

## nag\_median\_1var (g07dac)

### 1. Purpose

**nag\_median\_1var (g07dac)** finds the median, median absolute deviation, and a robust estimate of the standard deviation for a set of ungrouped data.

### 2. Specification

```
#include <nag.h>
#include <nagg07.h>

void nag_median_1var(Integer n, double x[], double y[], double *xme,
                    double *xmd, double *xsd, NagError *fail)
```

### 3. Description

The data consists of a sample of size  $n$ , denoted by  $x_1, x_2, \dots, x_n$ , drawn from a random variable  $X$ . **nag\_median\_1var** first computes the median

$$\theta_{\text{med}} = \text{med}_i\{x_i\}$$

and from this the median absolute deviation can be computed,

$$\sigma_{\text{med}} = \text{med}_i\{|x_i - \theta_{\text{med}}|\}.$$

Finally, a robust estimate of the standard deviation is computed,

$$\sigma'_{\text{med}} = \sigma_{\text{med}} / \Phi^{-1}(0.75)$$

where  $\Phi^{-1}(0.75)$  is the value of the inverse standard Normal function at the point 0.75. **nag\_median\_1var** is based upon the algorithm used in the function **LTMDVV** in the **ROBETH** library, see Marazzi (1987).

### 4. Parameters

**n**

Input: the number of observations,  $n$ .  
Constraint:  $\mathbf{n} > 1$ .

**x[n]**

Input: the vector of observations,  $x_1, x_2, \dots, x_n$ .

**y[n]**

Output: the observations sorted into ascending order.

**xme**

Output: the median,  $\theta_{\text{med}}$ .

**xmd**

Output: the median absolute deviation,  $\sigma_{\text{med}}$ .

**xsd**

Output: the robust estimate of the standard deviation,  $\sigma'_{\text{med}}$ .

**fail**

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

### 5. Error Indications and Warnings

**NE\_INT\_ARG\_GT**

On entry, **n** must not be greater than  $\langle \text{value} \rangle$ :  $\mathbf{n} = \langle \text{value} \rangle$ .

**NE\_INT\_ARG\_LE**

On entry, **n** must not be less than or equal to 1: **n** = *<value>*.

**6. Further Comments**

The function may be called with the same actual array supplied for parameters **x** and **y**, in which case the sorted data values will overwrite the original contents of **x**.

**6.1. Accuracy**

The computations are believed to be stable.

**6.2. References**

Huber P J (1981) *Robust Statistics* Wiley. Marazzi A (1987) *Subroutines for Robust Estimation of Location and Scale in ROBETH* Institut Universitaire de Médecine Sociale et Préventive, Lausanne (Cah Rech Doc IUMSP, No 3 ROB 1).

**7. See Also**

None.

**8. Example**

The following program reads in a set of data consisting of eleven observations of a variable **x**. The median, median absolute deviation and a robust estimate of the standard deviation are calculated and printed along with the sorted data in output array **y**.

**8.1. Program Text**

```

/* nag_median_1var(g07dac) Example Program
 *
 * Copyright 1990 Numerical Algorithms Group.
 *
 * Mark 1, 1990.
 *
 * Mark 3 revised, 1994.
 */

#include <nag.h>
#include <stdio.h>
#include <nag_stdlib.h>
#include <nagg07.h>
#define NMAX 25

main()
{
    Integer i, n;
    double x[NMAX], y[NMAX], xmd, xme, xsd;

    Vprintf("g07dac Example Program Results\n");
    /* Skip heading in data file */
    Vscanf("%*[\n]");
    Vscanf("%ld",&n);
    if (n<=NMAX)
    {
        for (i=0; i<n; ++i)
            Vscanf("%lf",&x[i]);
        g07dac(n, x, y, &xme, &xmd, &xsd, NAGERR_DEFAULT);
        Vprintf("Output y:\n");
        for (i=0; i<n; ++i)
            Vprintf("%6.3f%s", y[i], (i%11==10 || i==n-1) ? "\n": " ");
        Vprintf("\nxme = %6.3f, xmd = %6.3f, xsd = %6.3f\n", xme, xmd, xsd);
        exit(EXIT_SUCCESS);
    }
    else

```

```
{  
    Vfprintf(stderr, "n is out of range. n = %5ld\n",n);  
    exit(EXIT_FAILURE);  
}
```

**8.2. Program Data**

g07dac Example Program Data  
11  
13.0 11.0 16.0 5.0 3.0 18.0 9.0 8.0 6.0 27.0 7.0

**8.3. Program Results**

g07dac Example Program Results  
Output y:  
3.000 5.000 6.000 7.000 8.000 9.000 11.000 13.000 16.000 18.000 27.000  
  
xme = 9.000, xmd = 4.000, xsd = 5.930

---