



## RESEARCH REPORT

SURVO 76 EDITOR, A NEW TOOL  
FOR INTERACTIVE STATISTICAL COMPUTING,  
TEXT AND DATA MANAGEMENT  
by  
Seppo Mustonen

No. 19

August 1980

DEPARTMENT OF STATISTICS  
UNIVERSITY OF HELSINKI  
SF 00100 HELSINKI 10 FINLAND

## CONTENTS

1. Introduction	1
2. An introductory example	3
3. Use of edit keys	13
4. Text editing	15
4.1. Clearing the edit field	15
4.2. Moving parts of the edit field	17
4.3. Saving, loading and combining edit fields	18
4.4. Printout of the edit field and edit files	19
4.5. Making up and adjusting the text lines	20
4.6. Searching for strings	21
5. Editing operations for data matrices	22
5.1. Formatting	23
5.2. Sorting	23
5.3. Numerical operations	25
5.4. Loading and saving SURVO 76 files	31
6. Multiway tables	34
6.1. Definition of a multiway table	34
6.2. Table management and editing	35
6.3. Analysis of variance	39
6.4. Log-linear models for contingency tables	43
7. Correlations and regression analysis	46
REFERENCES	50

SURVO 76 EDITOR, a new tool  
for interactive statistical computing,  
text and data management

by

Seppo Mustonen  
Department of Statistics  
University of Helsinki

1. Introduction

SURVO 76 is an interactive statistical system intended to cover a wide range of activities in computational statistics. It has been designed especially for the needs of statisticians in both teaching and research work and its aims are slightly different from those of conventional statistical packages generally available for data analysis. In a certain sense the scope of SURVO 76 is wider permitting extended possibilities for data and text management, simulation, matrix computations and graphical analysis in interactive mode (see Mustonen 1977, 1980).

In the present form SURVO 76 has been implemented on the desk computer Wang 2200VP which provides suitable means for rapid interchange of information between the system and the user.

SURVO 76 is an interactive system and no special job describing language or code is needed. Using this system is like discussing with the computer; we speak about SURVO 76 conversations. The discussion is transmitted from the system to the user by a CRT display (speed is almost 5000 characters/sec.) and from the user to the system via a keyboard having also 32 programmable soft keys for control tasks. For a more precise and detailed output a line printer, a graphic CRT and a plotter are available.

SURVO 76 consists at the moment of about 60 statistical subsystems and programs and the total volume is almost 1 million bytes of program text. Formally the whole system is a single program written in the extended BASIC language of Wang.

In this paper we shall describe a new subsystem SURVO 76 EDITOR which is intended for various text and data editing activities in connection with statistical data analysis.

It is quite common that when writing a research report containing numerical tables the output from the computer cannot be used as such, but the results have to be retyped manually. This may happen even if the computer output is well designed, since the needs of the user may change during the reporting phase. In an interactive environment a good way of avoiding those editorial problems is to have text processing facilities in connection with the statistical system.

The main purpose of EDITOR is to lessen the burden of a statistician in data management and report writing. This editor can be used not only for standard text processing purposes, but also for various tasks encountered in statistical data processing like

- 1) input and editing unformatted data,
- 2) saving data in SURVO 76 files,
- 3) editing SURVO 76 files and results,
- 4) sorting and transforming data,
- 5) manipulating lists and multiway tables,
- 6) data analysis using techniques profitable in editing mode.

For instance, problems in analysis of variance and log-linear models for contingency tables can be easily handled in SURVO 76 EDITOR.

The whole editing process is controlled with the normal keyboard keys and programmable 'soft keys' (F-keys) which are used for simple text editing. For more complicated tasks several editing operations are available.

All the information is represented in an 'edit field' which consists, for example, of 100 columns and 250 rows. The field is always partially visible on the CRT. The editing operations are also typed in this field and they can be treated as normal text. Any operation can be activated by moving the cursor to the corresponding line and by pressing key CONTINUE. Whenever needed the contents of the edit field (tables, text and operations) can be saved in an edit file.

The edit field is like a notebook for the user, but it is much more flexible, since text and data in that notebook can be interpreted by editing operations and the results of these operations can be directed to any part of the field. Since the editing operations themselves are typed among the text and data the user can place them as he likes, usually near the object of operation.

If the user likes he can put the operations on adjacent lines and carry out them step by step as an editing program, but usually this is unnecessary. On the contrary, it is typical that during the editing process the edit field is filled by a mixture of text, data and operations, and the user scratches unessential ingredients when needed.

In the next chapters we try to illustrate with some practical examples of how to use EDITOR for typical data and text processing activities. In chapter 2 we give an introductory example related to analysis of contingency tables. Also the fundamentals of standard text editing are briefly described in this connection. Chapter 3 introduces the tasks of editing keys and chapters 4,5 the editing operations for normal text and data management. In chapters 6 and 7 some statistical operations are to be described.

2. An introductory example

When SURVO 76 is in use EDITOR will be called by name E like other SURVO 76 modules. At first a list of possible F-starts is displayed on the screen.

Disp.2.1

SURVO 76 EDITOR (C)1979 S.MUSTONEN, UNIVERSITY OF HELSINKI

F1: BASIC START  
F5: SAVE TEMPORARY FILE  
F6: LOAD TEMPORARY FILE  
F8: LOAD 'INDEX' FILE FROM THE DATA DISK

EDIT FIELD: 100 ROWS, 100 COLUMNS (MAX 262 x 100)

When starting with a new job usually the basic start F1 will be employed at first and then the upper left side corner of an empty edit field is displayed on the 24x80 screen:

Disp.2.2

1	SURVO 76 EDITOR	(C)1979 S.Mustonen	(100x100)
2	*		
3	*		
4	*		
5	*		
6	*		
7	*		
8	*		
9	*		
10	*		
11	*		
12	*		
13	*		
14	*		
15	*		
16	*		
17	*		
18	*		
19	*		
20	*		
21	*		
22	*		
23	*		

The cursor is blinking in the first position of the edit field and the user can start typing text, data values and editing operations according to his needs.

When the edit field is displayed the system can be operated like a normal typewriter and the text appears continuously on the screen.

Assume now that we have typed the following text: (This example has been taken from Payne 1977, p.106.)

Disp.2.3

```
1 SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)
2 *
3 *
4 *
5 * Example 1: Log-Linear models for Contingency tables
6 *
7 * The following data are taken from a survey of the
8 *political attitudes of a sample of British electors
9 *which is reported in Butler and Stokes (1974).
10 *We examine the the relationships among four variables:
11 *vote, sex, class and age._
12 *
13 *
14 *
15 *
16 *
17 *
18 *
19 *
20 *
21 *
22 *
23 *
```

Let us now consider some text editing facilities. On line 5 we have an obvious error in the word 'Contingency' where one 'n' is missing. To insert this letter we move the cursor from the present position on line 11 to the point of insertion by using F-keys with arrows indicating the direction, namely, F5:(upwards), F6:(downwards), F12:(to the right) and F13:(to the left).

We have then the following display (only essential lines are presented from now on):

Disp.2.4

```
5 * Example 1: Log-Linear models for Contingency tables
6 *
7 * The following data are taken from a survey of the
8 *political attitudes of a sample of British electors
9 *which is reported in Butler and Stokes (1974).
10 *We examine the the relationships among four variables:
11 *vote, sex, class and age.
12 *
```

and insertion takes place by pressing key F10:(insert) once:

Disp.2.5

```
5 * Example 1: Log-Linear models for Contingency tables
```

and typing the missing 'n':

Disp.2.6

```
5 * Example 1: Log-Linear models for Contingency tables
```

Correspondingly the extra 'the' on line 10 is removed by moving the cursor to that line at the second 'the':

Disp.2.7

```

5 * Example 1: Log-Linear models for Contingency tables
6 *
7 * The following data are taken from a survey of the
8 *political attitudes of a sample of British electors
9 *which is reported in Butler and Stokes (1974).
10 *We examine the the relationships among four variables:
11 *vote, sex, class and age.
12 *
```

and pressing key F9:(delete) four times:

Disp.2.8

```
10 *We examine the relationships among four variables:
```

For these correction and editing purposes EDITOR provides several means controlled by F-keys. A thorough description of those activities will be given in chapter 3.

Besides these 'soft keys' EDITOR includes many editing operations for more complicated tasks.

For instance, if we like to make the text wider in the edit field so that the present lines 7-11 should be longer up to 65 characters we achieve this situation by typing operation TRIM 7,11,65 on some empty line of the field (line 13 in this case):

Disp.2.9

```

5 * Example 1: Log-Linear models for Contingency tables
6 *
7 * The following data are taken from a survey of the
8 *political attitudes of a sample of British electors
9 *which is reported in Butler and Stokes (1974).
10 *We examine the relationships among four variables:
11 *vote, sex, class and age.
12 *
13 *TRIM 7,11,65_
14 *
```

and keeping now the cursor on line 13 and pressing key CONTINUE we obtain the following result

Disp.2.10

```

5 * Example 1: Log-Linear models for Contingency tables
6 *
7 * The following data are taken from a survey of the political
8 *attitudes of a sample of British electors which is reported in
9 *Butler and Stokes (1974). We examine the relationships among four
10 *variables: vote, sex, class and age.
11 *
12 *TRIM 7,11,65_
13 *
```

Since the new widened text requires one line less than the original one our TRIM operation locates now on line 12.

We may continue this trimming process and make the right edge of the text on lines 7-10 even by inserting extra blanks between some words. This is accomplished most easily by editing the TRIM operation on line 12 in the following form (only '2' has to be inserted):

Disp.2.11

```
12 *TRIM2_7,11,65
```

and activating this operation again with CONTINUE:

Disp.2.12

```
5 * Example 1: Log-Linear models for Contingency tables
6 *
7 * The following data are taken from a survey of the political
8 *attitudes of a sample of British electors which is reported in
9 *Butler and Stokes (1974). We examine the relationships among four
10 *variables: vote, sex, class and age.
11 *
12 *TRIM2_7,11,65
13 *
```

Observe that the editing operations are typed and edited like normal text, but each line containing text can be activated by moving the cursor to that line and by pressing CONTINUE. Then, if the line can be interpreted as a valid operation, this operation will be carried out, but if the line contains text which does not correspond to an editing operation, nothing will happen.

Activation can take place many times, since the operation text remains in the field until it is overwritten or removed by keys F25:(delete line), F8:(erase) or by editing operations like CLEAR and SCRATCH.

The position of the operation with respect to the object of the operation is immaterial in principle. In practice, however, it is typical to place the operations on lines close to the text and data to be handled. On the other hand it is profitable to collect some general operations like PRINT (printout of selected lines or chapters), SAVE (saving the edit field) and LOAD (loading some related fields) to the first lines of the field.

If we now like to save the current situation in the edit field on disk we can type the operation SAVE ELECTORS on the first line:

Disp.2.13

```
1 *SAVE ELECTORS_
2 *
3 *
4 *
5 * Example 1: Log-Linear models for Contingency tables
6 *
```

and by activating this operation the whole field will be saved in file ELECTORS on the user's data disk. The SAVE operation does not affect the contents of the field and we can continue the job from the present situation, but if we make a serious error like an unfortunate SCRATCH or CLEAR operation which partially or completely destroys our text, the situation before saving will be restored by typing and activating a LOAD ELECTORS operation.

The same procedure applies when we like to break the job for a while; we save the field and recall it later by a LOAD operation.

In order to print the lines 5-10 on paper we type the operation  
 PRINT 5,10 on some empty line:

Disp.2.14


---

1	SURVO 76 EDITOR	(C)1979 S.Mustonen	(100x100)
---	-----------------	--------------------	-----------

```

1 *SAVE ELECTORS
2 *
3 *PRINT 5,10_
4 *
5 * Example 1: Log-Linear models for Contingency tables
6 *
7 * The following data are taken from a survey of the political
8 *attitudes of a sample of British electors which is reported in
9 *Butler and Stokes (1974). We examine the relationships among four
10 *variables: vote, sex, class and age.
11 *
12 *TRIM2 7,11,65
13 *
```

and after activation obtain the following output on printer:

Example 1: Log-Linear models for Contingency tables

The following data are taken from a survey of the political attitudes of a sample of British electors which is reported in Butler and Stokes (1974). We examine the relationships among four variables: vote, sex, class and age.

In that output the line numbers and the control column filled with asterisks do not appear.

So far this has been only introduction to the normal text editing facilities of SURVO 76 EDITOR. Now we try to illustrate the possibilities for handling data sets and statistical operations.

We shall continue our example on contingency tables next by entering the four-way table to be analyzed:

Disp.2.15


---

1	SURVO 76 EDITOR	(C)1979 S.Mustonen	(100x100)
---	-----------------	--------------------	-----------

```

1 *SAVE ELECTORS
2 *
3 *PRINT 5,10_
4 *
5 * Example 1: Log-Linear models for Contingency tables
6 *
7 * The following data are taken from a survey of the political
8 *attitudes of a sample of British electors which is reported in
9 *Butler and Stokes (1974). We examine the relationships among four
10 *variables: vote, sex, class and age.
11 * The variables are defined and the observed frequencies are
12 *given in the four-way table below.
13 *
14 * OBSERVED FREQUENCIES FOR VOTE BY SEX BY CLASS BY AGE (N=1257)
15 *
16 *
17 *          Sex      Male      Female
18 *          Vote    Cons   Labour  Cons Labour
19 *Class   Age     *****
20 *
21 *upper   >73        4      0      10      0
22 *       51-73       27      8      26      9
23 *       41-50       27      4      25      9_
```

When entering the elements of the table we reach for the first time the last line of the screen, but we can continue typing normally,

since the field is scrolling automatically according to our needs and after typing the last lines of the table we have the display:

Disp.2.16

		SURVO 76 EDITOR		(C)1979 S.Mustonen		(100x100)	
16	*						
17	*	Sex	Male		Female		
18	*	Vote	Cons	Labour	Cons	Labour	
19	*Class	Age	*****				
20	*						
21	*upper	>73	4	0	10	0	
22	*	51-73	27	8	26	9	
23	*	41-50	27	4	25	9	
24	*	26-40	17	12	28	9	
25	*	<26	7	6	7	3	
26	*						
27	*lower	>73	8	4	9	2	
28	*	51-73	21	13	33	8	
29	*	41-50	27	12	29	4	
30	*	26-40	14	15	17	13	
31	*	<26	9	9	13	7	
32	*						
33	*work	>73	8	15	17	4	
34	*	51-73	35	62	52	53	
35	*	41-50	29	75	32	70	
36	*	26-40	32	66	36	67	
37	*	<26	14	34	18	33	

Observe that we can represent a multiway table in normal fashion. The structure of the table is defined by the classifiers Age, Class Vote and Sex which can be traced by the aid of the string \*\*\*\*\* on line 19. The number of asterisks points out the number of the maximal class name in use.

In order to manipulate this table with various editing and statistical operations we have to insert a TABLE specification to any empty line of the field (a natural place is in front of the table; here on line 16):

Disp.2.17

		SURVO 76 EDITOR		(C)1979 S.Mustonen		(100x100)	
16	*TABLE PAYNE,17,37,F_						
17	*	Sex	Male		Female		
18	*	Vote	Cons	Labour	Cons	Labour	
19	*Class	Age	*****				
20	*						
21	*upper	>73	4	0	10	0	
22	*	51-73	27	8	26	9	
23	*	41-50	27	4	25	9	
24	*	26-40	17	12	28	9	
25	*	<26	7	6	7	3	
26	*						
27	*lower	>73	8	4	9	2	
28	*	51-73	21	13	33	8	
29	*	41-50	27	12	29	4	
30	*	26-40	14	15	17	13	
31	*	<26	9	9	13	7	
32	*						
33	*work	>73	8	15	17	4	
34	*	51-73	35	62	52	53	
35	*	41-50	29	75	32	70	
36	*	26-40	32	66	36	67	
37	*	<26	14	34	18	33	

TABLE PAYNE,17,37,F on line 16 defines now a multiway table called PAYNE and locating on lines 17-37. The last parameter F in the TABLE

definition announces that the elements of this table should be interpreted as frequencies. (Another common alternative is a data table containing values of a certain variate according to a multiway classification; then we have X instead of F.)

EDITOR provides various means for modification of multiway tables. We can combine classes of any classifier, change places of classifiers in the table, form marginal tables by collapsing over some classifiers and compute with these modifications as well as with the original one. A set of TAB-operations enables a great variety of table representations and reductions.

Here we can, for instance, collapse over the Age variable by entering a TABD operation:

Disp.2.18

1	SURVO 76 EDITOR	(C)1979 S.Mustonen	(100x100)			
31	*	<26	9	9	13	7
32	*					
33	*work	>73	8	15	17	4
34	*	51-73	35	62	52	53
35	*	41-50	29	75	32	70
36	*	26-40	32	66	36	67
37	*	<26	14	34	18	33
38	*					
39	*TABD PAYNE,Age,40					
40	*					
41	*					
42	*					
43	*					
44	*					
45	*					

This operation has been typed on line 39 and the last parameter 40 indicates the first line for the result. This operation leads to a display:

Disp.2.19

1	SURVO 76 EDITOR	(C)1979 S.Mustonen	(100x100)			
31	*	<26	9	9	13	7
32	*					
33	*work	>73	8	15	17	4
34	*	51-73	35	62	52	53
35	*	41-50	29	75	32	70
36	*	26-40	32	66	36	67
37	*	<26	14	34	18	33
38	*					
39	*TABD PAYNE,Age,40					
40	*TABLE PAYNED,41,46,F					
41	*	Sex	Male	Female		
42	*	Vote	Cons	Labour	Cons	Labour
43	*Class	*****				
44	*upper		82	30	96	30
45	*lower		79	53	101	34
46	*work		118	252	155	227
47	*					

where we have the required three-way marginal table automatically labelled with an appropriate TABLE definition on line 40. The name of this new table is generated by adding the fourth letter from the TAB-operation used (here D) at the end of the original name PAYNE. Thus subsequent operations may refer to this new table by name PAYND.

For the analysis of contingency tables SURVO 76 EDITOR includes TABFIT operation which can be used for estimation of log-linear models. (For these models see e.g. Payne 1977, Bishop,Fienberg,Holland 1975.)

If we like to estimate a log-linear model for table PAYNED assuming that Sex, Class and Vote are independent we specify this model in the following way:

Disp.2.20

```

40 *TABLE PAYNED,41,46,F
41 *      Sex      Male      Female
42 *      Vote     Cons  Labour  Cons  Labour
43 *Class  *****
44 *upper          82    30    96    30
45 *lower          79    53   101    34
46 *work           118   252   155   227
47 *
48 *TABFIT PAYNED,49,50
49 *LOGLIN C,V,S
50 *

```

Here the TABFIT operation specifies the table to be analyzed (PAYNED), the line defining the model (49), and the line for the results (50). The independence of classifiers imply that the one-dimensional marginals for Class (C), Vote (V) and Sex (S) must be fixed in the estimation of the expected frequencies. This specification corresponds to the common  $\chi^2$ -test for independence and it is typed on the line 49 in the form LOGLIN C,V,S. Observe that the initials of each classifier should be used in this specification.

Activation of line 48 leads to the results:

Disp.2.21

```

40 *TABLE PAYNED,41,46,F
41 *      Sex      Male      Female
42 *      Vote     Cons  Labour  Cons  Labour
43 *Class  *****
44 *upper          82    30    96    30
45 *lower          79    53   101    34
46 *work           118   252   155   227
47 *
48 *TABFIT PAYNED,49,50
49 *LOGLIN C,V,S
50 *G^2= 165.04 DF= 7 P=0.000 FITTED: C,V,S (PAYNED)
51 *TABLE PAYNEDF,52,57,F
52 *      Sex      Male      Female
53 *      Vote     Cons  Labour  Cons  Labour
54 *Class  *****
55 *upper          58    58    61    61
56 *lower          65    65    69    68
57 *work           184   183   193   192
58 *

```

On line 50 we have the goodness-of-fit statistics for the model and on lines 51-57 the table of fitted frequencies. As expected this model is very poor and we have to continue by trying more complicated ones.

Here we make just one more effort and estimate a model where it is assumed that Class is independent of Sex given Vote. This implies fixing the two-dimensional marginals VS,VC.

To estimate this model we edit the lines 48 and 49 to the form

Disp.2.22

```

48 *TABFIT PAYNED,49,51
49 *LOGLIN VS,VC

```

indicating that the results are to appear from line 51 onwards.  
Activation of line 48 once again produces:

Disp.2.23

```

40 *TABLE PAYNED,41,46,F
41 *      Sex      Male      Female
42 *      Vote     Cons   Labour Cons   Labour
43 *Class *****
44 *upper          82     30     96     30
45 *lower          79     53    101     34
46 *work          118    252    155    227
47 *
48 *TABFIT PAYNED,49,51
49 *LOGLIN VS,VC
50 *G↓2= 165.04 DF= 7 P=0.000 FITTED: C,V,S (PAYNED)
51 *G↓2= 2.76 DF= 4 P=0.602 FITTED: VS,VC (PAYNED)
52 *TABLE PAYNEDF,53,58,F
53 *      Sex      Male      Female
54 *      Vote     Cons   Labour Cons   Labour
55 *Class *****
56 *upper          79     32     99     28
57 *lower          80     47    100     40
58 *work          121    256    152    223
59 *

```

The previous results on lines 51-57 are overwritten by the new ones.  
The fit described on line 51 is now good.

If we want to have the fitted frequencies more accurately this is achieved simply by editing at least one element in table PAYNED to a form corresponding the accuracy desired and by activating TABFIT again.

Thus by editing line 58 temporarily into the form

```
46 *work          118    252    155    227.00
```

and by repeating TABFIT we get the results in the form:

Disp.2.24

```

40 *TABLE PAYNED,41,46,F
41 *      Sex      Male      Female
42 *      Vote     Cons   Labour Cons   Labour
43 *Class *****
44 *upper          82     30     96     30
45 *lower          79     53    101     34
46 *work          118    252    155    227.00
47 *
48 *TABFIT PAYNED,49,51_
49 *LOGLIN VS,VC
50 *G↓2= 165.04 DF= 7 P=0.000 FITTED: C,V,S (PAYNED)
51 *G↓2= 2.76 DF= 4 P=0.602 FITTED: VS,VC (PAYNED)
52 *TABLE PAYNEDF,53,58,F
53 *      Sex      Male      Female
54 *      Vote     Cons   Labour Cons   Labour
55 *Class *****
56 *upper          78.70  32.11  99.30  27.89
57 *lower          79.59  46.56  100.41  40.44
58 *work          120.71 256.33 152.29 222.67
59 *

```

Thereafter the residuals can be formed simply by a TAB- operation:

```
60 *TAB- PAYNED,PAYNEDF,61_
```

and the result is:

Disp.2.25

```

59   *
60   *TAB- PAYNED,PAYNEDF,61_
61   *TABLE PAYNEDF-,62,67,F
62   *          Sex      Male      Female
63   *          Vote     Cons    Labour  Cons   Labour
64   *Class    *****
65   *upper        3.30  -2.11  -3.30   2.11
66   *lower        -0.59   6.44   0.59  -6.44
67   *work         -2.71  -4.33   2.71   4.33
68   *

```

These 'raw' residuals could be normalized (for instance by dividing them by the square roots of the expected frequencies) in a few editing steps, but we stop our analysis here.

At this stage the lines 1-67 of the edit field are filled by a mixture of text, data, results and editing operations. We may save the present situation on disk by returning to the first line of the field. This is achieved by pressing key F22:(previous page) a couple of times and by activating the old SAVE ELECTORS operation on line 1:

Disp.2.26

```

1   SURVO 76 EDITOR   (C)1979 S.Mustonen   (100x100)
1   *SAVE ELECTORS   -
2   *
3   *PRINT 5,10
4   *
5   * Example 1: Log-Linear models for Contingency tables
6   *
7   * The following data are taken from a survey of the political
8   *attitudes of a sample of British electors which is reported in
9   *Butler and Stokes (1974). We examine the relationships among four
10  *variables: vote, sex, class and age.
11  * The variables are defined and the observed frequencies are
12  *given in the four-way table below.
13  *
14  * OBSERVED FREQUENCIES FOR VOTE BY SEX BY CLASS BY AGE (N=1257)
15  *
16  *TABLE PAYNE,17,37,F
17  *          Sex      Male      Female
18  *          Vote     Cons    Labour  Cons   Labour
19  *Class    Age    *****
20  *
21  *upper   >73        4      0     10     0
22  *       51-73       27     8     26     9
23  *       41-50       27     4     25     9

```

To take a new printout of the essential lines, the PRINT operation on line 3 should be edited to form PRINT 5,67 and activated.

In order to make this output clean it can be purged from editing operations and technical lines with F25:(delete line) and F8:(erase line) and all the relevant comments may be inserted among the text. When more lines are needed for such insertions key F25:(insert line) is to be employed.

In many cases it is worth while to preserve the editing operations during the analysis stage and not to cancel them. If the editing operations are left in the edit field it is usually easy to work out the same problem with another data set just by changing the data lines and by activating the old operation lines.

In teaching situations it may be valuable to demonstrate the effects of various changes in the data set to the results by repeating the steps of analysis with data edited in various ways.

### 3. Use of edit keys

When EDITOR has been called from SURVO 76 and the basic start F1 is taken the F-keys operate as follows:

- F0: List of F-starts (as in other SURVO 76 modules)
- F1: extra character 1, default is 'ä'
- F2: extra character 2, default is 'ö'
- F3: UNDERLINING ON/OFF: The next characters to be typed will be printed (printer 2231-W3) underlined. After pressing F3 again the next characters will be printed without underlining.
- F4: (END) moves the cursor to the right end of the CRT.
- F5: (arrow down) moves the cursor one line downwards.
- F6: (arrow up) moves the cursor one line upwards.
- F7: (BEGIN) moves the cursor to the left of end of the CRT.
- F8: (ERASE) erases the line to the right from the cursor.
- F9: (DELETE) deletes one character.
- F10: (INSERT) inserts space for one character.
- F11: (--->) moves the cursor 5 steps to the right
- F12: (->) moves the cursor one step to the right.
- F13: (<-) moves the cursor one step to the left.
- F14: (<--->) moves the cursor 5 steps to the left.  
If the edge of the CRT is surpassed in these moves the visible part of the edit field will be moved correspondingly.
- F15: copies one line to the current line starting from the current position of the cursor. After pressing F15 the question LINE (NO.) TO BE COPIED? will first be displayed.
- F16: inserts missing words typed as the next line to a place pointed out by the cursor. Observe that a normal procedure for minor insertions is to use F26 (INSERT) repeatedly.
- F17: extra character 3, default is 'å'
- F18: extra character 4, default is 'ü'
- F19: types characters which are missing on the keyboard. After F19 the question <--HEXCODE? will be displayed and the user has to enter the hexcode of the character. The hexcodes are listed in appendix F of the '2200 VP Basic-2 Language Reference Manual'.
- F20: (END) moves the cursor to the bottom line of the CRT
- F21: (arrow down) displays the next page on the CRT
- F22: (arrow up) displays the previous page on the CRT
- F23: (BEGIN) moves the cursor to the first line of the CRT
- F24: (extra character) operates as F19 when used for the first time. Thereafter F24 types directly the selected character. To alter the extra character exit and select EDITOR again.
- F25: (DELETE) deletes the current line,
- F26: (INSERT) inserts a new empty line after the current line,
- F27: (--->) displays the right side of the edit field,
- F28: (->) displays the right side of the edit field,
- F29: (<-) moves the cursor to the start of the current line,
- F30: (<--->) moves the cursor to the start of the current line,

F31: moves the cursor to the control column 0 indicated normally by an asterisk \*. Exit from the control column takes place by using the arrow keys F11,F12. The characters in the control column have the following tasks:

char. task

- in PRINT operation the line will be printed with elongated characters,

s the next line feed (in printer 2231W-3) is 1/3 of a normal line feed. This feature can be used for typing exponents, indices etc.

76 \* For example

77 % Variables  $x_1^{(i)}, x_2^{(i)}, \dots, x_n^{(i)}$

80 \* (next line)

q as s, but only a 1/12 of a normal line feed,

r as s, but only a 1/6 of a normal line feed,

/ starts a new page when the extended form of PRINT operation is used.

4. Text editing

In this chapter we shall consider basic operations needed in text and data management.

4.1 Clearing the edit field

Clearing of single characters and lines takes place by the keys:

F9: delete one character,

F25: delete one line,

F8: erase line to the right of the cursor

If the user likes to clear the field partially or completely during the work he can employ SCRATCH, CLEAR, ERASE and DELETE operations.

SCRATCH

(without any parameters) clears the edit field from the SCRATCH line onwards.

CLEAR L1,L2

where L1 and L2 are line numbers and L1<=L2 clears the lines L1-L2.

CLEAR xy

clears a rectangular subfield indicated by the character x in the left upper corner and y in the right lower corner. The characters used for field indication (xy) must be uniquely determined in the edit field.

For instance, if we activate CLEAR \*+ in the following situation:

Disp.4.1

1 SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)					
38	*				
39	*				
40	*TABLE PAYNED,41,46,F				
41	*	Sex	Male	Female	
42	*	Vote	Cons	Labour	Cons
43	*Class	*****			
44	*upper		82	30	96
45	*lower		79	53	101
46	*work		118	252	155
47	*				227 +
48	*CLEAR *+				
49	*				

nothing happens, since \* is ambiguous, but insertion of \$ after asterisks on line 43

Disp.4.2

1 SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)					
38	*				
39	*				
40	*TABLE PAYNED,41,46,F				
41	*	Sex	Male	Female	
42	*	Vote	Cons	Labour	Cons
43	*Class	***** \$			
44	*upper		82	30	96
45	*lower		79	53	101
46	*work		118	252	155
47	*				227 +
48	*CLEAR \$+*				
49	*				

and activation of CLEAR \$+ on line 48 leads to  
Disp.4.3

```

1 SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)
38 *
39 *
40 *TABLE PAYNED,41,46,F
41 *      Sex     Male      Female
42 *      Vote    Cons   Labour Cons   Labour
43 *Class  *****
44 *upper
45 *lower
46 *work
47 *
48 *CLEAR $+_
49 *
```

A third form of CLEAR operation is CLEAR L1,L2,K which clears the lines L1-L2 for those columns which are indicated by non-blank characters in on image line K. Consider the situation:

Disp.4.4

```

1 SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)
40 *TABLE PAYNED,41,46,F
41 *      Sex     Male      Female
42 *      Vote    Cons   Labour Cons   Labour
43 *Class  *****
44 *upper          82     30     96     30
45 *lower          79     53     101    34
46 *work           118    252    155    227
47 *                  XXXXX      XXXXX
48 *CLEAR 42,46,47_
49 *
50 *
```

Execution of CLEAR 42,46,47 gives:

Disp.4.5

```

1 SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)
40 *TABLE PAYNED,41,46,F
41 *      Sex     Male      Female
42 *      Vote    Cons      Cons
43 *Class  *****
44 *upper          82     96
45 *lower          79     101
46 *work           118    155
47 *                  XXXXX      XXXXX
48 *CLEAR 42,46,47_
49 *
50 *
```

In EDITOR the columns to be processed are never referred to by column indices, but image lines are used to point out the columns involved.

The ERASE operation is to be used for erasing selected characters in the field.

ERASE <string>

erases all characters occurring in <string> from the edit field.

For instance, to erase all numbers execute ERASE 1234567890.

The DELETE operation is a generalization of key F9:(delete).

DELETE L1,L2

deletes the column indicated by the cursor from the lines L1-L2 simultaneously. Thus the current column will be lost and the

columns to the right from the cursor are moved one step to the left.

#### 4.2. Moving parts of the edit field

For various insertions in the edit field we can use following operations.

INSERT L1,L2

inserts one empty column to the current position of the cursor for the lines L1-L2 simultaneously. Thus the columns to the right of the cursor are moved one step to the right.

Thus INSERT is a generalization of key F10:(insert). For example, by activating INSERT 41,46 on line 48 five times

Disp.4.6

```

40 *TABLE PAYNED,41,46,F
41 *      Sex      Male          Female
42 *      Vote     Cons   Labour  Cons   Labour
43 *Class  *****
44 *upper           82    30    96    30
45 *lower           79    53   101    34
46 *work            118   252   155   227
47 *
48 *INSERT 41,46
49 *

```

we shall have:

Disp.4.7

```

40 *TABLE PAYNED,41,46,F
41 *      Sex      Male          Female
42 *      Vote     Cons   Labour  Cons   Labour
43 *Class  *****
44 *upper           82    30    96    30
45 *lower           79    53   101    34
46 *work            118   252   155   227
47 *
48 *INSERT 41,46
49 *

```

COPY L1,L2,L

moves lines L1-L2 to a new place starting from line L. The previous contents of lines L,L+1,...,L+L2-L1+1 will be destroyed. Observe that for moving single lines it is simplest to use key F15.

The MOVE operation is a generalization of COPY permitting various rectangular subfields to be copied in the edit field. MOVE has two forms.

MOVE xyz

moves a subfield indicated by character x in the left upper corner and character y in the right lower corner to a place indicated by character z so that x will be in the position of z. The indicators x,y,z must be unambiguous; otherwise the the MOVE xyz operation is not carried out.

MOVE L1,L2,K,L

moves a subfield from the lines L1-L2 to lines starting from L. The columns which are to moved and to which columns are specified by an image line K consisting of at least two X's and one Y. The columns to be moved are determined by the first and last X and the first column of the subfield in the new place by the first Y.

In the next example both MOVE operations on lines 48,49 will do the same task

Disp.4.8

```

40 *TABLE PAYNED,41,46,F
41 *      Sex <Male          Female      ↑
42 *      Vote Cons Labour Cons Labour
43 *Class *****
44 *upper           82   30   96   30
45 *lower           79   53   101  34
46 *work            118  252> 155  227
47 *                  X       X           Y
48 *MOVE <>↑
49 *MOVE 41,46,47,41

```

and the result is

Disp.4.9

```

41 *      Sex <Male          Female      <Male
42 *      Vote Cons Labour Cons Labour Cons Labour
43 *Class *****
44 *upper           82   30   96   30   82   30
45 *lower           79   53   101  34   79   53
46 *work            118  252> 155  227  118  252>
47 *                  X       X           Y
48 *MOVE <>↑
49 *MOVE 41,46,47,41

```

Note that after using MOVE <>↑ it is easy to cancel the auxiliary characters <>↑ from the edit field by an ERASE <>↑ operation.

4.3. Saving, loading and combining edit fields

The user can save edit fields on data disks and load them back for subsequent operations. Also parts of various edit files can be joined together in the present edit field. Observe, however, that in order to build up more comprehensive documents it is not necessary to reorganize the material in a right order before printout. The extended forms of PRINT operation enable the output to consist of any number of chapters located in various files in any order (see 4.4).

The present contents of the edit field is saved temporarily on the program disk by pressing F0 and thereafter F5. This temporary file is loaded into the edit field pressing F0 and thereafter F6.

During a long job it is recommended to employ this temporary saving procedure especially before activating operations which may harmfully spoil the results achieved.

Saving on user's data disk takes place by a SAVE operation.

SAVE XYZ

saves the the field in file XYZ. If the file is not in the catalog a new file XYZ will be automatically created and space will be reserved according the dimensions of the edit field. If the file XYZ is already in the catalog this file is used for saving.

LOAD XYZ

loads the edit file XYZ in the edit field and the previous contents of the field will be destroyed. The field will be also automatically redimensioned according to the file to be loaded.

During the job the user can alter the dimensions of the edit field with a REDIM operation. For instance, after execution of REDIM 124,80 the number of lines is 124 and the number of columns 80. The REDIM operation does not alter the contents of the edit field, unless the new dimensions are too small. Then the excessive lines and columns will disappear.

The JOIN operation can be used for connecting chapters and subfields from edit files to the present field. JOIN has two forms.

JOIN L1,L2,N,L

moves the lines L1-L2 from edit file N to the present edit field so that the first line will be on line L in the present edit field. The previous contents of lines L,L+1,...,L+L2-L1+i will be destroyed.

JOIN N,C1,C2,...,L

moves the subfields (chapters) C1,C2,... from edit file N to the present edit field so that the first line of the first subfield will be L. The chapters C1,C2,... have to be defined in file N by a DEF definition.

DEF <name of subfield>,L1,L2

defines a subfield consisting of lines L1-L2 in the present edit field. After saving the edit field in an edit file this subfield can be referred to in JOIN and PRINT operations.

To keep record of various edit files on disk the user can create an INDEX file which is like any other edit file, but INDEX can be loaded also simply by F8-start after EDITOR has been called or F0 has been pressed. In the INDEX file the user may have a list of various files on disk e.g. in the following form:

Disp.4.10

1	SURVO 76 EDITOR	(C)1979 S.Mustonen	(40x60)
1	*SAVE INDEX		
2	*		
3	*Index for disk 189/SM		
4	*		
5	*LOAD DATA1		
6	*LOAD DATA2		
7	*LOAD DATA3		
8	* data sets...		
9	*LOAD LETTER1		
10	*LOAD LETTER2		
11	* letters...		
12	*LOAD EX1		
13	*LOAD EX2		
14	* examples...		
15	*		

Thus any edit file may be loaded by moving the cursor to the corresponding LOAD line and by keying CONTINUE.

#### 4.4. Printout of the edit field and edit files

The PRINT operation has three alternative forms.

PRINT L1,L2

prints the lines L1-L2 on paper without line numbers and the control column as seen in Disp.1.14. The control column can be used for elongating the text on certain lines (- in control column) or for creating non-standard line feeds for indices etc. (see F31 in chapter 3).

PRINT L1,L2,<separator character>

works as PRINT above, but starts a new line whenever the selected separator character is encountered.

---

```

PRINT R1,R2,<1st page>,<left edge>,<page length>,<extra lines>
  <file 1>,<chapter 1>
  <file 2>,<chapter 2>
  -----
  END
prints a document consisting of several chapters in several edit
files so that the text will be automatically divided into pages
and the pages will be started by one or several header lines
with appropriate page numbers.

```

When the general PRINT operation is activated, the chapters to be printed need not to be in the current edit field.

The header lines for the pages are to be described on edit lines R1-R2 and the line R1 must contain the image of the page number in the form ###.

The pages will be numbered starting from the value <1st page>. In front of each line, number of columns pointed out by the parameter <left edge> will be left empty. The parameter <page length> specifies the maximum number of lines in one page (including the header lines). The parameter <extra lines> specifies the number of empty lines to be left between the pages.

The chapters to be printed must be defined by DEF's in their own edit files.

The page division can also be controlled by inserting / to the control column of those lines which are starting a new page. Each page will be filled up to the last line, unless otherwise stated by a / in the control column.

The printout can be checked in advance by using this PRINT operation with - in the place of <left edge>. The text will be displayed on the CRT and the output will stop when the page should be changed. After a stop of this kind the user can move the cursor in the control column by the keys F6 (upwards) and F5 (downwards) and set a new page change by pressing /. If the user is satisfied with the present suggestion, RETURN(EXEC) is pressed and the next possible point of page change will be displayed.

This checking can be repeated indefinitely, and the true output will take place by changing <left edge> to its final value and by activating PRINT once again.

For example this paper has been printed with a single PRINT operation of the form:

#### Disp.4.11

```

14  *S.Mustonen: SURVO 76 EDITOR      10.8.1980      ##
15  *-----
16  *
17  *_PRINT 14,16,1,-,66,6
18  *_EDIO,A
19  *_EDI1,A
20  *_EDI2,A
21  *_EDI3,A
22  *_EDI4,A
23  *_EDI5,A
24  *_EDI6,A
25  *END

```

#### 4.5. Making up and adjusting the text lines

Three TRIM operations are available for changing and adjusting the line length of the text in the edit field.

##### TRIM L1,L2,C

makes up the lines L1-L2 using the line width C without splitting words, numbers and other contiguous character strings.

If new lines are needed they will be automatically inserted and,

on the other hand, unnecessary lines will be deleted.

After TRIM only one space will be left between the 'words' and the right edge will not necessarily be even. To make it straight use TRIM2 immediately after TRIM.

The lines starting with at least one space will be preserved as new lines.

TRIM2 L1,L2,C

makes the right edge of the lines L1-L2 straight in the column position C without splitting or connecting the lines by inserting spaces evenly between the 'words'.

If the line is short for more than 5 characters nothing will be changed on that line.

TRIM3 L1,L2,C

operates as TRIM, but splits the words when needed according to the rules of Finnish. TRIM3 operates rather satisfactorily also in other languages. Incorrect splittings are easy to modify afterwards by using standard editing keys.

An example of TRIM operations is presented already in chapter 2 (Displays 2.9-12).

#### 4.6. Searching for strings and replacing them

FIND <string>

finds the first occurrence of <string> in the edit field.

The cursor will then point the first character of <string>.

By pressing CONTINUE immediately again the next occurrence will be found etc.

FIND L1,L2,K

works as FIND above but scans only lines L1-L2. The string to be found is the contents of image line K.

REPLACE L1,L2,K1,K2

searches for the string described on line K1 from lines L1-L2.

If CONTINUE is pressed thereafter the string K1 will be replaced by the string on line K2 and the next occurrence of K1 will be searched for. If CLEAR is pressed K1 is not replaced, but the next K1 will be searched for.

### 5. Editing operations for data matrices

The SURVO 76 EDITOR can be used in many ways for data input and management. Edited data sets may be saved in SURVO 76 data files. On the other side, SURVO 76 data files and various special files, like correlation files and MATRI files can be loaded in the edit field and modified there for subsequent operations in SURVO 76 or for final printout with text and comments.

For illustration, we consider the following small data set of 12 observations:

Yearly consumption of certain beverages per inhabitant  
in some European countries in 1972-1976  
(Source: Statistical Abstract of Sweden 1979)

	Coffee (kg)	Tea (kg)	Beer (l)	Wine (l)	Spirits (l)
Sweden	12.9	0.30	58.3	7.9	2.9
Denmark	11.8	0.41	113.9	10.4	1.7
Finland	12.5	0.15	54.7	7.6	2.7
Norway	9.4	0.19	43.5	3.1	1.8
France	5.2	0.10	44.5	104.3	2.5
Ireland	0.2	3.73	124.5	3.8	1.9
Italy	3.6	0.06	13.6	106.6	2.0
Holland	9.2	0.58	75.5	9.7	2.7
Portugal	2.2	0.03	27.5	89.3	0.9
Switzerland	9.1	0.25	73.5	44.9	2.1
Spain	2.5	0.03	43.6	73.2	2.7
England	1.8	3.49	113.7	5.1	1.4

This data set can be typed in the edit field as it is, but if the user likes save time and work it is not necessary to do that exactly in straight columns, since this can be achieved easily afterwards.

Thus after input of data we may have the following display:

#### Disp.5.1

```

1 SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)
8 *      Coffee Tea Beer Wine Spirits
9 *      (kg) (kg) (l) (l) (l)
10 *
11 *Sweden 12.9 0.3 58.3 7.9 2.9
12 *Denmark 11.8 0.41 113.9 10.4 1.7
13 *Finland 12.5 0.15 54.7 7.6 2.7
14 *Norway 9.4 0.19 43.5 3.1 1.8
15 *France 5.2 0.10 44.5 104.3 2.5
16 *Ireland 0.2 3.73 124.5 3.8 1.9
17 *Italy 3.6 0.06 13.6 106.6 2
18 *Holland 9.2 0.58 75.5 9.7 2.7
19 *Portugal 2.2 0.03 27.5 89.3 0.9
20 *Switzerland 9.1 0.25 73.5 44.9 2.1
21 *Spain 2.5 0.03 43.6 73.2 2.7
22 *England 1.8 3.49 113.7 5.1 1.4_
23 *
24 *
25 *

```

### 5.1. Formatting

In order to obtain a neat tabular form for this data matrix we use a FORM operation:

#### Disp.5.2

```

20  *Switzerland 9.1 0.25 73.5 44.9 2.1
21  *Spain 2.5 0.03 43.6 73.2 2.7
22  *England 1.8 3.49 113.7 5.1 1.4
23  *
24  * XXXXXXXXXXXX 12.1 1.12 123.1 123.1 1.1
25  *FORM 11,22,24_

```

FORM operates on lines 11-22 and the new format of these lines is specified on image line 24. The numeric fields are defined by numbers in the desired format and the alpha fields by letters. In many cases one of the operand rows edited to the final form can be used as an image line. Now activation of line 25 leads to the result:

#### Disp.5.3

	SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)						
8	*	Coffee	Tea	Beer	Wine	Spirits	
9	*	(kg)	(kg)	(l)	(l)	(l)	
10	*						
11	*	Sweden	12.9	0.30	58.3	7.9	2.9
12	*	Denmark	11.8	0.41	113.9	10.4	1.7
13	*	Finland	12.5	0.15	54.7	7.6	2.7
14	*	Norway	9.4	0.19	43.5	3.1	1.8
15	*	France	5.2	0.10	44.5	104.3	2.5
16	*	Ireland	0.2	3.73	124.5	3.8	1.9
17	*	Italy	3.6	0.06	13.6	106.6	2.0
18	*	Holland	9.2	0.58	75.5	9.7	2.7
19	*	Portugal	2.2	0.03	27.5	89.3	0.9
20	*	Switzerland	9.1	0.25	73.5	44.9	2.1
21	*	Spain	2.5	0.03	43.6	73.2	2.7
22	*	England	1.8	3.49	113.7	5.1	1.4
23	*						
24	*	XXXXXXXXXX	12.1	1.12	123.1	123.1	1.1
25	*	FORM	11,22,24_				

### 5.2. Sorting

Formatted data matrices can be sorted in numeric or alphanumeric order using any column as a sort key.

SORT L1,L2,K

sorts the lines L1-L2 in the order determined by image line K.  
If K has the form "XXXXXX", the lines will be  
sorted in alphanumeric order using the XXXXXX column as a sort  
key.

If K has the form " 111111 ", the lines will be  
sorted in ascending numeric order using 111111 column as a sort  
key.

-SORT L1,L2,K

sorts the lines L1-L2 in descending order in case of numeric  
sorting.

For example, to sort the countries in alphanumeric order we type the SORT operation:

Disp.5.4

```

8   *          Coffee Tea  Beer  Wine Spirits
9   *          (kg)   (kg)  (l)   (l)   (l)
10  *
11  * Sweden    12.9 0.30 58.3   7.9 2.9
12  * Denmark   11.8 0.41 113.9  10.4 1.7
13  * Finland   12.5 0.15 54.7   7.6 2.7
14  * Norway    9.4 0.19 43.5   3.1 1.8
15  * France    5.2 0.10 44.5  104.3 2.5
16  * Ireland   0.2 3.73 124.5  3.8 1.9
17  * Italy     3.6 0.06 13.6  106.6 2.0
18  * Holland   9.2 0.58 75.5   9.7 2.7
19  * Portugal  2.2 0.03 27.5  89.3 0.9
20  * Switzerland 9.1 0.25 73.5  44.9 2.1
21  * Spain     2.5 0.03 43.6  73.2 2.7
22  * England   1.8 3.49 113.7  5.1 1.4
23  *
24  * XXXXXXXXXX
25  *SORT 11,22,24_

```

and after the execution the display is:

Disp.5.5

```

8   *          Coffee Tea  Beer  Wine Spirits
9   *          (kg)   (kg)  (l)   (l)   (l)
10  *
11  * Denmark   11.8 0.41 113.9  10.4 1.7
12  * England   1.8 3.49 113.7  5.1 1.4
13  * Finland   12.5 0.15 54.7   7.6 2.7
14  * France    5.2 0.10 44.5  104.3 2.5
15  * Holland   9.2 0.58 75.5   9.7 2.7
16  * Ireland   0.2 3.73 124.5  3.8 1.9
17  * Italy     3.6 0.06 13.6  106.6 2.0
18  * Norway    9.4 0.19 43.5   3.1 1.8
19  * Portugal  2.2 0.03 27.5  89.3 0.9
20  * Spain     2.5 0.03 43.6  73.2 2.7
21  * Sweden    12.9 0.30 58.3   7.9 2.9
22  * Switzerland 9.1 0.25 73.5  44.9 2.1
23  *
24  * XXXXXXXXXX
25  *SORT 11,22,24_

```

Similarly, to obtain the order according to wine consumption we edit the lines 24,25 to form

```

24  *          11111
25  *-SORT 11,22,24_

```

and by activating the line 25 again we obtain the result:

Disp.5.6

		Coffee	Tea	Beer	Wine	Spirits
	*	(kg)	(kg)	(l)	(l)	(l)
8	*					
9	*					
10	*					
11	* Italy	3.6	0.06	13.6	106.6	2.0
12	* France	5.2	0.10	44.5	104.3	2.5
13	* Portugal	2.2	0.03	27.5	89.3	0.9
14	* Spain	2.5	0.03	43.6	73.2	2.7
15	* Switzerland	9.1	0.25	73.5	44.9	2.1
16	* Denmark	11.8	0.41	113.9	10.4	1.7
17	* Holland	9.2	0.58	75.5	9.7	2.7
18	* Sweden	12.9	0.30	58.3	7.9	2.9
19	* Finland	12.5	0.15	54.7	7.6	2.7
20	* England	1.8	3.49	113.7	5.1	1.4
21	* Ireland	0.2	3.73	124.5	3.8	1.9
22	* Norway	9.4	0.19	43.5	3.1	1.8
23	*					
24	*			11111		
25	*-SORT 11,22,24_					

The columns of tables can also be rearranged using the PERM operation.

PERM L1,L2,L

permutes the fields (columns) on lines L1-L2 according to the fields on the lines L,L+1,L+2,...

The numbers of fields on line L+i and L+i must be the same and the fields on line L+i will be permuted in a new order corresponding to the alphabetic order of the L+i fields.

If some of the lines L,L+1,... is empty the remaining lines will be permuted according to the last non-empty line. For example, when changing the order of columns in a table the line L should point out the new order and the line L+1 should be empty.

5.3. Numerical operations

Data matrices and other numeric tables can be transformed and edited with several numeric operations. At first auxiliary operations SET and COUNT are introduced.

To create constant columns (numeric or alpha) the SET operation is to be employed

SET L1,L2,K

inserts the non-blank characters of image line K to corresponding columns on lines L1-L2.

For creating line numbers or other index columns with a constant increment or cyclic behaviour the COUNT operation is available. It has three forms:

COUNT L1,L2,K,C

inserts numbers C,C+1,C+2,... on lines L1-L2 in the format indicated by an image line K of the form " -123.1 ".

COUNT L1,L2,K,C,D

works similarly, but inserts the numbers C,C+D,C+2D,...

COUNT L1,L2,K,C,D,E

with E>0 inserts numbers

C,C+D,C+2D,...,C+(E-1)D,C,C+D,C+2D,...,C+(E-1)D,C,C+D,C+2D,...  
on lines L1-L2 in the format indicated by image line K thus generating a cyclical count.

If E<0, the numbers inserted will be

C,C,...,C,C+D,C+D,...,C+D,C+2D,C+2D,...,C+2D,....

where each of values C,C+D,C+2D is repeated -E times.

For instance, annotations for a monthly time series in the form

```
year:month
1980: 1
1980: 2
1980: 3
-----
-----
1980:11
1980:12
1981: 1
1981: 2
etc.
```

are easily generated by COUNT and SET operations. Assume that the edit field is empty. We start by entering the years with COUNT.

#### Disp.5.7

	1 SURVO 76 EDITOR	(C)1979 S.Mustonen	(100x100)
1	*		
2	*		
3	*		
4	*COUNT 7,23,5,1980,1,-12_		
5	* 1234		
6	*		
7	* 1980		
8	* 1980		
9	* 1980		
10	* 1980		
11	* 1980		
12	* 1980		
13	* 1980		
14	* 1980		
15	* 1980		
16	* 1980		
17	* 1980		
18	* 1980		
19	* 1981		
20	* 1981		
21	* 1981		
22	* 1981		
23	* 1981		

Thereafter

```
4 *COUNT 7,23,5,1,1,12_
5 * 12
```

creates the month indices:

Disp.5.8

```
1  SURVO 76 EDITOR  (C)1979 S.Mustonen (100x100)
1  *
2  *
3  *
4  *COUNT 7,23,5,1,1,12_
5  *      12
6  *
7  * 1980 1
8  * 1980 2
9  * 1980 3
10 * 1980 4
11 * 1980 5
12 * 1980 6
13 * 1980 7
14 * 1980 8
15 * 1980 9
16 * 1980 10
17 * 1980 11
18 * 1980 12
19 * 1981 1
20 * 1981 2
21 * 1981 3
22 * 1981 4
23 * 1981 5
```

Finally, to place the missing ':' between year and month we perform a SET operation:

```
4  *SET 7,23,5_
5  *      :
```

The operations C+,C-,C\* and C/ are to be used for making new columns by addition, subtraction, multiplication and division, respectively.

C+ L1,L2,K

where image line K=" XXXX XXXX XXXX -123.12 "  
computes the sum of XXXX-columns to the column indicated  
by -123.12 for each of the lines L1-L2.

C\* L1,L2,K

works as C+, but the product of XXXX-columns is computed.

C- L1,L2,K

where K=" XXXX YYY -123.12 "  
subtracts the YYY-column from the XXXX-column.

C/ L1,L2,K

divides the XXXX-column by the YYY-column.

For example, let us create a column for the total alcohol consumption in the following display:

Disp.5.9

```

1 SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)
1 *
2 *
3 *
4 * Alcoholic beverages consumed in Finland 1969-1978
5 * (litres per inhabitant, 100 % alcohol)
6 *
7 *      strong   wines   long    beer
8 *          liquors      drinks
9 * 1969     1.58     0.48    0.04   2.11
10 * 1970    1.75     0.53    0.04   1.98
11 * 1971    2.07     0.56    0.04   2.05
12 * 1972    2.19     0.57    0.06   2.28
13 * 1973    2.46     0.69    0.10   2.35
14 * 1974    2.90     0.89    0.18   2.48
15 * 1975    2.81     0.77    0.20   2.41
16 * 1976    2.95     0.72    0.20   2.44
17 * 1977    2.99     0.68    0.22   2.49
18 * 1978    2.82     0.72    0.20   2.48

```

By using the C+ operation

```

2 *C+ 9,18,3_
3 *      XXXX   XXXX   XXXX   XXXX   1.12

```

we obtain:

Disp.5.10

```

1 SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)
1 *
2 *C+ 9,18,3_
3 *      XXXX   XXXX   XXXX   XXXX   1.12
4 * Alcoholic beverages consumed in Finland 1969-1978
5 * (litres per inhabitant, 100 % alcohol)
6 *
7 *      strong   wines   long    beer
8 *          liquors      drinks
9 * 1969     1.58     0.48    0.04   2.11   4.21
10 * 1970    1.75     0.53    0.04   1.98   4.30
11 * 1971    2.07     0.56    0.04   2.05   4.72
12 * 1972    2.19     0.57    0.06   2.28   5.10
13 * 1973    2.46     0.69    0.10   2.35   5.60
14 * 1974    2.90     0.89    0.18   2.48   6.45
15 * 1975    2.81     0.77    0.20   2.41   6.19
16 * 1976    2.95     0.72    0.20   2.44   6.31
17 * 1977    2.99     0.68    0.22   2.49   6.38
18 * 1978    2.82     0.72    0.20   2.48   6.22

```

The user can also define more complicated column transformations by means of a FUNC operation, which has three forms.

FUNC L1,L2,K

where image line K has a form "XXXX XXXX XXXX -123.12" computes a numerical function of XXXX-columns as a new column indicated by -123.12 on the image line for the lines L1-L2. The function must be defined on program lines 100-119 as a subroutine in the form

```
100 Y=function(X(1),X(2),...,X(M)):RETURN
```

where M is the number of XXXX-columns.

If M=1 (function of single variable), the subroutine may have the simple form 100 Y=function(X):RETURN

If the EDITOR program is protected (#ERR A06), the general form of FUNC described below must be employed.

```
FUNC N,L1,L2,K
```

where K=" XXXX XXXX XXXX -123.12 "

works as FUNC above, but employs a function subroutine N which is saved on the data disk under name N in the form:

```
99 GOTO 1415
```

```
100 Y=function(X(1),X(2),...,X(M)):RETURN
```

```
FUNC N,L1,L2,K,C(1),C(2),...
```

works as FUNC N,L1,L2,K above, but uses constants C(1),C(2),... as auxiliary parameters. Now the subroutine must be saved on the data disk under name N and in the form

```
99 GOTO 1415
```

```
100 Y=function(X(1),X(2),...,X(M);C(1),C(2),...):RETURN
```

For example, the following subroutine computes the sum of squares of any columns in a table (e.g. the communalities of a factor matrix):

```
99 GOTO 1415:REM "COMM"
```

```
100 Y=0:FOR I=1 TO C(1):Y=Y+X(I)↑2:NEXT I:RETURN
```

If this subroutine is saved on user's data disk under name "COMM" it can be called in EDITOR by the FUNC operation of the form

FUNC COMM,L1,L2,K,M where K is the image line specifying M columns filled with X's

The following displays show how our FUNC COMM operates for computation of communalities. Activation of FUNC COMM

#### Disp.5.11

1 SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)						
20	*Factor matrix on 48 athletes in Decathlon:					
20	*FUNC COMM,22,34,21,4_					
21	*	XXXXXXX	XXXXXXX	XXXXXXX	XXXXXXX 1.1234	
22	*POINTS	0.3914	0.6913	0.4496	0.1532	
23	*100M	-0.0561	0.7453	-0.3773	-0.2139	
24	*BR.JUMP	0.0797	0.5958	0.4356	0.0017	
25	*SHOTPUT	0.8521	-0.0128	-0.0906	-0.0605	
26	*HEIGHT	0.3345	-0.3018	0.5465	0.0725	
27	*400M	-0.4020	0.6098	-0.1976	0.4181	
28	*HURDLES	0.2286	0.6324	-0.1086	0.1471	
29	*DISCUS	0.8616	0.0124	0.0428	-0.1672	
30	*POLEVLT	-0.3184	0.1773	0.2089	-0.6447	
31	*JAVELIN	0.1203	-0.0381	0.7079	0.2428	
32	*1500M	-0.6032	-0.2191	-0.0245	0.5554	
33	*LENGTH	0.8026	-0.0491	-0.2760	0.3152	
34	*WEIGHT	0.8629	-0.1230	-0.2792	0.0942	

leads to the result:

#### Disp.5.12

1 SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)						
20	*Factor matrix on 48 athletes in Decathlon:					
20	*FUNC COMM,22,34,21,4_					
21	*	XXXXXXX	XXXXXXX	XXXXXXX	XXXXXXX 1.1234	
22	*POINTS	0.3914	0.6913	0.4496	0.1532 0.8567	
23	*100M	-0.0561	0.7453	-0.3773	-0.2139 0.7467	
24	*BR.JUMP	0.0797	0.5958	0.4356	0.0017 0.5510	
25	*SHOTPUT	0.8521	-0.0128	-0.0906	-0.0605 0.7381	
26	*HEIGHT	0.3345	-0.3018	0.5465	0.0725 0.5068	
27	*400M	-0.4020	0.6098	-0.1976	0.4181 0.7473	
28	*HURDLES	0.2286	0.6324	-0.1086	0.1471 0.4856	
29	*DISCUS	0.8616	0.0124	0.0428	-0.1672 0.7722	
30	*POLEVLT	-0.3184	0.1773	0.2089	-0.6447 0.5920	
31	*JAVELIN	0.1203	-0.0381	0.7079	0.2428 0.5759	
32	*1500M	-0.6032	-0.2191	-0.0245	0.5554 0.7209	
33	*LENGTH	0.8026	-0.0491	-0.2760	0.3152 0.8221	
34	*WEIGHT	0.8629	-0.1230	-0.2792	0.0942 0.8465	

For moving one or more columns downwards with respect to the other columns a LAG operation is available.

LAG L1,L2,K,C

where image line K=" XXXX XXXX XXX " moves the columns indicated by X's on the image line C steps downwards. The C first lines on the corresponding columns will be cleared. Missing C implies C=1.

The DIFF and %DIFF operations can be used for computing differences in time series.

DIFF L1,L2,K,C

where K=" XXXX -123.12 " computes on lines L1-L2 the C step differences of X-column according to the image -123.12. Missing C implies C=1.

%DIFF L1,L2,K,C

works as DIFF, but computes the differences in percentages.

For example, to compute the yearly proportional change of total consumption in Display 5.9 we enter the %DIFF operation

```
2 *%DIFF 9,18,3_
3 * XXXX +12.12
```

and after execution obtain the result:

Disp.5.13

	SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)						
1	*						
2	*%DIFF 9,18,3_						
3	*					XXXX +1.12	
4	* Alcoholic beverages consumed in Finland 1969-1978						
5	* (litres per inhabitant, 100 % alcohol)						
6	*						
7	*	strong	wines	long	beer	total	%
8	*	liquors		drinks			
9	*	1969	1.58	0.48	0.04	2.11	4.21
10	*	1970	1.75	0.53	0.04	1.98	4.30 +2.13
11	*	1971	2.07	0.56	0.04	2.05	4.72 +9.76
12	*	1972	2.19	0.57	0.04	2.28	5.10 +8.05
13	*	1973	2.46	0.69	0.10	2.35	5.60 +9.80
14	*	1974	2.90	0.89	0.18	2.48	6.45 +15.17
15	*	1975	2.81	0.77	0.20	2.41	6.19 -4.03
16	*	1976	2.95	0.72	0.20	2.44	6.31 +1.93
17	*	1977	2.99	0.68	0.22	2.49	6.38 +1.10
18	*	1978	2.82	0.72	0.20	2.48	6.22 -2.50

(The labels 'total' and '%' on line 7 are inserted afterwards.)

The operations L+,L-,L\* and L/ compute sums, differences, products and ratios of numbers located on consecutive lines in the same columns.

L+ L1,L2,K,L

where image line K=" -123.12 -123.12 -123.12 " computes the sum of lines L1-L2 for the columns indicated by the image line and inserts the sums to line L according to the formats on image line.

L\* L1,L2,K,L

works as L+, but the product is computed.

L- L1,L2,K,L

computes the difference of lines L1 and L2 to line L.

L/ L1,L2,K,L

computes the ratio of lines L1 and L2 to line L.

For instance, if we like to compute the sums of the frequencies over 'Class' in the following display:

Disp.5.14

```

40 *TABLE PAYNED,41,46,F
41 *      Sex      Male      Female
42 *      Vote     Cons  Labour  Cons  Labour
43 *Class  *****
44 *upper          82    30    96    30
45 *lower          79    53   101    34
46 *work           118   252   155   227
47 *
48 *
49 *

```

the operation

```

49 *                      111    111    111    111
50 *L+ 44,46,49,47_

```

will do that for us and the result is:

Disp.5.15

```

40 *TABLE PAYNED,41,46,F
41 *      Sex      Male      Female
42 *      Vote     Cons  Labour  Cons  Labour
43 *Class  *****
44 *upper          82    30    96    30
45 *lower          79    53   101    34
46 *work           118   252   155   227
47 *                  279   335   352   291
48 *
49 *                  111    111    111    111
50 *L+ 44,46,49,47_

```

5.4. Loading and saving SURVO 76 files

The data matrices typed in the edit field can be saved in SURVO 76 data files for subsequent analysis with various SURVO 76 modules.

On the other side, parts of SURVO 76 data files, correlation files, and various matrix files can be loaded in the current edit field for editing and output or for statistical operations which are controlled from EDITOR.

Before saving a new data set we have to create a suitable file on user's data disk. Thus, if the data is located already in the edit field, save it temporarily, exit from EDITOR (press F0 and F16), and select the DATA module for creating of a data file. At this stage it is not necessary to define all variables needed; it is only important that the file created is large enough for the data set in question. Thereafter return to EDITOR and load the edit field to be used.

The data set to be saved in a SURVO 76 file must be in a matrix form, the columns corresponding to variables and the rows to observations. Each observation line has to start with a name (a contiguous string; only 8 first characters will be saved) and somewhere in the field (typically just above the data set) a header line declaring the names of the variables (columns) must be typed. Also the names of the variables have to be contiguous strings and the 8 first characters of each are used as names for variables in the SURVO 76 file.

For instance, among the preceding displays the data sets in displays 5.3 and 5.9 have the desired structure. In display 5.3 we have 5 variables (Coffee, Tea, Beer, Wine, Spirits) and 12 observations with names Sweden, Denmark, Finland etc. When saving this data matrix in a SURVO 76 file the line 8 could be used as a header line which defines the names of the variables also in that SURVO 76 data file. In display 5.9 we have 10 observations of 4 variables and line 7 will serve as a header line.

To save the data in display 5.3 in an empty SURVO 76 file DRINKS the following SAVE operation should be activated:

Disp.5.16

```

1  SURVO 76 EDITOR   (C)1979 S.Mustonen   (100x100)
8  *      Coffee Tea  Beer  Wine Spirits
9  *      (kg)   (kg)  (1)   (1)   (1)
10 *
11 * Sweden    12.9 0.30 58.3  7.9 2.9
12 * Denmark   11.8 0.41 113.9 10.4 1.7
13 * Finland   12.5 0.15 54.7  7.6 2.7
14 * Norway    9.4 0.19 43.5  3.1 1.8
15 * France    5.2 0.10 44.5 104.3 2.5
16 * Ireland   0.2 3.73 124.5  3.8 1.9
17 * Italy     3.6 0.06 13.6 106.6 2.0
18 * Holland   9.2 0.58 75.5  9.7 2.7
19 * Portugal  2.2 0.03 27.5  89.3 0.9
20 * Switzerland 9.1 0.25 73.5  44.9 2.1
21 * Spain     2.5 0.03 43.6  73.2 2.7
22 * England   1.8 3.49 113.7  5.1 1.4
23 *
24 *SAVE 11,22,8,DRINKS_
25 *

```

The general structure of this SAVE operation is:

SAVE L1,L2,K,N

where N is the name of a SURVO 76 data file, L1-L2 are the observations to be saved and K is line specifying the names of the variables.

The SAVE operation can be used for saving new data matrices as well as for enlarging and editing SURVO 76 data files. Eventual new variables and observations will be automatically defined in the SURVO 76 file.

For instance, to alter some data values in SURVO 76 file DRINKS after the initial saving in Display 5.16 it is sufficient to make those alterations in the edit field and thereafter reactivate the LOAD operation. Of course in many cases it is more natural to make corrections directly in the data file DRINKS by means of the SURVO 76 module DATA. If the corrected data file is needed later in the edit field it is easy to load it back by using a LOAD operation.

A LOAD operation without any parameters can be used for loading parts of SURVO 76 data files and results of SURVO 76 conversations saved on disk in a matrix form.

When LOAD has been activated the edit field disappears temporarily and a typical SURVO 76 conversation follows. At first the user has to select the type of the file to be loaded into the edit field:

Disp.5.17

LOADING VARIOUS SURVO 76 FILES:

- 1: DATA FILE
  - 2: CORRELATION FILE
  - 3: MATRIX FILE (I.E. CORRELATION FILE ETC.)
  - 4: 'MATRI' FILE
  - 5: TABLE FILE (MULTIWAY TABLE)
- SELECT 1,2,3,4,5? \_

If alternative 1 (DATA FILE) is selected, the user has to enter the name of the file and select the variables and observations to be loaded. Also the first line in the edit field for use of the selected data set is to be specified. Default is always the first empty line in the field. Moreover the user can determine the format for each variable separately in the form -###.###

For example, if we load the 5 first observations and variables Wine,

Beer from the file DRINKS using format ####.## for data values we shall have a following display:

Disp.5.18

```
1  *
2  *LOAD_
3  *DATA DRINKS,6,10,4
4  *      Beer      Wine
5  *
6  * Sweden    58.30    7.90
7  * Denmark   113.90   10.40
8  * Finland   54.70    7.60
9  * Norway    43.50    3.10
10 * France    44.50   104.30
11 *
```

Observe that EDITOR specifies this data set with a DATA line which is placed automatically above the data matrix. In statistical operations working on data matrices a DATA specification will be used in the same way as TABLE specification was used already in the introductory example (see also chapter 7).

The other types of SURVO 76 files can be loaded in a similar fashion. During a short SURVO 76 conversation the dimensions and the format of the output will be defined and thereafter the matrix selected will appear with proper labels in the edit field.

## 6. Multiway tables

The most important statistical operations working directly in the edit field are dealing with multiway tables. EDITOR provides operations for analysis of variance (ANOVA) and log-linear models for contingency tables (TABFIT). Several technical operations are also available for table editing and modification. By means of these operations it is possible to obtain typical derived tables needed in analysis.

Before presenting the statistical operations we introduce the general structure of a multiway table in edit field and various auxiliary operations to be used for table management.

### 6.1. Definition of a multiway table

A multiway table is an n-dimensional array consisting either of data values or frequencies. The n dimensions governing this array are usually variables in nominal scale. They are often called factors. We use here a general term classifier.

When  $n=1$  or 2 the table can readily be presented in a simple tabular form, but when  $n>2$  difficulties may arise, since no multidimensional representation is possible directly in two dimensions.

Usually three- and more dimensional tables are represented by entering some of the classifiers in a nested form like in Display 6.1 where we have Class and Age as row classifiers and Sex and Vote as column classifiers. (This display is same as Disp.2.17.)

#### Disp.6.1

		1 SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)	
16	*TABLE PAYNE,17,37,F_	Sex	Male Female
17	*	Vote	Cons Labour Cons Labour
18	*		
19	*Class Age	*****	
20	*		
21	*upper >73	4	0 10 0
22	* 51-73	27	8 26 9
23	* 41-50	27	4 25 9
24	* 26-40	17	12 28 9
25	* <26	7	6 7 3
26	*		
27	*lower >73	8	4 9 2
28	* 51-73	21	13 33 8
29	* 41-50	27	12 29 4
30	* 26-40	14	15 17 13
31	* <26	9	9 13 7
32	*		
33	*work >73	8	15 17 4
34	* 51-73	35	62 52 53
35	* 41-50	29	75 32 70
36	* 26-40	32	66 36 67
37	* <26	14	34 18 33

In principle, we can have any number of row and column classifiers the total number of them being the dimension n.

The one- and two-dimensional subtables can be treated simply by standard operations introduced before. For management of a multiway table as an integrated entity we must provide a TABLE specification which is on line 16 in the previous example.

The general form of a TABLE specification is

TABLE <name of table>,L1,L2,<type>

where L1-L2 are lines for labels and elements in the table, and <type> is either X (=elements are data values) or F (=elements are frequencies).

In the sequel we shall speak about X- and F-tables. Most of the auxiliary operations treat all tables similarly, but in certain cases

the treatment varies depending on the type.

Observe that the TABLE specification tells nothing about the dimension and the classifiers. EDITOR can "see" this information directly from the layout by the aid of the auxiliary line (line 19 in the example) containing one or several asterisks (\*) as guidance.

The names of the row classifiers (Class, Age) are typed on the same line to the left from \*'s and the names of the column classifiers are above the \*'s in the same column. The number of \*'s defines the maximum length of labels.

The class names used as labels for columns and rows must be continuous strings. They can be typed hierarchically as above, or completely. The contents of the table must always obey the hierarchy specified by the order of classifiers and the number of distinct classes (class names) in label rows and columns. In case of ambiguous labels the general structure interpreted by EDITOR defines the position of each element in the n-dimensional table.

To make the table structure clearer for the reader extra empty lines (as lines 26 and 32 in the example) may be inserted between the rows.

## 6.2. Table management and editing

The following TAB-operations are available for table modifications and they insert the modified table from line L onwards and label it by name <old name><4th letter of operator>.

```
TABC <table>,L
copies a table.

TABS <table>,S,L
displays the table with S row classifiers (S=0,1,2,...,N).

TABM <table>,A,B,L
changes the positions of the classifiers A and B.

TABI <table>,C,C1,C2,L
changes the positions of classes C1 and C2 for classifier C.

TABD <table>,C,L
deletes classifier C.

In a frequency table the elements of the modified table are the marginal sums of C-classification; In this case TABD means collapsing over classifier C and the result is an (n-1)-dimensional table.

In a data table the C-classification will be changed to a N-classification where the C-class names are replaced by indices 1,2,... The modified table has still n-dimensional and has the old structure and contents; only the name C has replaced by N. If TABD is applied to a data table which already includes an N-classification, the C-classification will be united to the N-classification and now the dimension of the table will be decreased by 1 but the number of elements remains the same.
```

The rationale behind TABD is obvious in case of F-tables, but in case of X-tables some comments are needed. The classifier N with class names 1,2,3,... corresponds to pure ordering of replications in experimental designs. When deleting a classifier the elements are still preserved, but they are arranged within an augmented N-classification. The result of using TABD successively for all classifiers in a X-table will be a one-dimensional table with a single vector of all the original elements along an N-classification.

```
TABJ <F-table>,C,C1,C2,Cn,L
combines (in a frequency table) the classes C1 and C2 in the C-classification and calls the united class by Cn. The class Cn takes the position of C1 and C2-class will be cancelled.

TAB+ <F-table 1>,<F-table 2>,L
forms the sum of tables 1 and 2.
```

TAB- <F-table 1>, <F-table 2>, L  
 forms the difference of tables 1 and 2.

As an example, how these operations work in practice we shall make diverse modifications of the following, artificial 4-dimensional table:

Disp.6.2

```

1  SURVO 76 EDITOR      (C)1979 S.Mustonen      (100x100)
1  *
2  *
3  *
4  *TABLE TEST,5,11,X
5  *          A  A1      A2      A3
6  *          B  B1 B2 B3 B1 B2 B3 B1 B2 B3
7  * C  D    **
8  * C1 D1      1  5  9 13 17 21 25 29 33
9  * D2      2  6 10 14 18 22 26 30 34
10 * C2 D1     3  7 11 15 19 23 27 31 35
11 * D2      4  8 12 16 20 24 28 32 36
12 *

```

In this representation A and B are column classifiers with 3 classes each while C and D are dichotomous row classifiers.

If we like to have also C as a column classifier this situation will be reached by a TABS operation:

```

13 *TABS TEST,1,14_
which gives to us:

```

Disp.6.3

```

12 *
13 *TABS TEST,1,14_
14 *TABLE TESTS,15,20,X
15 *   A   A1           A2           A3
16 *   B   B1   B2   B3   B1   B2   B3   B1   B2   B3
17 *   C   C1 C2 C1 C2 C1 C2 C1 C2 C1 C2 C1 C2 C1 C2
18 *D  ** 
19 *D1      1  3  5  7  9 11 13 15 17 19 21 23 25 27 29 31 33 35
20 *D2      2  4  6  8 10 12 14 16 18 20 22 24 26 28 30 32 34 36
21 *

```

or when we like to have all classifiers on the row side it is enough to change the parameter 1 to 4 in the TABS operation and after the execution we obtain the result:

Disp.6.4

```

12 *
13 *TABS TEST,4,14
14 *TABLE TESTS,15,51,X
15 *A  B  C  D  **
16 *A1 B1 C1 D1      1
17 *          D2      2
18 *          C2 D1      3
19 *          D2      4
20 *          B2 C1 D1      5
21 *          D2      6
22 *          C2 D1      7
23 *          D2      8

```

In this form the table has 36 rows. Only 8 first of them are now visible, but the user can scan the table easily by scrolling the field with edit keys.

Observe that this representation is closely related to the normal data matrix. By means of standard editing operations it is easy to modify this table into the form:

Disp.6.5

```

12   *
13   *TABS TEST,4,14
14   *
15   *      A   B   C   D       X
16   * 1     1   1   1   1       1
17   * 2     1   1   1   2       2
18   * 3     1   1   2   1       3
19   * 4     1   1   2   2       4
20   * 5     1   2   1   1       5
21   * 6     1   2   1   2       6
22   * 7     1   2   2   1       7
23   * 8     1   2   2   2       8

```

and then we have a standard data matrix of 36 observations and 5 variables. On the other hand, if we load in the edit field a SURVO 76 data file having this structured representation it is possible to reach other representations by the TABS operation.

In order to change the positions of two classifiers the TABM operation is to be employed. For example,

13 \*TABM TEST,A,D,14\_

changes the places of A- and D-classification giving:

Disp.6.6

```

12   *
13   *TABM TEST,A,D,14_
14   *TABLE TESTM,15,23,X
15   *      D   D1      D2
16   *      B   B1 B2 B3 B1 B2 B3
17   *C  A  **
18   *C1 A1      1 5 9 2 6 10
19   *  A2      13 17 21 14 18 22
20   *  A3      25 29 33 26 30 34
21   *C2 A1      3 7 11 4 8 12
22   *  A2      15 19 23 16 20 24
23   *  A3      27 31 35 28 32 36
24   *

```

Moreover, if some classes ought to be interchanged the TABI will do that. For instance, in the modified table TESTM above the the classes D1 and D2 are interchanged (in situ) with

13 \*TABI TESTM,D,D1,D2,14\_

and we obtain

Disp.6.7

```

12   *
13   *TABI TESTM,D,D1,D2,14
14   *TABLE TESTMI,15,23,X
15   *      D   D2      D1
16   *      B   B1 B2 B3 B1 B2 B3
17   *C  A  **
18   *C1 A1      2 6 10 1 5 9
19   *  A2      14 18 22 13 17 21
20   *  A3      26 30 34 25 29 33
21   *C2 A1      4 8 12 3 7 11
22   *  A2      16 20 24 15 19 23
23   *  A3      28 32 36 27 31 35
24   *

```

Observe that table TESTMI now overwrites TESTM.

The preceding operations work similarly on X- and F-tables, but in TABD the result is different depending on whether an X or F is appearing as the 4th parameter in TARF specification.

An example of TABD was presented already in chapter 2 (Displays 2.18-19). Now let us consider X-table TEST in Disp.6.2 again.

Assume that we like to delete both row classifiers C,D in order to have the data set structured only by classifiers A,B. This will be achieved in two steps. At first perform a TABD operation

13 \*TABD TEST,D,14\_

and obtain:

Disp.6.8

```
12 *
13 *TABD TEST,D,14_
14 *TABLE TESTD,15,21,X
15 *      A   A1   A2   A3
16 *      B   B1 B2 B3 B1 B2 B3 B1 B2 B3
17 *C   N   **
18 *C1  1      1  5  9 13 17 21 25 29 33
19 *    2      2  6 10 14 18 22 26 30 34
20 *C2  1      3  7 11 15 19 23 27 31 35
21 *    2      4  8 12 16 20 24 28 32 36
22 *
```

Observe now that the effect was simply changing the D-classification into N-classification with classes 1,2. Now to delete also C-classification, edit line 13 into form

13 \*TABD TESTD,C,14\_

and activation of this operation gives finally:

Disp.6.9

```
12 *
13 *TABD TESTD,C,14_
14 *TABLE TESTDD,15,21,X
15 *      A   A1   A2   A3
16 *      B   B1 B2 B3 B1 B2 B3 B1 B2 B3
17 *N   **
18 * 1      1  5  9 13 17 21 25 29 33
19 * 2      2  6 10 14 18 22 26 30 34
20 * 3      3  7 11 15 19 23 27 31 35
21 * 4      4  8 12 16 20 24 28 32 36
22 *
```

and hence we have a dummy N-classification left in order to keep record of similar observations with respect to the remaining A- and B-classifications.

Of course, the same result could have been reached in this case by simple editing.

The TABJ operation works only on F-tables. If we like to combine classes 'upper' and 'lower' in 'Class'-classification the TABJ operation does that as follows:

Disp.6.10

```

38 *
39 *TABJ PAYNE,Class,upper,lower,middle,40_
40 *TABLE PAYNEJ,41,53,F
41 *                               Sex      Male      Female
42 *                               Vote     Cons    Labour   Cons   Labour
43 *Class   Age    *****
44 *middle  >73           12      4       19      2
45 *      51-73          48      21      59      17
46 *      41-50          54      16      54      13
47 *      26-40          31      27      45      22
48 *      <26            16      15      20      10
49 *work   >73          8       15      17      4
50 *      51-73          35      62      52      53
51 *      41-50          29      75      32      70
52 *      26-40          32      66      36      67
53 *      <26            14      34      18      33
54 *

```

6.3. Analysis of variance

The ANOVA operation computes the analysis of variance table for any X-table according to the standard n-factorial layout. Thus analysis of variance for one-way, two-way, three-way etc. classification with equal number of observations in each combination of factors can be computed. Also Latin squares and some other balanced experiments can be handled by pooling results from one or more ANOVA operations (see e.g. Afifi,Azen, 1979).

In SURVO 76 EDITOR analysis of variance tables for those more special experiments may be formed by numerical editing operations after some ANOVA operations.

The ANOVA operation has a simple structure

ANOVA <X-table>,L

where L is the first line for the ANOVA table.

For example, to compute the results for the following 3x2x2x2 factorial experiment:

Disp.6.11

1	SURVO 76 EDITOR	(C)1979 S.Mustonen	(100x100)
1	*		
2	*		
3	*TABLE EXP,4,11,X		
4	*      A  1      2      3		
5	*      B  1  2      1  2      1  2		
6	* C  D  *		
7	* 1  1      1.3 1.2  1.8 1.5  2.3 2.1		
8	* 2      1.2 1.0  1.7 1.5  2.4 2.0		
9	*		
10	* 2  1      1.5 1.4  2.2 1.9  2.6 2.4		
11	* 2      1.5 1.3  2.3 2.0  2.9 2.7		
12	*		

we activate the ANOVA operation

13 \*ANOVA EXP,14\_

and obtain the results:

Disp.6.12

```

13 *ANOVA EXP,14_
14 *ANALYSIS OF VARIANCE ON TABLE EXP
15 *SOURCE OF      SUM OF      DEGREES OF    MEAN
16 *VARIATION       SQUARES     FREEDOM      SQUARE
17 * A             5.0625      2            2.5313
18 * B             0.3038      1            0.3038
19 * AB            0.0175      2            0.0088
20 * C             0.9204      1            0.9204
21 * AC            0.0608      2            0.0304
22 * BC            0.0004      1            0.0004
23 * ABC           0.0058      2            0.0029
24 * D             0.0038      1            0.0038
25 * AD            0.0625      2            0.0313
26 * BD            0.0038      1            0.0038
27 * ABD           0.0075      2            0.0038
28 * CD            0.0504      1            0.0504
29 * ACD           0.0108      2            0.0054
30 * BCD           0.0004      1            0.0004
31 * ABCD          0.0058      2            0.0029
32 * TOTAL          6.5163      23           0.2833
33 *

```

Next we consider the following example taken from Hald (1952), p.444.

Disp.6.13

```

1 SURVO 76 EDITOR   (C)1979 S.Mustonen   (100x100)
1 *
2 *
3 *
4 * The quality of paper from a paper mill was tested at regular inter-
5 *vals by taking samples and determining their breaking strength.
6 *We test the hypothesis that the variation of breaking strength is ran-
7 *dom by using the following data set consisting of 11 sets of samples
8 *and 5 determinations of breaking strength made in each set.
9 *
10 *TABLE PAPER,11,23,X
11 *      N  1  2  3  4  5
12 * Set  ***
13 * 1      9.1 8.5 8.6 7.2 7.6
14 * 2      8.1 8.2 7.6 7.7 8.0
15 * 3      8.5 9.0 8.1 8.5 8.0
16 * 4      7.4 7.2 8.3 8.5 8.5
17 * 5      9.0 7.9 8.2 7.6 8.7
18 * 6      8.6 8.8 9.2 8.8 9.8
19 * 7      8.5 8.1 8.8 8.6 8.5
20 * 8      7.8 8.8 7.6 8.9 8.7
21 * 9      8.7 8.2 8.7 9.2 8.7
22 * 10     8.5 8.9 7.7 8.1 9.0
23 * 11     7.3 7.9 7.9 8.5 9.2_

```

At first we execute an ANOVA operation and obtain the following results:

Disp.6.14

```

1 SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)
21 * 9 8.7 8.2 8.7 9.2 8.7
22 * 10 8.5 8.9 7.7 8.1 9.0
23 * 11 7.3 7.9 7.9 8.5 9.2
24 *
25 *ANOVA PAPER,26_
26 *ANALYSIS OF VARIANCE ON TABLE PAPER
27 *SOURCE OF SUM OF DEGREES OF MEAN
28 *VARIATION SQUARES FREEDOM SQUARE
29 * N 0.8764 4 0.2191
30 * S 5.0873 10 0.5087
31 * NS 11.7636 40 0.2941
32 * TOTAL 17.7273 54 0.3283
33 *

```

At this stage we have the results in the form of two-factor (N,S) layout, but since the variation within sets is assumed to be random the sums of squares and degrees of freedom for N and NS variation must be pooled by simple addition.

This is done by means of a L+ operation (see Disp.5.15) as follows:

Disp.6.15

```

24 *
25 *ANOVA PAPER,26_
26 *ANALYSIS OF VARIANCE ON TABLE PAPER
27 *SOURCE OF SUM OF DEGREES OF MEAN
28 *VARIATION SQUARES FREEDOM SQUARE
29 * N 0.8764 4 0.2191
30 * S 5.0873 10 0.5087
31 * NS 11.7636 40 0.2941
32 * TOTAL 17.7273 54 0.3283
33 *
34 * N 0.8764 4 0.2191
35 * NS 11.7636 40 0.2941
36 * 12.6400 44
37 * 11.1111 11
38 *L+ 34,35,37,36_
39 *
40 *

```

We have copied (by F15) the lines 29 and 31 to lines 34-35 and computed the required sums by the L+ operation. The result corresponding to the variation within sets is then on line 36.

By inserting new labels and rearranging the lines in the ANOVA table the result may be displayed in the traditional form:

Disp.6.16

```

26 *ANALYSIS OF VARIANCE ON TABLE PAPER
27 *SOURCE OF SUM OF DEGREES OF MEAN
28 *VARIATION SQUARES FREEDOM SQUARE
29 * Between sets 5.0873 10 0.5087
30 * Within sets 12.6400 44 0.2873
31 * Total 17.7273 54
33 *

```

As a third example we consider a Latin square with 5 treatments.  
(Also this one is taken from Hald 1952, p.508.)

Disp. 6.17

	SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)						
11	*TABLE HALD,12,18,X						
12	*	B	1	2	3	4	5
13	*	A	*				
14	*	1	7.6	11.8	17.6	8.8	17.9
15	*	2	21.4	12.9	12.4	15.0	20.6
16	*	3	16.0	9.7	7.4	18.4	16.6
17	*	4	16.0	18.3	23.6	27.4	25.2
18	*	5	23.3	30.5	25.8	24.5	26.6
19	*						
20	*	1	C	E	A	D	B
21	*	2	B	D	E	C	A
22	*	3	A	C	D	B	E
23	*	4	E	B	C	A	D
24	*	5	D	A	B	E	C
25	*						

These results of an experiment have been entered as an X-table HALD according to two criteria A,B. The 5 replications of the 5 treatments are distributed over this square according to the Latin square represented on lines 14-18; we call the corresponding factor C.

To create the ANOVA table for this Latin Square we perform two ANOVA operations; the first of them directly on table HALD and the second on a modified table (HALDC) where the rows of table HALD have been permuted according to the lines 14-18 by a PERM operation. Thus we shall have two ANOVA tables; the first on factors A,B and the second on factors A,C and the final results for the Latin square experiment will be obtained as a simple combination.

The whole story with comments inserted to the right side can be summarized as follows:

Disp.6.18

Display on the CRT (lines entered by the user are indicated by -)	Comments
-11 *TABLE HALD,12,18,X	<u>Lines 11-18:</u>
-12 * B 1 2 3 4 5	Entering the observations according to the levels of factors A and B as a table HALD
-13 *A *	
-14 *1 7.6 11.8 17.6 8.8 17.9	
-15 *2 21.4 12.9 12.4 15.0 20.6	
-16 *3 16.0 9.7 7.4 18.4 16.6	
-17 *4 16.0 18.3 23.6 27.4 25.2	
-18 *5 23.3 30.5 25.8 24.5 26.6	
19 *	
-20 *1 C E A D B	<u>Lines 20-24:</u>
-21 *2 B D E C A	Entering the third factor C as a Latin square
-22 *3 A C D B E	
-23 *4 E B C A D	
-24 *5 D A B E C	
25 *	
-26 *TABC HALD,28	Copy HALDC of HALD on 28-35
-27 *PERM 31,35,20	PERM permutes the copy HALDC
28 *TABLE HALDC,29,35,X	according to the Latin square.
29 * C 1 2 3 4 5	B is replaced by C.
30 *A *	
31 *1 17.6 17.9 7.6 8.8 11.8	
32 *2 20.6 21.4 15.0 12.9 12.4	
33 *3 16.0 18.4 9.7 7.4 16.6	
34 *4 27.4 18.3 23.6 25.2 16.0	
35 *5 30.5 25.8 26.6 23.3 24.5	
36 *	

```

-37 *ANOVA HALD,38
38 *ANALYSIS OF VARIANCE ON TABLE HALD
39 *SOURCE OF      SUM OF      DEGREES OF
40 *VARIATION      SQUARES     FREEDOM
41 * B             77.154      4
42 * A             660.342     4
43 * BA            301.890     16
44 * TOTAL         1039.386    24
45 *
-46 *ANOVA HALDC,47
47 *ANALYSIS OF VARIANCE ON TABLE HALDC
48 *SOURCE OF      SUM OF      DEGREES OF
49 *VARIATION      SQUARES     FREEDOM
50 * C             181.546      4
51 * A             660.342     4
52 * CA            197.498     16
53 * TOTAL         1039.386    24
54 *
55 *ANALYSIS OF VARIANCE ON LATIN SQUARE HALD
56 * A             660.342      4
57 * B             77.154      4
58 * C             181.546      4
59 * RESIDUAL      120.344     12
60 * TOTAL         1039.386    24
61 *
62 *
-63 *L+ 56,58,62,61
-64 *L- 60,61,62,59

```

ANOVA computes the ANOVA table  
for HALD.

ANOVA computes the ANOVA table  
for HALDC.

Copying by F15:  
copy of line 38 (modified)  
copy of line 42  
copy of line 41  
copy of line 50  
copy of line 44

image for L-operations below  
forming 61 as a sum of 56-58  
forming residual line 59

#### 6.4. Log-linear models for contingency tables

For the estimation of log-linear models for F-tables the TABFIT operation is available. We have used this operation already in the introductory example (see chapter 2, especially displays 2.20-25).

To characterize the structure of the TABFIT operation assume that we have a 4-way F-table like TABLE PAYNE in displays 2.17 and 6.1. Here the classifiers are Age, Class, Vote and Sex. When specifying a log-linear model these variables are referred to by the initials as A,C,V and S in this case.

The general form of the TABFIT operation is

TABFIT <F-table>,K,L

where K is an image line specifying the model and L is the first line for the results.

Model specification on line K has the structure:

LOGLIN <list of marginals to be fitted>

Only hierarchical hypotheses may be tested (see e.g. Payne 1977, Bishop, Fienberg, Holland 1975), and hence it is enough to list the sufficient marginals in the model specification. For instance, if the three-dimensional marginal ACV will be fitted there is no need to mention the marginals AC,AV,CV,A,C and V in the specification.

For example, to estimate log-linear model for table PAYNE with marginals VSC,VA,SA,CA fitted we enter the TABFIT operation:

39 \*TABFIT PAYNE,40,41

40 \*LOGLIN VSC,VA,SA,CA

and activate it. During the estimation the edit field disappears from the screen and we will have a display telling us how the iterative proportional fitting (IPS) procedure is acting:

Disp.6.19

FITTING MARGINALS BY ITERATION (IPS METHOD):

# OF ITERATION	FIT
1	7.85676411443
2	.4350143992
3	.0542655821
4	.0074416167
5	.0010219528
6	.0001402384

and after the desired accuracy is reached the edit field will appear again with results:

Disp.6.21

```

1 SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)
39 *TABFIT PAYNE,40,41 -
40 *LOGLIN VSC,VA,SA,CA
41 *G42= 37.62 DF= 28 P=0.106 FITTED: VSC,VA,SA,CA (PAYNE)
42 *TABLE PAYNEF,43,60,F
43 *          Sex      Male      Female
44 *          Vote     Cons    Labour  Cons   Labour
45 *Class    Age    *****
46 *upper   >73           6     1     6     1
47 *       51-73          25    7     30    7
48 *       41-50          23    9     25    8
49 *       26-40          21   10     26   10
50 *       <26            7     4     9     4
51 *lower   >73          9     3     10    2
52 *       51-73          23   13     31    8
53 *       41-50          22   15     26    9
54 *       26-40          15   13     21    9
55 *       <26            10    9     13    6
56 *work    >73          11   10     14    9
57 *       51-73          36   61     48   56
58 *       41-50          33   72     40   61
59 *       26-40          26   71     36   68
60 *       <26            13   37     17   33
61 *

```

On the first line for the results (41) the goodness-of-fit statistics are displayed. On the next lines is the table of fitted frequencies. These results could be presented more accurately by editing at least one of the original frequencies to a form corresponding to the desired accuracy and by activating TABFIT again.

The TABFIT operation applies also to incomplete multiway tables containing structural zeros. For the structural zeros the common notation "-" should be used.

The following example is taken from Bishop, Fienberg, Holland (1975), p.224.

Disp.6.21

```

1 SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)
40 *
41 * Meadow Spittlebug Color Forms by Sex and Location
42 * (Color forms MAR,LAT and FLA are absent in the male sex.)
43 *
44 *
45 *TABLE BUG,46,53,F
46 *      COL POP TYP TRI MAR LAT FLA ALB
47 *SEX LOC *****
48 *M   high    23 55  2  -  -  -  1
49 *   low     67 186 30  -  -  -  0
50 *   grass   24 60 12  -  -  -  0
51 *F   high    28 61 12 11  3  1  3
52 *   low     63 148 17  9  6  1  3
53 *   grass   40 75 15 11  4  0  0.00
54 *

```

To fit a model with Sex-Locale two-factor effect we enter the TABFIT operation

```

55 *TABFIT BUG,56,57
56 *LOGLIN SL,C

```

and obtain the result:

Disp.6.22

```

54 *
55 *TABFIT BUG,56,57
56 *LOGLIN SL,C
57 *G^2= 32.17 DF= 21 P=0.056 FITTED: SL,C (BUG)
58 *TABLE BUGF,59,66,F
59 *      COL      POP      TYP      TRI      MAR      LAT      FLA      ALB
60 *SEX LOC *****
61 *M   high    21.45  51.23  7.71  0.00  0.00  0.00  0.61
62 *   low     74.96 178.98 26.92  0.00  0.00  0.00  2.14
63 *   grass   25.43  60.71  9.13  0.00  0.00  0.00  0.73
64 *F   high    28.68  68.48 10.30  7.22  3.03  0.47  0.82
65 *   low     59.53 142.15 21.38 14.98  6.28  0.97  1.70
66 *   grass   34.95  83.45 12.55  8.80  3.69  0.57  1.00

```

Observe that the fitted frequencies are now displayed with 2 decimals, since one element in table BUG was entered in the form 0.00.

When estimating several alternative models on the same data it is useful to place the results on consecutive lines for comparison as we already have done in the introductory example (see display 2.24). Then also the conditional G<sup>2</sup> statistics can be computed by means of an L- operation.

### 7. Correlations and linear regression analysis

The SURVO 76 includes several modules for regression analysis. There is no urgent need to perform computations of this type directly in EDITOR, since the results from various regression programs can be easily taken into the edit field for final reporting.

It might be, however, useful to have correlation and regression analysis in editor mode to some extent. This would be useful especially with small data sets in teaching.

Therefore EDITOR provides some operations from this area, too. These operations (CORR and REGRAN) assume the data set to be in the form of a data matrix and defined with a DATA specification. Each data set loaded from a SURVO 76 data file to the edit field has this specification automatically (see display 5.18).

The DATA specification is:

DATA <name of data set>,L1,L2,K

where L1-L2 are the lines for the data values in a matrix form and K is a line for the names of variables (columns).

The data set is to be represented in the form described in 5.4.

Observe that (typically for EDITOR) the DATA specification may be located on any line outside the data itself. Thus various subsets of the same data may be defined simultaneously by different DATA specifications. Normally, of course, DATA is typed above the data matrix.

The data appearing in display 5.16 is now defined with a following DATA specification:

Disp.7.1

	SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)				
1					
2	*				
3	*DATA DRINKS,11,22,8				
4		Coffee	Tea	Beer	Wine Spirits
5		(kg)	(kg)	(l)	(l)
6	*				
7	* Sweden	12.9	0.30	58.3	7.9 2.9
8	* Denmark	11.8	0.41	113.9	10.4 1.7
9	* Finland	12.5	0.15	54.7	7.6 2.7
10	* Norway	9.4	0.19	43.5	3.1 1.8
11	* France	5.2	0.10	44.5	104.3 2.5
12	* Ireland	0.2	3.73	124.5	3.8 1.9
13	* Italy	3.6	0.06	13.6	106.6 2.0
14	* Holland	9.2	0.58	75.5	9.7 2.7
15	* Portugal	2.2	0.03	27.5	89.3 0.9
16	* Switzerland	9.1	0.25	73.5	44.9 2.1
17	* Spain	2.5	0.03	43.6	73.2 2.7
18	* England	1.8	3.49	113.7	5.1 1.4
19	*				
20	*				

The CORR operation for computing means, standard deviations and correlations can be used as follows. To have the results for all variables in DRINKS we enter:

Disp.7.2

20	* Switzerland	9.1	0.25	73.5	44.9 2.1
21	* Spain	2.5	0.03	43.6	73.2 2.7
22	* England	1.8	3.49	113.7	5.1 1.4
23	*				
24	*	XXXX	XXXX	XXXXX	XXXXX XXX
25	*CORR DRINKS,24,26				

where 24 is the line specifying the columns (variables) and 26 is the line for the results. Thus by activating CORR we shall have:

Disp.7.3

```

24   *      XXXX XXXX XXXXX XXXXX XXX
25   *CORR DRINKS,24,26_
26   *          MEAN           STD.DEV.
27   * Coffee     6.700000    4.606319
28   * Tea        0.776666    1.334352
29   * Beer       65.566666   35.702618
30   * Wine       38.825000   42.506365
31   * Spirits    2.108333    0.611196
32   *
33   *CORR:    Coffee Tea   Beer   Wine   Spirit
34   *Coffee   1.000 -0.511 -0.022 -0.445  0.465
35   *Tea      -0.511  1.000  0.764 -0.453 -0.311
36   *Beer     -0.022  0.764  1.000 -0.690 -0.147
37   *Wine     -0.445 -0.453 -0.690  1.000 -0.100
38   *Spirits  0.465 -0.311 -0.147 -0.100  1.000
39   *

```

The REGRAN operation computes the parameters and the residuals of a linear regression model and has the general form:

REGRAN <name of data set>,K,L

where image line K specifies the model and L is the first line for the results.

K has the form " YYYY XXXX XXXX XXXX -123.12 ", where the Y's indicate the column of the dependent variable, X's indicate the columns of independent variables (regressors) and -123.12 indicates the place and the format for the residuals which will be displayed as a new column in the data matrix.

To have a constant term in the model a column of 1's has to be inserted as a regressor in the data set.

For example, if we like to estimate a model for the dependence of beer consumption on the consumption of other beverages in DRINKS, we shall proceed as follows:

At first a column of 1's is generated by a SET operation:

Disp.7.4

	SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)						
1	*						
2	*						
3	*						
4	*						
5	*SET 11,22,6_						
6	*				1		
7	*DATA DRINKS,11,22,8						
8	*	Coffee	Tea	Beer	Wine	Spirits	Constant
9	*	(kg)	(kg)	(1)	(1)	(1)	
10	*						
11	* Sweden	12.9	0.30	58.3	7.9	2.9	1
12	* Denmark	11.8	0.41	113.9	10.4	1.7	1
13	* Finland	12.5	0.15	54.7	7.6	2.7	1
14	* Norway	9.4	0.19	43.5	3.1	1.8	1
15	* France	5.2	0.10	44.5	104.3	2.5	1
16	* Ireland	0.2	3.73	124.5	3.8	1.9	1
17	* Italy	3.6	0.06	13.6	106.6	2.0	1
18	* Holland	9.2	0.58	75.5	9.7	2.7	1
19	* Portugal	2.2	0.03	27.5	89.3	0.9	1
20	* Switzerland	9.1	0.25	73.5	44.9	2.1	1
21	* Spain	2.5	0.03	43.6	73.2	2.7	1
22	* England	1.8	3.49	113.7	5.1	1.4	1
23	*						

Also the label 'Constant' has been typed on line 8.

To perform the regression analysis we enter the REGRAN operation in the form:

Disp.7.5

```

7 *DATA DRINKS,11,22,8
8 *          Coffee Tea Beer Wine Spirits Constant
9 *          (kg) (kg) (1) (1) (1)
10 *
11 * Sweden    12.9 0.30 58.3 7.9 2.9   1
12 * Denmark   11.8 0.41 113.9 10.4 1.7   1
13 * Finland   12.5 0.15 54.7 7.6 2.7   1
14 * Norway    9.4 0.19 43.5 3.1 1.8   1
15 * France    5.2 0.10 44.5 104.3 2.5   1
16 * Ireland   0.2 3.73 124.5 3.8 1.9   1
17 * Italy     3.6 0.06 13.6 106.6 2.0   1
18 * Holland   9.2 0.58 75.5 9.7 2.7   1
19 * Portugal   2.2 0.03 27.5 89.3 0.9   1
20 * Switzerland 9.1 0.25 73.5 44.9 2.1   1
21 * Spain     2.5 0.03 43.6 73.2 2.7   1
22 * England   1.8 3.49 113.7 5.1 1.4   1
23 *           XXXX XXXX YYYYYY XXXXX XXX   X      -123.1
24 *REGRAN DRINKS,23,25_

```

and after the execution the display is:

Disp.7.6

1 SURVO 76 EDITOR (C)1979 S.Mustonen (100x100)									
7	*DATA DRINKS,11,22,8								
8	*	Coffee	Tea	Beer	Wine				
9	*	(kg)	(kg)	(1)	(1)				
10	*								
11	*	Sweden	12.9	0.30	58.3	7.9	2.9	1	-16.1
12	*	Denmark	11.8	0.41	113.9	10.4	1.7	1	35.4
13	*	Finland	12.5	0.15	54.7	7.6	2.7	1	-15.0
14	*	Norway	9.4	0.19	43.5	3.1	1.8	1	-19.2
15	*	France	5.2	0.10	44.5	104.3	2.5	1	5.7
16	*	Ireland	0.2	3.73	124.5	3.8	1.9	1	4.7
17	*	Italy	3.6	0.06	13.6	106.6	2.0	1	-19.9
18	*	Holland	9.2	0.58	75.5	9.7	2.7	1	7.4
19	*	Portugal	2.2	0.03	27.5	89.3	0.9	1	-5.0
20	*	Switzerland	9.1	0.25	73.5	44.9	2.1	1	12.4
21	*	Spain	2.5	0.03	43.6	73.2	2.7	1	17.8
22	*	England	1.8	3.49	113.7	5.1	1.4	1	-8.2
23	*		XXXX	XXXX	YYYYYY	XXXXX	XXX	X	-123.1
24	*REGRAN DRINKS,23,25_								
25	*REGRESSION ANALYSIS: REGRESSAND:'Beer' DATA:'DRINKS'								
26	* TOTAL VARIANCE= 1274.676 DF= 11								
27	* RESIDUAL VARIANCE= 456.039 DF= 7 R <sup>2</sup> =0.7723								
28	* VARIABLE REGR.COEFF. STD.DEVIATION T								
29	* Coffee 3.983421 3.61848 1.100								
30	* Tea 26.583812 12.08591 2.199								
31	* Wine -0.015917 0.36583 -0.043								
32	* Spirits -4.596942 12.02483 -0.382								
33	* Constant 28.540865 48.56771 0.587								
34	*								

From this point on it is simple to continue the analysis by means of various editing and numerical operations. For example, the predicted values for 'Beer' can be computed as a difference between 'Beer' and the residuals by a C- operation.

In this environment it is simple to consider the data set ordered in different ways. For instance, to see the order of countries according the "overconsumption" of beer the data set can be sorted using the residual column as a sort key and the result after a SORT operation is:

Disp.7.7

	SURVO 76 EDITOR		(C)1979 S.Mustonen				(100x100)		
5	*-SORT 11,22,6						11111		
6	*								
7	*DATA DRINKS,11,22,8								
8	*	Coffee	Tea	Beer	Wine	Spirits	Constant	Residual	
9	*	(kg)	(kg)	(l)	(l)	(l)			
10	*								
11	*	Denmark	11.8	0.41	113.9	10.4	1.7	1	35.4
12	*	Spain	2.5	0.03	43.6	73.2	2.7	1	17.8
13	*	Switzerland	9.1	0.25	73.5	44.9	2.1	1	12.4
14	*	Holland	9.2	0.58	75.5	9.7	2.7	1	7.4
15	*	France	5.2	0.10	44.5	104.3	2.5	1	5.7
16	*	Ireland	0.2	3.73	124.5	3.8	1.9	1	4.7
17	*	Portugal	2.2	0.03	27.5	89.3	0.9	1	-5.0
18	*	England	1.8	3.49	113.7	5.1	1.4	1	-8.2
19	*	Finland	12.5	0.15	54.7	7.6	2.7	1	-15.0
20	*	Sweden	12.9	0.30	58.3	7.9	2.9	1	-16.1
21	*	Norway	9.4	0.19	43.5	3.1	1.8	1	-19.2
22	*	Italy	3.6	0.06	13.6	106.6	2.0	1	-19.9

Observe also how simple it is to make alterations in the data set and in the model definition, and see their their effects after reactivation of REGRAN.

REFERENCES

- Afifi,A.A., Azen,S.P. (1979), Statistical Analysis, a Computer Oriented Approach, (2nd edition), Academic Press
- Bishop,Yvonne M.M., Fienberg,S.E., and Holland,P.W. (1975), Discrete Multivariate Analysis, Cambridge, Massachusetts: MIT Press
- Butler,D.E., and Stokes,D. (1974), Political Change in Britain, (2nd edition), London: Macmillan
- Hald,A. (1952), Statistical Theory with Engineering Applications, Wiley
- Mustonen,S. (1977), SURVO 76, a statistical data processing system, Research Report No.6, Dept.of Statistics, University of Helsinki
- Mustonen,S. (1980), On interactive statistical data processing, 8th Nordic Conference on Mathematical Statistics
- Mustonen,S. (1980), Interactive analysis in SURVO 76, to appear in COMPSTAT 1980: Proceedings in computational statistics, ed. by M.M.Barritt and D.Wishart, Physica-Verlag
- Payne,C. (1977), The log-linear model for contingency tables, in The Analysis of Survey Data, ed. by C.A.O'Muircheartaigh and C.Payne, Volume 2, pp.105-144, Wiley