

# NAG Library Function Document

## nag\_median\_test (g08acc)

### 1 Purpose

nag\_median\_test (g08acc) performs the Median test on two independent samples of possibly unequal size.

### 2 Specification

```
#include <nag.h>
#include <nagg08.h>

void nag_median_test (Integer n1, const double x[], Integer n2,
    const double y[], Integer *below, Integer *above, double *p,
    NagError *fail)
```

### 3 Description

The Median test investigates the difference between the medians of two independent samples of sizes  $n_1$  and  $n_2$ , denoted by:

$$x_1, x_2, \dots, x_{n_1} \quad \text{and} \quad x_{n_1+1}, x_{n_1+2}, \dots, x_n, \quad n = n_1 + n_2.$$

The hypothesis under test,  $H_0$ , often called the null hypothesis, is that the medians are the same, and this is to be tested against the alternative hypothesis  $H_1$  that they are different.

The test proceeds by forming a  $2 \times 2$  frequency table, giving the number of scores in each sample above and below the median of the pooled sample:

	Sample 1	Sample 2	Total
Scores $\leq$ pooled median	$i_1$	$i_2$	$i_1 + i_2$
Scores $\geq$ pooled median	$n_1 - i_1$	$n_2 - i_2$	$n - (i_1 + i_2)$
Total	$n_1$	$n_2$	$n$

Under the null hypothesis,  $H_0$ , we would expect about half of each group's scores to be above the pooled median and about half below, that is, we would expect  $i_1$  to be about  $n_1/2$  and  $i_2$  to be about  $n_2/2$ .

nag\_median\_test (g08acc) returns:

- the frequencies  $i_1$  and  $i_2$ ;
- the probability,  $p$ , of observing a table at least as 'extreme' as that actually observed, given that  $H_0$  is true. If  $n < 40$ ,  $p$  is computed directly ('Fisher's exact test'); otherwise a  $\chi_1^2$  approximation is used.

$H_0$  is rejected by a test of chosen size  $\alpha$  if  $p < \alpha$ .

### 4 References

Siegel S (1956) *Non-parametric Statistics for the Behavioral Sciences* McGraw-Hill

## 5 Arguments

- 1: **n1** – Integer *Input*  
*On entry:* the size of the first sample,  $n_1$ .  
*Constraint:* **n1**  $\geq$  1.
- 2: **x[n1]** – const double *Input*  
*On entry:* the elements of **x** must be set to the data values in the first sample.
- 3: **n2** – Integer *Input*  
*On entry:* the size of the second sample,  $n_2$ .  
*Constraint:* **n2**  $\geq$  1.
- 4: **y[n2]** – const double *Input*  
*On entry:* the elements of **y** must be set to the data values in the second sample.
- 5: **below** – Integer \* *Output*  
*On exit:* the number of scores in the first sample which lie below the pooled median,  $i_1$ .
- 6: **above** – Integer \* *Output*  
*On exit:* the number of scores in the first sample which lie above the pooled median,  $i_2$ .
- 7: **p** – double \* *Output*  
*On exit:* the tail probability,  $p$ , corresponding to the observed dichotomy of the two samples.
- 8: **fail** – NagError \* *Input/Output*  
The NAG error argument (see Section 2.7 in How to Use the NAG Library and its Documentation).

## 6 Error Indicators and Warnings

### NE\_ALLOC\_FAIL

Dynamic memory allocation failed.

### NE\_INT\_ARG\_LT

On entry, **n1** must not be less than 1: **n1** =  $\langle value \rangle$ .

On entry, **n2** must not be less than 1: **n2** =  $\langle value \rangle$ .

## 7 Accuracy

The probability returned should be accurate enough for practical use.

## 8 Parallelism and Performance

nag\_median\_test (g08acc) is not threaded in any implementation.

## 9 Further Comments

The time taken by nag\_median\_test (g08acc) is small, and increases with  $n$ .

## 10 Example

This example is taken from page 112 of Siegel (1956). The data relate to scores of ‘oral socialisation anxiety’ in 39 societies, which can be separated into groups of size 16 and 23 on the basis of their attitudes to illness.

### 10.1 Program Text

```

/* nag_median_test (g08acc) Example Program.
 *
 * NAGPRODCODE Version.
 *
 * Copyright 2016 Numerical Algorithms Group.
 *
 * Mark 26, 2016.
 */

#include <stdio.h>
#include <nag.h>
#include <nag_stdlib.h>
#include <nagg08.h>

int main(void)
{
    Integer above, below, exit_status = 0, i, n1, n2;
    NagError fail;
    double p, *x = 0, *y = 0;

    INIT_FAIL(fail);

    printf("nag_median_test (g08acc) Example Program Results\n");

    /* Skip heading in data file */
#ifdef _WIN32
    scanf_s("%*[\n]");
#else
    scanf("%*[\n]");
#endif

    n1 = 16;
    n2 = 23;
    if (!(x = NAG_ALLOC(n1, double))
        || !(y = NAG_ALLOC(n2, double)))
    {
        printf("Allocation failure\n");
        exit_status = -1;
        goto END;
    }

    for (i = 1; i <= n1; ++i)
#ifdef _WIN32
        scanf_s("%lf", &x[i - 1]);
#else
        scanf("%lf", &x[i - 1]);
#endif
    for (i = 1; i <= n2; ++i)
#ifdef _WIN32
        scanf_s("%lf", &y[i - 1]);
#else
        scanf("%lf", &y[i - 1]);
#endif
    printf("\nMedian test\n\n");
    printf("Data values\n\n");
    printf("    Group 1  ");
    for (i = 1; i <= n1; ++i)
        printf("%4.0f%s", x[i - 1], i % 8 ? "" : "\n");
    printf("\n");
    printf("    Group 2  ");
    for (i = 1; i <= n2; ++i)

```

```

    printf("%4.0f%s", y[i - 1], i % 8 ? "" : "\n          ");
    printf("\n");
    /* nag_median_test (g08acc).
    * Median test on two samples of unequal size
    */
    nag_median_test(n1, x, n2, y, &above, &below, &p, &fail);
    if (fail.code != NE_NOERROR) {
        printf("Error from nag_median_test (g08acc).\n%s\n", fail.message);
        exit_status = 1;
        goto END;
    }

    printf("\n");
    printf("%6" NAG_IFMT "%s\n", above, " scores below median in group 1");
    printf("%6" NAG_IFMT "%s\n", below, " scores below median in group 2");
    printf("\n%s%8.5f\n", "          Significance ", p);
END:
    NAG_FREE(x);
    NAG_FREE(y);
    return exit_status;
}

```

## 10.2 Program Data

```

nag_median_test (g08acc) Example Program Data
 13  6 12  7 12  7 10  7 10  7 10  8  9  8
 17  6 16  8 15  8 15 10 15 10 14 10 14 11 14 11
 13 12 13 12 13 12 12

```

## 10.3 Program Results

```

nag_median_test (g08acc) Example Program Results

```

Median test

Data values

Group 1	13	6	12	7	12	7	10	7	10	7	10	8	9	8
Group 2	17	6	16	8	15	8	15	10	15	10	14	10	14	11
	13	12	13	12	13	12	12							

```

13 scores below median in group 1
 6 scores below median in group 2

```

```

Significance    0.00088

```

---