THE ICL REGRESSION PACKAGE
(PROGRAM XDS3)

by

Robert Engle, T.C.I. Ryan
and
Jonathan Abbatt

DISCUSSION PAPER NO. 174

For: University of Nairobi Computing Centre
and Institute for Development Studies

August, 1973.

INSTITUTE FOR DEVELOPMENT STUDIES
UNIVERSITY OF NAIROBI

THE I.C.L. REGRESSION PACKAGE

(XDS3)

TABLE OF CONTENTS

THE I.C.L. REGRESSION PACKAGE

1. INTRODUCTION

Regression analysis assumes that there is a relationship between a series of independent variables and a single dependent variable. The independent variables can be thought of as ones that get their values outside of the particular experiment (such as amount of rainfall or quantity of fertilizers in an agricultural experiment). while the dependent variablee.g size of plant growth is determined primarily by the levels of the independent variable with some additional contributions by unexplained (i.e. not included in analysis) and random forces. The object of the analysis is to determine the magnitude of the effects of each independent variable on the dependent variable. The relationship is assumed to be linear although by judicious transformations of the data, many non-linear functions can be made into linear ones.

At this time there are many excellent books on linear regression at both a simple and advanced level. Perhaps the most clear introductions are by Wonnacott and Wonnacott Econometrics and Introductionary Statistics or Kimenta Elements of Econometrics, none of which require advanced mathematics for a clear and intuitive understanding. Somewhat more advanced are Christ Econometric Models & Methods, Johnston Econometric Methods and Goldberger Econometric Theory.
These are all available in the University Library.

The I.C.L. regression package, comprising a program named XDS3, has a variety of options, too many indeed. This set of instructions explains how to write data cards and program cards to get a regression run. It will not explain all the options, but will focus on the main requirements and some of the limits of the program. There will be extensive discussions of the variety of transformations available including methods for creating dummy variables. A description of facilities for partitioning the data so that the analysis can be performed on only a portion of the observations and a detailed explanations of the output are also provided.

For a discussion of the many other options see the I.C.L. manual at the Computer Center.* It describes facilities for handling missing observations, magnetic tape input and a variety of regression procedures as well as other statistical capabilities such as analysis of variance, factor analysis, discriminant analysis, principle components and spectral analysis.

II. DATA CARDS

Program XDS3 can read the user's data in either one of two formats, called Method 1 Format and Method 2 Format in the ICL manual. The format described below is a restricted version of Method 1. It is a "fixed" format in the sense that the values of each variable will be found in the same positions (i.e. card columns) in all the records, or observations, of the user's data. This format is recommended when coding your data because it is more likely to be compatible with other packages.

If your data has already been coded, or is already held on cards, paper tape or magnetic tape, it is still possible that it is compatible with this package, or that it can be easily converted. See section XIX on compatibility with other packages, see also the ICL manual and the Computing Center staff for further details.

Sometimes there may be situations where some observations on some variables cannot be obtained. Section XVI of this report, on missing values, shows how these situations can be handled by XDS3.

The restrictions on the form of the data are the same for cards, paper tape and magnetic tape. We will describe them for cards. For the use of other media see section XVIII and the ICL manual. The annotated coding sheets in Appendix 1 give examples of the layout of data.

* Statistical Analysis MK 2, ICL Technical Publication No. 4301 (3rd Edition) copies may also be purchased from International Computers Ltd., P.O. Box 30293, Nairobi.

Cols. 3-8 contain the name or identification sequence of that observation. It is recommended but not essential* that it contain letters as well as numbers, at least one number must be present. This name or identification sequence must be unique to that observation. If two cards are identical in cols. 3-8, the computer reads the second as a continuation of the first, with added variables.

These sets of character(s) and number(s) need not be continuous, nor monotonic (always larger, or smaller). It is assumed that the set identifies the individual record, e.g. contains the interview number. It is useful to use the identification number of the basic record (interview number), preceded by some letter identifying the study in order to facilitate locating the original data for checking, correcting errors, or even recoding damaged cards. Do not use punctuation marks in these six columns. Blanks (space characters) should only appear at the end of the observation name.

If there are two or more cards of data per record (interview), we cannot indicate that in cols. 3-8 because these columns must be the same for all data cards of a single record, or observation. Hence, put the card number in col. 80 where the computer does not read it, but where it is available to help keep the data in order.

* Contrary to what is said in the ICL manual.

If there are three cards for each interview, then the first nine cards would read:

Cols. 3-8	Col. 80
M00001	1
M00001	2
M00001	3
M00002	1
M00002	2
M00002	3
M00003	1
M00003	2
M00003	3

The spacing of data would be different on each card of a triplet, but the same for all those noted as card 1 in col. 80, etc. If the cards are dropped, the number in col. 80 is helpful in getting them in the right order again.

Col. 9 must be blank.

Cols. 10-74 contain the data (and they can be continued onto cols. 10-74 of additional cards which must be identical in cols. 1-9 with the first card). No datum (variable) should be split between the end of one card and the beginning of another.

The data begin in col. 10, separated by at least one blank column between each variable and the next. If values of the same variable are of different size, leading zeroes are not required on the smaller numbers, but they should end in the same column (technically speaking numbers are right justified) so that all values of one variable are lined up directly underneath each other. Thus each variable needs the number of spaces for the digit number, one space for a minus sign if used, one for a decimal if used and one to separate it from the next variable.

There can be no more than six digits before or six digits after the decimal point for a total of 12 digits. Thus the largest integer number is also six digits. By measuring in thousands or millions, this obstacle can be easily avoided.

N.B. Other packages available within the computing centre may only process integer variables. For further details see section XIX.

Cols. 76-80 contain the sequence number of the card within the set of data cards for each interview or observation.

III. PROGRAM STRUCTURE

The program cards instruct the computer to read data, do calculations and print results. The program order is very flexible in that calculations can be done at any time after the data has been read and any necessary preliminary calculations have been performed. Similarly, the results of any calculations can be printed at any subsequent point in the program by merely referring to the name of the calculation. Therefore each calculation and print command must refer by name to a set of data or previous calculation.

The program cards (called Control Blocks in the ICL manual) may be written in either Fixed Format or Free Format.

The program cards will be described here in Free Format, which allows us more liberty where we put things on the card because of the reduced chance for error. Briefly described examples are presented on coding sheets in Appendix 2. This is a very convenient way to follow the details of the program.

Each program card begins in column 1 with a four letter word identifying the desired operation. This is followed by a variety of names and numbers all separated by commas without any empty spaces as the names and numbers may be of different length, they do not have to be in a particular space. They must of course be in the correct order i.e. in the order that the computer recognises and therefore to leave a blank entry (see examples under Printing Output), you must put two commas in adjacent columns of the card.

To write a program you should copy the standard information from the appendix example page 3 onto a coding form and fill in your own names and information where required. Use only capital letters, one per square, and to ensure understanding by the key punchers use the symbols Ø, I, Z for the letters, O 1 2 for numbers and ∇ for a blank. A very common mistake is to use a zero in a name in one place and "Ø" in another. Each line of the coding sheet will be punched on a separate card. Take the coding sheets to the computer centre for punching.

IV. NAMES

In several places you are requested to supply names for variables, matrices, etc. These are for your convenience and should preferably indicate to you the particular variable or data in question. They can be up to six characters long but must not have internal punctuation or spaces. Each name must be unique. They must always be spelled exactly the same since only then will the computer recognize them as the same. For example DATA01 and DATA0l will look almost identical in printout but not to the computer.

V. TITLE CARD

The first card merely gives a name (up to six characters) to the program and requests error messages on the line printer (LP).

Example
TRIAL,LP

VI. READING THE DATA

The data is read in the form of a matrix where each row is an observation or case and each column is a variable. Four control cards, a list of variable names, and the data itself are required to read the data.

1. On the observation matrix card (0BSE) you give a name to the data following two commas..

Example
0BSE,,RURAL

2. The column names (C0L) card includes the same name as in the 0BSE card. It tells the computer that the next card will be column or variable names.

Example
C0L,RURAL

3. The list of variable names appears in the next card or cards. Each card must begin with two blank spaces and contain a maximum of twelve names, each of which is at most five characters long, separated by commas. Any number of cards may be used but it is very important

that the total number of names is exactly equal to the number of variables coded in each observation or record. Otherwise it will seek too many or too few elements for each row of the observations matrix and will print an error message for each row. Later you can indicate which ones you wish to analyse.

For example if your card contains observations on five variables you must have five column names even if you do not want to use all the variables in your analysis.

Examples

```
INCOM,FARM,TRACT,WIVES,X100,TAX,INVST,CHILD
V1,V2,V3,V4,V5,V6,V7,V8,V9,V10,V11,V12
```

N.B. The name CONST should not be used for a variable name as it has a special meaning.

4. The matrix (MATR) card must again contain the name of the observations preceded by two commas.

Example

```
MATR,,RURAL
```

5. The data cards are placed next.

6. The end of data (END) card is last.

Example

```
END
```

VII. TRANSFORMATIONS - SIMPLE MATHEMATICAL EXPRESSIONS

The observations matrix can at any time be transformed to obtain a new observations matrix and new variables which are mathematical functions of the original variables. Any analysis can be performed on this new data merely by specifying its new name rather than the name of the original matrix. The set of transformations is introduced by a transformations (TRAN) card which gives names for the original data, the set of transformations and the transformed data. The last two are new names of up to six characters which you invent.

Example

TRAN,RURAL,TRFORM,RURIRA

This is followed by any number of transformations, each of which is one or more cards long and starts in column 3. Each transformation is merely an equation with a new variable name on the left of an equal sign and a mathematical expression on the right. This expression can be written in terms of the names of the original input variable, or previously defined transformed variables, or numbers. Numbers can have at most 6 digits before and six digits after the decimal. If you want to define some mathematical expression which is not to be part of the new matrix but which you will use in many of the transformations, you can define this "work variable" by using a colon instead of an equals sign. There is no restriction on the permissible names of the new variables except that they can be,at most, five characters long. Even the original names can be used again. However, if a name which has been given to both an original and a transformed variable is used in a subsequent transformation expression, the values are taken from the original data.

The form of the mathematical expressions is the same as in FORTRAN. The operations symbols are the following:

+	additions
-	subtractions
*	multiplications
/	division
**	exponentiation (raising to a power)

In addition, basic external and intrinsic FORTRAN functions can be used. The most useful of these are:

ALOG(X)	which means	$\ln (X)$
ALOG10(X)	which means	$\log_{10} (X)$
SQRT(X)	which means	$\sqrt{X}$
EXP(X)	which means	e^X

EXP10(X) which means 10^X

AMIN1(X,Y) which means minimum of (X,Y)

AMAX1(X,Y) which means maximum of (X,Y)

ABS(X) which means absolute value of X,

where X and Y may be variables or constants.

When an expression is evaluated by the computer, the first operations done are the FORTRAN functions and expressions within parentheses (or brackets). Next, numbers with exponents are raised to the specified power. Then multiplications, divisions, additions and finally subtractions are computed. Careful use of parentheses will eliminate many mistakes.

If you use the transformation data, then only variables you have explicitly created plus the CONST which the computer constructs, will be available for your analysis matrix. Therefore, it is necessary to list all the variables which you will want to use in the regression analysis in the transformations. If you want to include some of the original variables, simply make up a new name or even use the old name and set it equal to itself. The variable names to the left of the equal signs will be the column names of the transformed data matrix.

Examples,

```
∇∇Z1=V1+V3
∇∇Z2:V1*V2
∇∇Z3=100*V2/INCOM
∇∇Z4=(V1*200.105)**V3 i.e. (200.105V1)^V3
∇∇Z5=V2*V1+Z4
∇∇Z6=ALOG(Z4)
∇∇ V2=V2
∇∇ INCOM=INCOM
∇∇ TRACT=TRACT
∇∇FARMD=AMIN1( 1.0, FARM)
∇∇ INVST=INVST
∇∇FAMLY=WIVES+CHILD+1
```

VIII. TRANSFORMATIONS - DUMMY VARIABLES

By careful use of the transformations facilities, it is possible to construct a series of dummy (0 , 1) variables for a variety of situations. Dummy variables are often useful in regression analysis for dividing the population into two sub-groups.

A common problem is to construct a variable which is one for large values of a variable X and zero for small values. If K is the number at which the split between "large" and "small" is made we can write this transformation as two cards.

Example:

```
WORK:X-K
Z=(WORK+ABS(WORK))/(2*WORK)
```

Notice that if X is bigger than K,Z will be one while if X is smaller than K,Z will be zero. However, if any value of X equals K there will be an error message and the program will stop. Thus to separate above and below 12, let K = 12.000001. The difference between two such dummy variables for different K's will be a variable which is only 1 between the two numbers. A similar problem is to construct a dummy variable which is one for only one of a series of integers. Suppose K is the desired integer.

```
ABOVE:.5-X+K
BELOW:.5+X-K
Z=(ABOVE+ABS(ABOVE))*(BELOW+ABS(BELOW))
```

The first term will be zero for any integer X greater than K and the second will be zero for any X less than K. Both will equal one for X = K.

By using these and other simple operations, a wide variety of dummy variables can be constructed.

Further simple examples follow:

To transform 0 into 0
1 into 1
9 into 0

use Z=X*(9-X)/8

To indicate whether X or Y or both are one where X and Y are (0,1) variables

Z = AMAX1(X,Y). (Z will be zero when both X and Y are zero.) To set all values greater than 1 equal to one and leave others unchanged.

Z= AMIN1(1.0,X).

IX CROSS PRODUCT MATRIX - PARTITIONING THE DATA

In order to do a regression or complete a correlation matrix, a cross product matrix must first be computed. This can be done simply by writing CROS followed by a comma and the name of the data matrix you wish to use. If you are using transformed data, the name will be that of the transformed data.

Example
CROS,RURTRA

When the cross product is constructed it is possible to limit the number of variables or the number of observations. Both of these may be very useful but they require a new name for the matrix preceeded by three commas if the subset or weighting options are used.

Example
CROS,RURTRA,,,RURSB
These commands also produce the corresponding data matrix which is called by this same new name. Therefore these observations can be printed or means calculated just as for any other data matrix. See Section XI below, on printing output.

To limit the number of variables in the cross product matrix, a subset (SUBS) card follows directly after the cross product card. This is then followed by a list of variables names separated by commas starting in column 3 with up to twelve names per card. Only these variables will be included in the cross product matrix. As the computer storage is limited, only about 70 variables can be included. However, in the regression output there are statistics for all the included

variables so it is wise to substantially limit the number by a subset card. The CONST is automatically included.

Example

CRΘS,RURTRA,,,RURSB

SUBS

FARMD,~~2~~,~~4~~,INCΘM,INVST

The cross product matrix can also be computed using only a subset of the observations. This is most easily done by specifying a weighting variable in the cross product card. If the variable is a zero-one (0,1) variable, this has the effect of eliminating all the observations for which the weight is zero. The name of the weighting variable follows an additional two commas.

Example

CRΘS, RURTRA,,,RURFM,,FARMD (RURFM is the name you have given to this new matrix derived from RURTRA, and including only observations for which FARMD is one)

Finally, observations preceding a particular row name, say HG0023,may be excluded by listing that name after the first comma following the data matrix name, that is, in the third position. The subset of data can now be used in the usual way, using the old matrix name, in this case RURTRA.

Example

CRΘS,RURTRA,HG0023

X. CORRELATION

The covariance (CΘVA) and correlation (CΘRR) matrices may be successively computed merely by giving the name of the appropriate cross product matrix which has already been computed. It is unnecessary to complete these unless their output is desired specifically.

Example

CΘVA,RURFM

CΘRR,RURFM

XI. PRINTING OUTPUT

Printed output can be requested at anytime after calculations are complete. All printing is of course optional. There are five similar print commands.

Print Observations (PΘBS)
Print Cross Product (PCRP)
Print Covariance (PCΘV)
Print Correlations (PCΘR)
Print Means (PMNS)

Each is followed by a format code which may be left blank and the name of the matrix which is to be output, all separated by commas. The format code is just two digits, the first specifying the number of places to the left of the decimal point and the second the number of spaces to the right. A blank will give all output in exponential format. The use of a particular format may greatly reduce the number of pages of output and facilitate its perusal. For example a 13 format is generally appropriate for a correlation matrix.

The PMNS command also produces the maximum and minimum of each variable and the variance. If instead of the variance, the standard deviation is required, two commas and an S after the cross product matrix name will produce it.

Examples

PΘBS,,RURAL e.g. will give observations in exponential form
PMNS,,RURTRA,,S
PCRP,50,RURFM
PCΘV,33,RURFM
PCΘR,13,RURFM
PMNS,41,RURFM;
PMNS,,RURSB

XII. REGRESSION ANALYSIS

A regression analysis is initiated by a regression analysis (REGR) control card which includes the name of the cross product matrix to be used and the word CRΘS.

Example

REGR,RURFM,CROS

This is then followed by a dependent variable card (DEPE), an independent variable card (INDE), a list of independent variables, and a print regressions (PREG) card. These four cards may be repeated as often as desired but all are necessary for each regression. If the residuals are of interest then a fifth card (RESI) must be included for each regression.

The dependent variable (DEPE) also includes the name of the dependent variable. It must of course be one of the variables in the cross product matrix.

Example

DEPE,INVST

The independent variables card (INDE) includes also the number 99. This is a code to indicate that all the independent variables which you list are to be included in the regression. If a smaller number is written, only variables significant at that confidence level are included. Another option is to obtain the best subset of K regressors. These two procedures are generally not appropriate for economic research and the first does not necessarily have a unique answer. Hence we recommend 99.

Example

INDE, 99

This card is followed by a list of the names of the independent variables separated by commas, starting in column 3 and with no more than 12 per card. Generally CONST will be one of these variables unless there are strong reasons to believe that the dependent variable is zero when all the independent ones are also zero. The independent variable names can be on several cards, with one or more names on each card.

Example

CONST,INCOM,FAMLY

V6

The print regression (PREG) and residuals cards (RESI) require no additional information.

Examples

PREG

RESI

XIII. REGRESSION OUTPUT

The output from each regression includes information about not only the variables included in the regression but also those excluded from the regression but in the cross product matrix.

For each variable included in the regression set you will get:

a. the estimate of the regression coefficient,
b. the standard error of the regression coefficient,
c. Student's t statistic,
d. Partial correlation coefficient of the variable with the dependent variable, assuming other independent variables in the regression set to be held constant.

e. Under the heading Multiple Correlation are columns of the square roots of R^2, where the variable is excluded from the regression set. The R^2 is calculated on two bases. The second, which is more familiar, is one minus the sum of the squared residuals divided by the sum of squares of the dependent variable around its mean. If there is no CONST in the regression, then the first measure of R^2 may be more appropriate in which the sum of squared residuals is divided by the raw-sum of squares of the dependent variables. As we know the former can be interpreted as the correlation between the actual and fitted values of the dependent variable.

f. The error sum of squares of the regression if the variable were omitted.

For each variable excluded from the regression, the output includes:

a. the t - statistic if it were included
b. the partial correlation coefficient with the dependent variable
c. the two multiple correlations if it were included.
d. the error sum of squares if it were included.

In addition, for the regression as it stands printed at the bottom of the page are: the error sum of squares, residual error (standard error of the regression), and both measures of the multiple correlation (square root of R^2).

If residuals are requested (using RESI card) the output includes

a. actual value
b. fitted value
c. residual
d. First and Second differences
e. percentage of this squared residual to the error sum of squares
f. Ratio of fitted to actual value
g. Durbin-Watson statistic if there is a constant.

If the data is not a time series, then d and g will be inappropriate However, the other outputs may be very useful in analysing the regression.

XIV. GETTING OFF

The last card of the program is a get off (GET) card indicating no more commands. Before you put in the (GET) card, consider adding a new data matrix at this point. If you have two jobs to run, you can save the computer time by beginning the second immediately after the first. Merely postpone the (GET) card and omit the TITLE card and you are on your way.

XV. SUBMITTING THE PROGRAM - SYSTEM CARDS

In order to run your program, four more cards must be placed at the very beginning of the deck of cards to tell the computer who you are and to instruct it to prepare the package (XDS3) for your program. These four cards start in column one and leave spaces only when indicated by ∇ .

1. JOB∇ a , b , c
   a. Job number - 4 characters long
   b. Your account number - 3 characters long
   c. Your name - up to 12 characters long

2. RUN∇XDS3,ED2

3. ****

4. DOCV DATAXDS3

The "JOB" of card one is prepunched in the computer center and can be recognised by its yellow edge. Use this type of card if you do your own punching, otherwise the punching staff will provide it for you. Following the last of these cards, put all the data and program cards described in this report in the correct order. At the very end should go a duplicate of 3.
i.e.

Last card ****

The completed deck may be submitted by enclosing in a rubber band (or for many cards, a computer card box) and placing it either through the door beside the keypunches, or in the metal rack under the input heading just inside the computer center. The output will be placed in this rack when the job has been run.

Experience suggests that it is worthwhile to check the data and program cards carefully before submitting the job. A very small error in spelling or spacing may make the entire run useless and it may sometimes take a day or more to get it back even with no results.

XVI. MISSING VALUES

If an observation on a particular variable is not available for some reason, it may still be possible to include other data from that particular questionaire or interview by using the missing value facility.

In the data card, the missing value is replaced by a single * (asterisk) anywhere within the columns normally filled by the record of that observation when it is present. A pair of asterisks (**) indicates that all remaining data in that questionaire or interview are missing.

Example:

Q00001	100	20.2	126
Q00002	106	*	131
Q00003	109	**	

The fact that missing observations may appear in the data must be signified on the matrix card by an M preceeded by a comma.

Example

MATR,,RURAL,M

Print means (PMNS) and Print observations (POBS) statements may be used directly. No transformation however can be undertaken on a matrix with missing values. If transformations are required, it may be possible to use a missing values card (MISS) to fill in the missing values. A variety of schemes are available; however, which is appropriate depends on the statistics of the problem. The simplest is to replace each missing observation by the mean of that variable so that it has no effect on either mean or covariance except through the incorrect number of elements. The new matrix is given a new name (COMPRU in the example below). Note that if the transformation is non-linear, the mean of the transformed variable will not be the transform of the mean.

Example

MISS,RURAL,,COMPRU

A better procedure is to calculate the cross product matrix directly from the original matrix. This can be done in two ways, named A and B. Method A eliminates all observations which have any missing values. Method B, computes the cross product by only including values for which neither of the pair of variables is missing. Thus some elements in the cross product matrix will be composed of more observations than others. This will not bias the result of a regressions at least for large samples. It will however overestimate the confidence in the result if there are missing observations on the dependent variable. In this case Method A is preferable.

Examples

CROS,RURAL,,B,RURALB

CROS,RURAL,,A,RURALA

A regression may be run directly on either RURALB or RURALA with no change in cards.

XVII ERROR MESSAGES

There are a variety of possible error messages which the program may give. These are listed at the end of the ICL manual. Several common ones are given here.

N.B. If fifty errors are detected during the input of one observation matrix, the program will stop.

1. Incorrect Number of Elements

HG0023

This means that on card labelled HG0023 there are either more or fewer bits of data than the number of column names. If you get this message for every row, you probably have given the wrong number of column names. Otherwise it may be just a card or two which is mispunched. The program will skip such cards and continue.

2. Incorrect Format

HG0023

Probably there is a letter punched where data should be. The program will replace it with a number very close to zero and proceed. Check for an Θ instead of 0.

3. No Matrix Found

4. No Variable Found

These two messages can be given in a variety of situations most likely this means that you have spelled the name of the variable or matrix differently from the way you previously spelled it. Again check for Θ and 0 or other similar appearing letters. Alternatively you may have failed to correctly name the variable or matrix when you first defined it. A message 3 usually stops the program while it will manage to go past a message 4.

5. Error in Transformation

This signifies an error in one of the transformations and will stop the program. A specific message is given after the offending statement. These may arise because the name of a variable which is to be transformed is spelled wrong or because the expression is incorrectly written.

XVIII USING OTHER STORAGE MEDIA

It is possible for XDS3 to read program instructions and data from either paper tape or cards (although not from both in one run). The choice of input medium is determined by a program switch setting, see ICL manual for details. When the user runs the program in the way described in this report, the switch will always be set for card input.

However, XDS3 can read the data alone (i.e. just the observation matrix) from magnetic tape. Data held on this medium can be processed much more quickly than from cards or paper tape and is not subject to the kind of wear that produces card jams. It is therefore an advantage to use magnetic tape when there is a large volume of data and many computer runs are required.

The observation matrix must first be punched in XDS3 format onto cards or paper tape. It can then be transferred onto magnetic tape (in the standard ICL subfile format required by XDS3), using the data handling program S05L. See the program description* and Computing Centre bulletins for details.

The matrix is processed from magnetic tape by using the TAPE card in place of the data and END cards described in this report. The TAPE card carries the 12 character name of the magnetic tape file and the matrix name. These names must be the same as those given in the instructions to S05L*. Thus, if the users data is held on a tape named EXAMPLETAPE1 in a matrix named MXNAME, then following the matrix card (MATR,,MXNAME) will come the card:-

```
TAPE,EXAMPLETAPE1,0,MXNAME
```

Following the TAPE card the user can place cards transforming and analysing matrix MXNAME as already described.

XDS3 writes the observation matrix and other matrices created during the program run to work files on magnetic tape or magnetic disc and there are facilities for reading and processing these files in later runs.

* I.D.S. Discussion Paper No. 132 by R.W. Tacker

XIX COMPATIBILITY WITH OTHER PACKAGES

The method of coding data described in this report uses a fixed format. Therefore besides being acceptable to XDS3, data coded in this way can be processed fairly easily by user written programs in FORTRAN or other languages. It may also be acceptable to other packages available at the University Computing Centre, e.g.

a) the ICL Survey Analysis package (program XDSB) +,

b) the IDS programs S04A (one - way analysis of variance) and S04B (cross tabulation of means)*,

c) the Computing Centre Survey Analysis programs SFPE (column counts) and SFPF (cross tabulation)**.

Note however that of these programs only XDSB can process numeric values containing a decimal point.

Programs S04A and S04B only process signed or unsigned integer values. Programs SFPE and SFPF only process unsigned integer values (although + and - could be input as non-numeric codes). Where these restrictions apply, the relevant variables must be coded in integer form (other variables can be ignored). Alternatively these programs can ignore the decimal point and treat the fractional and integral parts of decimal values as seperate integer variables, i.e. the fraction can be truncated, or a leading decimal point omitted, scaling up the values.

It is not true though, that data acceptable to these other packages will generally be valid input to XDS3. All the above packages distinguish between separate values in each data record by means of a format statement input at run time. This allows them to ignore some variables (which are not required or permitted to be processed) and also allows close packing of variables within each record. XDS3 on the other hand, uses seperators (a blank column or a comma) to distinguish between the values of different variables and cannot ignore fields within a record.

+ See "Survey Analysis", ICL Technical Publication No. 4335 (4th Edition)

* See I.D.S. Discussion Papers Nos. 101 by D.S. Shepard and 106 by D.S. Shepard and K. Prewitt.

** See Computing Centre description "Survey Analysis Programs SFPE and SFPF" by J.P.S. Abbatt.

But it is often not practical to attempt to code all the variables in a users data file in XDS3 format, particularly with survey data coded from questionnaires. The restriction of using only 65 columns for data within each card, and of leaving a blank column between each variable, can be severe when there are a large number of variables using only 1-3 columns each.

Previously it has been necessary in cases like this, either to code the data required for analysis by XDS3 seperately, or else to write a special conversion program to extract the data required from the users file and reformat it for XDS3.

A general conversion program is now available however, named SAFI, which will extract observations on variables held in card or magnetic tape formats not acceptable to XDS3 and write them to magnetic tape in an acceptable form*. The layout of the parameters (instruction cards) is very similar to that of the general data handling program S05L and the data can then be processed by the ICL package using the TAPE option (see previous section of this report).

* See Computing Centre description "SAFI - Statistical Analysis Format Interface" by V. Bhatt.

APPENDIX I

80 COL. | DATA SHEET | JOB: _____ | DATE _____ | PAGE No. 1 | OF 2

EXAMPLES OF DATA CARDS

▽▽a b ▽c ... ▽ d

a. A letter; part of the identification sequence.
b. A five-place number, or combination of numbers and letters. Do not use punctuation marks. Zero-fill, if necessary. Blanks can only appear at the end.
c. Value of first variable. Follow this with a blank and then continue with data, putting blanks in-between successive elements. Do not go beyond column 74
d. Sequence number, if there is more than one card for each observation.

```
Q00001  0.5   5.3  0.02
Q00002 -0.2  12.7  0.15
Q00003  1.2 -10.2  0.00
Q00004 -1.1 100.0  0.10
```

PUNCHED VERIFIED

80 COL. DATA SHEET JOB: DATE PAGE No. 2 OF 2

```
The following data cards have two cards for each observation.  Sequence numbers are in column 80

SHP101  0.0206  534.0  236.000  0.4419  -3.8825  0.0  0.0  0.0  0.0  1.0  0.0  0.0    1
SHP101  0.0  0.0  0.0  1.0                                                             2
SHP102  0.2173  359.0  200.000  0.5571  -1.5266  0.0  1.0  0.0  0.0  1.0  1.0  0.0    1
SHP102  0.0  1.0  0.0  1.0                                                             2
SHP103  0.0148  135.0  80.625  0.4226  -4.2121  0.0  0.0  1.0  0.0  1.0  1.0  1.0    1
SHP103  1.0  0.0  0.0  1.0                                                             2


When writing program and data cards there is no need to leave any empty rows between them as has been done in this
appendix to make it easier to write comments on particular entries .
```

PUNCHED VERIFIED

APPENDIX 2.

80 COL. DATA SHEET JOB: _____ DATE _____ PAGE No. 1 OF 6

The following is a complete regression program. Information in parentheses and in small letters is to explain the example briefly. The capital letters indicate the appearance of the actual card. Every other line is skipped to give more room for explanation. This will not be necessary of course when you write your own program. Two programs will be described. The first which has no special features and is very short. The other employs a variety of options.

```
JOB AAAA,BBB,YOURNAME        (aaaa is your job number, bbb is your account number and yourname is
                             ( your name. The first two are obtained from the computer center staff.

RUN XDS3,ED2

****

DOC DATAXDS3                 (these cards are exactly as written. They introduce you to the computer
                             (and request XDS3

TRIAL,LP                     (trial is your name for the program. This is the first card of the
                             (regression program itself.
```

PUNCHED VERIFIED

80 COL. DATA SHEET JOB: _____ DATE _____ PAGE No 2 OF 6

```
OBSE,,RURAL          (This is the first card for entering data.  rural is your name
                     (for the observation matrix

COL,RURAL            (introduces column names for rural

   INCOM,FARM,TRACT,WIVES,X100,TAX,INVST,CHILD     (These are your names for columns
                                                   (In this case you have 20 columns
   V1,V2,V3,V4,V5,V6,V7,V8,V9,V10,V11,V12          (in your observation matrix rural

MATR,,RURAL          (Prepares the computer for data cards

                     (Your data cards must be inserted between the matrix card and the
                     (end card

END                  (Signifies the end of the data cards

POBS,,RURAL          (To check that the data has been punched and read correctly
                       it is a good idea to print observations once.
```

PUNCHED VERIFIED

80 COL. DATA SHEET JOB: _____ DATE _____ PAGE No. 3 OF 6

```
PMNS,,RURAL                  This also helps to check the data since the maximum and
                             minimum of each variable is printed as well as the mean and variance.

CROS,RURAL                   An instruction to compute the cross product matrix

REGR,RURAL,CROS              (The name rural tells the computer which cross product matrix to use for
                             (the regression analysis.

DEPE,INVST                   (The dependent variable is to be invst

IND=,99                      (The independent variables are income, farm, v6, and const.

    INCOM,FARM,V6,CONST

PREG                         (print the regression output

RESI                         (Compute and print the residuals

FET                          (The program is finished

****                         (Always the last goodbye to the computer
```

PUNCHED VERIFIED

80 COL. DATA SHEET JOB: _____ DATE _____ PAGE No. 4 OF 6

In this second program, the first two pages of the first program are repeated. We will now continue on page three	
TRAN,RURAL,TFORM,RURTRA	rural is the name of the data to be transformed and rurtra is the result. trform is a name for the transformations
FAMLY=WIVES+CHILD+1	This set of transformations constructs new variables and logs of those variables for use in regression or other calculations. It also constructs a dummy variable for observations where farm is at least 1, farmd is one. All other observations are assumed not to be farmers and have farmd = 0 since acres of land is less than one.
LINCM=ALOG(INCOM)	
LAND:FARM-1.0001	
FARMD=(LAND+ABS(LAND))/(2*LAND)	
LINVS=ALOG(INVST)	
LFAML=ALOG(FAMLY)	
INCOM=INCOM	
INVST=INVST	
V6=V6	
LV6=ALOG(V6)	
CROS,RURTRA,,,,RURFM,,FARMD	A cross product matrix named rurfm is constructed from data called rurtra. All observations for which farmd is zero are excluded.
COVA,RURFM	The covariances matrix is constructed
CORR,RURFM	The correlation matrix is constructed
CROS,RURTRA	The ordinary cross product including all observations is constructed

PUNCHED VERIFIED

80 COL. DATA SHEET JOB: _____ DATE _____ PAGE No. 5 OF 6

```
POBS,,RURFM                 (We might wish to see this set of observations of transformed data where
                            (farmd is one
PMNS,,RURFM,,S              (We might want the means and standard deviation
PCOR,13,RURFM               (Print the correlation matrix with one place to the left of the decimal and
                            (three to the right.
REGR,RURFM,CROS
DEPE,INVST
INDE,99
  INCOM,FAMLY,V6            (List of the four independent variables.  These can be on the same or
  CONST                     (different cards.
PREG
RESI
DEPE,LINVS
```

PUNCHED VERIFIED

UNIVERSITY OF NAIROBI

COMPUTING CENTRE

80 COL. DATA SHEET JOB: _____ DATE _____ PAGE No. 6 OF 6

```
INDE,99
    LINCM,LFAML,LV6,CONST
PREG
RESI
REGR,RVRTRA,CROS
DEPE,LINVS
INDE,99
    LINCM,LFAML,LV6,CONST
PREG
RESI
GET
****
```

PUNCHED VERIFIED

APPENDIX 3 EXAMPLE PRINTOUT

REGRESSION ANALYSIS CROS RURTRA CUT OFF PARAMETER .10000 E-5
DEPENDENT VARIABLE LINVS DEGREES OF FREEDOM 320
INDEPENDENT VARIABLES AT SIGNIFICANT LEVEL 99.00%

CONST LINCM LFAML LV6
VARIABLES IN THE REGRESSION SET

VAR NAME	REGRESSION COEFF	STANDARD ERROR	CONFIDENCE INTERVAL	T STAT	PART CORR	MULTIPLE CORRELATION		ESS
CONST	42.7631864	.184933E 1		23.12	0.78	0.978		.299948E 5
LINCM	1.1119332	.684077E 0		1.63	0.09	0.991	0.495	.109662E 5
LFAML	- 0.1805593	.627077E-1		2.88	-0.15	0.991	0.482	.111470E 5
LV6	0.0599186	.117316E-1		5.11	0.27	0.991	0.440	.117166E 5

VARIABLES NOT IN REGRESSION SET

VAR NAME			T STAT	PART CORR	MULTIPLE CORRELATION		ESS
FAMLY FARMD			5.17	-0.27	0.992	0.553	.106412E 5
INCOM			0.01	0.12	0.991	0.527	.108800E 5
INVST			1.28	0.30	0.992	0.535	.108118E 5
V6			10.95	0.75	0.994	0.561	.103391E 5
E.S.S.	.108817E 5						.106925E 5
RESIDUAL ERROR	.565729E 1						
MULT CORR	0.991	0.501					

This example is for an equation $Y = b_0 + b_1X_1 + b_2 X_2 + b_3 X_3 + \varepsilon$ where

$$E(\varepsilon_i) = 0$$
$$E(\varepsilon_i \varepsilon_j) = 0 \quad (i \neq j)$$
$$E(\varepsilon_i^2) = \sigma^2$$

Where $\hat{b}_1$, $\hat{b}_2$ and $\hat{b}_3$ are estimators of the true coefficients of X_1, X_2 and X_3; $\hat{b}_o$ is the estimator for the constant when all the independent variables, X, are zero.

Standard Error, Error Sum of Squares (E.S.S.) and Residual Error are all in floating point format. i.e.,184933E 1 is 1.84933 and .627077E - 1 is .0627077.

The standard errors in the two variable case are computed:

$$s.e._{\hat{b}} = \sqrt{\frac{\sum (Y - \hat{Y})^2}{(n - 2)\sum (X - \bar{X})^2}}$$

where $\hat{Y}$ are the fitted values of Y from the regression estimate.

The confidence interval will only be printed if a significance level less than 99% is requested.

The t statistic is computed

$$t_i = \frac{\hat{b}_i}{s.e._{\hat{b}_i}}$$

The partial correlation coefficients which show the relationship of that variable with the dependent variable, are calculated

$$r_{X_1Y.X_2X_3} = \frac{r_{X_1Y} - r_{YX_2}\, r_{YX_3}\, r_{X_1X_2}\, r_{X_1X_3}}{(1 - r_{YX_2}^2)(1 - r_{YX_3}^2)(1 - r_{X_1X_2}^2)(1 - r_{X_1X_3}^2)}$$

where $r_{X_1Y} = \dfrac{\sum (X_1 - \bar{X}_1)(Y - \bar{Y})}{\sqrt{\sum (X_1 - \bar{X}_1)^2 \sum (Y - \bar{Y})^2}}$

The numbers under the headings Multiple Correlation and ESS in the table of "Variables in the Regression Set" all show what would happen to those statistics were that variable to be dropped from the regression set.

The first number under Multiple Correlation is the square root of:

$$R_1^2 = 1 - \frac{\sum (Y - \hat{Y})^2}{\sum \hat{Y}^2}$$

This is the only number that will be printed if there is no constant term included in the regression set.

The second number is the square root of R^2 calculated in the more normal way:

$$R_2^2 = 1 - \frac{\sum (Y - \hat{Y})^2}{\sum (Y - \bar{Y})^2}$$

ESS, the Error Sum of Squares, is calculated

$$ESS = \sum (Y - \hat{Y})^2$$

In the table "Variables not in the Regression Set." the t statistic relates to the significance of the regression coefficient of that variable - this coefficient is not printed. In the same table the partial correlation coefficient shows the relationship of that variable with the dependent variable.

The numbers under Multiple Correlation and ESS in this table, unlike in the earlier table, show what would happen to those statistics were that variable introduced into the regression set at the next iteration.

Below these two tables are three (or four) statistics: The error sum of squares (E.S.S.) of the regression, the residual error, the two values of Multiple Correlation and, in the case where regressions are performed on the covariance matrix rather than the cross product matrix, an intercept term is also output.

The Residual Error is often called the standard error of the estimate and is calculated:

$$\text{Residual Error: } S_{y.x} = \sqrt{\frac{\sum (Y - \hat{Y})^2}{n - k - 1}}$$

where $(n - k - 1)$ are the degrees of freedom.