************************************************
C
C******************************************************************************
1020 IF(.NOT.IFCH1) GO TO 1035
1030 WRITE(LIST,301) IDENT
CALL JUTPLT(PNCH)
C
C******************************************************
1035 READ(INPUT5,117) PREDICT
IF(.NOT.PREDICT) GO TO 100
CALL PREDIC
1040 GO TO 100
C
C******************************************************
2001 WRITE(LIST,1306)
STOP
C
C******************************************************
8001 FORMAT(1H1)
8002 FORMAT(1H2)
110 FORMAT (I2,13A6)
111 FORMAT (1H1,13A6,A2)
117 FORMAT(7L1,11,L1)
118 FORMAT(1H7L1,11,L1)
170 FORMAT(33H THERE IS NO BO TERM IN THE MODEL)
190 FORMAT(26H THERE IS A BO TO ESTIMATE )
230 FORMAT(4012)
235 FORMAT(11H NTERM(K) = /(1H 3014))
240 FORMAT(25H THE TRANSFORMATIONS ARE /(1H 5(4I4,5X)))
257 FORMAT( 11L1, F3.3)
260 FORMAT(5E15.7)
262 FORMAT(19H THE CONSTANTS ARE /(1H 8G15.7))
282 FORMAT(2014)
283 FORMAT(1H 2014)
284 FORMAT(11H THERE ARE I5,16H REPLICATE SETS )
300 FORMAT(13A6,1A2)
301 FORMAT (1H 13A6,A2)
370 FORMAT(1H0,29H OBSERVED VARIABLES, WEIGHT = G14.6,6X,15H OBSERVATION 1 = ,I5)
380 FORMAT(1H 9G14.6)
460 FORMAT(1H ,37H TERMS OF THE EQUATION, OBSERVATION = ,I5)
462 FORMAT(18H K** REPLICATE SET IS 15,3X,100(1H*))
4671 FORMAT(1H125(1H*))
467 FORMAT(14H DEP. VAR. I6,8H SSQ=G14.7,8H SUM=G14.7,8H M XEAN= G14.7)
540 FORMAT(1H 8G14.7)
560 FORMAT(21H2X TRANSPOSE X MATRIX )
670 FORMAT(25H2CORRELATION COEFFICIENTS )
700 FORMAT(32H2X TRANSPOSE X) INVERSE MATRIX )
986 FORMAT(39H THE NUMBER OF DEPENDENT VARIABLES WAS 13,83H IT IS BE XING SET TO ONE AND THE REJECTION OPTION EXERCISED ON DEPENDENT VAR XIABLE 1 )
1005 FORMAT(39H THERE IS NO EVIDENCE OF A REGRESSION, / X 74H USE THE MEAN RESPONSE FOR THE BEST ESTIMATE OF THE DEPEND XENT VARIABLE(S) )
1282 FORMAT(314.15,14)
1283 FORMAT(1H 314,15)
1306 FORMAT(40H REPLICATE SETS ARE NOT GROUPED PROPERLY )
1307 FORMAT(11H THE FIRST 12,64H TERMS OF THE MODEL WILL BE RETAINED R XEGARDLESS OF SIGNIFICANCE )
END
52
**SUBROJTINE MATINV**

**PURPOSE**

1) Compute eigenvalues and eigenvectors of \((X - \text{TRANSPOSE } X)\) matrix if requested. (\text{STORYX}=.TRUE.)

2) Compute \((X \text{ TRANSPOSE } X)^{-1}\)

3) Compute regression coefficients

4) Compute other regression statistics

**SUBROUTINES NEEDED**

- BORD
- LOC
- EIGEN
- INVXTX
- RECT
- RSTATS
- TRIANG

**REMARKS**

The eigenvalues are computed as an aid in determining the condition of the system of equations for the regression coefficients. Examination of them and their associated eigenvectors may show that certain sets of independent variables are highly correlated and not easily liable to independent study.

**SUBROJTINE MATINV**

**COMMON/BIG/B(60,9),SUMXY(60,9),SUMXX(1830),SUMXXI(1830)**

**DOUBLE PRECISION SUMXY,SUMXX,SUMXXI**

**COMMON/MED/BO(9),SUMX(69),SUMX2(69),SUMY2(9),ZEAN(69),**

- \(X CON(99),ERRMS(9),IDENT(13),IDOUT(60),NCON(100),NTERM(69),\)

- \(NTRAN(100),NXCOD(100),POOLED(9),REPVAR(9),RESMS(9),X(99)\)

**DOUBLE PRECISION B0,SUMX,SUMY2,ZEAN**

**COMMON /FMTS/ FMT(13),FMTTRI(14)**

**COMMON /SMALL/ BYPASS,BZERO,DELETE, FIRST, IFCHI, IFSSR,**

- \(X IFTT, IFWT, INPUT, INPUT5, INTER,\)

- \(X ISTAT, JCOL, KONNO, LENGTH, LIST,\)

- \(X NERRR, NODEP, NOOB, NOTERM,\)

- \(X NOVAR, NPDEG, NRES, NTRANS, NWHERE,\)

- \(X P, PREDCT, REPS, RWT,\)

- \(X STORYI, STORYG, STORYX, TOTWT, WEIGHT,\)

- \(X EKRFKD, ECONMY, IOUT, ICOL\)

**LOGICAL ECONMY**

**LOGICAL BYPASS, BZERO, DELETE, IFCHI,**

- \(X IFSSR, IFTT, IFWT, REPS, PREDCT,\)

- \(X STORY, STORYX, STORYY, FIRST, EKRFKD\)

**DOUBLE PRECISION RWT, TOTWT, WEIGHT**

**DIMENSION A(1),C(1),XTX(3)**

**EQUIVALENCE (A,SUMXI), (C,SUMXX1(915))**

**DOUBLE PRECISION SUM**

**DATA (XTX(I),I=1,3) /6HX TRAN, 6HPOSE, 6HX **

**IORDER= NOTERM**

**IF(NOTERM-1) 10,10,12**
10 SUMXXI(I) = 1.0/SUMX2(I)
GO TO 350

C

12 IF(*,NJT, STORYX) GO TO 30
DO 14 I=1,LENGTH
A(I) = SUMXX(I)
14 CONTINUE

16 CALL EIGEN(A,C , IORDER,0)
WRITE(LIST,17)(XTX(I),I=1,3)
J=0
DO 18 I=1,IORDER
J=J+1
18 A(I) = A(J)
WRITE(LIST,19) (A(I),I=1,IORDER)
WRITE(LIST,20)
CALL RECT(IORDER,IORDER,IORDER,IORDER,C,X ,FMTTRI,1)

30 DO 35 I=1,LENGTH
35 SUMXXI(I) = SUMXX(I)

C

C*************************************************************************
C NO SJBMODELS TO ANALYZE SO INVERT A DIRECTLY BY GAUSS
49 IF(IFSSR) GO TO 50
CALL INVTXX(SUMXXI,NOTERM,D,1.0)
GO TO 60

C

C*************************************************************************
C SUBMODELS HAVE BEEN REQUESTED SO WE USE BORDERING
50 IORDER = 0
55 IORDER = IORDER + 1
CALL BORD(IORDER,SUMXXI)
60 CONTINUE

C

C*************************************************************************
C COMPUTE COEFFICIENTS AND PRINT THEM
350 DO 373 J=1,NODEP
DO 373 K=1,IORDER
B(K,J) = 0.000
DO 373 L=1,IORDER
CALL LOC(L,K,IR)
B(K,J) = B(K,J) + SUMXXI(IR)*SUMXY(L,J)
370 CONTINUE

C

WRITE(LIST,380) IDENT
WRITE(LIST,382)
IF(*,NJT,BZERO) GO TO 400
DO 395 J=1,NODEP
SUM = 0.000
KBAR = NOTERM + J
DO 385 K=1,IORDER
SUM = SUM + B(K,J)*ZEAN(K)
385 CONTINUE

385 BO(J) = ZEAN(KBAR) - SUM
390 CONTINUE
WRITE(LIST,395)
WRITE(LIST,397) (BO(K),K=1,NODEP)
400 WRITE(LIST,410)
DO 433 J=1,IORDER
WRITE(LIST,432) IDOUT(J), (B(J,K),K=1,NODEP)
430 CONTINUE

C
**C**
**COMPUTE REGRESSION STATISTICS IN RSTATS**

**C**
**CALL RSTATS(IORDER)**

**C**
**IF IORDER IS LESS THAN NOTERM WE HAVE USED THE BORDERING OPTION**
**AND MUST GO BACK TO FINISH.**

**C**
**IF(IORDER-NOTERM) 55,500,500**

**500**
**RETURN**

**17**
**FORMAT(34H2) THE FOLLOWING ARE EIGENVALUES OF 2A6,A1, 7H MATRIX**

**19**
**FORMAT(1H8G16.7)**

**20**
**FORMAT(132H2) THIS IS THE MODAL MATRIX OR MATRIX OF EIGENVECTORS.**
**EIGENVECTORS ARE WRITTEN IN COLUMNS LEFT TO RIGHT IN SAME ORDER AS**
**2EIGENVALUES )**

**380**
**FORMAT(1H1,13A6,1A2)**

**382**
**FORMAT(61H EACH COLUMN CONTAINS THE COEFFICIENTS FOR ONE DEPENDEN**
**XT TERM )**

**395**
**FORMAT(20H CONSTANT TERM (B0) )**

**397**
**FORMAT(4X,9G14.6)**

**410**
**FORMAT(36H REGRESSION COEFFICIENTS (B1,...,BK) )**

**432**
**FORMAT(1H13,9G14.6)**

**END**

---

**SIBFCT TT-STX**

**C**
**SUBROJTINE TTEST**

**C**
**PURPOSE**

**C**
**COMPUTE THE T-STATISTICS FOR EACH REGRESSION TERM AND**
**ITS TWO TAILED SIGNIFICANCE LEVEL. THEN DETERMINE THE**
**TERM WITH THE LEAST SIGNIFICANCE AND RETURN THIS**
**INFORMATION TO NEWRAP**

**C**
**SUBROJTINE TTEST(*,NTKEEP)**

**C**
**COMMON /FRMTS/ FMT(13),FMTTRI(14)**

**COMMON/BIG/B(60,9),SUMXY(60,9),SUMX(1830),SUMXX(1830)**

**X ,SUMMY(1)**

**DOUBLE PRECISION B,SUMXY,SUMXX,SUMXI**

**COMMON/MED/BO(9),SUMX(69),SUMX2(69),SUMY(69),SUMY2(69),ZEN(69),**

**X ,CON(99),EERRMS(9),IDENT(13),IDOUT(60),NCON(200),NTERM(69),**

**X ,NTRAN(100),NXCON(100),POOLED(9),REPVAR(9),RESMS(9),X(99)**

**DOUBLE PRECISION BO, SUMX, SUMX2, SUMY2, ZEN**

**COMMON/SMALL/ BYPASS,BZERO,DELETE, FIRST, IFCHI, IFSSR,**

**X ,IFITT, IFW, INPUT, INPUTS, INTER,**

**X ,ISTRAT, JCOL, KONNO, LENGTH, LIST,**

**X ,NERROR, NOUEP, NOOB, NOTERM,**

**X ,NOVAR, NPGEG, NRES, NTRANS, NWHERE,**

**X ,P, PREDICT, REPS, RWT,**

55
| DATA (PLEVEL(JJ), JJ=1,13) | 0.10, 0.20, 0.30, 0.40, 0.50, 0.60, 0.70, 0.80, 0.90, 0.95, 0.98, 0.99, 0.999 / |
| DATA (T(1,JJ), JJ=1,13) | 0.158, 0.325, 0.510, 0.727, 1.000, 1.376, |
| 1.963, 3.078, 6.314, 12.706, 31.821, 63.057, 63.6619, |
| 0.142, 0.289, 0.445, 0.617, 0.816, 1.061, |
| 1.386, 1.886, 2.920, 4.3027, 6.965, 9.925, 31.598, |
| 0.137, 0.277, 0.424, 0.584, 0.765, 0.978, |
| 1.250, 1.638, 2.353, 3.1825, 4.541, 5.841, 12.924, |
| 0.134, 0.271, 0.414, 0.569, 0.741, 0.941, |
| 1.190, 1.533, 2.132, 2.776, 3.747, 4.604, 8.610, |
| 0.132, 0.267, 0.408, 0.559, 0.727, 0.920, |
| 1.156, 1.476, 2.015, 2.570, 3.365, 4.032, 6.869, |
| 0.131, 0.265, 0.404, 0.553, 0.718, 0.906, |
| 1.134, 1.440, 1.943, 2.446, 3.143, 3.707, 5.959, |
| 0.130, 0.263, 0.402, 0.549, 0.711, 0.896, |
| 1.119, 1.415, 1.895, 2.364, 2.998, 3.499, 5.408, |
| 0.130, 0.262, 0.399, 0.546, 0.706, 0.889, |
| 1.108, 1.397, 1.860, 2.306, 2.896, 3.355, 5.041, |
| 0.129, 0.261, 0.398, 0.543, 0.703, 0.883, |
| 1.100, 1.389, 1.833, 2.262, 2.821, 3.250, 4.781, |
| 0.129, 0.260, 0.397, 0.542, 0.700, 0.879, |
| 1.093, 1.372, 1.812, 2.281, 2.764, 3.019, 4.587, |
| 1.088, 1.363, 1.796, 2.101, 2.718, 3.106, 4.437, |
| 0.128, 0.259, 0.395, 0.539, 0.695, 0.873, |
| 1.083, 1.356, 1.782, 2.178, 2.681, 3.055, 4.318, |
| 0.128, 0.259, 0.394, 0.538, 0.694, 0.870, |
| 1.079, 1.350, 1.771, 2.164, 2.650, 3.012, 4.221, |
| 0.128, 0.258, 0.393, 0.537, 0.692, 0.868, |
| 1.076, 1.345, 1.761, 2.148, 2.624, 2.977, 4.140, |
| 0.128, 0.258, 0.392, 0.536, 0.691, 0.866, |
| 1.074, 1.341, 1.753, 2.135, 2.602, 2.947, 4.073, |
| 0.128, 0.258, 0.392, 0.535, 0.690, 0.865, |
| 1.071, 1.337, 1.746, 2.128, 2.583, 2.921, 4.015, |
| 0.128, 0.257, 0.392, 0.534, 0.689, 0.863, |
| 1.069, 1.333, 1.740, 2.109, 2.567, 2.898, 3.965, |
| 0.127, 0.257, 0.392, 0.534, 0.688, 0.862, |
| 1.067, 1.330, 1.734, 2.100, 2.552, 2.878, 3.922, |
| 0.127, 0.257, 0.391, 0.533, 0.688, 0.861, |
| 1.066, 1.328, 1.729, 2.093, 2.539, 2.861, 3.883, |
| 0.127, 0.257, 0.391, 0.533, 0.687, 0.860, |
| 1.064, 1.325, 1.725, 2.086, 2.528, 2.845, 3.850, |
| 0.127, 0.257, 0.391, 0.532, 0.686, 0.858, |
| 1.065, 1.323, 1.721, 2.079, 2.518, 2.831, 3.819, |
| 0.127, 0.257, 0.390, 0.532, 0.686, 0.858, |
| 1.064, 1.321, 1.717, 2.073, 2.510, 2.819, 3.789, |
| 0.127, 0.257, 0.389, 0.531, 0.685, 0.857, |
| 1.063, 1.319, 1.713, 2.069, 2.502, 2.806, 3.761, |
| 0.127, 0.257, 0.388, 0.530, 0.684, 0.855, |
| 1.062, 1.317, 1.709, 2.065, 2.494, 2.793, 3.734, |
| 0.127, 0.257, 0.388, 0.529, 0.683, 0.854, |
| 1.061, 1.315, 1.705, 2.061, 2.487, 2.780, 3.707, |
| 0.127, 0.257, 0.387, 0.528, 0.682, 0.853, |
| 1.060, 1.313, 1.701, 2.058, 2.480, 2.767, 3.680, |
| 0.127, 0.257, 0.387, 0.527, 0.681, 0.852, |
| 1.059, 1.311, 1.697, 2.054, 2.473, 2.754, 3.653, |
| 0.127, 0.257, 0.386, 0.526, 0.680, 0.851. |
1.061, 1.321, 1.717, 2.0739, 2.508, 2.819, 3.792

(T(23, JJ), JJ=1, 13) /0.127, 0.256, 0.390, 0.532, 0.685, 0.858,

1.060, 1.319, 1.714, 2.0687, 2.500, 2.807, 3.767

(T(24, JJ), JJ=1, 13) /0.127, 0.256, 0.390, 0.531, 0.685, 0.857,

1.059, 1.318, 1.711, 2.0639, 2.492, 2.797, 3.745

(T(25, JJ), JJ=1, 13) /0.127, 0.256, 0.390, 0.531, 0.684, 0.856,

1.058, 1.316, 1.708, 2.0595, 2.485, 2.787, 3.725

(T(26, JJ), JJ=1, 13) /0.127, 0.256, 0.390, 0.531, 0.684, 0.856,

1.058, 1.315, 1.706, 2.0555, 2.479, 2.779, 3.707

(T(27, JJ), JJ=1, 13) /0.127, 0.256, 0.389, 0.531, 0.684, 0.855,

1.057, 1.314, 1.703, 2.0518, 2.473, 2.771, 3.690

(T(28, JJ), JJ=1, 13) /0.127, 0.256, 0.389, 0.530, 0.683, 0.855,

1.056, 1.313, 1.701, 2.0484, 2.467, 2.763, 3.674

(T(29, JJ), JJ=1, 13) /0.127, 0.256, 0.389, 0.530, 0.683, 0.854,

1.055, 1.311, 1.699, 2.0452, 2.462, 2.756, 3.659

(T(30, JJ), JJ=1, 13) /0.127, 0.256, 0.389, 0.530, 0.683, 0.854,

1.055, 1.310, 1.697, 2.0423, 2.457, 2.750, 3.646

(T(31, JJ), JJ=1, 13) /0.126, 0.255, 0.388, 0.529, 0.681, 0.851,

1.050, 1.303, 1.684, 2.0211, 2.423, 2.704, 3.551

(T(32, JJ), JJ=1, 13) /0.126, 0.254, 0.387, 0.527, 0.679, 0.848,

1.046, 1.296, 1.671, 2.0003, 2.390, 2.660, 3.460

(T(33, JJ), JJ=1, 13) /0.126, 0.254, 0.386, 0.526, 0.677, 0.845,

1.041, 1.289, 1.658, 1.9799, 2.358, 2.617, 3.373

(T(34, JJ), JJ=1, 13) /0.126, 0.253, 0.385, 0.524, 0.674, 0.842,

1.036, 1.282, 1.645, 1.9600, 2.326, 2.576, 3.291

(T(1, JJ), JJ=1, 13) /0.10, 0.20, 0.30, 0.40, 0.50, 0.60, 0.70

T(II, JJ) IS THE T-STATISTIC AT THE TABULATED DEGREES OF FREEDOM

II = 13 DEGREES OF FREEDOM, EXCEPT FOR

II = 31 IS FOR 40 DEGREES

II = 32 IS FOR 60

II = 33 IS FOR 120

II = 34 IS FOR INFINITY

<table>
<thead>
<tr>
<th>JJ</th>
<th>PROBABILITY LEVEL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.10</td>
</tr>
<tr>
<td>2</td>
<td>0.20</td>
</tr>
<tr>
<td>3</td>
<td>0.30</td>
</tr>
<tr>
<td>4</td>
<td>0.40</td>
</tr>
<tr>
<td>5</td>
<td>0.50</td>
</tr>
<tr>
<td>6</td>
<td>0.60</td>
</tr>
<tr>
<td>7</td>
<td>0.70</td>
</tr>
</tbody>
</table>

**CALCULATE T STATISTICS**

220 WRITE (LIST, 230)

230 FORMAT (1H, 23H CALCUATED T STATISTICS /75H THE T STATISTICS CAN BE

1 USED TO TEST THE NET REGRESSION COEFFICIENTS b(I). )

DO 260 J = 1, NERROR

DO 240 K = 1, NODEP

TT(J, K) = ABS(B(J, K) / DEVB(J, K))

240 CONTINUE

WRITE (LIST, 250) (TT(J, K), K = 1, NODEP)

250 FORMAT (1H 9G14.6)

260 CONTINUE

**SEARCH THE TABLE OF TABULATED DEGREES OF FREEDOM**

NDEG = NERROR

**TABLE FOR 120 OF TABULATED DEGREES OF FREEDOM**

**STATISTICS FOR THE T STATISTICS CAN BE**

**AND AT THE TABULATED PROBABILITY LEVELS (JJ).**

**T(II, JJ) IS THE T-STATISTIC AT THE TABULATED DEGREES OF FREEDOM**

**II = 32 IS FOR 60**

**II = 33 IS FOR 120**

**II = 34 IS FOR INFINITY**

**CALCULATE T STATISTICS**

**SEARCH THE TABLE OF TABULATED DEGREES OF FREEDOM**

57
MAKENJ=.FALSE.
IF(NDEG=30)GO TO 400
IF(NDEG=40)GO TO 310
FINV=1.0/40.0
FM1INV=1.0/30.0
MAKENJ=.TRUE.
II=31
GO TO 400
IF(NDEG=60)GO TO 340
FINV=1.0/60.0
FM1INV=1.0/40.0
MAKENJ=.TRUE.
II=32
GO TO 400
IF(NDEG=120)GO TO 370
FINV=1.0/120.0
FM1INV=1.0/60.0
MAKENJ=.TRUE.
II=33
GO TO 400
II=34
FINV=0.0
FM1INV=1.0/120.0
MAKENJ=.TRUE.
WRITE(LIST,410)
410 FORMAT(104H UNDER NULL HYPOTHESIS THE INTERVAL (-T,T) WHERE T IS GIVEN ABOVE, HAS APPROX PROBABILITY LISTED BELOW. /42H MINUS SIGN INDICATES PROB EXCEEDS .999. )
IF(.NOT.MAKENU) GO TO 430
FNDEG=NDEG
DO 420 JJ=1,13
   T(35,JJ)=T(II,JJ)+((1.0/FNDEG - FINV)/(FM1INV-FINV))*(T(II-1,JJ) - T(II,JJ))
CONTINUE
II=35
DO 550 J=1,NOTERM
   DO 540 K=1,NODEP
      DO 440 JJ=1,13
         JJ=JJ
         IF(T(II, JJ) - TT(J, K))440,450,460
CONTINUE
   PROB(J, K)=-0.999
   GO TO 540
450 PROB(J, K)=PLEVEL(JJ)
   GO TO 540
460 IF(JJ.LE.9) GO TO 470
   JJ1=JJ-2
   JJ2=JJ-1
   JJ3=JJ
   GO TO 490
470 IF(JJ.LE.4) GO TO 480
   JJ1=JJ-1
   JJ2=JJ
   JJ3=JJ+1
   GO TO 490
480 JJ1=JJ
   JJ2=JJ+1
   JJ3=JJ+2
GO TO 490
PERFORM A THREE-POINT LAGRANGE INTERPOLATION

490 X=ALOS(TT(J,K))
X1=AL3S(T(I1,J11))
X2=ALJS(T(I1,J21))
X3=ALJS(T(I1,J31))
IF(TT(J,K).LE.1.0) GO TO 500
Y1=AL3G(1.0-LEVEL(JJ1))
Y2=AL3G(1.0-LEVEL(JJ2))
Y3=AL3G(1.0-LEVEL(JJ3))
GO TO 510
500 Y1=AL3G(LEVEL(JJ1))
Y2=AL3G(LEVEL(JJ2))
Y3=AL3G(LEVEL(JJ3))
510 PROB(J,K)=((X-X2)*(X-X3)*Y1)/((X1-X2)*(X1-X3)) + ((X-X1)*(X-X3))
     1 *Y2)/((X2-X1)*(X2-X3)) + ((X-X1)*(X-X2)*Y3)/((X3-X1)*(X3-X2))
     IF((TT(J,K)-1.0).LE.0.520.530
     GO TO 540
520 PROB(J,K)=EXP(PROB(J,K))
     GO TO 540
530 PROB(J,K)=1.0-EXP(PROB(J,K))
540 CONTINUE

WRITE THE PROBABILITIES (1.0-ALPHA)
WRITE(LIST,550) IDOUT(J),(PROB(J,K),K=1,NUDEP)
550 FORMAT(1H13,9(8X,F6.3))
560 CONTINUE

LIST THE DESIRED VALUE OF PROBABILITY (PWANT)

PERCENT=PWANT*100.0
WRITE(LIST,580) PERCENT
580 FORMAT(1H100,36HTHE DESIRED VALUE OF PROBABILITY IS ,F5.1, 8H PERCENT
     1)

DELETE THE TERM WITH THE LOWEST COMPUTED PROBABILITY IF THAT
PROBABILITY IS LESS THAN THAT DESIRED (PWANT)

IF(.NOT.DELETE) GO TO 660
IF(NTKEEP.EQ.NOTERM) GO TO 660
JOUT=J
590 AMIN=PWANT
JLO=MAXO(1,NTKEEP)
DO 620 J=JLO,NOTERM
IF(ABS(PROB(J,1))-PWANT)600,620,620
600 IF(ABS(PROB(J,1))-AMIN)610,620,620
610 AMIN=ABS(PROB(J,1))
JOUT=J
620 CONTINUE
IF(IDJT) 660,660,630
630 WRITE(LIST,550) IDOUT(IOUT)
650 FORMAT(1H1U1X,11HTHE TERM X(12,18H) IS BEING DELETED )
     GO TO 670
5 ALL VARIABLES REMAINING HAVE BEEN CONCLUDED SIGNIFICANT
660 RETURN
670 RETURN
END
SUBROUTINE RSTATS

PURPOSE
1) Compute and print the analysis of variance tables on regression and lack-of-fit if appropriate.
2) Compute and print r-squared and standard error of estimate.
3) Compute and print sums of squares due to each variable if it were last to enter regression.
4) Compute and print the standard deviations of each regression coefficient.

SUBROUTINE RSTATS(IORDER)

COMMON/BIG/B(60,9),SUMXY(60,9),SUMXX(1830),SUMXXI(1830)
X ,DUMMY(1)
DOUBLE PRECISION B,SUMXY,SUMXX,SUMXI
COMMON/MED/60(9),SUMX(69),SUM2(69),SUMY2(9),ZEAN(69),
X CON(99),ERRMS(9),IDENT(13),IDOUT(60),NCOND(200),NTERM(69),
X NTRAN(100),NXCOD(100),POoled(9),REPVAR(9),RESMS(9),X(99)
DOUBLE PRECISION BD,SUMX,SUMXX,SUMX2,SUMY2,ZEAN
COMMON /FRMTS/ FMT(13)
X FRMTTRI (14)
COMMON / SMALL/ 
X IFTT, IFWT, INPUT, INPUT5, INTER,
X ISTRAT, JCOL, KONNO, LENGTH, LIST,
X VERROR, NODEP, NOOB, NOTERM,
X NVAR, NPDEG, NRES, NTRANS, NWHERE,
X P, PREDCT, REPS, RWT,
X STORY, STORYC, STORYX, TOTWT, WEIGHT,
X ERRFXD, ECONMY, IOUT, ICOL
LOGICAL ECONMY
LOGICAL SATTRD
LOGICAL BYPASS, BZERO, DELETE, IFCHI, IFSSR,
X IFIT, IFWT, INPUT, INPUT5, INTER,
X ISTRU, JCOL, KONNO, LENGTH, LIST,
X VERROR, NODEP, NOOB, NOTERM,
X NVAR, NPDEG, NRES, NTRANS, NWHERE,
X P, PREDCT, REPS, RWT,
X STORY, STORYC, STORYX, TOTWT, WEIGHT,
LOGICAL ECONMY
LOGICAL SATTRD
LOGICAL BYPASS, BZERO, DELETE, IFCHI, IFSSR,
X IFIT, IFWT, INPUT, INPUT5, INTER,
X ISTRU, JCOL, KONNO, LENGTH, LIST,
X VERROR, NODEP, NOOB, NOTERM,
X NVAR, NPDEG, NRES, NTRANS, NWHERE,
X P, PREDCT, REPS, RWT,
X STORY, STORYC, STORYX, TOTWT, WEIGHT,
LOGICAL ECONMY
LOGICAL SATTRD
LOGICAL BYPASS, BZERO, DELETE, IFCHI, IFSSR,
X IFIT, IFWT, INPUT, INPUT5, INTER,
X ISTRU, JCOL, KONNO, LENGTH, LIST,
X VERROR, NODEP, NOOB, NOTERM,
X NVAR, NPDEG, NRES, NTRANS, NWHERE,
X P, PREDCT, REPS, RWT,
X STORY, STORYC, STORYX, TOTWT, WEIGHT,
LOGICAL ECONMY
LOGICAL SATTRD
LOGICAL BYPASS, BZERO, DELETE, IFCHI, IFSSR,
X IFIT, IFWT, INPUT, INPUT5, INTER,
X ISTRU, JCOL, KONNO, LENGTH, LIST,
X VERROR, NODEP, NOOB, NOTERM,
X NVAR, NPDEG, NRES, NTRANS, NWHERE,
X P, PREDCT, REPS, RWT,
X STORY, STORYC, STORYX, TOTWT, WEIGHT,
LOGICAL ECONMY
LOGICAL SATTRD
LOGICAL BYPASS, BZERO, DELETE, IFCHI, IFSSR,
X IFIT, IFWT, INPUT, INPUT5, INTER,
X ISTRU, JCOL, KONNO, LENGTH, LIST,
X VERROR, NODEP, NOOB, NOTERM,
X NVAR, NPDEG, NRES, NTRANS, NWHERE,
X P, PREDCT, REPS, RWT,
X STORY, STORYC, STORYX, TOTWT, WEIGHT,
LOGICAL ECONMY
LOGICAL SATTRD
LOGICAL BYPASS, BZERO, DELETE, IFCHI, IFSSR,
X IFIT, IFWT, INPUT, INPUT5, INTER,
X ISTRU, JCOL, KONNO, LENGTH, LIST,
X VERROR, NODEP, NOOB, NOTERM,
X NVAR, NPDEG, NRES, NTRANS, NWHERE,
X P, PREDCT, REPS, RWT,
X STORY, STORYC, STORYX, TOTWT, WEIGHT,
LOGICAL ECONMY
LOGICAL SATTRD
LOGICAL BYPASS, BZERO, DELETE, IFCHI, IFSSR,
X IFIT, IFWT, INPUT, INPUT5, INTER,
X ISTRU, JCOL, KONNO, LENGTH, LIST,
X VERROR, NODEP, NOOB, NOTERM,
X NVAR, NPDEG, NRES, NTRANS, NWHERE,
X P, PREDCT, REPS, RWT,
X STORY, STORYC, STORYX, TOTWT, WEIGHT,
LOGICAL ECONMY
LOGICAL SATTRD
LOGICAL BYPASS, BZERO, DELETE, IFCHI, IFSSR,
X IFIT, IFWT, INPUT, INPUT5, INTER,
X ISTRU, JCOL, KONNO, LENGTH, LIST,
X VERROR, NODEP, NOOB, NOTERM,
X NVAR, NPDEG, NRES, NTRANS, NWHERE,
X P, PREDCT, REPS, RWT,
X STORY, STORYC, STORYX, TOTWT, WEIGHT,
LOGICAL ECONMY
LOGICAL SATTRD
LOGICAL BYPASS, BZERO, DELETE, IFCHI, IFSSR,
X IFIT, IFWT, INPUT, INPUT5, INTER,
X ISTRU, JCOL, KONNO, LENGTH, LIST,
X VERROR, NODEP, NOOB, NOTERM,
X NVAR, NPDEG, NRES, NTRANS, NWHERE,
X P, PREDCT, REPS, RWT,
X STORY, STORYC, STORYX, TOTWT, WEIGHT,
LOGICAL ECONMY
LOGICAL SATTRD
LOGICAL BYPASS, BZERO, DELETE, IFCHI, IFSSR,
X IFIT, IFWT, INPUT, INPUT5, INTER,
X ISTRU, JCOL, KONNO, LENGTH, LIST,
X VERROR, NODEP, NOOB, NOTERM,
X NVAR, NPDEG, NRES, NTRANS, NWHERE,
X P, PREDCT, REPS, RWT,
X STORY, STORYC, STORYX, TOTWT, WEIGHT,
LOGICAL ECONMY
LOGICAL SATTRD
LOGICAL BYPASS, BZERO, DELETE, IFCHI, IFSSR,
X IFIT, IFWT, INPUT, INPUT5, INTER,
X ISTRU, JCOL, KONNO, LENGTH, LIST,
X VERROR, NODEP, NOOB, NOTERM,
X NVAR, NPDEG, NRES, NTRANS, NWHERE,
X P, PREDCT, REPS, RWT,
X STORY, STORYC, STORYX, TOTWT, WEIGHT,
LOGICAL ECONMY
LOGICAL SATTRD
LOGICAL BYPASS, BZERO, DELETE, IFCHI, IFSSR,
X IFIT, IFWT, INPUT, INPUT5, INTER,
X ISTRU, JCOL, KONNO, LENGTH, LIST,
X VERROR, NODEP, NOOB, NOTERM,
X NVAR, NPDEG, NRES, NTRANS, NWHERE,
X P, PREDCT, REPS, RWT,
X STORY, STORYC, STORYX, TOTWT, WEIGHT,
LOGICAL ECONMY
LOGICAL SATTRD
LOGICAL BYPASS, BZERO, DELETE, IFCHI, IFSSR,
X IFIT, IFWT, INPUT, INPUT5, INTER,
X ISTRU, JCOL, KONNO, LENGTH, LIST,
X VERROR, NODEP, NOOB, NOTERM,
X NVAR, NPDEG, NRES, NTRANS, NWHERE,
X P, PREDCT, REPS, RWT,
X STORY, STORYC, STORYX, TOTWT, WEIGHT,
LOGICAL ECONMY
LOGICAL SATTRD
LOGICAL BYPASS, BZERO, DELETE, IFCHI, IFSSR,
X IFIT, IFWT, INPUT, INPUT5, INTER,
X ISTRU, JCOL, KONNO, LENGTH, LIST,
X VERROR, NODEP, NOOB, NOTERM,
X NVAR, NPDEG, NRES, NTRANS, NWHERE,
X P, PREDCT, REPS, RWT,
X STORY, STORYC, STORYX, TOTWT, WEIGHT,
RLOF=1.0/FLOAT(NLOF)
GO TO 100
90 SATRD=.TRUE.
100 CONTINUE
  NXT=RM=ORDER
  RNO=RT
C
C***********************************************************************
 C COMPUTE RESIDUAL SUM OF SQUARES, RESIDUAL VARIANCE, VARIANCE
 C FROM REPETITIONS IF APPROPRIATE, AND THE F-RATIO OF MEAN SQUARE
 C LACK-OF-FIT AND MEAN SQUARE RESIDUALS.
C DO 213 J=1,NODEP
    SSQRES(J)=0.0
    DO 20 J=1,NXTERM
      SSQREG(J)=SSQREG(J)+B(I,J)*SUMXY(I,J)
  CONTINUE
    SSQRES(J)=SUMY2(J)-SSQREG(J)
    REGMS(J)= SSQREG(J)*RNREG
    RESMS(J)= SSQRES(J)*RNRES
    RSQD(J)=SSQREG(J)/SUMY2(J)
    R(J)=SQRT(RSQD(J))
    IF((SATRD.OR.REP1)) GO TO 210
    XLOF(J)=SSQRES(J)-POOL(J)
    XLOFMS(J)= XLOF(J)+RNLOF
    FRATIJ(J)=XLOFMS(J)/REPVAR(J)
  210 CONTINUE
C
C***********************************************************************
 C DETERMINE WHICH ESTIMATE OF SIGMA SQUARED SHOULD BE USED IN
 C HYPOTHESIS TESTS. PUT THE PROPER ONE IN ERRMS AND SET ERRFXD
 C TO TRUE IF THE PRESENT VALUE IS TO BE USED FOR ALL FOLLOWING
 C TESTS AND T-STATISTICS.
C IOUT=ISTRAT
  IF(ERRFXD) GO TO 250
  IF(ISTRAT.GT.3) GO TO 214
  211 DO 213 J=1,NODEP
    ERRMS(J)= RESMS(J)
    NERROR = NRES
    IOUT=3
    GO TO 250
  214 IF(ISTRAT.LE.1) GO TO 218
  IF(NXTRA) GO TO 211
  DO 215 J=1,NODEP
  215 ERRMS(J)= REPVAR(J)
    NERROR= NPDEG
    ERRFXD=.TRUE.
    IOUT=1
    GO TO 250
  218 IF(FIRST.AND.(IORDER.EQ.NOTERM)) GO TO 220
  GO TO 211
  220 ERRFXD=.TRUE.
  DO 222 J=1,NODEP
  222 ERRMS(J)= RESMS(J)
    NERROR = NRES
    ISTRAT=2
    IOUT=2
C
C***********************************************************************
 C WRITE ANOVA TABLES
C DO 500 J=1,NODEP
  IF(ERRMS(J).EQ.0.0) ERRMS(J)=1.0E-30
  WRITE(LIST,1001) J
WRITE(LIST,1002)
WRITE(LIST,1003) SSQREG(J), NREG, RESMS(J)
WRITE(LIST,1004) SSQRES(J), NRES, RESMS(J)
WRITE(LIST,1005)
WRITE(LIST,1006) SUMY2(J), NTOT
WRITE(LIST,1007)
WRITE(LIST,1500) RSD(J), R(J)
STD=SQRT(RESMS(J))
WRITE(LIST,1600) STD
WRITE(LIST,1700) IOUT, ERRMS(J), NERROR
F=REGM(J)/ERRMS(J)
WRITE(LIST,1750) F, NREG, NERROR
IF((.NOT.REPS).OR.SATRTD) GO TO 500
WRITE(LIST,2001)
WRITE(LIST,1002)
WRITE(LIST,2005) XLOF(J), NLOF, XLOFMS(J)
WRITE(LIST,2006) POOL(J), NPDEG, REPVAR(J)
WRITE(LIST,1004) SSQRES(J), NRES, RESMS(J)
WRITE(LIST,1005)
WRITE(LIST,2008) FRATIO(J)
WRITE(LIST,1007)
500 CONTINUE
C
C******************************************************************************
C COMPUTE CONTRIBUTION OF EACH INDEPENDENT VARIABLE TO REG SUM
C OF SQUARES AS IF IT WERE LAST TO ENTER
C
WRITE(LIST,370)
IR= 0
DO 8635 K=1,NXTERM
IR= IR+K
DO 8632 J=1,NODEP
8632 SSQSLST(J)= U(K,J)**2/SUMXI(IR)
WRITE(LIST,380) IDOUT(K),(SSQSLST(J),J=1,NODEP)
8635 CONTINUE
C
C******************************************************************************
C COMPUTE STANDARD DEVIATION OF REGRESSION COEFFICIENTS
C
WRITE(LIST,375)
IF(.NOT.BZERO) GO TO 959
DO 910 J=1,NXTERM
R(J)=0.0
DO 910 I=1,NXTERM
CALL LOC(I,J,IR)
R(J)=R(J)+ZEAN(I)*SUMXI(IR)
910 CONTINUE
XXT=0.0
DO 920 J=1,NXTERM
920 XXT=XXT+ZEAN(J)*R(J)
DO 930 K=1,NODEP
930 DEVB(1,K)=SQRT(ERRMS(K)*(RNOOB+XKT))
K=0
WRITE(LIST,380) K,(DEVB(1,J),J=1,NODEP)
959 IR=IR+J
DO 970 J=1,NXTERM
IR= IR+J
DO 960 K=1,NODEP
DEV(K) =SQRT(ERRMS(K)*SUMXI(IR))
960 CONTINUE
WRITE(LIST,380) JOUT(J), (DEV(J,K),KR=1,NODEP)
970 CONTINUE
RETURN
C
CFORMATS

1001 FORMAT(42H ANOVA OF REGRESSION ON DEPENDENT VARIABLE 15)
1002 FORMAT(1H 79(1H*)/79H SOURCE SUMS OF SQUARES DEG
XRES OF FREEDOM MEAN SQUARES /1H 79(1H-))
1003 FORMAT(17H REGRESSION G20.8, 5X, I10, 5X, G20.8)
1004 FORMAT(17H RESIDUAL G20.8, 5X, I10, 5X, G20.8)
1005 FORMAT(1H 79(1H-))
1006 FORMAT(17H TOTAL G20.8, 5X, I10)
1007 FORMAT(1H 79(1H*))

2001 FORMAT(1X/1X/22H ANOVA OF LACK OF FIT )
2002 FORMAT(17H LACK OF FIT G20.8, 5X, I10, 5X, G20.8)
2003 FORMAT(1X/22H REPLICATION G20.8, 5X, I10, 5X, G20.8)
2004 FORMAT(28H F = MS(LDF)/MS(REPS) = F10.3 )

370 FORMAT(74H1 SUMS OF SQUARES DUE TO EACH VARIABLE IF IT WERE LAST T
X0 ENTER REGRESSION )
375 FORMAT(115H2 STANDARD DEVIATION OF REGRESSION COEFFICIENTS (DERIVE
X0 FROM DIAGONAL ELEMENTS OF (X TRANSPOSE X) INVERSE MATRIX ))
380 FORMAT(1H 13.9G14.6)
1500 FORMAT(40H R SQUARED = SSQ(REG) / SSQ(TOT) = F8.6,
X 5X, 4HR = F7.6)
1600 FORMAT(34H STANDARD ERROR OF ESTIMATE G14.6)
1700 FORMAT(24H USING POOLING STRATEGY 12, 29H THE ERROR MEAN SQUARE =
X G14.7, 26H WITH DEGREES OF FREEDOM = 16)
1750 FORMAT(5X, 19HF = MS(REG)/MS(ERR) = F6.2, 5X, 13HCMPARE TO F(12, 1H, 13, 1
XH))

980 WRITE(LIST, 981)
981 FORMAT(41H ZERO RESIDUAL DEGREES OF FREEDOM. STOP. )
STOP
END

LIBFTC RECTX

SUBROUTINE  RECT(IROW, JJCOL, IMAX, JMAX, A, B, FMT, II)
DIMENSION  A(IMAX, JMAX), FMT(14), XOUT(8)
DOUBLE PRECISION  B, DXOUT
DIMENSION  B(IMAX, JMAX), DXOUT(8)
COMMON/SMAI/L/DUM(15), LIST
DATA  J8/8/
LOGICAL  OUT
OUT =.FALSE.
JTIMES=0
JCOL=JJCOL
5 JNXT=JCOL-J8
IF(JNXT) 10, 20, 30
10 JP=JCOL
GO TO 40
20 JP=J8
GO TO 40
30 JCOL=JNXT
JP=J8
GO TO 50
40 OUT=.TRUE.
50 DO 100 I=1, IROW
GO TO (55, 75), II
55 CONTINUE
DO 60 J=1, JP
JJ=JTIMES +J

63
SUBROUTINE PREDIC

PURPOSE
1) READ INPUT LEVELS OF INDEPENDENT VARIABLES AND COMPUTE
   A PREDICTED RESPONSE FROM THE ESTIMATED REGRESSION EQUATION.
2) COMPUTE VARIANCE AND STANDARD DEVIATION OF THE PREDICTED
   MEAN VALUE AND A SINGLE FURTHER OBSERVATION.

SUBROUTINES NEEDED
TRANS
LOC

REMARKS
VALUES FOR DEPENDENT VARIABLES ARE NOT NECESSARY FOR THE
PREDICTING OF VALUES. HOWEVER, A DUMMY VALUE MAY NEED TO
BE SUPPLIED IF A ZERO (BLANK) INPUT VALUE WILL CAUSE AN
IMPOSSIBLE OPERATION TO BE ATTEMPTED DURING THE
TRANSFORMATIONS.

******************************************************************************

SUBROUTINE PREDIC
COMMON /FRMTS/FMT(13),FMTTRI(14)
COMMON/BIG/B(60,9),SUMXY(60,9),SUMXX(1830),SUMXXI(1830)
DOUBLE PRECISION B,SUMXY,SUMXX,SUMXXI
COMMON/MED/B(60,9),SUMX(69),SUMX2(69),SUMY(9),Z(69),
   X CON(99),ERRMS(9),IDENT(13),IDOUT(60),NCON(200),NTERM(69),
X NTR(100),NXGDD(100),POOLED(9),REPVAR(9),RESMS(9),X(99)
DOUBLE PRECISION B0,SUMX,SUMX2,SUMY,Z
COMMON/SMALL/ BYPASS,BZERO,DELETE, FIRST, IFCHI, IFSSR,
   X IFFT, IFWT, INPUT, INPUT5, INTER,
X ISTRAT, JCOL, KUNNO, LENGTH, LIST,
X NERROR, NODEP, NODEP, NOTERM,
X NOVAR, NPDEG, NRES, NTRANS, WHERE,
X P, PREDC, REPS, RMT,
X STORY, STORY, STORYX, TAMI, WEIGHT,
X ERKXO, ECUMNY, IDOUT, ICOL
LOGICAL ECONOMY
LOGICAL BYPASS, BZERO, DELETE, IFCHI, IFSSR,
XIFSSR, IFFT, IFWT, REPS, PREDC, PREDICT,
XSTORY, STORY, STORY, FIRST, ERKXO

64
DOUBLE PRECISION WEIGHT, RWT, TOTWT
DIMENSION YCALC(9), V(60), VARC(9), SEEM(9), XVARP(9), SEEPE(9)
DOUBLE PRECISION XXT, v
EQUIVALENCE (YCALC(1), SUMXX(1)), (V(1), SUMXX(150)),
X (VARP(1), SUMXX(71)), (SEEM(1), SUMXX(80)), (VARP(1), SUMXX(89))
X (SEEPE(1), SUMXX(98))
EQUIVALENCE (RNOOB, RWT)

C
C********************************************************************
C
C IF (NOTERM.EQ.0) RETURN
WRITE (LIST,3)
READ (INPUT5,5) NPRED
C
DO 503 KK = 1, NPRED
C
105 READ (INPUT5, FMT) (X(I), I = 1, ICOL)
WRITE (LIST,110) (X(I), I = 1, ICOL)
125 CALL TRANS
DO 133 K = 1, JCOL
I = TERM(K)
X(K) = CON(I)
130 CONTINUE
WRITE (LIST,135) (X(I), I = 1, NOTERM)
C
C COMPUTE PREDICTED RESPONSE
140 DO 153 K = 1, NODEP
YCALC(K) = B0(K)
IF (NOTERM.EQ.0) YCALC(K) = 0.0
DO 153 J = 1, NOTERM
YCALC(K) = YCALC(K) + B(J,K)*X(J)
150 CONTINUE
C
C COMPUTE VARIANCE AND STANDARD DEVIATION OF REGRESSION LINE
C AND VARIANCE AND STANDARD DEVIATION OF PREDICTED VALUE
C AT THE POINT XO
C
DO 253 K = 1, NOTERM
V(K) = 3.000
DO 253 J = 1, NOTERM
CALL LOC(J,K,IR)
V(K) = V(K) + (X(J) - ZEAN(J)) * SUMXX(I,IR)
250 CONTINUE
XXT = 0.000
DO 273 K = 1, NOTERM
XXT = XXT + (X(K) - ZEAN(K)) * V(K)
275 CONTINUE
XRNUOB = RNUOB
IF (NOTERM.EQ.0) XRNUOB = 0.0
DO 300 K = 1, NODEP
VARM(K) = ERRMS(K) * (XRNUOB + XXT)
SEEM(K) = SQRT (VARM(K))
VARP(K) = ERRMS(K) + VARM(K)
SEEPE(K) = SQRT (VARP(K))
300 CONTINUE
WRITE (LIST,310) (YCALC(K), K = 1, NODEP)
WRITE (LIST,320) (VARM(K), K = 1, NODEP)
WRITE (LIST,320) (SEEM(K), K = 1, NODEP)
WRITE (LIST,320) (VARP(K), K = 1, NODEP)
WRITE (LIST,320) (SEEPE(K), K = 1, NODEP)
C
C
CONTINUE
RETURN
3 FORMAT(54H FOR EACH SET OF INDEP VARIABLES THERE IS COMPUTED...
20H PREDICTED RESPONSE
29H VARIANCE OF REGRESSION LINE
34H STANDARD DEVIATION OF REGRESSION
29H VARIANCE OF PREDICTED VALUE
39H STANDARD DEVIATION OF PREDICTED VALUE )
5 FORMAT(14)
110 FORMAT(39H INPUT DATA FOR THIS PREDICTED RESPONSE /(1H 9G14.6))
135 FORMAT(56HK INDEPENDENT TERMS ACCORDING TO FINAL REGRESSION MODEL
14H /(1H 9G14.6))
310 FORMAT( 55HK PREDICTED RESPONSE FOR ABOVE INDEP VARIABLES
14H /(1H 9G14.6))
320 FORMAT(1H 9G14.6)
END

SUBROUTINE SUMUPX

SUBROUTINE SUMUPS
PURPOSE
1) CALCULATE (X TRANSPOSE X) AND (X TRANSPOSE Y) MATRICES ONE
   OBSERVATION AT A TIME.
2) COMPUTE TOTAL OF THE WEIGHTS
   ** BOTH CALCULATIONS ARE IN DOUBLE PRECISION
SUBROUTINES NEEDED
LOCAL

C********************************************************************
SUBROUTINE SUMUP
COMMON/BIG/B(60,9),SUMXY(60,9),SUMxx(1830),SUMXI(1830)
DOUBLE PRECISION B,SUMXY,SUMxx,SUMXI
COMMON/MEU/B(9),SUMx(69),SUMx2(69),SUMy2(9),ZEN(69),
CON(99),ERRMS(9),IDENT(13),IDOUT(60),NCON(20G),NTERM(69),
NTRAN(100),NXCOD(100),POOLED(9),REPVAR(9),RESMS(9),X(99)
DOUBLE PRECISION SUMX,SUMx2,ZEAN
COMMON/SMALL/ BYPASS,BZERO,DELETE, FIRST, IFCHI, IFSSK,
X IFTT, IFWT, INPUT, INPUT5, INTER,
X ISTRAT, JCOL, KONNO, LENGTH, LIST,
X NERROR, NDDEP, NOOB, NOTERM,
X NVAR, NPDEG, NRES, NTRANS, NWHERE,
X P, PREDCT, REPS, RWT,
X STORYI, STORYY, STORYX, TOTWT, WEIGHT,
X ERRFXD, ECONMY, IOUT, ICOL
LOGICAL ECONMY
LOGICAL BYPASS, BZERO, DELETE, IFCHI,
X IFSSR, IFTT, IFWT, REPS, PREDCT,
X STORY, STORYX, STORYY, FIRST, ERRFXD
DOUBLE PRECISION RWT,TOTWT,WEIGHT
DOUBLE PRECISION DUB1,DUB2

DO 113 I=1,JCOL
SUMX(I)=SUMX(I)+X(I)*WEIGHT
113 CONTINUE

C********************************************************************
SUBROUTINE BORD

PURPOSE

TO COMPLETE THE INVERSION OF A SYMMETRIC POSITIVE DEFINITE
MATRIX A OF ORDER N GIVEN THAT THE UPPER LEFT SUB-
MATRIX OF ORDER N-1 HAS ALREADY BEEN INVERTED.

SUBROUTINES NEEDED

LOC

REMARKS

ONLY THE UPPER TRIANGULAR PART OF A IS STORED AS A
VECTOR IN THE ORDER A(1,1),A(1,2),A(2,2),A(1,3),...ETC

SUBROUTINE BORD(IORDER,A)

DIMENSION BETA(60),A(1)

DOUBLE PRECISION A,ALPHA, RALPHA ,BETA

ALPHA= 0.0D0
NM1= IORDER-1
IF(NM1) 100,100,200
100 A(1) = 1.0/A(1)
GO TO 600
200 M=NM1*(NM1+1)/2
LEN = M + IORDER
DO 400 I=1,NM1

CONTINUE
IR=0
DO 100 K=1,NOTERM
DUB1=X(K)
DO 90 J=1,NODEP
KBAR=J+NOTERM
DUB2=X(KBAR)
SUMXY(K,J)=SUMXY(K,J)+DUB1*DUB2*WEIGHT
90 CONTINUE
DO 50 I=1,K
IR=IR+L
DUBZ=K(I)
SUMXX(IR)=SUMXX(IR)+DUB1*DUB2*WEIGHT
50 CONTINUE
DO 15 J=1,NODEP
KBAR=NOTERM + J
DUB1=X(KBAR)
SUMYZ(J)=SUMYZ(J)+DUB1*DUB1*WEIGHT
15 CONTINUE
TOTWT= TOTWT+WEIGHT
RETURN
END
BETA(I) = 0.000
MI = M+I
DO 350 J=1,NM1
CALL LOG(I,J,II)
MJ = M+J
BETA(I) = BETA(I) - A(II)*A(MJ)
350 CONTINUE
ALPHA = ALPHA + A(MI)*BETA(I)
400 CONTINUE

C
C
ALPHA = ALPHA + A(LEN)
RALPHA = 1.000/ALPHA
A(LEN) = RALPHA
450 CONTINUE

C
DO 500 I=1,NM1
DO 500 J=I,NM1
CALL LOG(I,J,II)
A(II) = A(II) + BETA(I)*BETA(J)*RALPHA
500 CONTINUE

C
DO 550 J=1,NM1
MJ = M+J
A(MJ) = BETA(J)*RALPHA
550 CONTINUE

C
600 CONTINUE
RETURN
END

$IBTC MFIXX

SUBROUTINE MFIX

C**************************************************************************
COMM /FRMTS/ FMT(13),FMTTRI(14)
COMMON/BIG/B(60,9),SUMXY(60,9),SUMXX(1830),SUMXI(1830)
DOUBLE PRECISION SUMXY,SUMXX,SUMXI
COMMON/MED/B(60,9),SUMX(60,9),SUMXX(60,9),SUMXY(60,9),ZEAN(60,9),
    X CON(99),ERRMS(9),IDENT(13),IDOUT(60),NCON(200),NTERM(69),
    X NTRANS(100),NXCOD(100),POOLED(9),REPVAR(9),RESMS(9),X(99)
DOUBLE PRECISION BYPASS, BZERO, DELETE, FIRST, IFCHI, IFSSR,
    X FTIT, IFWT, INPUT, INPUT5, INTER,
    X ISTRAT, JCOL, KONNO, LENGTH, LIST,
    X NERROR, NODEP, NQOB, NOTERM,
    X VOVAR, NPDEG, NRES, NTRANS, NWHERE,
    X P, PREDCT, REPS, RWT,
    X STORI, STORLC, STORYX, TOTWT, WEIGHT,
    X ERRFXD, ECONNY, IOOUT, ICOL,
LOGICAL ECONMY
DOUBLE PRECISION RWT, TOTWT, WEIGHT
LOGICAL BYPASS, HZERO, DELETE, IFCHI,
    X IFSSR, FTIT, IFWT, REPS, PREDCT,
    X STORI, STORYX, STORYI, FIRST, ERRFXD
C**************************************************************************
C IF(EC)25MY) GO TO 500
IF(BZERO) GO TO 500
WRITE(LIST,530)
WRITE (LIST,540) (SUMX (I),I=1,JCOL)
WRITE(LIST,560)
CALL TRIANG(X,SUMXX,NOTERM,8,FMTTRI,2)
WRITE(LIST,565)
CALL RECT(NOTERM,NODEP,60,9,X,SUMXY,FMTTRI,2)
C******************************************************************************
C COMPUTE AND PRINT MEANS. COMPUTE AND PRINT THE(X TRANSPOSE X)
C MATRIX IN TERMS OF DEVIATIONS FROM MEAN. THE DEVIATIONS FORM
C OF (X T X) IS THE VARIANCE-COVARIANCE MATRIX OF THE
C INDEPENDENT VARIABLES.
500 CONTINUE
RWT=l.000/TUTWT
DO 570 I=1,JCOL
570 ZEAN(I)=SUMX(I)*RWT
WRITE(LIST,580)
WRITE(LIST,540) (ZEAN(I),I=1,JCOL)
IR = J
DO 600 J=1,NODEM
IR=IR + J
IF(.NOT.BZERO) GO TO 601
SUMXX(J)=SUMXX(IR)-SUMX(J)**2 * RWT
GO TO 600
601 SUMXX(J)=SUMXX(IR)
600 CONTINUE
602 CONTINUE
IR=1
DO 620 J=1,NODEM
DO 618 K=1,NODEM
IF(.NOT.BZERO) GO TO 618
KBAR=V OTERM+K
SUMXY(J,K)=SUMXY(J,K)-SUMX(J)*SUMX(KBAR)*RWT
618 CONTINUE
619 DO 620 K=1,J
IF(.NOT.BZERO) GO TO 6191
SUMXX(I,R)=SUMXX(IR)-SUMX(K)*SUMX(J)*RWT
6191 SUMXX(I,R)=SUMXX(IR)/DSQRT(SUMX2(J)*SUMX2(K))
6194 IR=IR+1
620 CONTINUE
IF(.NOT.BZERO) GO TO 6220
DO 6210 J=1,NODEM
K=NOTERM+J
SUMY2(J)=SUMY2(J)-SUMX2(J)*RWT
6210 CONTINUE
6220 CONTINUE
C******************************************************************************
IF(EC)25NY) GO TO 622
IF(.NOT.BZERO) GO TO 621
WRITE(LIST,625)
CALL TRIANG(X,SUMXX,NOTERM,8,FMTTRI,2)
WRITE(LIST,630)
CALL RECT(NOTERM,NODEP,6C,9,X,SUMXY,FMTTRI,2)
621 WRITE(LIST,670)
CALL TRIANG(X,SUMXXI,NOTERM,8,FMTTRI,2)
622 CONTINUE
C******************************************************************************
530 FORMAT(1HO,32H SUMS OF INDEP AND DEP VARIABLES )
$IBFTC LOCXXX

SUBROUTINE LOC(I,J,IR)
IX= I
JX= J
20 IF(IX-JX) 22,24,24
22 IRX= IX + (JX*JX-JX)/2
GO TO 36
24 IRX= JX + (IX*IX - IX)/2
36 IR= IRX
RETURN
END

$IBFTC XTRANS

SUBROUTINE TRANS
C*******************************************************************************
C
COMMON/BIG/B(60,9),SUMXY(60,9),SUMXX(1830),SUMXXII(1830)
DOUBLE PRECISION B,SUMX,SUMXX,SUMXI
COMMON/MED/B0(9),SUMX(69),SUMXX(69),SUMXY(2)(9),ZENAN(69),
X CITIN(99),ERRMS(9),IDENT(13),DOUT(60),NCOL(2)(9),NTFRM(69),
X NTRAN(100),NXCOD(100),POOLED(9),REPVAR(9),REMS(9),X(99)
DOUBLE PRECISION B0,SUMX,SUMXX,SUMXI
COMMON/SMALL/ BYPASS,BZERO,DELETE, FIRST, IFCHI, IFSSR,
X IFFT, IFWT, INPUT, INPUT5, INTER,
X ISTRAT, JCOL, KUNNO, LENGTH, LIST,
X VERROR, NODEP, NOOB, NOTERM,
X NVAR, NPDEG, NRES, NTRANS, NWHERE,
X P, PREDICT, REPS, RWT,
X STORYI, STORYC, STORYX, TOTWT, WEIGHT,
X ERRFXD, ECONMY, IOUT, ICOL
LOGICAL ECONMY
LOGICAL BYPASS, CZERO, DELETE, IFCHI,
XIFSSR, IFFT, IFWT, REPS, PREDICT,
XSTORYC, STORYX, STORYI, FIRST, FRRFXD
DOUBLE PRECISION RWT,TOTWT,WEIGHT
C
C*******************************************************************************
C THIS SUBROUTINE PERFORMS TRANSFORMATIONS IF THIS OPTION IS
C REQUESTED.
C
TRANSFORMATION SET NUMBER.

CONSTANT NUMBER TO USE.

DERIVED CONSTANT.

NUMBER OF TRANSFORMATION REQUESTED.

VARIABLE NUMBER

80 DO 300 K=1,NTRANS
   I=NCON(2*K-1)
   IF(I)130,100,110
100 CONS=0.0
   GO TO 120
110 CONS=CONS(I)
120 I=NCCOD(K)
   Y=X(I)
   MTRAN = NTRAN(K)
   IF(MTRAN.LE.0) MTRAN=32
150 CONS=Y+CONS
   GO TO 460
160 CONS=Y*CONS
   GO TO 460
170 CONS=CONS/Y
   GO TO 460
180 CONS=EXP(Y)
   GO TO 460
190 CONS=Y+CONS
   GO TO 460
200 CONS=ALOG(Y)
   GO TO 460
210 CONS=ALOG10(Y)
   GO TO 460
220 CONS=SIN(Y)
   GO TO 460
230 CONS=COS(Y)
   GO TO 460
240 CONS=SIN(3.14159265*(CONS*Y)
   GO TO 460
250 CONS=COS(3.14159265*(CONS*Y)
   GO TO 460
260 CONS=1.0/Y
   GO TO 460
270 CONS=EXP(CONS/Y)
   GO TO 460
280 CONS=EXP(CONS/(Y*Y))
   GO TO 460
290 CONS=SQRT(Y)
   GO TO 460
300 CONS=1.0/SQRT(Y)
   GO TO 460
310 CONS=CONS**Y
   GO TO 460
320 CONS=10.0**Y
   GO TO 460
330 CONS=SINH(Y)
   GO TO 460
340 CONS=COSH(Y)
   GO TO 460
350 CONS=(1.0-COS(Y))/2.0
   GO TO 460
360 CONS=ATAN(Y)
    GO TO 460
370 CONS=ATAN2(Y,CONS)
    GO TO 460
380 CONS=Y*Y
    GO TO 460
390 CONS=Y*Y*Y
    GO TO 460
400 CONS=ARSIN(SQRT(Y))
    GO TO 460
410 CONS=2.0*3.14159265*Y
    GO TO 460
420 CONS=1.0/(2.0*3.14159265*Y)
    GO TO 460
430 CONS=ERF(Y)
    GO TO 460
440 CONS=SUMMAY(Y)
    GO TO 460
442 CONS=Y/CONS
    GO TO 460
450 CONS=Y
460 I=NCOV(2*K)
    CON(I)=CONS
    IF(I=53) 500,500,490
490 XI(I)=DNS
    CONTINUE
    RETURN
END

SIBFTC OUTPLX

C******************************************************************************
C
C******************************************************************************

C

SUBROUTINE OUTPLT(PNCH)
COMMON/BIG/B(60,9),SUMXY(60,9),SUMXX(1830),SUMXXI(1830)
DOUBLE PRECISION B, SUMXY, SUMXX, SUMXXI
COMMON/MED/B(60,9),SUMX(9),SUMX2(9),SUMY2(9),ZEN(9),
  X CON(99),ERRMS(9),IDENT(13),IDOUT(60),NCON(200),INTERM(99),
  X NTRAN(100),NXCOD(100),POOLED(9),REPVAR(9),REMS(9),X(99)
DOUBLE PRECISION BD, SUMX, Su MX2, SUMY2, ZEAN
COMMON/SMALL/
  BYPASS, BZERO, DELETE, FIRST, IFCHI, IFSSR,
  X IFFT, IFWT, INPUT, INPUT5, INTER,
  X ISTRAT, JCOL, KUNNO, LENGTH, LIST,
  X NERROR, NODEP, NOOB, NOTEKM,
  X NPVAR, NPDEG, NRES, NTRANS, NWHERF,
  X P, PREDCT, REPS, RWT,
  X STORI, STORIC, STORIX, TOTWT, WEIGHT,
  X ERRFD, ECONMY, IOUH, ICL
LOGICAL ECONMY
DOUBLE PRECISION RWT, TOTWT, WEIGHT
LOGICAL BYPASS, BZERO, DELETE, IFCHI,
XIFSSR, IFFT, IFWT, REPS, PREDCT,
XSTORI, STORIC, STORIX, FIRST, ERRFD
COMMON/MAK/MAXPLT
INTEGER CELLS, PLUS1
DIMENSION BOUND(45), CELLBD(21), OBS(20,9), RCT(212), X STDIRR(9), VAR(9), YCALC(9), YDIFR(9), X Z(9)
EQUIVALENCE (VAR, RESMS)
ODATA BOUND(45), 67448907, 67448907, 67448907, 67448907
1, 31853932, 67448907, 67448907, 67448907
2, 52439379, 84162001, 67448932, 67448922
3, 43072721, 67448922, 67448921, 67448921
4, 18001235, 36610621, 67448921, 67448921
5, 156756653, 146521688, 67448932, 67448932
6, 48877614, 67448930, 67448930, 67448930
7, 1, 13971028, 13971028, 13971028
8, 58945544, 76470731, 67448930, 67448930
9, 1, 13971028, 13971028, 13971028
A, 52440000, 67448801, 67448801, 67448801
B, 1, 28154233, 1, 64490172
LOGICAL SAVRES
DIMENSION RESPLT(1)
DIMENSION SKEW(9), SKUR(9)
EQUIVALENCE (RESPLT, SUMXY)
DIMENSION ICHAR(9), HCHAR(9)
DATA(ICHAR(I) = I = 1,9) / &
HRESIDUT, HALS, FD, HVAR
X, 96-1 VS 1, HREDICT, 6HED VAL, HUE, / &
LOGICAL PNC

JCOT = NOTERM + NODEP
NUVAR = NOTERM + 1
BYPASS = .FALSE.
COUNT = 0
SAVRES = .FALSE.
ITPLT = NOOB*(NWHERE+2*NODEP)
IF(ITPLT.LE.MAXPLT) SAVRES = .TRUE.
CALL LRLEGN(IDENT(10), 24, 0, 1, 4, 5, 1, 0)

IF(NO3B-20) 110, 120, 120
BYPASS = .TRUE.
GO TO 125
120 CELLS = NOOB/5
CELLS = MINO(CELLS, 20)
I = MOD(CELLS, 2)
IF(1, 4E0) CELLS = CELLS + 1
FCELLS = FLOAT(CELLS)
PLUS1 = CELLS + 1
MINUS1 = CELLS - 1
NESEGCH = CELLS - 3
IR = CELLS/2 - 1
IC = IR*(IR-1)/2
IS = IR + 2
DO 122 J = 1, IR
IC = IC + 1
IBC = IS - J
IRC = IS + J
CELLBD(I8C) = BOUND(IC)
CELLBD(IRC) = BOUND(IC)
CONTINUE
CELLBD(1)=-1.0E+37
CELLBD(PLUS1)=1.0E+37
CELLBD(IS)=0.0
DO 124 K=1,NODEP
DO 124 I=1,CELLS
OBS(I,K)=0.0
124 CONTINUE

DO 125 K=1,NODEP
SKEW(K)=0.0
SKUR(K)=0.0
STDERR(K)=SQRT(ERRMS(K))
125 CONTINUE
WRITE(LIST,135)

DO 140 J=1,NOOB
READ(INTER) (X(I),I=1,69),WEIGHT
IF(.N3T.SAVRES) GO TO 141
INPLT=WHERE
DO 140 I=1,INPLT
K=(I-1)*NOOB+J
140 RESPLT(K)=X(I)

DO 142 I=1,NOTERM
K=IDOUT(I)
X(I)=X(K)
142 CONTINUE

KBAR=WHERE
DO 143 I=1,NODEP
IC=NOTERM+I
KBAR(KBAR+I)=KBAR+1
X(IC)=X(KBAR)
143 CONTINUE

DO 150 K=1,NODEP
YCALC(K)=BO(K)
IF(.N3T.BZERO) YCALC(K)=0.0
KBAR(KBAR)=K+NOTERM
DO 150 I=1,NOTERM
YCALC(K)=YCALC(K)+B(I,K)*X(I)
150 CONTINUE

ACTDEV=X(KBAR)-YCALC(K)
YDYF(K)=ACTDEV
Z(K)=ACTDEV/STDERR(K)
A=ACTDEV**3
SKEW(K)=SKEW(K)+A
SKUR(K)=SKUR(K)+A*ACTDEV

IF(.N3T.SAVRES) GO TO 179
K=INPLT*NOOB+J
KBAR(KBAR)=K+NOOB
DO 175 I=1,NODEP
ITC=(I-1)*NOOB
ISC=K+ITC
IS=KBAR+ITC
RESPLT(IS)=YCALC(I)
RESPLT(IS)=Z(I)
175 CONTINUE
CONTINUE
WRITE(LIST,180) (X(K),K=NUVAR,JCOL)
WRITE(LIST,190) (YCALC(K),K=1,NODEP)
WRITE(LIST,200) (YDIFR(K),K=1,NODEP)
IF(PNZH) PUNCH 5250, J(YDIFR(K),YCALC(K),K=1,NODEP)
WRITE(LIST,210) (Z(K),K=1,NODEP)
IF(BYPASS) GO TO 410

C

DO 250 K=1,NODEP
DO 230 I=1,PLUS1
IF(Z(K)-CELLBD(I)) 220,220,230
220 OBS(I-1,K)=OBS(I-1,K)+1.0
GO TO 250
230 CONTINUE
250 CONTINUE

C

410 KOUNT = KOUNT +1
IF(KOUNT.LT.10) GO TO 430
WRITE(LIST,270) IDENT
KOUNT=0
430 CONTINUE

C

434 CONTINUE

C

439 CONTINUE

C

440 CONTINUE

C

450 CONTINUE

C

460 CONTINUE

C

470 CONTINUE

C

480 CONTINUE

C

490 CONTINUE

C

500 CONTINUE

C

510 CONTINUE

C

520 CONTINUE

C

530 CONTINUE

C

540 CONTINUE

C

550 CONTINUE

C

560 CONTINUE

C

570 CONTINUE

C

580 CONTINUE

C

590 CONTINUE

C

600 CONTINUE

C

610 CONTINUE

C

620 CONTINUE

C

630 CONTINUE

C

640 CONTINUE

C

650 CONTINUE

C

660 CONTINUE

C

670 CONTINUE

C

680 CONTINUE

C

690 CONTINUE

C

700 CONTINUE

C

710 CONTINUE

C

720 CONTINUE

C

730 CONTINUE

C

740 CONTINUE

C

750 CONTINUE

C

760 CONTINUE

C

770 CONTINUE

C

780 CONTINUE

C

790 CONTINUE

C

800 CONTINUE

C

810 CONTINUE

C

820 CONTINUE

C

830 CONTINUE

C

840 CONTINUE

C

850 CONTINUE

C

860 CONTINUE

C

870 CONTINUE

C

880 CONTINUE

C

890 CONTINUE

C

900 CONTINUE

C

910 CONTINUE

C

920 CONTINUE

C

930 CONTINUE

C

940 CONTINUE

C

950 CONTINUE

C

960 CONTINUE

C

970 CONTINUE

C

980 CONTINUE

C

990 CONTINUE

C

1000 CONTINUE

C

1010 CONTINUE

C

1020 CONTINUE

C

1030 CONTINUE

C

1040 CONTINUE

C

1050 CONTINUE

C

1060 CONTINUE

C

1070 CONTINUE

C

1080 CONTINUE

C

1090 CONTINUE

C

1100 CONTINUE

C

1110 CONTINUE

C

1120 CONTINUE

C

1130 CONTINUE

C

1140 CONTINUE

C

1150 CONTINUE

C

1160 CONTINUE

C

1170 CONTINUE

C

1180 CONTINUE

C

1190 CONTINUE

C

1200 CONTINUE

C

1210 CONTINUE

C

1220 CONTINUE

C

1230 CONTINUE

C

1240 CONTINUE

C

1250 CONTINUE

C

1260 CONTINUE

C

1270 CONTINUE

C

1280 CONTINUE

C

1290 CONTINUE

C

1300 CONTINUE

C

1310 CONTINUE

C

1320 CONTINUE

C

1330 CONTINUE

C

1340 CONTINUE

C

1350 CONTINUE

C

1360 CONTINUE

C

1370 CONTINUE

C

1380 CONTINUE

C

1390 CONTINUE

C

1400 CONTINUE

C

1410 CONTINUE

C

1420 CONTINUE

C

1430 CONTINUE

C

1440 CONTINUE

C

1450 CONTINUE

C

1460 CONTINUE

C

1470 CONTINUE

C

1480 CONTINUE

C

1490 CONTINUE

C

1500 CONTINUE

C

1510 CONTINUE

C

1520 CONTINUE

C

1530 CONTINUE

C

1540 CONTINUE

C

1550 CONTINUE

C

1560 CONTINUE

C

1570 CONTINUE

C

1580 CONTINUE

C

1590 CONTINUE

C

1600 CONTINUE

C

1610 CONTINUE

C

1620 CONTINUE

C

1630 CONTINUE

C

1640 CONTINUE

C

1650 CONTINUE

C

1660 CONTINUE

C

1670 CONTINUE

C

1680 CONTINUE

C

1690 CONTINUE

C

1700 CONTINUE

C

1710 CONTINUE

C

1720 CONTINUE

C

1730 CONTINUE

C

1740 CONTINUE

C

1750 CONTINUE

C

1760 CONTINUE

C

1770 CONTINUE

C

1780 CONTINUE

C

1790 CONTINUE

C

1800 CONTINUE

C

1810 CONTINUE

C

1820 CONTINUE

C

1830 CONTINUE

C

1840 CONTINUE

C

1850 CONTINUE

C

1860 CONTINUE

C

1870 CONTINUE

C

1880 CONTINUE

C

1890 CONTINUE

C

1900 CONTINUE

C

1910 CONTINUE

C

1920 CONTINUE

C

1930 CONTINUE

C

1940 CONTINUE

C

1950 CONTINUE

C

1960 CONTINUE

C

1970 CONTINUE

C

1980 CONTINUE

C

1990 CONTINUE

C

2000 CONTINUE

C

2010 CONTINUE

C

2020 CONTINUE

C

2030 CONTINUE

C

2040 CONTINUE

C

2050 CONTINUE

C

2060 CONTINUE

C

2070 CONTINUE

C

2080 CONTINUE

C

2090 CONTINUE

C

2100 CONTINUE

C

2110 CONTINUE

RETURN

135 FORMAT(51H FOR EACH DEPENDENT TERM AND OBSERVATION IS PRINTED
X /31H OBSERVED RESPONSE (Y OBSERVED)
X /29H CALCULATED RESPONSE (Y CALC)
CRSPLT PROGRAM

CRSPLT accepts a subset of the data used for a NEWRAP problem. It can be used as a preregression analysis program to help formulate model equations to be analyzed with NEWRAP or it may be used as a postregression program by using punched output from NEWRAP to obtain more complex residual plots than direct use of NEWRAP allows.

When used as a preregression program, it will compute an $X'X$ and $C$ matrix including all the terms (independent and dependent) if requested. It can also compute eigenvalues and eigenvectors of the submatrix of $X'X$ corresponding to the independent variables.

When used as a postregression program, the punched output of residuals and predicted values from NEWRAP can be plotted against new functions of the independent variables.

The input is much the same as for NEWRAP. The seven sets of input are as follows:

(1) **IDENTIFICATION** (I, IDENT)(I2, 13A6): IDENT is Hollerith data used to identify the problem. I indicates the number of additional cards to be read for identification (columns 1 to 78).

(2) **PROBLEM SIZE** (NOVAR, NODEP, NOTERM, NOOB)(314, I5)

<table>
<thead>
<tr>
<th>Input</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOVAR</td>
<td>Number of input independent variables</td>
</tr>
<tr>
<td>NODEP</td>
<td>Number of input dependent variables</td>
</tr>
<tr>
<td>NOTERM</td>
<td>Number of terms in model equation</td>
</tr>
<tr>
<td>NOOB</td>
<td>Number of observations</td>
</tr>
</tbody>
</table>

(3) **TRANSFORMATIONS**: This input is the same as the transformations of NEWRAP except for the upper limit of 150 transformations.

(4) **FORMAT** (INPUT, FMT)(I2, 13A6): INPUT specifies the unit number the input data is stored on and FMT indicates the format for reading it.

$x /28H$ RESIDUAL $(Y_{OBS} - Y_{CALC})/Y_{DIF}$

$x /28H$ STUDENTIZED RESIDUAL $(Z)$

180 FORMAT(12H KY OBSERVED, 9G13.4)
190 FORMAT(12H Y CALC, 9G13.4)
200 FORMAT(12H Y DIF, 9G13.4)
5250 FORMAT(16, 4E16.8/(6X, 4E16.8))
210 FORMAT(12H STUDENTIZED, 9G13.4)
270 FORMAT(1H113A6rA2)
280 FORMAT(27H CHI-SQUARE STATISTIC WITH 16, 22H DEGREES OF FREEDOM =
$x \cdot G14.6/12H$ SKEWNESS = $G14.6/12H$ KURTOSIS = $G14.6$)
(5) PLOTTING REQUESTS

NOPLTS (14)
(INPLT, IYPLT)

One card supplies NOPLTS, the number of plots desired. The following cards supply pairs of integers indicating which pairs of terms to plot. The format is 4012 (i.e., 20 plots per card). The first number of the pair (INPLT) specifies the sequence number of the term to be used as the abscissa. The second number (IYPLT) specifies the sequence number of the term to be used as the ordinate. As an example, the following sequence of transformations and subsequent plotting requests would cause \( x_1^2 \) to be plotted against \( x_1 \) as well as against \( x_3^3 \):

```
MODEL SIZE 0003
TERMS 616263
TRANSFORMATIONS 0100061 01026162 01026263
CONSTANTS blank card
NOPLTS 0002
(INPLT, IYPLT) 0102 0302
```

(6) MATRIX REQUESTS (XTXC, EIGENC)(2L1): If XTXC is F, no matrix calculations are executed. If it is T, than an \( X'X, X'X \) deviation and a correlation matrix of all the NOTERM + NODEP terms appearing on the TERMS card are computed. If EIGENC is T, then the eigenvalues and eigenvectors of the submatrix corresponding to the independent terms (the first NOTERM terms) are calculated.

(7) DATA: Same usage as in NEWRAP.

An illustrative set of input is given, followed by the corresponding sample of output and a main program listing. The subprograms TRIANG, RECT, and EIGEN are required and are the same as in NEWRAP.

15 SAMPLE GRSFLT PROBLEM
DATA IS FROM DRAPER AND SMITH
APPLIED REGRESSION ANALYSIS (REFERENCE 4 OF NEWRAP REPORT)
CHAPTER 7

INITIAL MODFL EQ WAS
Y = CHAMBER PRESSURE
X1 = TEMPERATURE OF CYCLE
X2 = VIBRATION LEVEL
X3 = DROP(SHOCK)
X4 = STATIC FIRE
Y = \( b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_4 + \epsilon \)
THE FOLLOWING TERMS ARE BEING CREATED FOR RESIDUAL PLOTS

1  X1  2  X2  3  X3  4  X4
5  X1*X2  6  X1*X3  7  X1*X4  8  X2*X3
9  X2*X4  10  X3*X4  11  Y(PRED)  12  UNIT NO.

RESIDUALS

UNIT NO.  1  -75  0  -65  1.4
1  -0.3655047E+01  C  50550470E+01
2  175  0  C  15C  26.3
2  -0.7633190E+00  C  27633190E+02
7  0  0  -65  15C  29.4
3  0.2336681E+01  C  27633190E+02
8  0  0  C  165  -65  9.7
4  0.4644953E+01  C  50550470E+01
0  0  0  C  15C  32.9
5  0.5836681E+01  C  27633190E+02
10  -75  -75  0  C  15C  26.4
6  0.50644449E+00  C  25893355E+02
11  175  175  0  -65  8.4
7  0.6150389E+00  C  77849610E+01
14  -75  -75  -65  15C  28.4
8  0.25066446E+01  C  25893355E+02
15  175  175  165  -65  11.5
9  0.3715039E+01  C  77849610E+01
18  0  0  -65  15C  1.3
10  -0.37550470E+01  C  50550470E+01
19  0  0  C  165  15C  21.4
11  -0.56363189E+01  C  270633190E+02
20  0  -75  -65  -65  0.4
12  -0.34850838E+01  C  38850838E+01
21  0  175  165  15C  22.9
13  -0.63932328E+01  C  29793233E+02
24  0  0  0  -65  3.7
14  -0.1355470E+01  C  50550470E+01
3  0  -75  0  15C  26.5
15  0.664439E+00  C  258933550E+02
5  0  -75  0  15C  23.4
16  -0.24933555E+01  C  258933550E+02
16  0  -75  0  15C  26.5
17  0.60644439E+00  C  258933550E+02
4  0  175  0  -65  5.8
18  -0.19849611E+01  C  77849610E+01
6  0  175  0  -65  7.4
19  -0.3849611E+00  C  77849610E+01
17  0  175  0  -65  5.8
20  -0.19849611E+01  C  77849610E+01
12  0  -75  -65  15C  28.8
21  0.29066443E+01  C  258933550E+02
22  0  -75  -65  15C  26.4
22  0.56644494E+00  C  258933550E+02
13  0  175  165  -65  11.8
23  0.40150389E+01  C  77849610E+01
23  0  175  165  -65  11.4
BIRFTC CRSPLX

COMMON/BL1/ X(99),CUN(99),SUMX(70),SUMXX(2485),A(75,7)
X,XDATA(12000)
COMMON/A2/AMAT(70),XSTD(70),SUMX2(7C),NTRANS,NCON(300),
X NTERM(71),NTRAN(15C),NXCOD(15C)
DIMSILN IDENT(13),FMT(13),FMTTRK(14),CORR(1),FMTSGL(3)
DATA(FMTTRK(11),I=1,3)/6H(1H16,6H8G15,S2H6)/
DATA(FMTSGL(1),I=1,3)/6H(1H16,6H8G15,S2H6)/
LOGICAL XTC,GEIGEN
EQUIVALENCE (CORR,A)
DATA ICHAR(216),VS /
COMMON IXPIT(450),IYPLT(450)
DIMENSION ICHAR(3)

C**************************************************************************************
1C READ(5,110) IDENT
WRITE(6,111) IDENT
DO 100 J=1,19863
X(J)=0.0
100 CONTINUE
110 IF(I(1)EQ.120,120,115
115 READ(5,300) FMT
WRITE(6,301) FMT
I=I-1
GO TO 113
12C READ(5,112) NVAR,NUDEP,NTERM,NJOB
WRITE(6,305) NVAR,NUDEP,NTERM,NJOB
MVTERM=MTERM+NUDEP
L=MVTERM+NUDEP
IF(L.GT.12000) GO TO 10C0
READ (5,282) NTRANS,KONNU
IF(NTRANS)255,255,22C
22C READ(5,239) (NTERM(K),K=1,MVTERM)
WRITE(6,235) (NTERM(K),K=1,MVTERM)
READ(5,230) (NXCOD(I),NTRAN(I),NCON(2*I-1),NCON(2*I),I=1,NTRANS)
WRITE(6,240) (NXCOD(I),NTRAN(I),NCON(2*I-1),NCON(2*I),I=1,NTRANS)
IF(KNNU.NQ.255,255,25C
25C READ(5,260) (CON(I),I=1,KNNU)
WRITE(6,262) (CN(I),I=1,KNNU)
255 READ(5,3000) IUIN,FMT
WRITE(6,3001) IUIN,FMT
READ(5,282) NPLTS
IF(NPLTS.LE.0) GO TO 32C
IF(NPLTS.GT.300) GO TO 20CC
READ(5,230) (IXPIT(I),IYPLT(I),I=1,NPLTS)
WRITE(6,5000) (IXPIT(I),IYPLT(I),I=1,NPLTS)
32C CONTINUE
READ(5,4000) XTC,GEIGEN
NORD=NVAR+NUDEP
C**************************************************************************************
KNOOB = 1,N FLOAT(KNOOB)
DO 590 J=1,NOOB
READ(IUIN,FMT) (X(I),I=1,NORD)
IF(NTRANS).GT.450/.GT.450,.340
340 CALL TRANS
DO 430 K=1,MVTERM
I = NTERM(K)
X(K)=CN(I)
430 CONTINUE

C**************************************************************************************
CONTINUE
IF(.NOT.XTXC) GO TO 620
IR=0
DO 610 I=1,MVTERM
SUMX(I) = SUMX(I) + X(I)
DO 600 I=1,I
IR=IR+1
SUMXX(IR) = SUMXX(IR) + X(I)*X(I)
600 CONTINUE
610 CONTINUE
620 CONTINUE
IA = J - NOOB
DO 650 I=1,MVTERM
IX = IA + [I*NOOB
XDATA(IX) = X(I)
650 CONTINUE
690 CONTINUE
IF(.NOT.XTXC) GO TO 720
WRITE(6,1010)
CALL RECT(NOOB,MVTERM,NOOB,MVTERM,XDATA,FMTSGL)
WRITE(6,1020)
CALL TRIANG(SUMXX,MVTERM,8,FMTTRI)
R = 0
DO 700 I=1,MVTERM
IR = IR+1
SUMX2(IR) = SUMX2(IR) - SUMX(I)**2*NOOB
700 XMEAN(I) = SUMX(I)*NOOB
R = 1
DO 710 J=1,MVTERM
DO 710 K=1,J
SUMX(X) = SUMX(IR) - SUMX(K)*SUMX(J)*NOOB
CORR(IR) = SUMX2(IR)/ SQRT(SUMX2(J)*SUMX2(K))
IR = IR+1
710 CONTINUE
WRITE(6,1025)
CALL TRIANG(SUMXX,MVTERM,8,FMTTRI)
WRITE(6,1030)
CALL TRIANG(CORR,MVTERM,8,FMTTRI)
IF(.NOT.EIGENC) GO TO 720
CALL EIGEN(SUMXX,A,NOTERM,C)
WRITE(6,1040)
J = 0
DO 718 I=1,NOTERM
J = J+1
718 SUMXX(I) = SUMXX(J)
WRITE(6,1050) (SUMXX(I),I=1,NOTERM)
WRITE(6,1060)
CALL RECT(NOTERM,NOTERM,NOTERM,NOTERM,A,FMTTRI)
720 CONTINUE
CALL LRLEGN(IDENT,54,C,15,C,0)
CALL LRLEGN(IDENT(1C),24,0,14,5,1,0)
DO 990 K=1,NOPLTS
ICCHAR(1)=IXPLT(K)
ICCHAR(2)=LYPLT(K)
IS1=(IXPLT(K)-1)*NOOB+1
IS2=(LYPLT(K)-1)*NOOB+1
CALL LCVRT(ICHAR(1),L,ICHAR(1),1,6,C)
CALL LCVRT(ICHAR(2),L,ICHAR(2),1,6,C)
CALL LRLEG(ICHAR,18)
CALL LRPLT(XDATA(IS1),XDATA(IS2),NOOB)
990 CONTINUE
GO TO 10
IF (WRITE(6,45))L
STOP
55 FORMAT(62H THE REQUIRED NUMBER OF LOCATIONS EXCEEDS THE 12000 AVAILABLE)
STOP
WRITE(6,2*05) NOPLTS
STOP
265 FORMAT(25H THE MAX. NO. OF PLOTS IS 300) I8)
110 FORMAT(12,13A6)
111 FORMAT(1H1,13A6)
112 FORMAT(314,15)
120 FORMAT(2L1)
50°C FORMAT(1HK/(15(1X,12,1X,12,2X)))
301 FORMAT(1H 13A6, A2)
305 FORMAT(1H 316)
282 FORMAT(2I4)
230 FORMAT(4012)
235 FORMAT(11H TERMS ARE /(1H 3014))
245 FORMAT(25H THE TRANSFORMATIONS ARE /(1H 5(414,5X))]
26F FORMAT( 5F15.7)
262 FORMAT(19H THE CONSTANTS ARE /(1H 8G15.7))
161 FORMAT(16H THE DATA MATRIX)
172 FORMAT(26H THE X TRANSPOSE X MATRIX)
195 FORMAT(32H THE X TRANSPOSE X DEVIATIONS MATRIX)
1030 FORMAT(38H THE CORRELATION MATRIX)
1046 FORMAT(46H THE FOLLOWING ARE EIGENVALUES OF X TRANS X MATRIX)
1050 FORMAT(1H 8G15.7)
106F FORMAT(53H THE EIGENVECTORS BY COLUMNS IN SAME ORDER AS EIGENVALUES)
3010 FORMAT(12,13A6)
3061 FORMAT(1H 1E,2X,13A6)
FND

SIBFTX TRANSX

SUBROUTINE TRANSX
C******************************************************************************
C
C COMMON/(H1/,X(99),CUN(99),SUMX(7C),SUMXX(2485),A(7C,70)
X,XDATA(12000)
COMMON/H2/XMEAN(70),XSTD(7C),SUMX2(7C),NTRANS,NCON(30C),
X NTERM(70),NTRAN(12C),NXCOD(150)
C
C******************************************************************************
C
C THIS SUBROUTINE PERFORMS TRANSFORMATIONS IF THIS OPTION IS
C REQUESTED.
C
C
C K
C NCON(2*K-1) TRANSFORMATION SET NUMBER.
C NCON(2*K) CONSTANT NUMBER TO USE.
C NCON(2*K) DERIVED CONSTANT.
C NTRANS(K) NUMBER OF TRANSFORMATION REQUESTED.
C NXCOD(K) VARIABLE NUMBER.
C
AC NO 500 K=1, NTRANS
I=NCON(2*K-1)
IF(I(1100,10C,110

81
23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44
45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66
67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84
SAMPLE CASPLT PROBLEM
DATA IS FROM DRAPER AND SMITH
APPLIED REGRESSION ANALYSIS (REFERENCE 4 OF NEWRAP REPORT)
CHAPTER 7
INITIAL MODEL EU was
Y= CHAMBER PRESSURE
X1= TEMPERATURE OF CYCLE
X2= VIBRATION LEVEL
X3= DROP SHOCK
X4= STATIC FIRE

THE FOLLOWING TERMS ARE BEING CREATED FOR RESIDUAL PLOTS
1  X1  2  X2  3  X3  4  X4
4  X1*X2  6  X1*X3  7  X1*X4  8  X2*X3
9  X2*X4  10  X3*X4  11  YIPRED  12  UNIT NO.
13 RESIDUALS
6  1  12

THEM TRANSFORMATIONS ARE
6  1  13  3  13  4  13  5  13  6  13  7  13  8  13  9  13  10  13  11  13  12  13

THM DATA MATRIX
1  -75.66000
2  175.00000
3  -50.00000
4  -50.00000
5  150.00000
6  -75.00000
7  175.00000
8  -75.00000
9  175.00000
10  -50.00000
11  150.00000
12  -75.00000
13  175.00000
14  -50.00000
15  -75.00000
16  -75.00000
17  -75.00000
18  -50.00000
19  -75.00000
20  -75.00000
21  -75.00000
22  -75.00000
23  -75.00000
24  -75.00000

THM GO TO 460
CONIS=VINI
IF11410.470.470.490
47F CON1I+CONS
GO TO 500
45C (IN S = CONS=6AMHAlV
GO TO 4611
CONS=Y
INF (114m.470.4
RO CON I I = CONS
GO TO 500
CONI I I = CONS
IF (1-60) 5fl0.500.4
XI I = CONS
CONTINUE
K
E T LIRN
END
### THE X TRANSPOSE X MATRIX

<table>
<thead>
<tr>
<th>1</th>
<th>104750,0</th>
<th>29000C,0</th>
<th>18870C,0</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>33700,0</td>
<td>33500,0</td>
<td>-82250,0</td>
</tr>
<tr>
<td>3</td>
<td>14175,0</td>
<td>15050,0</td>
<td>32070,0</td>
</tr>
<tr>
<td>4</td>
<td>-12150,0</td>
<td>-53120,0</td>
<td>-872450,0</td>
</tr>
<tr>
<td>5</td>
<td>93125,0</td>
<td>-3394058E+07</td>
<td>521125,0</td>
</tr>
<tr>
<td>6</td>
<td>59375,0</td>
<td>2421050E+07</td>
<td>6644525,0</td>
</tr>
<tr>
<td>7</td>
<td>-59435,0</td>
<td>-521125,0</td>
<td>6445752,0</td>
</tr>
<tr>
<td>8</td>
<td>99130,0</td>
<td>-2349250E+07</td>
<td>-721425,0</td>
</tr>
<tr>
<td>9</td>
<td>372750,0</td>
<td>2063750E+07</td>
<td>4644540,0</td>
</tr>
<tr>
<td>10</td>
<td>-51375,0</td>
<td>-3645450,0</td>
<td>-9219750E+07</td>
</tr>
<tr>
<td>11</td>
<td>-32650,0</td>
<td>-3645450,0</td>
<td>-9219750E+07</td>
</tr>
<tr>
<td>12</td>
<td>-12660,0</td>
<td>-3645450,0</td>
<td>-9219750E+07</td>
</tr>
<tr>
<td>13</td>
<td>12660,0</td>
<td>-3645450,0</td>
<td>-9219750E+07</td>
</tr>
<tr>
<td>14</td>
<td>12660,0</td>
<td>-3645450,0</td>
<td>-9219750E+07</td>
</tr>
<tr>
<td>15</td>
<td>12660,0</td>
<td>-3645450,0</td>
<td>-9219750E+07</td>
</tr>
<tr>
<td>16</td>
<td>12660,0</td>
<td>-3645450,0</td>
<td>-9219750E+07</td>
</tr>
<tr>
<td>17</td>
<td>12660,0</td>
<td>-3645450,0</td>
<td>-9219750E+07</td>
</tr>
<tr>
<td>18</td>
<td>12660,0</td>
<td>-3645450,0</td>
<td>-9219750E+07</td>
</tr>
<tr>
<td>19</td>
<td>12660,0</td>
<td>-3645450,0</td>
<td>-9219750E+07</td>
</tr>
<tr>
<td>20</td>
<td>12660,0</td>
<td>-3645450,0</td>
<td>-9219750E+07</td>
</tr>
<tr>
<td>21</td>
<td>12660,0</td>
<td>-3645450,0</td>
<td>-9219750E+07</td>
</tr>
<tr>
<td>22</td>
<td>12660,0</td>
<td>-3645450,0</td>
<td>-9219750E+07</td>
</tr>
<tr>
<td>23</td>
<td>12660,0</td>
<td>-3645450,0</td>
<td>-9219750E+07</td>
</tr>
<tr>
<td>24</td>
<td>12660,0</td>
<td>-3645450,0</td>
<td>-9219750E+07</td>
</tr>
</tbody>
</table>

### THE CORRELATION MATRIX

<table>
<thead>
<tr>
<th>1</th>
<th>1.000000</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.379765</td>
</tr>
<tr>
<td>3</td>
<td>0.714513</td>
</tr>
<tr>
<td>4</td>
<td>0.369485</td>
</tr>
<tr>
<td>5</td>
<td>0.310154</td>
</tr>
<tr>
<td>6</td>
<td>0.369485</td>
</tr>
<tr>
<td>7</td>
<td>0.310154</td>
</tr>
<tr>
<td>8</td>
<td>0.369485</td>
</tr>
<tr>
<td>9</td>
<td>0.310154</td>
</tr>
<tr>
<td>10</td>
<td>0.369485</td>
</tr>
<tr>
<td>11</td>
<td>0.714513</td>
</tr>
<tr>
<td>12</td>
<td>0.310154</td>
</tr>
<tr>
<td>13</td>
<td>0.310154</td>
</tr>
</tbody>
</table>

### X TRANSPOSE X DEVIATIONS MATRIX

<table>
<thead>
<tr>
<th>1</th>
<th>10500,0</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>6250,0</td>
</tr>
<tr>
<td>3</td>
<td>11500,0</td>
</tr>
<tr>
<td>4</td>
<td>-2650,0</td>
</tr>
<tr>
<td>5</td>
<td>-53750,0</td>
</tr>
<tr>
<td>6</td>
<td>53750,0</td>
</tr>
<tr>
<td>7</td>
<td>-53750,0</td>
</tr>
<tr>
<td>8</td>
<td>316250,0</td>
</tr>
<tr>
<td>9</td>
<td>-316250,0</td>
</tr>
<tr>
<td>10</td>
<td>316250,0</td>
</tr>
<tr>
<td>11</td>
<td>-316250,0</td>
</tr>
<tr>
<td>12</td>
<td>316250,0</td>
</tr>
<tr>
<td>13</td>
<td>-316250,0</td>
</tr>
</tbody>
</table>

### RElations:**nil** FILE:**nil**
In optimum-seeking experimentation involving many independent variables, quadratic response surfaces are often used. A development of the most important aspect of the design and analysis of response surface experiments can be found in Davies (ref. 9) and Box and Hunter (ref. 10). A discussion of the interpretation of a quadratic surface fitted to a large experiment is given in reference 11.

The general form of a quadratic surface is given by

\[ y = b_0 + b'X + X'BX + \epsilon \]

\[ = b_0 + (b_1, b_2, \ldots, b_p) \begin{pmatrix} X_1 \\ \vdots \\ X_p \end{pmatrix} + \begin{pmatrix} \frac{1}{2}b_{11} & \frac{1}{2}b_{12} & \cdots & \frac{1}{2}b_{1p} \\ \frac{1}{2}b_{12} & b_{22} & \cdots & \frac{1}{2}b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{2}b_{1p} & \frac{1}{2}b_{2p} & \cdots & b_{pp} \end{pmatrix} \begin{pmatrix} X_1 \\ \vdots \\ X_p \end{pmatrix} + \epsilon \quad (26) \]

The analysis of such an equation is simplified by two calculations: (1) the calculation of the stationary point of the surface and (2) a transformation of axes to new independent variables which changes the \( B \) matrix to a diagonal matrix.

The stationary point of the surface is the solution \( X_s \) to

\[ \frac{\partial y}{\partial x_i} = 0 \quad i = 1, p \]

The transformation of the axes is given by the computation of the orthogonal matrix \( P \) which reduces \( B \) to a diagonal matrix; that is,
The $\lambda_i$ are the eigenvalues of $B$. The new variables are given by

$$z = P'X$$

where $P$ is the matrix whose columns are the eigenvectors of $B$. If two successive transformations of the variables are made as follows:

$$W = X - X_s$$

$$z = P'W$$

then equation (26) becomes

$$y - y_s = \lambda_1 Z_1^2 + \lambda_2 Z_2^2 + \ldots + \lambda_p Z_p^2$$  \hspace{1cm} (27)

where

$$y_s = b_0 + b'X_s + X_s'BX_s$$  \hspace{1cm} (28)

From examination of equation (27), some general conclusions can be drawn concerning the attainment of a maximized response. For example, consider just two of the possible results.

(1) Suppose all the $\lambda_i \leq 0$ and $X_s$ is near or in the region of $X$ at which the experiments were performed. Then clearly any deviation of $Z$ from $Z = 0$ will decrease the response. Thus $Z = 0$ (or equivalently $X = X_s$) is a maximum and is the combination of independent variables the experimenter seeks.

(2) Some $\lambda_i < 0$ and some $\lambda_i > 0$, and $X_s$ is close to the region of experimentation. Thus $X_s$ represents what is sometimes called a saddlepoint. Moving in some directions will cause a decrease in $y$ and moving in other directions will cause an increase in $y$. Thus the experimenter could move from $X_s$ in the direction that corresponds to the direction of the $Z$ which has the largest positive coefficient in equation (27). This will increase $y$ most rapidly from the value of $y_s$. 

86
The INPUT (Sample input is shown in fig. 8) is as follows:

1. Identification: One card, all 80 columns.
2. The number of independent variations JFAC. (14) (JFAC must be less than or equal to 15)
3. Variable Format. One card, all 80 columns.
4. $b_0$ (according to FORMAT in item (3)).
5. $b_1, \ldots, b_{JFAC}$ (according to format in item (3)).
6. $b_1, 1$

\[
\begin{array}{cccc}
    & b_{1,2} & b_{2,2} \\
    b_{1,3} & b_{2,3} & b_{3,3} \\
    \vdots & & & \vdots \\
    b_{1,JFAC} & b_{2,JFAC} & \ldots & b_{JFAC,JFAC}
\end{array}
\]
One READ statement for each line, according to the format in item (3). As an example, consider the following estimated equation:

\[ y = b_0 + b_1 x_1 + b_2 x_2 + b_3 x_3 + b_4 x_4 + b_{11} x_1^2 + b_{12} x_1 x_2 + b_{22} x_2^2 + b_{13} x_1 x_3 + b_{23} x_2 x_3 + b_{33} x_3^2 + b_{14} x_1 x_4 + b_{24} x_2 x_4 + b_{34} x_3 x_4 + b_{44} x_4^2 + b_{15} x_1 x_5 + b_{25} x_2 x_5 + b_{35} x_3 x_5 + b_{45} x_4 x_5 + b_{55} x_5^2 \]

\[ y = 147.2686 - 8.989120 x_1 - 6.817975 x_2 - 14.60964 x_3 + 9.248688 x_4 + 14.19698 x_5 + 1.805520 x_1^2 - 2.126719 x_1 x_2 - 1.475730 x_2^2 - 7.314218 x_1 x_3 - 1.620219 x_2 x_3 + 2.274270 x_3^2 + 1.310780 x_1 x_4 - 2.477188 x_2 x_4 + 2.236770 x_4^2 + 2.664464 x_1 x_5 + 1.827434 x_2 x_5 + 2.389934 x_3 x_5 + 6.014934 x_4 x_5 - 12.58198 x_5^2 \]

A canonical reduction of this is given in the sample output. The listing of the main program is supplied. The subprograms TRIANG, DGELG, EIGEN, and RECT are required. TRIANG and RECT are as in NEWRAP. DGELG and EIGEN are the double precision linear equation and eigenvalue routines from reference 12.
REAU(5,1001) FMT
READ(5,FMT) BZERO
WRITE(6,1005) BZERO
READ(5,FMT) (BL(I),I=1,JFAC)
IE=0
DO 5 I=1,JFAC
IE=IE+1
IS=IE-IE+1
READ(5,FMT) (BS(K),K=IS,IE)
DO 4 < IS,IE
BSAVE(K)=BS(K)
4 SBS(K)=SNGL(BS(K))
BLSAVE(I)=BL(I)
SBL(I)=SNGL(BL(I))
BL(I)=-BL(I)
5 CONTINUE
LENGTH = JFAC*(JFAC+1)/2
WRITE(5,1006)
WRITE(6,1007) (SBL(I),I=1,JFAC)
WRITE(6,1008)
CALL TRIANG(SBS,JFAC,8,FMTRI)
IJ=1
XIN(1)=BS(1)*2.000
DO 50 I=2,JFAC
II=I-1
IIK=I
IIJ=JFAC*II
50 CONTINUE
IIJ=IIJ+1
XIN(IIJ)=2.000*BS(IIJ)
5 CONTINUE
EPS=1.0E-10
CALL QGEGL(BL,XIN,JFAC,1,1,1,1,IE)
IF(IE,NE,0) WRITE(6,1009) 1ER
WRITE(6,1010)
WRITE(6,1007) (BL(I),I=1,JFAC)
IJ=0
YS=BZERO
DO 153 I=1,JFAC
YS=YS+BL(I)*SBL(I)
DO 150 J=1,I
150 CONTINUE
WRITE(5,1011) YS
110 CONTINUE
DO 163 I=1,LENGTH
160 SBS(IJ)=SNGL(BSAVE(IJ))
CALL EIGEN(BSAVE,XIN,JFAC,0)
120 CONTINUE
DO 203 I=1,JFAC
200 BL(I)=BSAVE(IJ)
WRITE(5,1012)
WRITE(6,1007) (BL(I),I=1,JFAC)
WRITE(6,1013)
JJ=JFAC*JFAC
DO 213 I=1,LENGTH
210 CONTINUE
WRITE(5,1011) YS
SAMPLE CANONICAL REDUCTION PROBLEM

THE COEFFS ARE BZERO  147.268600

<table>
<thead>
<tr>
<th>LINEAR</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>SECONOD ORDER</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

THE STATIONARY POINT IS 3.05811526 -1.96462563 0.12068067 -3.89057544 -0.93711068E-01

THE VALUE OF YS IS 120.589275


EIGENVECTORS

<table>
<thead>
<tr>
<th>ROW 1</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.300108</td>
<td>0.519711</td>
<td>0.750815</td>
<td>0.234709</td>
<td>-0.138316</td>
</tr>
<tr>
<td>ROW 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-0.549544</td>
<td>-0.275150</td>
<td>-0.293631</td>
<td>0.690293</td>
<td>-0.244071</td>
</tr>
<tr>
<td>ROW 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.779700</td>
<td>0.809764E-02</td>
<td>0.804860E-01</td>
<td>0.577909</td>
<td>-0.270383</td>
</tr>
<tr>
<td>ROW 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-0.219489E-02</td>
<td>0.796049</td>
<td>-0.582318</td>
<td>-0.228627E-02</td>
<td>-0.191618</td>
</tr>
<tr>
<td>ROW 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.105615E-02</td>
<td>0.173062</td>
<td>-0.669725E-01</td>
<td>0.364045</td>
<td>0.912707</td>
</tr>
</tbody>
</table>

Lewis Research Center,
National Aeronautics and Space Administration,
Cleveland, Ohio, January 25, 1972,
132-80.
APPENDIX - BORROWED Routines

Some of the routines used in the programs were taken from the literature. Both INVXTX and TRIANG are by Webb and Galley (ref. 13), and EIGEN and HIST are from the IBM programmer's manual (ref. 12), as are DGELG and the double precision version of EIGEN.

Listing of INVXTX and TRIANG are given here, as follows:

$IBFTC TRIANGX

SUBROJITINE TRIANG(A,B,NN,NKOL,FORMAT,II)
DIMENSION FORMAT(1)
DIMENSION A(1), B(1)
DOUBLE PRECISION E
1 FORMAT (1H1)
3 FORMAT(1H/1H/1H )
COMMON/SMALL/DUM(15), LIST
N = NN
NCOL = NKOL
KLUMPS = N/NCOL

C

KEEPCR = 0
K1 = 1
K2 = NCOL - 1
K3 = NCOL
IF (KLUMPS .EQ. 0) GO TO 120

C

DO 90 KLUMP=1,KLUMPS
ITR1 = KEEPCR
I = -1
ILO = (KLUMP-1)*NCOL + ITR1 + 1
DO 30 K=K1,K2
I = I + 1
ITR1 = ITR1 + K - 1
ILO = ILO + K - 1
IHI = ILO + 1
GO TO (26,28),II
26 WRITE(LIST,FORMAT) K,(A(J),J=ILO,IHI)
GO TO 30
28 WRITE(LIST,FORMAT) K,(B(J),J=ILO,IHI)
30 CONTINUE

C

KEEPCR = ITR1 + K2
DO 60 K=K3,N
ITR1 = ITR1 + K - 1
ILO = ILO + K - 1
IHI = ILO + NCOL - 1
GO TO(56,58),II
56 WRITE(LIST,FORMAT) K,(A(J),J=ILO,IHI)
GO TO 60
58 WRITE(LIST,FORMAT) K,(B(J),J=ILO,IHI)
60 CONTINUE

C

K1 = <1 + NCOL
K2 = <2 + NCOL
K3 = <3 + NCOL
90 WRITE(LIST,3)
120 ITRL = KEEPTR
    IF (K1 .GT. N) GO TO 180
    I = -1
    ILO = KLUMPS*NCOL + ITRL + 1
    DO 150 K=K1,N
       I = I + 1
       ITRL = ITRL + K - 1
       ILO = ILO + K - 1
       IHI = ILO + I
       GO TO (146,148,11,146)
146 WRITE(LIST,FORMAT) K, (A(J), J=ILO,IHI)
    GO TO 150
148 WRITE(LIST,FORMAT) K, (B(J), J=ILO,IHI)
150 CONTINUE
C
180 RETURN
END

$IBFTC INVXXX

SUBROUTINE INVXX(A, NN, D, FACT)
C
   ASSUMES THE MATRIX A IS SYMMETRIC AND POSITIVE DEFINITE,
   AND ONLY
   THE UPPER TRIANGLE IS STORED AS A ONE-DIMENSIONAL ARRAY IN THE
   ORDER A(1,1), A(1,2), A(2,2), A(1,3), A(2,3), A(3,3), ..., A(N,N).
   N IS THE ORDER N OF THE INPUT MATRIX A.
   D IS (ON EXIT) THE DETERMINANT OF A, DIVIDED BY FACTOR**NN.
C
   DIMENSION A(N)
   DOUBLE PRECISION A,PV,F
   N = NN
   ITRL = 0
   DO 145 K=1,N
      ITRL = ITRL+K-1
      KP1 = K+1
      KM1 = K-1
      KK = ITRL+K
      PV = 1.0DO0/A(KK)
C
   ITR2 = 0
   IF (K-1) 150,80,50
C
   REDUCE TOP PART OF TRIANGLE, LEFT OF PIVOTAL COLUMN
50 DO 60 J=1,KM1
      ITR2 = ITR2+J-1
      KJ = ITR2+J
      F = A(KJ)*PV
      DO 60 I=1,J
         IJ = ITR2+I
         IK = ITR1 + I
         60 A(IJ) = A(IJ) + A(IK)*F
C
   IF (K-N) 70,120,150
C
   REDUCE REST OF TRIANGLE, RIGHT OF PIVOTAL COLUMN
70 ITR2 = ITR1
DO 113 J=KP1,N
   ITR3 = ITR1
   ITR2 = ITR2+J-1
   KJ = ITR2+K
   F = A(KJ)*PV
   DO 100 I=1,J
      IF (I-K) 90,100,95
   90 IJ = ITR2+I
      IK = ITR1 + I
      A(IJ) = A(IJ) - A(IK)*F
      GO TO 100
   95 IJ = ITR2 + I
      ITR3 = ITR3 + I - 1
      IK = ITR3 + K
      A(IJ) = A(IJ) - A(IK)*F
   100 CONTINUE
   110 CONTINUE
   C
   DIVIDE PIVOTAL ROW-COLUMN BY PIVOT, INCLUDING APPROPRIATE SIGNS
   120 ITR2 = ITR1
   DO 140 I=1,N
      IF (I-K) 125,130,135
   125 IK = ITR1+I
      A(IK) = -A(IK)*PV
      GO TO 140
   C
   (REPLACE PIVOT BY RECIPROCAL)
   130 A(KK) = PV
      GO TO 140
   135 ITR2 = ITR2+I-1
      KI = ITR2+K
      A(KI) = A(KI)*PV
   140 CONTINUE
   C
   145 CONTINUE
   C
   150 RETURN
END
REFERENCES


"The aeronautical and space activities of the United States shall be conducted so as to contribute . . . to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."

—National Aeronautics and Space Act of 1958

**NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS**

**TECHNICAL REPORTS:** Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

**TECHNICAL NOTES:** Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

**TECHNICAL MEMORANDUMS:** Information receiving limited distribution because of preliminary data, security classification, or other reasons.

**CONTRACTOR REPORTS:** Scientific and technical information generated under a NASA contract or grant and considered an important contribution to existing knowledge.

**TECHNICAL TRANSLATIONS:** Information published in a foreign language considered to merit NASA distribution in English.

**SPECIAL PUBLICATIONS:** Information derived from or of value to NASA activities. Publications include conference proceedings, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

**TECHNOLOGY UTILIZATION PUBLICATIONS:** Information on technology used by NASA that may be of particular interest in commercial and other non-aerospace applications. Publications include Tech Briefs, Technology Utilization Reports and Technology Surveys.

Details on the availability of these publications may be obtained from:

**SCIENTIFIC AND TECHNICAL INFORMATION OFFICE**

**NATIONAL AERONAUTICS AND SPACE ADMINISTRATION**

Washington, D.C. 20546