NAG Toolbox

nag_stat_plot_histogram (g01aj)

1 Purpose

nag_stat_plot_histogram (g01aj) prints a histogram on a character printing device, allowing you control over size, positioning, and the range of data values included.

2 Syntax

```
[xstep, n1, multy, ifail, nstepx, nstepy, ispace, xmin, xmax] =
nag_stat_plot_histogram(x, nstepx, nstepy, 'n', n, 'itype', itype, 'ispace',
ispace, 'xmin', xmin, 'xmax', xmax)
```

```
[xstep, n1, multy, ifail, nstepx, nstepy, ispace, xmin, xmax] = g01aj(x,
nstepx, nstepy, 'n', n, 'itype', itype, 'ispace', ispace, 'xmin', xmin, 'xmax',
xmax)
```

Note: the interface to this routine has changed since earlier releases of the toolbox:

At Mark 23: **xmin** was made optional (default 0); **xmax** was made optional (default 0); **ispace** was made optional (default 0); **itype** was made optional (default 0); output parameters were reordered.

3 Description

A histogram is printed using vertical bars consisting of * characters. The output is directed to the current advisory message unit. It may be redirected by a call to nag_file_set_unit_advisory (x04ab) before calling nag_stat_plot_histogram (g01aj).

The following options are available:

(a) inclusion of all data values, or only of those lying within a specified range;

(b) specification of the size of the histogram in the vertical and horizontal directions, and of positioning in the horizontal direction;

(c) calculation of frequencies or cumulative frequencies in the histogram.

The maximum and minimum data values used, the (integral) number of observations represented by each * in the histogram, and the step size per character position in the horizontal direction, are returned.

The histogram is headed FREQUENCY or CUM.FREQ. depending on the option selected. Each line is annotated with the minimum frequency which a bar reaching that line represents. The data maximum and minimum are printed under the histogram, unless either exceeds 9999.99 in modulus, in which case they are not printed (although they are still returned).

4 References

None.

5 Parameters

5.1 Compulsory Input Parameters

1: **x**(**n**) – REAL (KIND=nag_wp) array

The data values.

2: **nstepx** – INTEGER

The number of character positions to be used in the horizontal direction, i.e., the number of categories in the histogram.

3: **nstepy** – INTEGER

The number of character positions to be used in the vertical direction, i.e., the maximum height of a histogram bar.

5.2 Optional Input Parameters

1: **n** – INTEGER

Default: the dimension of the array **x**.

The number of data values.

Constraint: $\mathbf{n} \geq 1$.

2: **itype** – INTEGER

Default: 0

Indicates whether a histogram of frequencies or cumulative frequencies is required.

itype $= 0$ for frequencies.

itype $\neq 0$ for cumulative frequencies.

The second option effectively displays the distribution function of the data rather than the density function.

3: **ispace** – INTEGER

Default: 0

Indicates how many spaces are to be inserted at the beginning of each output line.

If **ispace** < 0 on input, **ispace** $= 0$ is used

If **ispace** $+$ **nstepx** $+ 14 > 120$, then **ispace** $= 0$ is used.

The second condition imposes an effective line length limit of 120 characters.

4: **xmin** – REAL (KIND=nag_wp)

5: **xmax** – REAL (KIND=nag_wp)

Default: 0

If **xmin** $<$ **xmax**, only those values in **x** such that **xmin** $\leq \mathbf{x}(i) \leq$ **xmax**, for $i = 1, 2, \ldots, n$, are included in the histogram.

If **xmin** $\geq$ **xmax**, all the data are included.

5.3 Output Parameters

1: **xstep** – REAL (KIND=nag_wp)

The width of each class interval.

2: **n1** – INTEGER

The total number of observations actually included in the histogram.

3: **multy** – INTEGER

The number of observations represented by each * in the histogram.

4: **ifail** – INTEGER

ifail $= 0$ unless the function detects an error (see Section 5).

5: **nstepx** – INTEGER

If **nstepx** < 10 on entry, **nstepx** is reset to 10.

If **nstepx** > 99 on entry, **nstepx** is reset to 99.

Otherwise, **nstepx** is unchanged on exit.

6: **nstepy** – INTEGER

If **nstepy** < 10 on entry, **nstepy** is reset to 10.

If **nstepy** > 99 on entry, **nstepy** is reset to 99.

Otherwise, **nstepy** is unchanged on exit.

7: **ispace** – INTEGER

Default: 0

Unchanged unless **ispace** < 0 or **ispace** $+$ **nstepx** $+ 14 > 120$, in which case **ispace** is set to 0.

8: **xmin** – REAL (KIND=nag_wp)

9: **xmax** – REAL (KIND=nag_wp)

Default: 0

If **xmin** $<$ **xmax** on entry, then **xmin** and **xmax** are unchanged.

If **xmin** $\geq$ **xmax** then **xmin** and **xmax** contain the minimum and maximum data values respectively.

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= 1$

On entry, **n** < 1.

ifail $= -99$

An unexpected error has been triggered by this routine. Please contact NAG.

ifail $= -399$

Your licence key may have expired or may not have been installed correctly.

ifail $= -999$

Dynamic memory allocation failed.

7 Accuracy

Accuracy is limited by the number of plotting positions available.

8 Further Comments

The time taken by nag_stat_plot_histogram (g01aj) increases with **n**, **nstepx** and **nstepy**.

A total of $(\mathbf{nstepx} + \mathbf{ispace} + 14)$ character positions are used in the horizontal direction, not including the carriage control character. It is important that this total does not exceed the maximum line length available on the output device.

A total of $(\mathbf{nstepy} + 3)$ output lines are normally generated, one less if **xmin** and/or **xmax** is too large in modulus to be printed.

9 Example

A sample of 50 random numbers form the data, which correspond to a Normal distribution with mean and standard deviation both equal to 5.0. A histogram of the entire sample is drawn first, followed by a cumulative histogram of all data values lying between -10.0 and 5.0. The first histogram is indented 10 character positions. Note the use of nag_file_set_unit_advisory (x04ab) to direct the output to unit number 6.

9.1 Program Text

```
function g01aj_example

fprintf('g01aj example results\n\n');

x = [ 2.68     8.23     0.11     1.39    -2.77    -1.17      3.49     7.39 ...
      5.22     5.44     7.06     6.17    -4.44     5.98     14.02     8.06 ...
     10.96     0.99     7.87     4.85     9.76     3.3       4.45     7.02 ...
      5.43     3.57     1.98     3.09    -2.66     5.04     14.52    -0.61 ...
      5.04     6.85     3.94     4.87     3.82     5.93      7.89     9.48 ...
      6.81     6.91     7.76     6.21     0.55    -2.44     18.09     6.19 ...
      9.19    10.55];

nstepx = nag_int(10);
nstepy = nag_int(10);
ispace = nag_int(10);

[xstep, n1, multy, ifail, nstepx, nstepy, ispace, xmin, xmax] = ...
    g01aj( ...
    x, nstepx, nstepy, 'ispace', ispace);

% Display cumulative histogram and no indent
itype = nag_int(1);
ispace = nag_int(0);

% Display only data between -10 and 5
xmin = -10;
xmax = 5;

[xstep, n1, multy, ifail, nstepx, nstepy, ispace, xmin, xmax] = ...
    g01aj( ...
    x, nstepx, nstepy, 'itype', itype, 'ispace', ispace, ...
    'xmin', xmin, 'xmax', xmax);
```

9.2 Program Results

```
 g01aj example results

      FREQUENCY
         20  .          .   20
         18  .          .   18
         16  .          .   16
         14  .          .   14
         12  .    *     .   12
         10  .    **    .   10
          8  .   ***    .    8
          6  .   ***    .    6
          4  .* *****   .    4
          2  .******* * .    2
             ------------
           -4.44    18.09
```

```
 CUM. FREQ.
    30  .           .   30
    27  .           .   27
    24  .           .   24
    21  .          *.   21
    18  .          *.   18
    15  .         **.   15
    12  .         **.   12
     9  .        ***.    9
     6  .       ****.    6
     3  .     ******.    3
        ------------
      -10.00     5.00
```