

nag_prob_normal (g01eac)

1. Purpose

nag_prob_normal (g01eac) returns a one or two tail probability for the standard distribution.

2. Specification

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

```
double nag_prob_normal(Nag_TailProbability tail, double x, NagError *fail)
```

3. Description

The lower tail probability for the standard Normal distribution, $P(X \leq x)$ is defined by:

$$P(X \leq x) = \int_{-\infty}^x Z(X) dX$$

where

$$Z(X) = \frac{1}{\sqrt{2\pi}} e^{-X^2/2}, \quad -\infty < X < \infty.$$

The relationship

$$P(X \leq x) = \frac{1}{2} \operatorname{erfc} \left(-\frac{x}{\sqrt{2}} \right)$$

is used, where `erfc` is the complementary error function, and is computed using `nag_erfc (s15adc)`. For the upper tail probability the relationship $P(X \geq x) = P(X \leq -x)$ is used and for the two tail significance level probability twice the probability obtained from the absolute value of x is returned. When the two tail confidence probability is required the relationship

$$P(X \leq |x|) - P(X \leq -|x|) = \operatorname{erf} \left(\frac{|x|}{\sqrt{2}} \right),$$

is used, where `erf` is the error function, and is computed using `nag_erf (s15aec)`.

4. Parameters

tail

Input: indicates whether the upper or lower tail probability is required.

If **tail** = **Nag_LowerTail**, the lower tail probability is returned, i.e., $P(X \leq x)$.

If **tail** = **Nag_UpperTail**, the upper tail probability is returned, i.e., $P(X \geq x)$.

If **tail** = **Nag_TwoTailSignif**, the two tail (significance level) probability is returned, i.e., $P(X \geq |x|) + P(X \leq -|x|)$.

If **tail** = **Nag_TwoTailConfid**, the two tail (confidence interval) probability is returned, i.e., $P(X \leq |x|) - P(X \leq -|x|)$.

Constraint: **tail** = **Nag_UpperTail**, **Nag_LowerTail**, **Nag_TwoTailSignif** or **Nag_TwoTailConfid**.

x

Input: the value of the standard Normal variate, x .

fail

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

5. Error Indications and Warnings

If **fail.code** $\neq$ **NE_NOERROR**, then `nag_prob_normal` returns 0.0.

NE_BAD_PARAM

On entry, parameter **tail** had an illegal value.

6. Further Comments**6.1. Accuracy**

Accuracy is limited by *machine precision*. For detailed error analysis see nag_erfc (s15adc) and nag_erf (s15aec).

6.2. References

Abramowitz M and Stegun I A (1972) *Handbook of Mathematical Functions* Ch. 7.1, p.297 and Ch. 26.2, p. 931 Dover Publications, New York.

Hastings N A J and Peacock J B (1975) *Statistical Distributions* Butterworth.

7. See Also

nag_erfc (s15adc)

nag_erf (s15aec)

8. Example

Four values of **tail** and **x** are input and the probabilities calculated and printed.

8.1. Program Text

```
/* nag_prob_normal(g01eac) Example Program.
 *
 * Copyright 1996 Numerical Algorithms Group.
 *
 * Mark 4, 1996.
 *
 */

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

main()
{
    double prob;
    double x;

    Integer i;

    Nag_TailProbability tail;
    char tail_char;

    Vprintf(" g01eac Example Program Results\n");

    /*      Skip heading in data file */
    Vscanf("%*[^\\n] ");

    Vprintf("\n Tail      X      Probability \\n\\n");
    for (i = 1; i <= 4; ++i)
    {
        Vscanf("%c %lf ", &tail_char, &x);

        switch (tail_char)
        {
            case 'L':
                tail=Nag_LowerTail;
                break;
            case 'U':
```

```

        tail=Nag_UpperTail;
        break;
    case 'C':
        tail=Nag_TwoTailConfid;
        break;
    case 'S':
        tail=Nag_TwoTailSignif;
    }

    prob = g01eac(tail, x, NAGERR_DEFAULT);

    Vprintf("    %c    %4.2f    %6.4f\n", tail_char, x, prob);
}
exit(EXIT_SUCCESS);
}

```

8.2. Program Data

g01eac Example Program Data

L	1.96
U	1.96
C	1.96
S	1.96

8.3. Program Results

g01eac Example Program Results

Tail	X	Probability
L	1.96	0.9750
U	1.96	0.0250
C	1.96	0.9500
S	1.96	0.0500
