NAG Library Function Document

nag_cumul_normal (s15abc)

1 Purpose

nag_cumul_normal (s15abc) returns the value of the cumulative Normal distribution function, \( P(x) \).

2 Specification

```c
#include <nag.h>
#include <nags.h>
double nag_cumul_normal (double x)
```

3 Description

nag_cumul_normal (s15abc) evaluates an approximate value for the cumulative Normal distribution function

\[
P(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-u^2/2} \, du.
\]

The function is based on the fact that

\[
P(x) = \frac{1}{2} \text{erfc} \left( \frac{x}{\sqrt{2}} \right)
\]

and it calls nag_erfc (s15adc) to obtain a value of \( \text{erfc} \) for the appropriate argument.

4 References


5 Arguments

1:  \( x \) – double

On entry: the argument \( x \) of the function.

6 Error Indicators and Warnings

None.

7 Accuracy

Because of its close relationship with \( \text{erfc} \), the accuracy of this function is very similar to that in nag_erfc (s15adc). If \( \epsilon \) and \( \delta \) are the relative errors in result and argument, respectively, they are in principle related by

\[
|\epsilon| \simeq \left| \frac{x e^{-\frac{1}{2}x^2}}{\sqrt{2\pi} P(x)} \delta \right|
\]

so that the relative error in the argument, \( x \), is amplified by a factor \( \frac{x e^{-\frac{1}{2}x^2}}{\sqrt{2\pi} P(x)} \), in the result.
For $x$ small and for $x$ positive this factor is always less than one and accuracy is mainly limited by machine precision.

For large negative $x$ the factor behaves like $\sim x^2$ and hence to a certain extent relative accuracy is unavoidably lost.

However the absolute error in the result, $E$, is given by

\[ |E| \approx \left| \frac{x e^{-\frac{1}{2}x^2}}{\sqrt{2\pi}e} \right| \]

so absolute accuracy can be guaranteed for all $x$.

8 Parallelism and Performance

Not applicable.

9 Further Comments

None.

10 Example

This example reads values of the argument $x$ from a file, evaluates the function at each value of $x$ and prints the results.

10.1 Program Text

/* nag_cumul_normal (s15abc) Example Program.
   * Copyright 2014 Numerical Algorithms Group.
   * Mark 1, 1990.
   */
#include <nag.h>
#include <stdio.h>
#include <nag_stdlib.h>
#include <nags.h>
int main(void)
{
   Integer exit_status = 0;
   double x, y;

   /* Skip heading in data file */
#ifdef _WIN32
   scanf_s("%*[\n]");
#else
   scanf("%*[\n]");
#endif
   printf("nag_cumul_normal (s15abc) Example Program Results\n");
   printf(" x y\n");
#ifdef _WIN32
   while (scanf_s("%lf", &x) != EOF)
#else
   while (scanf("%lf", &x) != EOF)
#endif
{
   /* nag_cumul_normal (s15abc).
      * Cumulative Normal distribution function $P(x)$
      */
   y = nag_cumul_normal(x);
printf("%2.3e%2.3e\n", x, y);
}
return exit_status;
}

10.2 Program Data

nag_cumul_normal (s15abc) Example Program Data
  -20.0  
  -1.0  
   0.0  
   1.0  
   2.0  
  20.0

10.3 Program Results

nag_cumul_normal (s15abc) Example Program Results
  x    y
-2.000e+01  2.754e-89  
-1.000e+00  1.587e-01  
  0.000e+00  5.000e-01  
  1.000e+00  8.413e-01  
  2.000e+00  9.772e-01  
  2.000e+01  1.000e+00  