nag_5pt_summary_stats (g01alc)

1. Purpose

nag_5pt_summary_stats (g01alc) calculates a five-point summary for a single sample.

2. Specification

```
#include <nag.h>
#include <nagg01.h>

void nag_5pt_summary_stats(Integer n, double x[], double res[], NagError *fail)
```

3. Description

nag_5pt_summary_stats calculates the minimum, lower hinge, median, upper hinge and the maximum of a sample of n observations.

The data consists of a single sample of n observations denoted by x_i and let z_i, for $i = 1, 2, ..., n$ represent the sample observations sorted into ascending order.

Let $m = \dfrac{n}{2}$ if n is even and $\dfrac{(n+1)}{2}$ if n is odd,

and $k = \dfrac{m}{2}$ if m is even and $\dfrac{(m+1)}{2}$ if m is odd.

Then we have:

$$
\begin{aligned}
\text{Minimum} &= z_1, \\
\text{Maximum} &= z_n, \\
\text{Median} &= z_m && \text{if } n \text{ is odd,} \\
&= \frac{z_m + z_{m+1}}{2} && \text{if } n \text{ is even,} \\
\text{Lower hinge} &= z_k && \text{if } m \text{ is odd,} \\
&= \frac{z_k + z_{k+1}}{2} && \text{if } m \text{ is even,} \\
\text{Upper hinge} &= z_{n-k+1} && \text{if } m \text{ is odd,} \\
&= \frac{z_{n-k} + z_{n-k+1}}{2} && \text{if } m \text{ is even.}
\end{aligned}
$$

4. Parameters

n

Input: the number of observations in the sample, n.
Constraint: $\mathbf{n} \geq 5$.

x[n]

Input: the sample observations, $x_1, x_2, \ldots, x_n$.

res[5]

Output: **res** contains the five-point summary as follows:

res[0] = the minimum
res[1] = the lower hinge
res[2] = the median
res[3] = the upper hinge
res[4] = the maximum

fail
The NAG error parameter, see the Essential Introduction to the NAG C Library.

5. Error Indications and Warnings

NE_INT_ARG_LT
On entry, **n** must not be less than 5: **n** = ⟨*value*⟩.

NE_ALLOC_FAIL
Memory allocation failed.

NE_INTERNAL_ERROR
An internal error has occurred in this function. Check the function call and any array sizes. If the call is correct then please consult NAG for assistance.

6. Further Comments

The time taken by the routine is proportional to n.

6.1. Accuracy

The computations are stable.

6.2. References

Erickson B H and Nosanchuk T A (1985) *Understanding Data.* Open University Press, Milton Keynes.
Tukey J W (1977) *Exploratory data analysis.* Addison-Wesley.

7. See Also

nag_summary_stats_1var (g01aac)

8. Example

The example program calulates a five-point summary for a sample of 12 observations.

8.1. Program Text

```
/* nag_5pt_summary_stats(g01alc) Example Program.
 *
 * Copyright 1996 Numerical Algorithms Group.
 *
 * Mark 4, 1996.
 *
 */

#include <nag.h>
#include <stdio.h>
#include <nag_stdlib.h>
#include <nagg01.h>

#define NMAX 12

main()
{

  double x[NMAX], res[5];

  Integer i;
  Integer n;

  Vprintf("g01alc Example Program Results\n");
  /*     Skip heading in data file */
  Vscanf("%*[^\n] ");
  Vscanf("%ld ", &n);
  for (i = 1; i <= n; ++i)
    Vscanf("%lf ", &x[i - 1]);
```

```
  g01alc(n, x, res, NAGERR_DEFAULT);

  Vprintf("\n");
  Vprintf(" Maximum     %16.4f\n", res[4]);
  Vprintf(" Upper Hinge %16.4f\n", res[3]);
  Vprintf(" Median      %16.4f\n", res[2]);
  Vprintf(" Lower Hinge %16.4f\n", res[1]);
  Vprintf(" Minimum     %16.4f\n", res[0]);
  exit(EXIT_SUCCESS);
}
```

8.2. Program Data

```
g01alc Example Program Data
12
12.0  9.0  2.0  5.0  6.0  8.0  2.0  7.0  3.0  1.0  11.0  10.0
```

8.3. Program Results

```
g01alc Example Program Results

 Maximum             12.0000
 Upper Hinge          9.5000
 Median               6.5000
 Lower Hinge          2.5000
 Minimum              1.0000
```