NAG C Library Function Document

nag_mills_ratio (g01mbc)

1 Purpose

nag_mills_ratio (g01mbc) returns the reciprocal of Mills’ Ratio.

2 Specification

double nag_mills_ratio (double x)

3 Description

nag_mills_ratio (g01mbc) calculates the reciprocal of Mills’ Ratio, the hazard rate, \( \lambda(x) \), for the standard Normal distribution. It is defined as the ratio of the ordinate to the upper tail area of the standard Normal distribution, that is,

\[
\lambda(x) = \frac{Z(x)}{Q(x)} = \frac{1}{\sqrt{2\pi}} \frac{e^{-(x^2/2)}}{\int_x^\infty e^{-(t^2/2)} dt}.
\]

If \( x \leq 9.0 \), then the calculation is based on a Chebyshev expansion as described in nag_erfc (s15adc); otherwise, the method due to Swan (1969) is used.

4 References


5 Parameters

1: x – double

On entry: the argument of the reciprocal of Mills’ Ratio, \( x \).

6 Error Indicators and Warnings

None.

7 Accuracy

In the left-hand tail, \( x < 0.0 \), if \( \frac{1}{2} e^{-(1/2)x^2} \leq \) the safe range parameter (nag_real_safe_small_number (X02AMC)), then 0.0 is returned, which is close to the true value.

The relative accuracy is bounded by the effective machine precision. See nag_erfc (s15adc) for further discussion for the case \( x \leq 9.0 \).

8 Further Comments

If, before entry, \( x \) is not a standard Normal variable, it has to be standardized, and on exit, nag_mills_ratio (g01mbc) has to be divided by the standard deviation. That is, if the Normal distribution has mean \( \mu \) and variance \( \sigma^2 \), then its hazard rate, \( \lambda(x; \mu, \sigma^2) \), is given by

\[
\lambda(x; \mu, \sigma^2) = \lambda((x - \mu)/\sigma)/\sigma.
\]
9 Example

The hazard rate is evaluated at different values of $x$ for Normal distributions with different means and variances. The results are then printed.

9.1 Program Text

```c
/* nag_mills_ratio (g01mbc) Example Program.
 * Copyright 2001 Numerical Algorithms Group.
 */
#include <stdio.h>
#include <nag.h>
#include <nag_stdlib.h>
#include <nagg01.h>

int main(void)
{
    /* Scalars */
    double rm, x, xmu, xsig, z__;
    Integer exit_status, i;
    exit_status = 0;
    Vprintf("g01mbc Example Program Results
");
    /* Skip heading in data file */
    Vscanf("%*[^
] ");
    Vprintf("%2sMean%5sSigma%4sX%8sReciprocal", "", "", "", "");
    Vprintf("Mills Ratio
");
    for (i = 1; i <= 3; ++i)
    {
        Vscanf("%lf%lf%lf%*[\n] ", &x, &xmu, &xsig);
        z__ = (x - xmu) / xsig;
        rm = g01mbc(z__) / xsig;
        Vprintf("%7.4f%2s%7.4f%2s%7.4f%2s%7.4f", xmu, "", xsig, "", x, "", rm);
    }
    Vprintf("\n");
    return exit_status;
}
```

9.2 Program Data

<table>
<thead>
<tr>
<th>g01mbc Example Program Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0 0.0 1.0</td>
</tr>
<tr>
<td>-2.0 1.0 2.5</td>
</tr>
<tr>
<td>10.3 9.0 1.6</td>
</tr>
</tbody>
</table>

9.3 Program Results

<table>
<thead>
<tr>
<th>g01mbc Example Program Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
</tr>
<tr>
<td>-------</td>
</tr>
<tr>
<td>0.0000</td>
</tr>
<tr>
<td>1.0000</td>
</tr>
<tr>
<td>9.0000</td>
</tr>
</tbody>
</table>