TECHNICAL MANUAL NO. 3

STATISTICAL PACKAGES

R.J. Christiansen

UNIVERSITY OF QUEENSLAND
COMPUTER CENTRE

TECHNICAL MANUAL NO. 3

STATISTICAL PACKAGES

R.J. Christiansen

MNT-3
1 JULY 1975

This manual has been authorized by
the Director of the Computer Centre.

CONTENTS

1. PREPARING DATA FOR ANALYSIS

The data preparation phase of any statistical analysis project involving a computer is probably the most important phase, but unfortunately the one which receives least attention and is most likely to suffer first from any time schedule which may be imposed. Time spent in checking data (either manually, or with the computer), and arranging it in a convenient and easily handled form, can save many hours in both human and computer processing time at a later stage in the project.

This section contains hints on data preparation and checking, as well as references to other literature on the subject. Emphasis will be placed on "social science" type analysis, and in particular, the processing of survey data and questionnaires. 'Data preparation' in this context refers to the punching of data onto 80 column cards. It is assumed that the actual punching will be carried out by the Computer Centre data preparation service or some other such body.

It is expected that a "data entry" facility whereby data is directly entered onto magnetic storage will be available in the future.

1.1 QUESTIONNAIRE DESIGN

The specific theory and details of questionnaire design are adequately covered elsewhere. (See Moser & Kalton.) What will be covered here are the aspects of questionnaire design specifically relating to data preparation and computer processing.

The process of moving data from the point of collection to actual analysis is such that there is a great potential for the introduction of error. Naturally, any transcription process involving human participation increases this potential, and it should be an aim in the design of any questionnaire to minimize the frequency of such human intervention.

It is to this end, that the layout of a questionnaire should be given careful consideration. Naturally, if the questionnaire is to be completed by the respondent and not by an interviewer, then the design of the questionnaire should, as much as possible facilitate the complete and accurate answering by the respondent. It is possible to achieve this, and still have a layout conducive

to data preparation.

If data can be transferred into machine readable form (e.g. punched cards) directly from the source document, then time can be saved, and the risk of human error reduced.

1.1.1 PRE-CODED ANSWERS

The simplest way to facilitate data preparation is to use precoded answers to questions, and there are many ways of achieving this. (See Moser & Kalton Ch.13) e.g.

(A) DO YOU OWN A MOTOR CAR?
(Please put X in appropriate box)

YES [X] 1
NO [ ] 2

(26)

(B) IN WHAT STATE OF AUSTRALIA WERE YOU BORN?
(Write appropriate number in box)

1. QUEENSLAND
2. NEW SOUTH WALES
3. VICTORIA
4. TASMANIA
5. S. AUSTRALIA
6. W. AUSTRALIA
7. N. TERRITORY
8. NOT BORN IN AUSTRALIA

[1] (27)

(C) HOW DO YOU NORMALLY GET TO WORK? (OFFICE USE ONLY)
(please circle)

1. ON FOOT
2. BICYCLE
3. MOTOR CYCLE
4. CAR
5. PUBLIC TRANSPORT
6. TAXI
7. NOT APPLICABLE

(28)

In example (A) the respondent is required to place a "X" in the appropriate box. Each box has a number beside it, which is the figure actually punched onto the card.

In example (B), the respondent is required to write the appropriate code for his answer into the box provided.

In example (C), the respondent circles the appropriate answer, the code for which is later transcribed into a box in the right margin.

1.1.2 DESCRIPTIVE ANSWERS

Descriptive answers may not be directly processed by the computer, but may still be essential to the questionnaire. If processing is required, then they should be quantified in some way at a later stage e.g.

(1) WHAT DO YOU THINK ABOUT LOWERING THE MINIMUM AGE FOR OBTAINING A DRIVING LICENSE FROM 17 YEARS TO 15 YEARS? (OFFICE USE ONLY)

(24)

In this example the respondent is required to express an opinion in the space provided. At a later stage, a scrutineer may interpret this response according to the following categories, and then enter a code into the space provided.

1. Strongly agree
2. Agree
3. Dont care
4. Disagree
5. Strongly Disagree
6. Did not answer.

The respondent, on the other hand, may be asked to rate his own opinion on a ten point scale e.g.

Ø	1	2	3	4	5	6	7	8	9	1Ø
STRONGLY DISAGREE					DONT CARE					STRONGLY AGREE

(24)

1.1.3 USER CODED NUMERIC RESPONSES

Many data preparation errors result from this type of question, and where possible, they should be avoided.

E.g. Instead of:

(1) WHAT IS YOUR WEEKLY INCOME IN DOLLARS? [, ,] (24-27)

USE

(2) IN WHICH GROUP DOES YOUR WEEKLY INCOME LIE?

a. UP TO $50

b. $50 to $100

c. $100 to $200

d. OVER $200

[ ] (23)

If (1) is used, respondents may try to enter income as dollars and cents, in which case insufficient space is provided;also, they may left justify their numbers in the space provided, rather than right justify i.e.

[20] instead of [20]

Generally speaking, if classes can be provided with meaningful intervals, such as in the second example, they should be used. If this is not possible, then the actual coding should be left to scrutineers.

1.1.4 HINTS ON QUESTIONAIRE DESIGN

1. Where possible, use numeric codes rather than letters e.g.

USE YES 0 [ ] (25)

NO 1 [ ]

RATHER THAN

YES Y ☐ (25)

NO N ☐

This is because computers generally manipulate numbers better than characters, and also because data preparation is quicker for all numerics than for a mixture of alphabetics and numerics.

2. If respondents are to write codes in a box provided ensure there are sufficient positions for the largest code.

3. Indicate the position of each coded response on the punched card (or other recording medium). e.g.

YES Ø ☐ (64)

NO 1 ☐

- the coded answer for this question will be punched in column 64 on the card.

4. If respondents are to complete the questionnaire ensure that spaces for answers are as close to the right hand side of the page as possible so that the data preparation assistant can scan straight down the page.

5. If actual coding of the answers is to be carried out by a scrutineer on the questionnaire form, ensure that an adequate margin is left on the righthand side of the page.

6. If the questionnaire is in the form of sheets of paper pinned in the top left corner, print only on one side of the paper as such forms are difficult to handle. Questionnaires in booklet form may be printed on both sides of the paper.

7. Always manually check completed questionnaires before they are submitted for data preparation (See next section).

8. If in doubt consult the Computer Centre.

9. Ensure with precoded answers that all possible responses are provided for e.g. it should be possible to answer all

questions, even if the answer is "don't know", "not applicable" etc.

1.2 DATA CHECKING

Whether the data is to be punched directly from the source document, or from some intermediate document, the same checking procedures should be observed at all stages. This checking at least in the first case, will have to be done manually by scrutineers, however as soon as the data is in a machine readable form, the checking may be performed using the computer.

Basically, data should be checked for completeness and accuracy.

1.2.1 COMPLETENESS

It is important to first check that the data is complete. Incomplete data can be a problem particularly when the respondents themselves, actually fill out and code the questionnaire. Incomplete questionnaires should be set aside by scrutineers and some prescribed course of action taken. It is not the job of data preparation assistants to make assumptions about unanswered questions. If necessary, a code for "did not answer" should be provided, as most statistical analysis techniques make allowance for missing values.

1.2.2 ACCURACY

It is not enough to check that all questions have been answered; as far as possible data should be checked for consistency and accuracy. This type of check is probably best carried out using the computer once the data is in machine readable form. Useful accuracy checks include:

(1) Questions which are to be answered only on the basis of answers to previous questions should be checked carefully e.g.

(a) DO YOU SMOKE CIGARETTES? YES Ø ☐ (64)
NO 1 ☐

(b) IF YOU ANSWERED "YES" TO QUESTION (a), HOW MANY CIGARETTES DID YOU SMOKE YESTERDAY?

1. NONE
2. LESS THAN 10
3. 10 OR MORE BUT <20 ☐ (65)
4. 20 OR MORE
5. NOT APPLICABLE

You should check that if the respondent answered "no" to the first question, then he answered "not applicable" to the next; any other answer would be inconsistent, and would warrant further checking.

(2) Range checks should be made on all data items. This will also help to detect any punching errors which may not have been discovered.

Ensure that all precoded answers are within the bounds of the codes provided e.g.

For yes/no answers, where "yes" = 0 and "no" = 1, check that none of the codes are less than 0 or greater than 1.

For answers which require users to enter amounts which are not precoded (age, income etc), determine reasonable upper and lower bounds on these amounts, and carefully look at values which do not lie in these bounds. e.g.

If surveying first year students at university, ensure that the age given lies in a range of about 16 years to 40 years. Any answer given outside this range should be checked to ensure its validity.

1.3 PREPARING CODING SHEETS

Unless questionnaires have been designed with the intention of keypunching directly from them it is advisable that the results be encoded onto special coding forms. These may be of the standard 80 column type, available from the Computer Centre (see

fig.) or they may be designed for a special application. Users wishing to design their own special coding forms should consult with the computer centre beforehand.

When preparing coding sheets, the following rules should be observed.

(1) Use ink or biro rather than pencil.
If using pencil, use a soft grade (2B).

(2) If a mistake is made in a line, cross out the whole line, rather than trying to make untidy corrections.

(3) Ensure that there is no ambiguity of characters.
Observe the following conventions:

o - letter 'Oh'
I - letter I
Z - letter Z

Ø - zero
1 - one

(4) Take care in coding as keypunch operators will punch exactly what appears on the coding sheet and will not make any assumptions or corrections no matter how obvious any errors may appear.

1.4 STORING DATA IN MACHINE READABLE FORM

There are a number of ways of storing data in a form which can be read directly by the computer. The normal method which has been mentioned before, is to punch data onto 80 column computer cards. The data may then be transferred onto any one of the other storage media via this stage. It is hoped that a data preparation service which enters data directly onto magnetic disk, will be available in the near future.

1.4.1 PUNCHED CARDS

Punched cards are the traditional form of data storage, and are still widely used. They have 80 columns and 12 rows, and so may store up to 80 characters or digits or combination of both. Each character has a unique code which is seen as a number of punched holes in each column. Cards are read by the computer at the rate of 1000 cards per minute.

Punched cards have a number of advantages;

(1) Can be manipulated before being read by the computer (sorted etc)

(2) Easily read and interpreted by humans

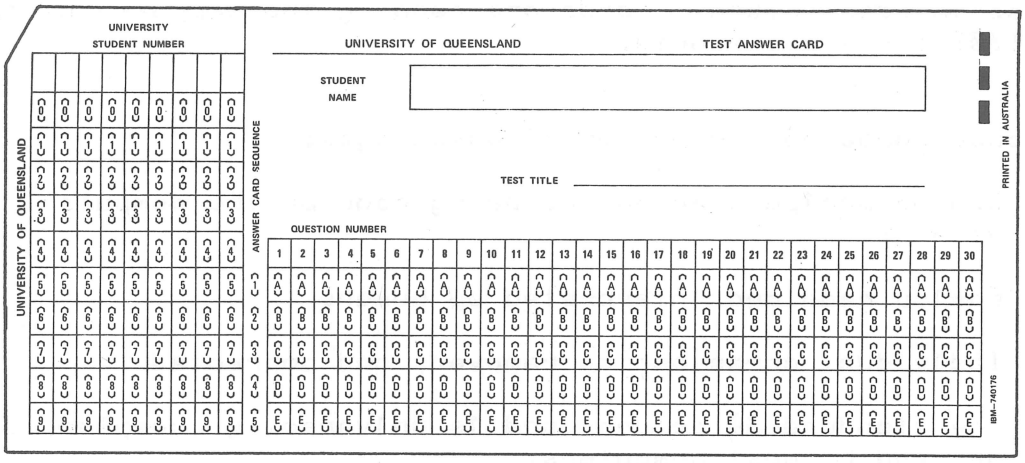
(3) Individual cards can be removed, and corrected.

(4) The standard 80 column card is universally accepted, and can be read by most computers.

However there are disadvantages such as bulk, durability and slow transfer rates which make other means of storage more practical. They are still useful as an initial means of transferring data into machine readable form.

1.4.2 OPTICAL MARK CARDS

These cards are similar in dimension and characteristics to punched cards, however, instead of punching holes to code characters, pencil marks are made in predefined fields.



These cards are useful for answering precoded question (e.g. multichoice exams) so that the respondent actually codes the answer directly onto the card. Generally speaking, users will require a special layout for application of these cards, and so should contact the computer centre for further information. Cards for the mark sense reader are printed with a special ink which does not interfere with the marks to be read, and users should be careful to ensure that the cards used are of this type. The mark sense card reader will read normal punched cards, provided that they are punched on the special cards mentioned before.

1.4.3 PAPER TAPE

Paper tape is one of the traditional storage mediums which is rapidly declining in use. Paper tape comes in 5, 6, 7 or 8 track forms, and the actual codes used for representing characters vary considerably.

Whilst some automatic data collection devices still punch paper tape output for computer processing, it should, where possible be avoided as a storage medium. Facilities are available at the Computer Centre for reading and punching paper tape. However, data obtained on paper tape should be transferred to some magnetic storage medium as soon as possible. Users with data on paper tape should consult the Computer Centre before attempting

to use it.

1.4.4 MAGNETIC TAPE

Magnetic Tape is a very commonly used storage medium for data on computers, but, the actual codes used vary among computer manufacturers. Generally, Magnetic tapes have either 7 or 9 tracks, and come in lengths of 1200 feet or 2400 feet. Whilst the computer centre does not encourage the use of magnetic tape for the storage of data, it is often a useful way of transferring data from computer to computer.

If users are obtaining data from some other source on magnetic tape, then as much as possible, the tapes should have the following characteristics.

1. Any length up to 2400 feet
2. Must be 7 track (or 9 track)
3. Packing Density of 200, 556 or 800 bits per inch.
4. Unlabelled
5. BCD or ASCII code

The Computer Centre has the capability to read most tapes that conform to these characteristics. If there is any doubt as to the type of magnetic tape, it is important to contact the computer centre before ordering. Data from magnetic tape will, if size permits, be transferred to disk. Other wise, it will be converted into a form easily read by the U.Q. machines, and put back onto magnetic tape. A nominal charge will be made for this service.

1.4.5 MAGNETIC TAPE CASSETTES

Some remote terminals have magnetic tape cassette facilities which enable users to enter data locally and store it on cassettes. This facility is only available to users with terminals of this kind. Unfortunately, there is no uniformity of code between different brands of cassette mechanism, and so they tend to be unsatisfactory for transportation of data between

machines.

1.4.6 MAGNETIC DISK

The PDP-10 computer system at the University of Queensland is a disk based system, providing users with a large but finite space for the storage of data. Magnetic disks provide the fastest, most sophisticated means of data storage yet discussed. It is generally inevitable that, whatever the original form, most data will end up on disk for some length of time during processing.

1.4.6.1 ONLINE DISK STORAGE

Data is stored on the PDP-10 disk system as named "files". (NOTE users who are not familiar with the file system on the PDP-10 should read MNT-2 Ch. 4 in detail before proceeding).

Each user has available an amount of "online" disk space in which to store files. This space is directly accessible any time the user logs onto the system (MNT-2 Ch 6). It is limited by a "logged out quota", which is the total amount of space a user's files may occupy when the user is "logged off" the system. A user is permitted to occupy more space while logged in - up to what is known as a "logged in quota". This extra space is provided to allow for the generation of temporary storage needed during the execution of a program, but is not available when the user logs off (See MNT-2 Ch 6.2).

It is possible to increase the size of a logged in quota on a temporary basis for very large jobs e.g. some SPSS jobs. This is explained in Chapter 4.

1.4.6.2 OFF LINE DISK STORAGE

"Offline" storage differs from "Online" storage in that files stored are not necessarily available to the user immediately after logging onto the system. It is however, cheaper than on line storage, with no real limit on the amount used. Offline disk storage can be effected in two ways.

(a) File Migration System

This system enables a user to request that one or more of his files be "migrated" or transferred from "online" disk storage to a general public "offline" disk storage area. Similarly a user may request that one or more files be transferred from the "offline" area to the "online" area. This is useful for storing infrequently used files, or files which are too large to leave on online storage (MNT-2 Ch 9).

(b) Private Disk Packs

A number of disk drives are available for use with privately owned disk packs. Users with very large data sets may wish to purchase or rent a disk pack for their exclusive use. (See MNT-1) The number of drives available for this purpose is limited, and so it is wise to book a drive ahead of when it will be needed. The amount of storage a user can have on a private disk pack is limited only by the capacity of the pack (30 million characters). The use of private disk packs is discussed in Sect. 1.4.6.4.

1.4.6.3 GETTING DATA ONTO DISK

There are a number of ways of transferring data to disk, however only two will be mentioned here. It is important that users refer to the appropriate sections in MNT-2.

(a) Creating a file with the EDITOR.

The "EDITOR" is a program which enables users to create and change files on disk. It is intended for use from a remote terminal (MNT-2 Ch 5.) and has the advantage of being fast and convenient when dealing with relatively small files. To "CREATE" a file, a user must first "LOG" onto the system (MNT-2 Ch 6.1) and run the editor (MNT-2 Ch 6.2.1). A more detailed description of the editor and all its facilities can be found in MNT-6 "A Line Editor for the PDP-10".

(b) Creating a file from a card deck.

It is desirable that data on punched cards be transferred to disk. This is particularly important where multiple analysis is to be performed on the same data set for although the card reader can read 1000 cards per minute, this is slow compared to the data transfer rates from magnetic disk. Also, punched cards are not a

very durable medium for storage, so as age and number of times read increases, the likelihood of of problems in reading them arises.

Using the computer via punched cards is called "batch processing". This means that instead of being able to input data directly and get an immediate response as with a remote terminal, users submit decks of cards which are processed in "batches" at some later stage. The user then receives the printout of results. (MNT-2 Ch 7. should be read at this stage). An example of a card deck to put data onto disk is given below.

```
$SEQUENCE
$JOB  [124,160]/NAME:NURK/COST:$2.00
$DECK  SURVEY.DAT
  .
  .
  .   (data cards)
  .
  .
$EOD
$EOJ
```

This copies the data on cards onto disk as a file called SURVEY.DAT. This file may then be manipulated in the same way as a file created from a remote terminal (See MNT-2 Ch 6.2.2 to 6.2.6).

1.4.6.4 USING PRIVATE DISK PACKS

Whilst the public disks are in operation at all times, a user wishing to utilize a private disk pack is required to mount the disk pack before using it (See MNT-16). Each private disk pack has a 4 character "logical" name which is allocated with the disk pack. When using a remote terminal, the "mount" command is used as follows.

```
.MOUNT EDUA: <cr>
```

This requests that a private disk pack called "EDUA" be mounted on a drive, and assigned to the user. If the job is to be run through Batch, the user should first consult the Computer Centre for advice on the methods available.

Bibliography

MOSER C.A. & KALTON G.
"Survey Methods in Social Investigation"
Heinemann Educational Books Ltd, London (1971).

2. BIOMEDICAL STATISTICAL PACKAGE (BMD)

2.1 INTRODUCTION

The BMD package is representative of the largest group of statistical packages - the set of "stand alone" programs. It was developed initially as a tool for research at the UCLA Medical Centre and catered as much as possible for the analytic problems of biomedical research. It has undergone a number of changes since it was introduced, including addition of new programs to cover new fields of analysis and refinements of existing programs.

There are at present about 55 individual programs in the BMD package, and these can be classified into six groups;

1. Description and Tabulation
2. Multivariate Analysis
3. Regression Analysis
4. Special Programs (Life & Contingency Tables)
5. Time Series Analysis
6. Variance Analysis

2.2 AVAILABLE PROGRAMS

The Computer Centre has made available a large number of the BMD programs, with a selection from each of the six classifications given before. It is possible to obtain an up to date list of the BMD programs which are available by typing the following command on a remote terminal, or placing a card with the same command punched onto it, in a batch run.

```
.DIR STA:BMD???
```

[STA: refers to the particular area of disk storage where the statistical programs are to be found.]

If it is wished to use a BMD program which is not currently available, the Computer Centre must be contacted, and assistance

may be given in obtaining the program.

2.3 USING A BMD PROGRAM

Before attempting to use any BMD program on the PDP-10, the BMD manual (published by the University of California Press) should be read carefully, particularly the chapter relating to the program to be used.

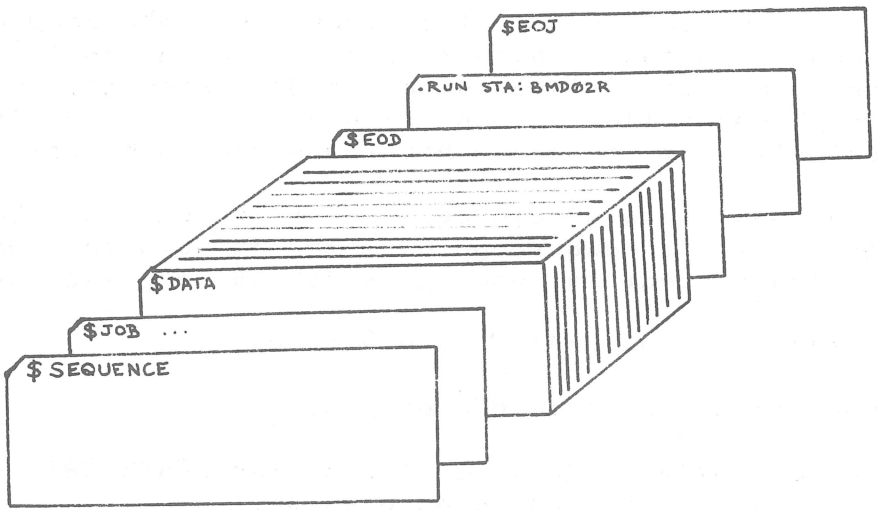
Users should note that some information in the BMD manual applies to run procedures for the particular IBM computer used at UCLA. This information should be ignored as different procedures apply for the PDP-10.

2.3.1 RUNNING A BMD PROGRAM THROUGH BATCH

2.3.1.1 DATA STORED ON CARDS

To run a BMD program through batch with data on cards, the following deck setup should be used.

```
$SEQUENCE
$JOB
$DATA
  .
  .
  .      (BMD control cards and input data as per
  .       BMD manual)
  .
  .
$EOD
.RUN STA:BMDØ2R          (or the name of the
                          particular
                          BMD program to be used)
$EOJ
```



2.3.1.2 DATA STORED ON DISK

To run a BMD program through batch with data on disk storage, the following procedure should be adopted.

1. Rename the file to FOR02.DAT i.e.

```
.RENAME FOR02.DAT=MYFILE.DAT
```

2. Assign logical unit 2 to disk i.e.

```
.ASSIGN DSK:2
```

3. Rename file back to original name (after running). The following deck set up will apply.

```
$SEQUENCE
$JOB [...,...]/NAME:SMITH/COST:$5.00
.RENAME FOR02.DAT=MYFILE.DAT
.ASSIGN DSK:2
.RUN STA:BMD05V
.RENAME MYFILE.DAT=FOR02.DAT
$EOJ
```

2.3.2 RUNNING A BMD PROGRAM FROM TERMINAL

Due to the volume of output produced from BMD programs, they are not really suited to direct running from a remote terminal. If, however, the output is directed to disk storage, and printed on the high speed line printer, the remote terminal is particularly convenient. The following procedure should be used.

(a) The input data should be in a file called FOR05.DAT. (If not, it is necessary to enter the data through the terminal as the program is running, which is very tedious.)

(b) Assign logical units 5 and 6 to disk.

(c) Run the appropriate BMD program

(d) The output will go to a file called FOR06.DAT, which may be printed on the high speed printer.

Example

```
.RENAME FOR05.DAT=MYFILE.DAT
Files renamed:
MYFILE.DAT

.ASSIGN DSK:5
DSK assigned

.ASSIGN DSK:6
DSK assigned

.RUN STA:BMD02R


EXECUTION TIME: 0.08 SEC.
TOTAL ELAPSED TIME:     0.52 SEC.
NO EXECUTION ERRORS DETECTED

EXIT


.PRINT FOR06.DAT
Total of 1 block in 1 file in LPT request

.RENAME MYFILE.DAT=FOR05.DAT
Files renamed:
FOR05.DAT
.
```

2.4 GENERAL HINTS ON RUNNING BMD PROGRAMS

(a) The amount of core memory available to individual users during prime shift (8 a.m. to 6 p.m.) is 32Kwords. Some of the BMD programs require more than 32Kwords of core memory and so must be run after 6.00 p.m., when 64Kwords of core is available. When using a remote terminal, the amount of core required can be determined by the following commands.

```
.GET STA:BMD02R
Job setup

.CORE
19+0/32K core
vir. core left=251K
.
```

The sum of these two figures will give the amount of core memory required (in this case, 19KWORDS). Batch users should specify the amount of core required by their job (if it is more than 32KWORDS) on the $JOB card. This prevents the job from actually being run until the required amount of memory is available. i.e.

```
$JOB [120,131]/NAME:SMITH/COST:$2.00/CORE:64K
```

(b) Users should endeavour to become familiar with the program they are going to use before committing a full set of data. A test run using a small set of data is particularly useful.

(c) Before seeing a consultant about problems in running a BMD program, check that the parameters specified on the PROBLM card are correct. Also, ensure that the data is in the appropriate format and there is the correct number of cards.

3. SCIENTIFIC SUBROUTINE PACKAGE (SSP)

3.1 INTRODUCTION

The Scientific Subroutine Package is a library of subroutines for programs written in FORTRAN IV. These subroutines cover most areas of statistical and numerical analysis, but require a main program written in FORTRAN IV to combine the necessary subroutines for an analysis.

Users should refer to the SSP manual before attempting to use any of the subroutines. A copy of this manual is available for viewing at the Computer Centre.

3.2 USING SSP

The SSP routines are kept on disk storage as a binary relocatable library file (i.e. a REL file). The SSP library file should be loaded with the user program before execution. i.e.

```
.EXECUTE/REL MYPROG,STA:SSP/SEARCH
```

The SSP routines contain no internal error reporting, and this should be taken care of in the user program.

> Note - SSP mathematical routines are also available, and may be located on the MAT: directory. Their method of use is the same as for the SSP statistical routines.

4. STATISTICAL PACKAGE FOR THE SOCIAL SCIENCES (SPSS)

4.1 INTRODUCTION

SPSS was one of the first statistical packages to employ a "total system" concept, whereby the package itself provided all or most of the necessary data handling and file manipulation facilities, as well as the required statistical analysis.

Ideally, such a package should minimize the amount of knowledge a user needs, about the particular computer system being used. Unfortunately, this is not really true in all circumstances, and so users who wish to perform analyses on large data sets in particular should endeavour to become familiar with the operation of the actual computer system being used. Such knowledge can often mean a considerable saving in time and expense. This section will discuss the use of SPSS on the PDP-10 both in simple applications, and more complex ones, with an emphasis on making most efficient use of SPSS and the PDP-10.

If any situations arise which are not covered in this manual, users should not hesitate to contact the Computer Centre.

4.2 THE PDP-10 IMPLEMENTATION OF SPSS

The PDP-10 implementation of SPSS (SPSS-10) was produced at the University of Pittsburgh and is a conversion of the National Opinion Research Centre's SPSSH Version 5.02. As well, some features of SPSS Version 6 are implemented in the PDP-10 version. SPSS-10 closely follows the documentation in:

NIE, BENT, HULL
STATISTICAL PACKAGE FOR THE SOCIAL SCIENCES
McGRAW-HILL INC. (1974)

Any differences between the PDP-10 implementation and that described in the McGraw-Hill manual are documented below.

4.2.1 RUNNING SPSS-10

SPSS-10 will run in a minimum of 32 KWORDS of core memory. Included in this is a "space" allocation of 1.5 KWORDS. This default of 1.5K is sufficient to run small jobs, however for larger jobs more space is required. The space requirements for a particular job can be calculated using the formulae given below, and then specified using the "SPACE" switch (see later).

Since the maximum memory allowed for a user during prime shift is 32K, only small SPSS-10 jobs may be run at this time. Jobs requiring more than 32K will be run after 6.00 p.m..

A typical batch deck for a simple SPSS job would be;

```
$SEQUENCE
$JOB [113,160]/NAME:SMITH/COST:$5.00
$SPSS
  .
  .
  .     (SPSS PROGRAM)
  .
  .
$EOD
$EOJ
```

To run an SPSS program which is on disk rather than cards, the following deck set up could be used.

```
$SEQUENCE
$JOB [113,160]/NAME:SMITH/COST:$5.00
.RU STA:SPSS
*LPT:=TEST.SPS
$EOJ
```

The general format is as follows;

```
.RU STA:SPSS

*Destination=Source,Switches
```

"Destination" and "Source" are the standard PDP-10 file specifications i.e.

```
        DEV:FILE.EXT[p,pn]<PROT>
```

where the following conventions will apply:-

	DESTINATION	SOURCE
DEV:	Default DSK:	Default DSK:
FILE	Defaults to source file name	no default (must specify)
EXT	Defaults to .LST	defaults to .SPS
[p,pn]	defaults to user's ppn	defaults to users ppn
<prot>	defaults to system standard	inappropriate

Example 1

```
        .RU STA:SPSS
        *LPT:=TEST.SPS[160,105]
```

This would look for the SPSS program in a file caled TEST.SPS on the [160,105] disk area. The output would be printed on the line printer.

Example 2

```
        .RU STA:SPSS
        *POVA:KRID[20,5]<155>=DSKD:DATA
```

This would look for the SPSS Program in a file called DATA.SPS on DSKD:. The output would be written to a file called KRID.LST which would be on the [20,5] area of a disk called POVA:. The file would be given a <155> protection.

SWITCHES

In SPSS-10, "switches" replace the specifications provided by the JCL cards in the IBM implementation (as described in the McGraw-Hill manual). The following switches are presently provided:-

/GET

- this switch follows a file specification and overrides any file specification found on the GETFILE control card (in the SPSS program). Although the specification field for a GETFILE card is overriden, use of the /GET switch still requires the occurrence of a GETFILE card in the SPSS program.

/HELP

- this switch causes a description of all switches to be typed on the terminal or in the log file.

/INPUT

- this switch follows a file specification, and overrides any file specification given on the INPUT MEDIUM control card in an SPSS program. The same conditions apply as for the /GET switch.

/OUTPUT

- this switch follows a file specification and overrides any file specification on the OUTPUT MEDIUM card. The same conditions apply as for the /GET and /INPUT switches.

/SAVE

- this switch follows a file specification and overrides any file specification on the SAVEFILE card in an SPSS program. The same coonditions apply as for the /GET, /INPUT and /OUTPUT switches.

/SCRATCH

- this switch follows a device specification and overrides the default scratch device which is used by SPSS to hold observations between statistical procedures. This file can e very large, and in fact for large SPSS jobs, may exceed the users logged in disk quota. If there is a risk of this occurring, the user should use a private disk pack or the system scratch pack (DSKS) for scratch purposes. NOTE: If only one statistical procedure and no save file is involved, then no scratch space is really necessary. Unfortunately SPSS does not take account of this, and will in fact still produce a large scratch area. To overcome this and

significantly reduce run time, use the null device (NUL:) as the scratch device.

/SPACE:n

- this switch is designed to allow users to specify memory requirements above the system default. The value of n is determined for each statistical procedure on the basis of formulae given below. If n > 225 it is assumed to mean "words" of memory. If n < 225, it is assumed to mean KWORDS (i.e. 1K=1024 words)

The following are examples of the use of switches both through batch and from a terminal.

(a) .RU STA:SPSS

*LPT:=TEST.SPS,DSKS:/SCRATCH,INPUT.DAT/INPUT/SPACE:5

(b) $SPSS DSKS:/SCRATCH,INPUT.DAT/INPUT/SPACE:5

The output would go to the line printer; the SPSS program would be read from DSK:TEST.SPS; the input data (i.e. INPUT MEDIUM card) would come from DSK:INPUT.DAT, irrespective of what file was actually specified on the INPUT MEDIUM card; the scratch file would be written to the public scratch pack DSKS: (ensure that DSKS: is mounted first), and 5140 words of memory would be provided for SPACE.

4.2.2 SPSS CONTROL CARDS

In SPSS-10 the general control card format is free field (as opposed to the McGraw-Hill manual which defines control cards in a fixed format), and is interpreted as follows;

1. If column 1 contains neither a blank nor a tab character, then all columns from col 1 up to a tab or two consecutive blanks, or to col 15 are considered the control field.

2. If the card begins with one or more blanks or tabs, then the card is a continuation card, and all characters are part of a specification field.

3. The specification field of a card may contain no more than 65 characters, irrespective of leading blanks.

4. Any tab which is encountered in the specification field is replaced by a single blank character, and the specification field is printed left justified.

5. If the 'numbered' option is specified, then the numbering field must begin in col 73.

The specification fields of some of the SPSS-10 conrol statements differ from those given in the McGraw-Hill manual. These differences are documented below.

(1) FILE NAME FILE NAME, FILE LABEL

The file name card should be considered as documentation only. Its specification fields are stored and retrieved with SPSS-SYSTEM save files and they are printed on listings. But they should not be confused with the file specifications which appear on the GETFILE or SAVEFILE cards and are seen in the user's directory listing. In SPSS-10 the SPSS-10 the number file name card may be used to change a previous file name and file label after a GETFILE and before a SAVEFILE. The information on the file name card is used by SPSS-10 for creating SPSS system save file names when the user has not explicitly given the information on the save file command. This procedure is not recommended.

(2) OF CASES NUMBER OR "UNKNOWN" OR "ESTIMATED NUMBER" OR NUMBER IN 1ST SUBFILE, 2ND SUBFILE, ...

In SPSS-10 the number following the keyword estimated may be smaller than the actual file size. It is used to allocate contiguous blocks of disk storage for scratch and save files. The version 6.00 keyword "unknown" may be used if no estimate is available.

(3) INPUT MEDIUM CARD OR FILE SPECIFICATION

The input medium card takes the standard decsystem-10 file specification. The keyword "card" may also be used as described in the McGraw-Hill Manual. The Keywords "Tape", "disk", and "other" are not recognized. If they are used, then the proper specification should be provided with the /input switch. The default device is DSK:. The file name and extension do not

default, the project-programmer number defaults to the user's and protection is inappropriate. The ability to reference an alternate project-programmer number is particularly useful for allowing several students or researchers to reference a common data base. Two buffers are used. The physical blocksize is the installation device default and may be changed with the SET BLOCKSIZE monitor command (see decsystem-10 User's Handbook).

(4) INPUT FORMAT FREEFIELD OR FIXED (FORMAT LIST)

The fixed format specification may contain the format control characters: A, E, E, G,O, T, and X. Variables read in with an A-type format are automatically given print format (A), therefore the print format card is required only if the variable is later recoded to F-Type. Up to five characters can be contained in an A-Type variable. If fewer are used then they will be left justified and blank filled. The freefield definition has been changed to correspond to DEC's forots list directed I/O definition. See Fortran-10 Language Manual. This means that each case must start on a new card. Freefield may be used in conjunction with # of cases unknown. A control-Z provides the end-of-file when input is from a terminal.

(5) MISSING VALUES VARIABLE LIST (VALUE LIST) / ...

THE KEYWORD "BLANK" MAY NOT BE USED TO SPECIFY MISSING VALUES.

(6) OSIRIS VARS VARIABLE NAME LIST

In SPSS-10 when reading an IBM SYSTEM/370 OSIRIS tape, an input medium card must precede the osiris vars card. A .set blocksize dev: n monitor command must be issued after the tape mount. Osiris dictionary files have a block size of 1600 characters and would therefore require n to be 400 words. If the data file block size is larger, then n should be set to the actual number of bytes divided by four, rounded up to the next full word. The tape must be positioned at the osiris dictionary file except that tape-label files will automatically be skipped. This may be accomplished using PIP. The data file is assumed to follow the dictionary file but may be separated from it by tape-label files.

(7) OUTPUT MEDIUM CARD OR FILE SPECIFICATION

The output medium card has been added as the output analog of the input medium card. It takes a file specification and is used to specify the destination of the write cases procedure, the aggregate procedure the matrix output options, and the optional output of fastmarg, etc. If output medium is absent then the

default is DSK:FOR09.DAT. When present the defaults are: device defaults to DSK:, file name defaults to the source name, and extension defaults to .DAT. If this file is later queued to the line printer and if it has no carriage control characters then it should be queued with switch /FILE:ASCII. Two buffers are used. The physical blocksize is the installation device default and may be changed with the .SET BLOCKSIZE monitor command.

(8) PRINT FORMATS VARIABLE NAME LIST (VALUE) / ...

In SPSS-10, variables read with an alphanumeric format code are automatically given print format (A), therefore the print format card is required for these variables only if they are later recoded to numeric values.

(9) SUBFILE LIST SUBFILE NAME LIST

(10) VARIABLE LIST VARIABLE NAME LIST

(11) VAR LABELS VARIABLE NAME, LABEL / ...

(12) FINISH

The finish card may be absent. SPSS-10 will then generate one. This is useful since input medium card will produce a more useful diagnostic if the number of data cards is short.

(13) KEYPUNCH 029

Keypunch 026 is not implemented since this is easily accomplished with the $MODE 026 batch command.

(14) PAGESIZE 'N' OR 'NOJECT'

The noject keyword is a version 6.00 feature intended for paper conservation. It is implemented in SPSS-10 and is the default. This default may be overriden by a pagesize command.

(15) COMMENT ANY TEXT

(16) COUNT RESULT VARIABLE = VARIABLE LIST (VALUE LIST)
*COUNT RESULT VARIABLE = VARIABLE LIST (VALUE LIST)

The keyword 'blank' is not recognized.

(17) DOCUMENT ANY TEXT

(18) NUMBERED 'YES' OR 'NO'

(19) PRINT BACK 'YES' OR 'NO' OR FORMAT OR CONTROL

(20) RUN NAME 64 CHARACTER LABEL

(21) TASK NAME TASK LABEL

(22) SAMPLE FACTOR / SEED = POSITIVE ODD INTEGER
*SAMPLE FACTOR / SEED = POSITIVE ODD INTEGER

In SPSS-10 the seed for the pseudo-random number generator is not generated from the hardware clock. Instead the sample and each *sample card sets its own seed and generates its own sequence of pseudo-random numbers. Thus separate runs employing sampling yield reproducible results. The seed = keyword has been added in order to get the seed. Its value should be a positive odd integer. If seed = is absent or if it is set to zero then a standard seed is used.

(23) SELECT IF (LOGICAL EXPRESSION)
*SELECT IF (LOGICAL EXPRESSION)

It should be noted that multiple SELECT IF and *SELECT IF cards act as if they were disjoined, that is, logical "OR"ed together.

(24) WEIGHT VARIABLE NAME
*WEIGHT VARIABLE NAME

(25) GET ARCHIVE

Archiving is not yet implemented.

(26) GET FILE FILE SPECIFICATION

In SPSS-10 the GET file card specifies the physical device from which the SPSS-system file will be fetched. It takes a standard decsystem-10 file specification. DEV: defalts to DSK . File and

extension if appropriate do not default. ppn defaults to the user's projectprogrammer number and protection is inappropriate. Two buffers are used. The physical blocksize is the installation device default and may be changed with the .set blocksize monitor command. The file name on the SPSS-system file is not verified against the specification field of the get file command.

(27) REORDER VARS VARIABLE LIST

(28) SAVE ARCHIVE

Archiving is not yet implemented.

(29) SAVE FILE FILE SPECIFICATION

The save file card specifies the physical device to which the SPSS-system file will be written. It takes a standard DECsystem-10 file specification DEV: defaults to DSK:. File and extension if appropriate do not default. ppn defaults to the user's project programmer number and protection defaults to the installation default. Two buffers are used. The physical blocksize is the installation device default and may be changed with the .set blocksize monitor command. In SPSS-10 the file label specification field of a save file command is ignored. To change a file name or file label in creating a new SPSS-system file use a file name command following the GET file.

(30) SORT CASES VARIABLE LIST (S), ...

SPACE = (NUMBER OF SORT KEYS + 1) NUMBER OF CASES

```
(31) AGGREGATE      GROUPVAR = VARIABLE LIST/
                    VARIABLES = VARIABLE LIST/
                    ACTIONS = SUM  OR
                              N   OR
                              NS   OR
                              MEAN OR
                              SD   OR
                              VALUE OR
                              PCTGT(CONST)  OR
                              PCTBIN(CONST1,CONST2)/
                       SET = YES OR NO/
                       RMISS = CONST/
                       FORMAT = STANDARD  OR  BINARY
```

Options:

1 Inclusion of missing data.

	2	Listwise Deletion of missing data.
	3	Create aggregated output file.
Statistics:	1	Complete tables printed (instead of summary).

Space = (10 * NV + NP) /4

NV = TOTAL numbers of variables on all variables = lists.
NP = TOTAL number of new variables on the output file.

(32) ANOVA DEPENDENT VARIABLE LIST BY INDEPENDENT VARIABLE (MIN,MAX) independent variable list (MIN,MAX) ... with covariate list/ dependent variable list ...

Options:

1	Include Missing Data.
2	Suppress Value labels.
3	Ignore 2-way and higher interactions.
4	Ignore 3-way and higher interactions.
5	Ignore 4-way and higher interactions.
6	Ignore the 5-way interaction.
7	Process covariates concurrently with main effects.
8	Process covariates after the main effects.
9	Assess all effects simultaneously.
10	Use Hierarchial procedure within nonmetric factors and covariates.

Statistics:

1	Multiple classification analysis.
2	Output standardized partial regression coefficients.

Space = SE + SM + 15 * MR + 2 * MN

SE = Sum of the values of E for each list
E = Product on 1 of V (I)
V (J) = Number of values specified for independent variable J
SM= Sum of the values of M
M = R * (R + 1) / 2
R = E - 1 + D + C
D = Number of dependent variables
C = Number of covariates
MR = Maximum R
MM = Maximum value of M

The current version of anova is a field test version from NORC.

(33) BREAKDOWN VARIABLE LIST BY VARIABLE LIST BY ...

1	Inclusion of missing data.
2	Exclusion of missing data for dependent variables.
3	Suppression of labels.
4	Condensed output format (default)
5	Original output format.

Statistics:

1	One way analysis of variance table.
2	Table of Linearity. (Cannot use 2 without 1).

Space = (MAXCELLS + 1) * (MD + 5)

MAXCELLS = Sum of unique combinations of the independent or control variables.
MD = Maximum number of uses of keyword by.

(34) CANCORR VARIABLES = VARIABLE LIST/

RELATE = VARLIST WITH VARLIST/...

Options:

1	Inclusion of missing data.
2	Pairwise deletion of missing data.
3	Correlation matrix input.
4	correlation matrix output.

Statistics:

1	Means and Standard deviations.
2	Correlation matrix.
3	Canonical correlation table.

For the First List:

SPACE = MAX (SPACE1, SPACE2)

SPACE1 = 2 * (NV * NV + 2 * NV + M * (M + 11) / 2 + M * MQ)

SPACE2 = 2 * (NV * NV + 3 * NV + QP * 2 * NV * NV)

NV = Number of variables following the variables =
M = Total number of variable in the relate = list
MQ = Larger number of variables on one side of with
QP = Ø for listwise deletion, 1 for pairwise deletion

For multiple relate = lists, space1 is the largest calculated for any out of all lists.

(35) CODEBOOK VARIABLE LIST OR ALL

Options:

1	Inclusion of missing data.
2	Suppression of labels.
3	Not used.
4	Generation of histograms.
5	Generation of histograms with summation of frequency tables.

Statistics:

1	Mean.
2	Standard Error.
3	Median.
4	Mode.
5	Standard deviation.
6	Variance.
7	Kurtosis.
8	Skewness.
9	Range.
10	Minimum.
11	Maximum.

Space is either:

Space = maxvals * (8 + 2 * NVARS) + 4 * NVARS + 15
Maxvals = non-missing values in largest variable.
Nvars = number of variables.

OR:

Space = Maxvars * (4 + 2 * NVALS) + 8 * NVALS + 15
Maxvars = Maximum number of variable for a value of nvals.
NVALS = Number of values in largest variable list.

(36) CONDESCRIPTIVE VARIABLE LIST OR ALL

Options:

1	Inclusion of missing data.
2	Suppression of variable labels.

Statistics:

1.2	See Corresponding statistics for codebook.
3-4	Not used.
5-11	See corresponding statistics for codebook.

Space = 13 * NVAR

(37) CROSSTABS VARIABLE LIST BY VARIABLE LIST BY ...

Options:

1	Inclusion of missing data.
2	Suppression of labels.
3	Deletion of row percentages.
4	Deletion of column percentages.
5	Deletion of total percentages.

Statistics:

1	Chi-Square
2	Phi for 2 x 2 table. cramer's v for larger tables.
3	Contingency coefficient.
4-5	Not used.
6	Kendall's Tau B.
7	Kendall's Tau C.
8	Gamma.
9	Somer's D (ASYMMETRIC).

Space = (MAXCELLS + 1) * (D + 3)

(38) DISCRIMINANT GROUP VARNAME(NSTEP,F1,F2,TOL) WITH
VARIABLE LIST OR
VARIABLE NAME1(CNT1)..../ OR
LEVELS=(M1,M2,.....Mj)

Options:

1	Suppression of labels.
2-3	Not used.
4	missing data included in calculations.
5	Not used.
6	Omit step output.
7	Omit discriminant functions.

Statistics:

1	Print group means.
2	Print standard deviations.
3	Compute and print covariance matrix.
4	Compute and print correlation matrix.

Space = NVAR * (2 + 2 * NGRP + NVAR + NVAR * NGRP)

NVAR = NUMBER OF VARIABLES FOLLOWING WITH
NGRP = NUMBER OF GROUPS

(39) FACTOR VARIABLES = VARIABLE LIST/
TYPE = PA1/DIAGONAL=VALUE LIST/ OR

```
            PA2/ OR
            RA0/ OR
            ALPHA/ OR
            IMAGE/ OR
            BYPASS/
      NFACTOR = NUMBER OF FACTORS DESIRED/
      MINEIGEN = DESIRED EIGENVALUE/
      ITERATE = MAXIMUM NUMBER OF ITERATIONS/
      STOPFACT = CONVERGENCE CRITERION/
      FACSCORE/ OR FACSCORE = MOP
      ROTATE = VARIMAX/ OR
             QUARTIMAX/ OR
             EQUIMAX/ OR
            OBLIQUE/ OR
            NOROTATE/
     DELTA = OBLIQUE ROTATION/ OR
            START, INCREMENT, END/
```

Options:

1	Inclusion of missing data.
2	Pairwise deletion of missing data.
3	Correlation matrix input.
4	Factor matrix and communalities input.
5	Correlation matrix output on output medium.
6	Factor matrix and communalities output on output medium.
7	Not used.
8	Means and standard deviations of variables on variables=list output on output medium. (Cannot be used if either/both of options 3 and 4 are used.
9	Matrix Processing mode.
10	Not used.
11	Factor score records sequenced in columns 1-20 output on output medium.
12	If option 2 specified, option 12 produces weighted factor score records on output medium.

Statistics:

1	Means and Standard deviations.
2	Correllation matrix.
3	Inverse and determinant of matrix.
4	Communalities, eigenvalues and proportion of total and common variance.
5	Initial factor matrix.
6	Rotated factor matrix and transformation matrix.
7	Factor-score cooefficient matrix.
8	Plot of rotated factors.

Space = 2 * (9 * M * M + M)

M = TOTAL NUMBER OF VARIABLE FOLLOWING VARIABLE =

Bug: Edit will produce spurious error messages for factor. To avoid this error do not use edit on factor runs; rather check the control cards prior to a live run.

Bug: Factor, keywords PA2 and NFACTOR. If NFACTOR is set equal to the number of variables on the variable=list and the correlational matrix is ill-conditioned, SPSS will enter an endless loop. To avoid this problem do not set NFACTOR equal to the number of variables, or specify type=PA1.

(40) FASTABS VARIABLES = VARIABLE LIST(LOW,HIGH)/
TABLES = VARLIST BY VARLIST BY.../...

Options:

1	Inclusion of missing data.
2	Suppression of labels.
3	Deletion of row percentages.
4	Deletion of column percentages.
5	Deletion of total percentages.
6	Not used.
7	Missing values included in fastabs tables but excluded from calculation of statistics.

Statistics:

1	Chi square.
2	Phi for 2 x 2 table.
3	Contingency coefficient.
4	Lambda symmetric and asymmetric.
5	Uncertainty coefficient symmetric and asymmetric.
6	Kendall's Tau B.
7	Kendall's Tau C.
8	Gamma.
9	Somer's D symmetric and asymmetric.
10	Eta.

Space = Maxcells + 10 * NVARS

(41) FASTBREAK VARABLES = VARLIST(LOW,HIGH)/
tables - varlist by ... BY VARLIST

Options:

1	Inclusion of missing data.
2	Exclusion of missing data for dependent variables only.

3 Suppression of labels.

Statistics:
1 One way analysis of variance table.
2 Table of Linearity. (Cannot use 2 without 1).

(42) FASTMARG VARIABLE LIST(LOW,HIGH)

Options:
1 Inclusion of missing data.
2 Suppression of value labels.
3 8 1/2" x 11" output format.
4 All output on output medium.
5 Deletion of frequency distributions.

Statistics: 1-11 see corresponding statistics for codebook.

Option 4 of fastmarg writes fortran carriage control characters at the front of each line. If the output medium is not LPT: then these characters may be properly interpreted using the /P switch in PIP.

(43) GUTTMAN SCALE SCALE NAME = VARNAME(DIVISION POINT) ...

Options:
1 Inclusion of missing data.
2 Suppression of variable labels.
3 Suppresses ordering of variables.

Statistics:
1 Inter-item and part-whole correlation coefficients.
2 Coefficient of reproducibility.
3 Minimum marginal reproducibility.
4 Percent improvement achieved by Guttman scale.
5 Coefficient of scalability.

(44) LIST CASES CASES = NUMBER/VARIABLES = VARIABLE NAME LIST

(45) MARGINALS VARIABLE LIST OR ALL

Options:
1 Inclusion of Missing data.
2 Suppression of variable labels.
3 Suppression of cumulative frequencies.
4 Not used.

5 Suppression of frequency table.
6 Missing value included in tables but deleted from statistics.

1-11 See corresponding statistics for codebook.

SPACE = 2 * NVARS + MAXVALS * (2 * NVARS + 3)

(46) NONPAR CORR VARIABLE LIST WITH VARIABLE LIST OR VARIABLE LIST

Options:
1 Inclusion of missing data.
2 Listwise deletion of missing data.
3 One and two tailed tests of significance.
4 Correlation matrix output. (with must not be used).
5 Kendall correlations.
6 Kendall and Spearman correlations.

Statistics:
No additional statistics available.

SPACE = MXCASE * (NVAR + 1)

or if the default missing value option is selected:

SPACE = MXCASE * (NVAR + 3)

(47) ONEWAY GROUPS(K)= GROUP COUNTS/ OR
SUBFILES/ OR
VARIABLE NAME/
VARIABLES = VARIABLE LIST/
POLYNOMIAL= N/
CONTRAST=COEFF. LIST/...
RANGES = RANGE SPEC./..

Options:
1 Inclusion of missing data.
2 Listwise deletion of missing data.
3 Suppression of variable labels.
4 Group counts, means, standard deviations written on output medium.
5 Not used.
6 Use value labels of group indicator variable.
7 Input is group counts, means, standard deviations, not raw data.
8 Input is group counts, means, pooled variance.

Statistics:

1	Group counts, means, std. deviations, standard errors, minimum, maximum, 95% confidence interval, interval for the mean.
2	Fixed and random effect measures.
3	Cochran's C. Bartlett Box F, Max/Min variance ratio.

SPACE = NG * (2 + 5 * NV * NP2)

NV = NUMBER OF VARIABLES FOLLOWING VARIABLE =
NG = NUMBER OF GROUPS
NP2 = IF POLYNOMIAL THEN DEGREE OF POLYNOMIAL + 2 ELSE 0.

BUG: Groups which have no cases are considered valid groups and groups with a variable value of 0 are ignored. To avoid this error, prior to using oneway, obtain marginals and recode variables accordingly. For example, if a variable has cases for the following non-missing values: 0, 2, 3, 5, 8, it should be recoded to (0=1) (5=4) (8=5). No update is available at this time: please use the bypass.

(48) PARTIAL CORR VARIABLE LIST WITH VARIABLE LIST BY OR VARIABLE LIST (ORDER VALUES) OR MATRIX TYPE VARIABLE LIST

options:

1	Inclusion of missing data.
2	Pairwise deletion of missing data.
3	One and two tailed tests of significance.
4	Correlation matrix input.
5	Correlation matrix output.
6	Matrix Processing mode.
7	Rectangular matrix output (default).
8	Original output format.

Statistics:

1	Correlations with degrees of freedom and significance.
2	Means, standard deviations.
3	Forced printing of correlation matrix when non-computable correlations are encountered.

(49) PEARSON CORR VARIABLE LIST WITH VARIABLE LIST OF VARIABLE LIST

Options:

1	Inclusion of missing data.

2 Listwise deletion of missing data.
3 One and two tailed tests of significance.
4 Correlation matrix output.
5 Rectangular matrix output (default).
6 Original output format.

Statistics:
1 Means and standard deviations.
2 Cross-product deviations and covariance.

(50) REGRESSION VARIABLES = VARIABLE LIST/
REGRESSION = DEPENDENT VARIABLE.
(PARAMETERS) WITH INCLUSION LIST
(INCLUSION LEVEL)
RESID = M

Options:
1 Inclusion of missing data.
2 Pairwise deletion of missing data.
3 Suppression of variable labels.
4 Correlation matrix input.
5 Correlation matrix input plus means and standard deviations as input. (Cannot use 5 without 4).
6 Suppression of step-by-step output.
7 Suppression of summary table.
8 Correlation matrix output.
9 Matrix processing mode.
10 Sequencing information placed in columns 1-20 of residual/predictor records.
11 Predictors output on output medium.
12 Do not output residuals.
13 Weighted standardised predictors output.
14 Suppress printing of axes on plots.
15 Means and standard deviations of variables on variables = list(s) output on output medium.
20 Regression through the origin.

Option 20 was provided by Prof. David A. Specht of Iowa State University.

Statistics:
1 Correlation matrices.
2 Means, standard deviations, and number of valid cases.
3 Forced printing of correlation matrix when non-computable correlations are encountered.
4 Plot standardised predictor against standardised residual.

5 Durbin-Watson test statistic output.
Case order plot of the standardised residual calculated by the last resid = Ø in the regression procedure.
20 Sequential F-Test.

Statistic 20 was provided by Prof. David A.Specht of Iowa State University.

Space = MAXØ (SPACEØ, SPACE1).

SPACEØ = 2 * (NVARS + MATRIX)

NVARS = THE NUMBER OF VARIABLES IN ALL VARIABLES LISTS.
MATRIX = SUMATION ON I OF 3 * N (I) * N (I)
IF OPTION 2 IS SPECIFIED.
N (I) = NUMBER OF VARIABLES IN ITH LIST.
MATRIX = SUMMATION ON I OF N (I) * N (I) +
MAXØ on I OF (MAXØ ON J OF M (I, J) * N (I))
IF LISTWISE DELETION OF MISSING VALUES.
M (I, J) = NUMBER OF VARIABLE IS JTH REGRESSION LIST
ØF ITH VARIABLE LIST.

SPACE1 = 5.5 * V + 3 * I + 17.5 * N + Q4 * 1Ø36 * N

V = SUM OF VARIABLES ON EACH VARIABLE LIST RELATED
TO REGRESSION LISTS FOR WHICH RESIDUALS WERE REQUESTED.
I = NUMBER OF INDEPENDENT VARIABLES.
N = NUMBER OF RESID = REQUESTS
Q4 = 1 IF STATISTIC 4 ELSE Ø.

(51) RELIABILITY VARIABLES = VARIABLES LIST/
SCALE LABEL) = SCALE LIST/MODEL = /
SCALE LABEL) = SCALE LIST/ ...
VARIABLES = VARIABLE LIST/ ... etc.

Options:

1 Include missing data.
2 Not used.
3 Don't search for labels.
4 Input covariance matrix.
6 Input matrix is triangular.
7 Input means.
8 Punch covariance matrix.
9 Index matrices using SPSS master variable list.
10 Stop after punching covariance matrix.
11 Punch means.
12 Triangular input matrices are punched.
As singular vector (option 6
must also be specified and

option 9 may not be used)
13 Punch matrices as single vector.
rather than by rows.

Statistics:

1	Item means, std. dev.
2	Covariance matrix.
3	Correlation matrix.
4	Scale mean, scale variance.
5	Statistics on means.
6	Statistics on variances.
7	Statistics on covariances.
8	Statistics on correlations.
9	Item - total statistics.
10	Analysis of variance.
11	Tukey test for additivity.
12	Hotelling's T-squared.
13	Input std. dev. and correlation matrix.

SPACE = 2 * (TRIANGLE + MATRIX + 5 * LABEL + TUCKEY)

TRIANGLE = Ø IF ONLY ONE SCALE LIST IS SPECIFIED, ELSE
TRIANGLE = SUMMATION ON I OR N (I) * (N (I) + 1) / 2
N (I) = NUMBER OF VARIABLE IN ITH VARIABLE LIST
MATRIX = MAX * (MAX + 3)
MAX = NUMBER OF VARIABLE IN LONGEST SCALE LIST
LABEL = Ø IF OPTION 3 IS SPECIFIED, ELSE
LABEL = SUMMATION ON I OF N (I).
TUKEY = Ø IF STATISTIC 11 IS NOO REQUESTED, ELSE
TUKEY = SUMMATION ON I AND J OF M (I, J)
M (I, J) = NUMBER OF VARIABLES IN JTH SCALE LIST OF
THE ITH VARIABLE LIST.

Subprogram reliability was provided by Prof. David A. Specht of Iowa State University.

(52) SCATTERGRAM VARIABLE LIST OR
VARIABLE LIST WITH VARIABLE LIST OR
VARIABLE LIST (LOW,HIGH) OR
VARIABLE LIST (LOW,HIGH) WITH OR
VARIABLE LIST(LOW,HIGH)

Options:

1	Inclusion of missing data.
2	Listwise deletion of missing cases.
3	Suppression of variable labels.
4	Suppression of Plot grid lines.
5	Diagonal grids provided.
6	Two tailed test of significance.
7	Automatic scaling

8 Plot as many cases as possible.

Statistics:
1 Pearson's R.
2 R Squared.
3 Significance of R.
4 Standard error of the estimate.
5 Intercept with the vertical axis.
6 Slope.

SPACE = 2 * NV + NV * MXCASE + MXCASE

NV = NUMBER OF VARIABLES ON SCATTERGRAM CARD.
MXCASE = MAXIMUM NUMBER OF CASES.

The version 6.00 feature to bypass plots if too many cases are implemented. See Option 8.

(53) T-TEST GROUPS = GROUP SPECIFICATION/
VARIABLES = VARIABLE LIST
PAIRS = VARIABLE LIST OR
PAIRS = VARLIST WITH VARLIST

Options:
1 Inclusion of missing data.
2 Listwise deletion of missing data.
3 Suppression of variable labels.

Statistics:
No additional statistics available.

SPACE = 2 * (IP + 6 * NV + 6 * NP)
IP = NUMBER OF ARGUMENTS ON T-TEST CARD.
NV = NUMBER OF VARIABLES FOLLOWING =.
NP = NUMBER OF PAIRS FOLLOWING PAIRS =.

(54) WRITE CASES (FORMAT LIST) VARIABLE NAME LIST

Options:
1 Listwise deletion of missing data.

The letter O (Octal) is a valid conversion code.
Note that the letter I (Integer) is not.

4.3 RUNNING SPSS FROM TERMINAL

As well as running SPSS-10 from a terminal and using a program in tle form of a disk file (as described earlier), it is possible to enter an SPSS program directly.

When the source device is TTY:, SPSS-10 prompts the user with a right angle bracket ">". A command or a procedural request with accompanying statistics and options specifications may then be entered in free field format. After the last continuation line (if any), typing an "escape" or "altmode" will cause SPSS to execute the request immediately.

This feature is convenient when used in conjunction with a save file, for applying repeated statistical analysis to a common data set.

4.4 SOME EXAMPLES OF SPSS RUNS ON THE PDP-10

(a) A simple batch run using an SPSS program with data stored on cards.

```
$SEQUENCE
$JOB [60,105]/NAME:SMITH/COST;$5.00
$SPSS
  .
  .
  .     SPSS program (including data)
  .
  .
$EOD
$EOJ
```

(b) A simple batch run, but with a very large number of data cards.

In this case, it is recommended that the actual input data is first read onto a disK file so that for later runs, all of the cards need not be read in again.

```
$SEQUENCE
$JOB etc.
$DECK INDOT.DAT
  .
  .
  .    (actual input data)
  .
  .
$EOD
$SPSS

input medium INDOT.DAT

$EOD
$EOJ
```

-to use the same data set again in subsequent runs.

```
$SEQUENCE
$JOB
$SPSS

input medium INDOT.DAT

$EOD
$EOJ
```

If the input data is on a private disk pack, then this pack must be available when the job is being run.

4.5 GENERAL HINTS ON RUNNING SPSS-10

(a) Unless otherwise specified, a default time limit of 5 minutes processor time is allocated for each job on the PDP-10. As a general "rule-of-thumb", one minute of processor time should be allowed for each $2.00 of cost limit for a standard priority run (1 minute per $1.00 at low priority).

Thus if a job is expected to cost approximately $20, a time limit of 10 minutes should be allowed (in standard priority) i.e.

```
$JOB [60,105]/NAME:SMITH/COST:$20.00/TIME:20
```

(Note: It is more desirable for a job to stop with "COST LIMIT EXCEEDED" than "TIME LIMIT EXCEEDED".)

(b) It is difficult to estimate the actual cost of a particular SPSS run, because of the fact that few SPSS users are alike. Also, the cost of an SPSS run does not necessarily increase proportionally with the number of variables, cases, options or statistics etc. Generally, some "order of magnitude" cost estimate can be given, however, if in doubt, the limit should be set high rather than low.

The effects of "COST LIMIT EXCEEDED" can be minimized in three ways;

(i) The user should try to keep a record of all SPSS runs including information on the number of variables, statistics, options, procedures, priorities, observations etc. This will assist in the making of more accurate estimates.

(ii) As far as possible break large jobs up into small subjobs. For example, if a job required to process all of 80 subfiles, the job should be split up into 4 smaller jobs processing 20 subfiles.

(iii) If the job is being run from a remote terminal, and it stops because of an exceeded cost limit, the user may reset the cost limit and continue, e.g.

```
.RU STA:SPSS

*LPT:=TEST.SPS


?COST LIMIT EXCEEDED


EXIT

.SET COST +$10.00

.CONTINUE
```

If it is not wished to continue processing, but to check what results have already been obtained, the files should be "CLOSEd" as follows;

```
.RU STA:SPSS

*LST:= TEST.SPS

?COST LIMIT EXCEEDED

.SET COST +$2.00

.CLOSE

.PRINT TEST.LST
```

Because of the iterative nature of most SPSS jobs, the output obtained up to the point of stopping is useful, and so it may not be necessary to return the whole job.

If the job is running through batch, and a "COST LIMIT EXCEEDED" occurs, the above procedure of closing the files will be undertaken automatically.

(c) When embarking on a new SPSS project, try all procedures to be used, on a small test deck of data. Experience gained in doing this may save considerable time and expense on a later run using a full set of data.

(d) Confusion exists regarding the use of filename extensions in SPSS. Both the input command file and the SPSS system file generated by a SAVE FILE command take ".SPS" as default extension. Using the same filename (with no extension) for both the command file and the system file can cause the overwriting of the command file. To avoid confusion either use different filenames for command and system files or give explicit extensions in both cases.

(e) Users are reminded that SPSS expects all numbers input to be in floating-point format. Therefore "I" format should not be used. (Use of I format is likely to result in all data being regarded as zero.)

(f) Before seeing a consultant about an SPSS problem, ensure that all files and programs are available, and that an up-to-date listing is obtained. Experience has shown that many SPSS errors are caused by not strictly adhering to the procedures and limitations as documented in the manual, so it is wise to

closely check these in relation to the particular statistical procedure being employed.