

## NAG Library Function Document

### nag\_prob\_studentized\_range (g01emc)

#### 1 Purpose

nag\_prob\_studentized\_range (g01emc) returns the probability associated with the lower tail of the distribution of the Studentized range statistic.

#### 2 Specification

```
#include <nag.h>
#include <nagg01.h>
double nag_prob_studentized_range (double q, double v, Integer ir,
    NagError *fail)
```

#### 3 Description

The externally Studentized range,  $q$ , for a sample,  $x_1, x_2, \dots, x_r$ , is defined as:

$$q = \frac{\max(x_i) - \min(x_i)}{\hat{\sigma}_e},$$

where  $\hat{\sigma}_e$  is an independent estimate of the standard error of the  $x_i$ 's. The most common use of this statistic is in the testing of means from a balanced design. In this case for a set of group means,  $\bar{T}_1, \bar{T}_2, \dots, \bar{T}_r$ , the Studentized range statistic is defined to be the difference between the largest and smallest means,  $\bar{T}_{\text{largest}}$  and  $\bar{T}_{\text{smallest}}$ , divided by the square root of the mean-square experimental error,  $MS_{\text{error}}$ , over the number of observations in each group,  $n$ , i.e.,

$$q = \frac{\bar{T}_{\text{largest}} - \bar{T}_{\text{smallest}}}{\sqrt{MS_{\text{error}}/n}}.$$

The Studentized range statistic can be used as part of a multiple comparisons procedure such as the Newman-Keuls procedure or Duncan's multiple range test (see Montgomery (1984) and Winer (1970)).

For a Studentized range statistic the probability integral,  $P(q; v, r)$ , for  $v$  degrees of freedom and  $r$  groups can be written as:

$$P(q; v, r) = C \int_0^\infty x^{v-1} e^{-vx^2/2} \left\{ r \int_{-\infty}^\infty \phi(y) [\Phi(y) - \Phi(y - qx)]^{r-1} dy \right\} dx,$$

where

$$C = \frac{v^{v/2}}{\Gamma(v/2) 2^{v/2-1}}, \quad \phi(y) = \frac{1}{\sqrt{2\pi}} e^{-y^2/2} \quad \text{and} \quad \Phi(y) = \int_{-\infty}^y \phi(t) dt.$$

The above two-dimensional integral is evaluated using numerical quadrature with the upper and lower limits computed to give stated accuracy (see Section 7).

If the degrees of freedom  $v$  are greater than 2000 the probability integral can be approximated by its asymptotic form:

$$P(q; r) = r \int_{-\infty}^\infty \phi(y) [\Phi(y) - \Phi(y - q)]^{r-1} dy.$$

This integral is evaluated using nag\_1d\_quad\_inf\_1 (d01smc).

## 4 References

Abramowitz M and Stegun I A (1972) *Handbook of Mathematical Functions* (3rd Edition) Dover Publications

Lund R E and Lund J R (1983) Algorithm AS 190: probabilities and upper quartiles for the studentized range *Appl. Statist.* **32(2)** 204–210

Montgomery D C (1984) *Design and Analysis of Experiments* Wiley

Winer B J (1970) *Statistical Principles in Experimental Design* McGraw–Hill

## 5 Arguments

- 1: **q** – double *Input*  
*On entry:*  $q$ , the Studentized range statistic.  
*Constraint:*  $q > 0.0$ .
- 2: **v** – double *Input*  
*On entry:*  $v$ , the number of degrees of freedom for the experimental error.  
*Constraint:*  $v \geq 1.0$ .
- 3: **ir** – Integer *Input*  
*On entry:*  $r$ , the number of groups.  
*Constraint:*  $ir \geq 2$ .
- 4: **fail** – NagError \* *Input/Output*  
 The NAG error argument (see Section 2.7 in How to Use the NAG Library and its Documentation).  
 If on exit **fail.code** = NE\_INT or NE\_REAL, then nag\_prob\_studentized\_range (g01emc) returns to 0.0.

## 6 Error Indicators and Warnings

### NE\_ACCURACY

Warning – There is some doubt as to whether full accuracy has been achieved.

### NE\_ALLOC\_FAIL

Dynamic memory allocation failed.

See Section 3.2.1.2 in How to Use the NAG Library and its Documentation for further information.

### NE\_INT

On entry, **ir** =  $\langle value \rangle$ .

Constraint:  $ir \geq 2$ .

### 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 contact NAG for assistance.

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

See Section 3.6.6 in How to Use the NAG Library and its Documentation for further information.

**NE\_NO\_LICENCE**

Your licence key may have expired or may not have been installed correctly.  
See Section 3.6.5 in How to Use the NAG Library and its Documentation for further information.

**NE\_REAL**

On entry,  $q = \langle value \rangle$ .  
Constraint:  $q > 0.0$ .

On entry,  $v = \langle value \rangle$ .  
Constraint:  $v \geq 1.0$ .

**7 Accuracy**

The returned value will have absolute accuracy to at least four decimal places (usually five), unless **fail.code** = NE\_ACCURACY. When **fail.code** = NE\_ACCURACY it is usual that the returned value will be a good estimate of the true value.

**8 Parallelism and Performance**

nag\_prob\_studentized\_range (g01emc) is threaded by NAG for parallel execution in multithreaded implementations of the NAG Library.

Please consult the x06 Chapter Introduction for information on how to control and interrogate the OpenMP environment used within this function. Please also consult the Users' Note for your implementation for any additional implementation-specific information.

**9 Further Comments**

None.

**10 Example**

The lower tail probabilities for the distribution of the Studentized range statistic are computed and printed for a range of values of  $q$ ,  $v$  and  $r$ .

**10.1 Program Text**

```

/* nag_prob_studentized_range (g01emc) Example Program.
 *
 * NAGPRODCODE Version.
 *
 * Copyright 2016 Numerical Algorithms Group.
 *
 * Mark 26, 2016.
 */

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

int main(void)
{
    /* Scalars */
    Integer exit_status = 0;
    double q, v, valp;
    Integer i, ir;
    NagError fail;

    INIT_FAIL(fail);

    printf("nag_prob_studentized_range (g01emc) Example Program Results\n");

```

```

/* Skip heading in data file */
#ifdef _WIN32
scanf_s("%*[\n] ");
#else
scanf("%*[\n] ");
#endif
printf("\n%s\n\n", " q      v      ir      Quantile ");
for (i = 1; i <= 3; ++i) {
#ifdef _WIN32
scanf_s("%lf%lf%" NAG_IFMT "%*[\n] ", &q, &v, &ir);
#else
scanf("%lf%lf%" NAG_IFMT "%*[\n] ", &q, &v, &ir);
#endif
/* nag_prob_studentized_range (g01emc).
 * Computes probability for the Studentized range statistic
 */
valp = nag_prob_studentized_range(q, v, ir, &fail);
if (fail.code != NE_NOERROR) {
printf("Error from nag_prob_studentized_range (g01emc).\n%s\n",
fail.message);
exit_status = 1;
goto END;
}
printf("%7.4f%2s%4.1f%1s%3" NAG_IFMT "%1s%10.4f\n", q, "",
v, "", ir, "", valp);
}
}

END:
return exit_status;
}

```

## 10.2 Program Data

nag\_prob\_studentized\_range (g01emc) Example Program Data

4.6543	10.0	5
2.8099	60.0	12
4.2636	5.0	4

## 10.3 Program Results

nag\_prob\_studentized\_range (g01emc) Example Program Results

q	v	ir	Quantile
4.6543	10.0	5	0.9500
2.8099	60.0	12	0.3000
4.2636	5.0	4	0.9000

---