Raabo.

A/S REGNECENTRALEN

SCANDINAVIAN INFORMATION PROCESSING SYSTEMS

RCSL No: 53-S1

Edition: April 1970

Author: Søren Henckel

Title: data survey

Part 1

Keywords: RC 4000, Software, Statistical, Simple Data Description, Data Screening, Histogram, Fractile Diagram

Abstract: The program data survey performs a simple statistical description of a number of observations of an arbitrary number of variables. The description of one variable consists of a histogram, and fractile diagrams in the normal - and exponential distribution may be drawn. The program has facilities for specifying grouplimits, transgenerations, and subsets of a variable. 16 pages.

CONTENTS: Page

APPENDIX:

1. STATISTICAL THEORY.

It is assumed, that the observations (of the same variable) are stochastically independent and identically distributed with a distribution having central moments of order <= 4,
(e.g. EX < oo and E(X - EX)xxj) < oo for j = 2, 3, 4).

If we put mean = m1 = sum(1 <= i <= n) (X(i)) / n and
S(j) = sum(1 <= i <= n)((X(i) - m1)xxj) for j = 2, 3, 4, the program computes:

m2 = variance = S(2) / (n - 1)
std.dev. = sqrt(m2)
m3 = skewness = S(3) / (S(2)xx1.5)
m4 = kurtosis = S(4) / (S(2)xx2)
Students' t = m1/ (std.dev. / sqrt(n)).

The 95 pct. confidence interval for mean is defined as:
m1-1.96xstd.dev./sqrt(n) < mean < m1+1.96xstd.dev./sqrt(n).

The histograms are made in such a way, that the column heigth (and not as usual the column area) is proportional to the group density.

The chi square distribution test is based on
chi square = sum(1<=i<=k)((obs(i) - nxp(i))xx2/(nxp(i)))
= -n + sum(1<=i<=k)(obs(i)xx2/(nxp(i))),
with n = number of observations, k = number of groups, hxp(i) = the expected and obs(i) = the observed number of observations in the i-th group. For nxp(i) -> oo chi square is approximately chi square distributed with k-3 degrees of freedom, but the adaption is only acceptable when min(1<=i<=k) (nxp(i)) > 5.0, and the program therefore unites groups (by deleting some of the grouplimits) in order to let this condition be true.

The program use the following approximations of the normal distribution cumulative function phi(y) (see ref (4)):

(1) phi(y) eq. exp(-yxx2/2)x.39894x((.9372980xp-.1201676)xp + .4361836)xp
with p = 1/(abs(y)x.33267+1.0) for y <=0.

(2) $phi^{-1}(y)$ eq. -p+(2.30753+.27061×p)/((.04481×p+.99229)×p+1.0)
with p = sqrt(ln(y)×(-2.0)) for 0.0 <= y <= 0.5

which combined with phi(-y) = 1- phi(y) for x real

and $phi^{-1}(y)$ = - $phi^{-1}(1-y)$ for 0 <= y <= 1 gives total approximations of phi and phi^{-1}

2.1. INPUT/OUTPUT SYSTEM:

For facilitating the problems about input/output, the two standard zones in/out (connected to current input/output) are used for input/output on character level. Output is made for format A4 vertical and produced with 8 leading spaces on each line.

Besides the computed output, the output will contain small notes about the different parts of input, and possibly also some error messages.

2.2. SYNTAX OF INPUT DATA.

The input file (= current input) must contain data in accordance with the following syntax:

<input file> ::= <identification><data set>$^{*}_{0}$ <end>$^{1}_{0}$ EM

<identification> ::= < <legal character>$^{118}_{0}$ <

<legal character> ::= <letter>|<digit>|+|-|×|/|=|;|(|)|'|:|.|,|NL|SP

<end> ::= e

<data set> ::= <variables>|<variables><specification>

<specification> ::= <groups>|<execute mark>|<transgenerations>|
<compute>|<subset>

<variables> ::= <experiment number>,<variable number>,<number of cases><name> < <observation>$^{3000}_{2}$ <

<name> ::= <empty>|<letter><legal character>$^{40}_{0}$ <

<observation> ::= <real>, |<observation><checksum> s

<groups> ::= g <grouplimit>$^{48}_{2}$ <

<grouplimit> ::= <real bigger than preceding (if one)>,

<execute mark> ::= <

<transgenerations> ::= t <trans. spec.>$_{1}^{8}$ <

<trans. spec.> ::= <trans. type>,<constant 1>,<constant 2>,

<trans. type> ::= 1|2|3

<compute> ::= c <indicator normal>,<indicator exponential> <

<subset> ::= s <first case>,<last case> <

<constant> ::= <real>

<experiment number> ::= <integer>

<variable number> ::= <integer>

<number of cases> ::= <integer>

<indicator> ::= <integer>

<first case> ::= <integer>

<last case> ::= <integer>

Extra commas, spaces, and new line are allowed everywhere, and are blind outside texts. Tape feed, all holes, FF, HT, and VT are blind everywhere in <input file>.

2.3. SEMANTICS OF INPUT.

It appears from the definition of <data set>, that one set of observations might be object for several examinations with different <specification>'s because every <execute mark> causes an examination in accordance with the <specification>'s before the <execute mark>. Every part of <specification> will be valid as long as it can be interpreted correctly, or until it is actively altered by writing a new part of <specification> of the same kind. The <specification>'s are always attached to the preceding variable, and if several <specification>'s of the same kind are specified without an <execute mark> between, the last <specification> of each kind will be valid at <execute mark>.

Check sums among observations may appear everywhere (and may be omitted totally). The notion <indicator> has the following explanation: if indicator > 0 the corresponding fractile diagram will appear in output, otherwise it will be suppressed. The program is initialized by c 1, -1 <, and the histograms can never be suppressed. If the <specification>'s do not contain any <groups> (before a certain <execute mark>), the corresponding examination is performed with grouplimits computed by the program (these grouplimits will be reasonable in the most cases).

If no <subset> has been specified, the corresponding examination will of caurse cover all the observations in the actual <data set>, and it is evident that the <data set>'s not need to have the same number of observations (cases).

By using <groups> it is important to notice, that the grouplimits must be given in increasing order, whereas the groups not need to be equal in length. The grouplimits and the groupdensities are printed in output in order to obtain correct interpretation in case of groups not equal in length.

By using <transgenerations> it is <u>very important</u> to notice, that the very observations not are stored separately, but are changed by transgenerations, and that all transgenerations are made successively. The program gives possibilities of three kinds of transgenerations, and they are:

trans. type = 1 <=> y:= ln(y + constant1)×constant2
trans. type = 2 <=> y:= (y + constant1)××constant2
trans. type = 3 <=> y:= (y + constant1)×constant2.

If you want to examine y:= ((ln(y + 3.5) + 4.2)××2.1 this can be done by writing: t 1, 3.5, 1, 2, 4.2, 2.1 < before the actual <execute mark>, which means y:= ln(y + 3.5)×1; followed by y:= (y + 4.2)××2.1;

The syntax of input data is made in accordance with the syntax which was valid in 1969 for the GIER programs on regression analysis developed by Alex Jessen on A/S Regnecentralen.

2.4. ERROR MESSAGES.

The program can generate the following error messages:

(a) checksum error: computed sum = dddd check = dddd
means that the sum made by adding observations in <input file> does not agree with the <checksum> given in <input file>.

(b) cases on tape = ddd   cases = ddd
means that the number of cases actual found in <input file> does not agree with the <number of cases> given in the variable head

(c) but these are rejected
means that the limits in <groups> not has been given in increasing order (if necessary the program will compute some suitable limits).

(d) a missing execute mark of end of data is generated
means that <EM> is met before the sequence <e and this causes an extra examination performed with the current <specification>'s

(e) extra examination (not specified in input)
means that two sets of <variables> has been given without any <specification> between (not even an <execute mark>), and this causes an extra examination with the current <specification>'s

These error messages do not terminate the run, and as a general rule, the run is continued with the information actually found in <input file>.

The program can produce 7 other error messages, which all will terminate the run with a suitable short error message followed by a copy of 250 characters of <input file> from the place where the error has been detected.

The essence of these error messages are:

(1) error in art of information
(2) error in subsets
(3) error in number of constants in transgeneration information
(4) error in art of transgeneration
(5) which has too many observations
(6) error in number of grouplimits
(7) identification not terminated by < .

3. PROGRAM TAPE, STORAGE REQUIREMENTS, CAPACITY AND RUNNING TIME.

The program data survey is written in ALGOL 5, and is available on 8 channel paper tape in the normal ISO form (with even parity).

Compilation of the program can be done with the following file processor commands (see ref (2))

datasurvey = set 44
datasurvey = algol tre

Reading and compilation of the program takes approx. 27 seconds and the compilated program occupies 44 or 43 segments on backing storage corresponding to compilation with index.yes or index.no.

Compilation requires a process on 12 k bytes, whereas run requires 23 k bytes in order to obtain acceptable speed at running time. The capacity of the program is 3000 observations (exclusive checksums) in each <data set>, 48 grouplimits in each part of <grous> and 48 numbers in each part of <transgenerations> (corresponding to 8 successive transgenerations). These maximal limits can easily be altered by altering the declaration of the arrays intens, group, obs and trngen in the main program block.

Because of the structure of RC 4000 the run time cannot be given exactly, but as a rough guide it can be mentioned that:

reading and compilation takes approx. 27 seconds,
computation of moments takes about 1000 observations/second,
transgeneration takes 500-3000 observations/second,
grouping takes about 800 observations/second, and
that printer output occupies approx. 60-80 pct. of the total running time (excl. compilation).

4.1. PROGRAM STRUCTURE.

The program data survey has been made with extensive use of procedures, and the greater part of these have no parameters. This implies that the procedures often use global variables for storing results. In order to get a clearer coding almost all variables are called by names which shows their use. To obtain that the names also could be pronounced, many of the identifiers are rather long.

The reading part of the program is placed between the labels data: and execute:, and is formed as a case statement.

In every case, the actual information is controlled as far as possible and a short message about the information is given in output. When a hard error occurs, the program calls procedure error, which uses the global integer inftyp, and returns to label exit_program:.

The errors mentioned in part 2.4 ((a) to (e)) do not terminate the run but gives reasonable error messages and -reactions.

The output is made for format A4 vertical with 8 leading spaces in each line. Pages and <execute mark>'s are in output counted 1, 2, and so on, and every page in putput will contain examination number, page number, the text given in <identification>, <variable number>, and <name> in the actual <variables>.

4.2. METHOD.

The two central procedures in the program are

(1) procedure grouping;
the procedure groups the observations stored in array obs(first:last) according to the grouplimits placed in array group(1:groupnumber) (with group(groupnumber):= oo) and place the result in the integer array intens(1:groupnumber).
The automatical determination of grouplimits (controlled by the boolean groups) is made according to the following rules:
grouplength is based on standard deviation/q with q:= if number of observations < 100 then 2 else 3. This basic length is rounded according to a logaritmic scale in this way
basic length [1.25, 2.5[=> grouplength:= 2.0,
basic length [2.5, 6.0[=> grouplength:= 5.0 and
else grouplength:= 1.0 or 10 (according to the logarithmic scale).
The groups are placed in multipla of the grouplength, with at least one observation in the two outermost groups.

The basic grouping is performed in two steps:

(a) a rough partitioning according to a division of the groups in three disjoint parts

(b) the exact grouping is then performed in the classes defined by obs class(i) <=> obs > group(i - 1) and obs <= group(i) (with group(0) := - oo and group(groupnumber):= + oc)

(2) procedure moments;

the computation of moments is done on the observations stored in obs(first:last). The results are placed in variables min, max, m1, stdev, m2, m3, and m4. The following numerical algorithm is used:

Let n:= last - first + 1 and S(j):=
sum(first<i<=last)(obs(i)-obs(first))××j/n for j = 1, 2, 3 and 4, then we obtain (m3 and m4 only for m2 > 0.0)

m4:= (S(4)-4×S(3)×S(1)+6×S(2)×S(1)××2-3×S(1)××4)/m2××2
m3:= (S(3)-3×S(2)×S(1)+2×S(1)××3)/m2××1.5
m2:= S(2)-S(1)××2
m1:= S(1)+obs(first)

Finally we obtain
variance:= m2×n/(n-1) and stdev:= sqrt(m2)

If m2 = 0 the examination is terminated with an error message, and the program returns to label new:.

5. REFERENCES.

(1) Søren Lauesen: ALGOL 5 Users Manual
A/S Regnecentralen Copenhagen July 1969.

(2) Søren Lauesen: File Processor Users Manual
A/S Regnecentralen Copenhagen April 1968.

(3) A. Hald: Statistical Theory with Engineering Applications,
Wiley, London 1952.

(4) Handbook of Mathematical Functions,
National Bureau of Standards. Chap. 26.2.

6. USER'S EXAMPLE.

6.1. Preparation of input tape.

We have measured the height of 74 soldiers in centimeters and want to examine whether the observations can be described by a normal distribution.

Besides we want some different histograms according to different methods of grouping the observations.

We shall make 2 examinations:

1: The automatical determination of grouplimits has to be tried, and we want only fractile diagram in the normal distribution.

2: Only normal fractile diagram with grouplimits = 160, 163, 166, 169, 172, 175, and 181 is wanted.

Because the program gives 8 leading spaces in each line we may let every line in <identification> have 8 leading spaces (this will be nice, but is not necessary).

These claims will give an input tape (file) like this:

```
<        identification for testdata to
        program: data survey  april 1970<

35,   1,   74<
heigth of soldiers 67/68 in centimeters.<

171,170,176,170,172,168,169,167,167,165,165,179,
161,160,183,175,178,174,174,173,169,176,168,172,
172,172,170,175,170,170,171,171,171,170,169,168,
168,165,162,173,178,174,173,172,169,175,177,172,
177,173,173,172,171,170,171,173,171,168,167,167,
164,164,173,166,177,180,174,172,171,168,169,176,
172,178<<

c 1,0<   g 160,163,166,169,172,175,178,181<<
   end of  <input file>.
```

6.2. USER'S EXAMPLE

Output, interpretation, and run time.

The test run gave this output (4 pages) on current output:

17 11 70 examination number 1 page 1
søh. data-survey

identification for testdata to
program: data survey april 1970

variable number 1 heigth of soldiers 67/78 in centimeters.

input of observations: total 74 cases without checksum control
execute mark

number of cases	minimum	maximum
74	160.0000	183.0000

mean	variance	stand.dev.	skewness	kurtosis
171.1622	$1939.800_{10}-2$	4.404316	-0.046365	3.234582

t-test for mean=0 is t = 334.307
which has 73 degrees of freedom.

95 pct. confidence interval is 170.1587 < mean < 172.1657

histogram: every x represents 1 observation

number of cases	upper class-limit	
3	162.0000	XXX
2	164.0000	XX
4	166.0000	XXXX
10	168.0000	XXXXXXXXXX
12	170.0000	XXXXXXXXXXXX
17	172.0000	XXXXXXXXXXXXXXXXX
11	174.0000	XXXXXXXXXXX
6	176.0000	XXXXXX
6	178.0000	XXXXXX
2	180.0000	XX
0	182.0000	
1		X
total 74		

17 11 70 examination number 1 page 2
søh. data-survey

identification for testdata to
program: data survey april 1970

variable number 1 heigth of soldiers 67/78 in centimeters.

fractile diagram in the normal distribution

```
                                        estimates of
fraction   upper class-                 position parameter =    171.1622
in pct.    limit                           scale parameter =      4.4043

                       -2.0      -1.0       0.0       1.0       2.0
                    .    .    .    .    .    .    .    .    .    .    .

  4.05    162.0000          X

  6.76    164.0000            X

 12.16    166.0000               X

 25.68    168.0000                    X

 41.89    170.0000                         X

 64.86    172.0000                               X

 79.73    174.0000                                   X

 87.84    176.0000                                       X

 95.95    178.0000                                           X

 98.65    180.0000                                                X

 98.65    182.0000                                                X

                    .    .    .    .    .    .    .    .    .    .    .
```

chi square test for the normal distribution is chisq = 2.9918
which has 5 degrees of freedom.

17 11 70 examination number 2 page 3
søh. data-survey

identification for testdata to
program: data survey april 1970

variable number 1 heigth of soldiers 67/78 in centimeters.

output speification: histogram, fractile normal
group specification: limits=

160.0000	163.0000	166.0000	169.0000	172.0000
175.0000	178.0000	181.0000		

execute mark

number of cases	minimum	maximum
74	160.0000	183.0000

mean	variance	stand.dev.	skewness	kurtosis
171.1622	1939.800$_{10}$-2	4.404316	-0.046365	3.234582

t-test for mean=0 is t = 334.307
which has 73 degrees of freedom.

95 pct. confidence interval is 170.1587 < mean < 172.1657

histogram: every x represents 1 observation

number of cases	upper class-limit	
1	160.0000	X
2	163.0000	XX
6	166.0000	XXXXXX
15	169.0000	XXXXXXXXXXXXXXX
24	172.0000	XXXXXXXXXXXXXXXXXXXXXXXX
14	175.0000	XXXXXXXXXXXXXX
9	178.0000	XXXXXXXXX
2	181.0000	XX
1		X
total 74		

17 11 70 examination number 2 page 4
søh. data-survey

identification for testdata to
program: data survey april 1970

variable number 1 heigth of soldiers 67/78 in centimeters.

fractile diagram in the normal distribution

estimates of
position parameter = 171.1622
scale parameter = 4.4043

fraction in pct.	upper class-limit		-2.0		-1.0		0.0		1.0		2.0	
1.35	160.0000		X									
4.05	163.0000			X								
12.16	166.0000				X							
32.43	169.0000					X						
64.86	172.0000							X				
83.78	175.0000								X			
95.95	178.0000									X		
98.65	181.0000										X	

chi square test for the normal distribution is chisq = 2.2205
which has 3 degrees of freedom.

The interpretation of this may be:

The height of soldiers is very nicely described by a normal distribution with parameters $(\bar{x}, \text{sigma}^2)$ = (171.16, (4.40)xx2). The emperical distribution is nearly symmetric (skewness = -0.046), and the chi square distribution test is not significant even at 50 pct. level. It is noticed that the description seems a little better when using grouplimits at 160 + 3xh instead of 160 + 2xh, but that might be because the grouplength 2 is too small when only having 74 cases.

You might be interested in examining whether a log normal distribution can be used and this can be done by adding t 1,0.0,1000 < just before end of <input file> (this means that y is substituted by ln(y) × 1000.0;) If you only want to examine the exponential distribution, you may write c 0, 1 < and if you want to examine both distributions, you may write c1,1<.

Running time for this example was approx.:

Program reading and compilation	27 sec.
Reading, computing, and output to printer	8 sec.
Total	35 sec.