D01BAF – NAG Fortran Library Routine Document

Note. Before using this routine, please read the Users' Note for your implementation to check the interpretation of bold italicised terms and other implementation-dependent details.

1 Purpose

D01BAF computes an estimate of the definite integral of a function of known analytical form, using a Gaussian quadrature formula with a specified number of abscissae. Formulae are provided for a finite interval (Gauss–Legendre), a semi-infinite interval (Gauss–Laguerre, Gauss–Rational), and an infinite interval (Gauss–Hermite).

2 Specification

```
real FUNCTION D01BAF(D01XXX, A, B, N, FUN, IFAIL)
INTEGER                N, IFAIL
real                   A, B, FUN
EXTERNAL               D01XXX, FUN
```

3 Description

3.1 General

This routine evaluates an estimate of the definite integral of a function $f(x)$, over a finite or infinite range, by n-point Gaussian quadrature (see Davis and Rabinowitz [1], Froberg [2], Ralston [3] or Stroud and Secrest [4]). The integral is approximated by a summation

$$\sum_{i=1}^{n} w_i \, f(x_i)$$

where the w_i are called the weights, and the x_i the abscissae. A selection of values of n is available. (See Section 5.)

3.2 Both Limits Finite

$$\int_a^b f(x)\,dx.$$

The Gauss–Legendre weights and abscissae are used, and the formula is exact for any function of the form:

$$f(x) = \sum_{i=0}^{2n-1} c_i x^i.$$

The formula is appropriate for functions which can be well approximated by such a polynomial over $[a,b]$. It is inappropriate for functions with algebraic singularities at one or both ends of the interval, such as $(1+x)^{-1/2}$ on $[-1,1]$.

3.3 One Limit Infinite

$$\int_a^\infty f(x)\,dx \quad \text{or} \quad \int_{-\infty}^a f(x)\,dx.$$

Two quadrature formulae are available for these integrals.

(a) The Gauss–Laguerre formula is exact for any function of the form:

$$f(x) = e^{-bx} \sum_{i=0}^{2n-1} c_i x^i.$$

This formula is appropriate for functions decaying exponentially at infinity; the parameter b should be chosen if possible to match the decay rate of the function.

(b) The Gauss–Rational formula is exact for any function of the form:

$$f(x)=\sum_{i=2}^{2n+1}\frac{c_i}{(x+b)^i}=\frac{\sum_{i=0}^{2n-1}c_{2n+1-i}(x+b)^i}{(x+b)^{2n+1}}.$$

This formula is likely to be more accurate for functions having only an inverse power rate of decay for large x. Here the choice of a suitable value of b may be more difficult; unfortunately a poor choice of b can make a large difference to the accuracy of the computed integral.

3.4 Both Limits Infinite

$$\int_{-\infty}^{+\infty}f(x)\,dx.$$

The Gauss–Hermite weights and abscissae are used, and the formula is exact for any function of the form:

$$f(x)=e^{-b(x-a)^2}\sum_{i=0}^{2n-1}c_ix^i.$$

Again, for general functions not of this exact form, the parameter b should be chosen to match if possible the decay rate at $\pm\infty$.

4 References

[1] Davis P J and Rabinowitz P (1975) *Methods of Numerical Integration* Academic Press

[2] Froberg C E (1970) *Introduction to Numerical Analysis* Addison–Wesley

[3] Ralston A (1965) *A First Course in Numerical Analysis* McGraw–Hill 87–90

[4] Stroud A H and Secrest D (1966) *Gaussian Quadrature Formulas* Prentice–Hall

5 Parameters

1: D01XXX — SUBROUTINE, supplied by the NAG Fortran Library. *External Procedure*

The name of the routine indicates the quadrature formula:

D01BAZ, for Gauss–Legendre quadrature on a finite interval;
D01BAY, for Gauss–Rational quadrature on a semi-infinite interval;
D01BAX, for Gauss–Laguerre quadrature on a semi-infinite interval;
D01BAW, for Gauss–Hermite quadrature on an infinite interval.

2: A — ***real*** *Input*

3: B — ***real*** *Input*

On entry: the parameters a and b which occur in the integration formulae:

Gauss–Legendre:
a is the lower limit and b is the upper limit of the integral. It is not necessary that $a<b$.

Gauss–Rational:
b must be chosen so as to make the integrand match as closely as possible the exact form given in Section 3.3(b). The range of integration is $[a,\infty)$ if $a+b>0$, and $(-\infty,a]$ if $a+b<0$.

Gauss–Laguerre:
b must be chosen so as to make the integrand match as closely as possible the exact form given in Section 3.3(a). The range of integration is $[a,\infty)$ if $b>0$, and $(-\infty,a]$ is $b<0$.

Gauss–Hermite:
a and b must be chosen so as to make the integrand match as closely as possible the exact form given in Section 3.4.

Constraints:

Gauss–Rational: $A + B \neq 0$,
Gauss–Laguerre: $B \neq 0$,
Gauss–Hermite: $B > 0$.

4: N — INTEGER *Input*

On entry: the number of abscissae to be used, n.

Constraint: N = 1,2,3,4,5,6,8,10,12,14,16,20,24,32,48 or 64.

5: FUN — ***real*** FUNCTION, supplied by the user. *External Procedure*

FUN must return the value of the integrand f at a given point.

Its specification is:

```
real FUNCTION FUN(X)
real             X
```

1: X — ***real*** *Input*

On entry: the point at which the integrand must be evaluated.

Some points to bear in mind when coding FUN are mentioned in Section 7.
FUN must be declared as EXTERNAL in the (sub)program from which D01BAF is called. Parameters denoted as *Input* must **not** be changed by this procedure.

6: IFAIL — INTEGER *Input/Output*

On entry: IFAIL must be set to 0, −1 or 1. Users who are unfamiliar with this parameter should refer to Chapter P01 for details.

On exit: IFAIL = 0 unless the routine detects an error or gives a warning (see Section 6).

For this routine, because the values of output parameters may be useful even if IFAIL ≠ 0 on exit, users are recommended to set IFAIL to −1 before entry. **It is then essential to test the value of IFAIL on exit**. To suppress the output of an error message when soft failure occurs, set IFAIL to 1.

6 Error Indicators and Warnings

Errors or warnings specified by the routine:

IFAIL = 1

The N-point rule is not among those stored. If the soft fail option is used, the answer is evaluated for the largest valid value of N less than the requested value.

IFAIL = 2

The value of A and/or B is invalid.

Gauss–Rational: $A + B = 0$.

Gauss–Laguerre: $B = 0$.

Gauss–Hermite: $B \leq 0$.

If the soft fail option is used, the answer is returned as zero.

7 Accuracy

The accuracy depends on the behaviour of the integrand, and on the number of abscissae used. No tests are carried out in the routine to estimate the accuracy of the result. If such an estimate is required, the routine may be called more than once, with a different number of abscissae each time, and the answers compared. It is to be expected that for sufficiently smooth functions a larger number of abscissae will give improved accuracy.

Alternatively, the range of integration may be subdivided, the integral estimated separately for each sub-interval, and the sum of these estimates compared with the estimate over the whole range.

The coding of the function FUN may also have a bearing on the accuracy. For example, if a high-order Gauss–Laguerre formula is used, and the integrand is of the form

$$f(x) = e^{-bx} g(x)$$

it is possible that the exponential term may underflow for some large abscissae. Depending on the machine, this may produce an error, or simply be assumed to be zero. In any case, it would be better to evaluate the expression as:

$$f(x) = \exp(-bx + \ln g(x))$$

Another situation requiring care is exemplified by

$$\int_{-\infty}^{+\infty} e^{-x^2} x^m \, dx = 0, \quad m \text{ odd.}$$

The integrand here assumes very large values; for example, for $m = 63$, the peak value exceeds 3×10^{33}. Now, if the machine holds floating-point numbers to an accuracy of k significant decimal digits, we could not expect such terms to cancel in the summation leaving an answer of much less than 10^{33-k} (the weights being of order unity); that is instead of zero, we obtain a rather large answer through rounding error. Fortunately, such situations are characterised by great variability in the answers returned by formulae with different values of n. In general, the user should be aware of the order of magnitude of the integrand, and should judge the answer in that light.

8 Further Comments

The time taken by the routine depends on the complexity of the expression for the integrand and on the number of abscissae required.

9 Example

This example program evaluates the integrals

$$\int_0^1 \frac{4}{1+x^2} \, dx = \pi$$

by Gauss–Legendre quadrature;

$$\int_2^{\infty} \frac{1}{x^2 \ln x} \, dx = 0.378671$$

by Gauss–Rational quadrature with $b = 0$;

$$\int_2^{\infty} \frac{e^{-x}}{x} \, dx = 0.048901$$

by Gauss–Laguerre quadrature with $b = 1$; and

$$\int_{-\infty}^{+\infty} e^{-3x^2-4x-1} \, dx = \int_{-\infty}^{+\infty} e^{-3(x+1)^2} e^{2x+2} \, dx = 1.428167$$

by Gauss–Hermite quadrature with $a = -1$ and $b = 3$.

The formulae with $n = 4, 8, 16$ are used in each case.

9.1 Program Text

Note. The listing of the example program presented below uses bold italicised terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     D01BAF Example Program Text
*     Mark 14 Revised.  NAG Copyright 1989.
*     .. Parameters ..
      INTEGER          NOUT
      PARAMETER        (NOUT=6)
*     .. Local Scalars ..
      real             A, ANS, B
      INTEGER          I, IFAIL
*     .. Local Arrays ..
      INTEGER          NSTOR(3)
*     .. External Functions ..
      real             D01BAF, FUN1, FUN2, FUN3, FUN4
      EXTERNAL         D01BAF, FUN1, FUN2, FUN3, FUN4
*     .. External Subroutines ..
      EXTERNAL         D01BAW, D01BAX, D01BAY, D01BAZ
*     .. Data statements ..
      DATA             NSTOR/4, 8, 16/
*     .. Executable Statements ..
      WRITE (NOUT,*) 'D01BAF Example Program Results'
      WRITE (NOUT,*)
      WRITE (NOUT,*) 'Gauss-Legendre example'
      DO 20 I = 1, 3
         A = 0.0e0
         B = 1.0e0
         IFAIL = 1
*
         ANS = D01BAF(D01BAZ,A,B,NSTOR(I),FUN1,IFAIL)
*
         IF (IFAIL.NE.0) THEN
            WRITE (NOUT,99998) 'IFAIL = ', IFAIL
            WRITE (NOUT,*)
         END IF
         IF (IFAIL.LE.1) WRITE (NOUT,99999) NSTOR(I),
     +       ' Points     Answer = ', ANS
   20 CONTINUE
      WRITE (NOUT,*)
      WRITE (NOUT,*)
      WRITE (NOUT,*) 'Gauss-Rational example'
      DO 40 I = 1, 3
         A = 2.0e0
         B = 0.0e0
         IFAIL = 1
*
         ANS = D01BAF(D01BAY,A,B,NSTOR(I),FUN2,IFAIL)
*
         IF (IFAIL.NE.0) THEN
            WRITE (NOUT,99998) 'IFAIL = ', IFAIL
            WRITE (NOUT,*)
         END IF
         IF (IFAIL.LE.1) WRITE (NOUT,99999) NSTOR(I),
     +       ' Points     Answer = ', ANS
   40 CONTINUE
      WRITE (NOUT,*)
      WRITE (NOUT,*)
```

```
      WRITE (NOUT,*) 'Gauss-Laguerre example'
      DO 60 I = 1, 3
         IFAIL = 1
         A = 2.0e0
         B = 1.0e0
*
         ANS = D01BAF(D01BAX,A,B,NSTOR(I),FUN3,IFAIL)
*
         IF (IFAIL.NE.0) THEN
            WRITE (NOUT,99998) 'IFAIL = ', IFAIL
            WRITE (NOUT,*)
         END IF
         IF (IFAIL.LE.1) WRITE (NOUT,99999) NSTOR(I),
     +       ' Points     Answer = ', ANS
   60 CONTINUE
      WRITE (NOUT,*)
      WRITE (NOUT,*)
      WRITE (NOUT,*) 'Gauss-Hermite  example'
      DO 80 I = 1, 3
         A = -1.0e0
         B = 3.0e0
         IFAIL = 1
*
         ANS = D01BAF(D01BAW,A,B,NSTOR(I),FUN4,IFAIL)
*
         IF (IFAIL.NE.0) THEN
            WRITE (NOUT,99998) 'IFAIL = ', IFAIL
            WRITE (NOUT,*)
         END IF
         IF (IFAIL.LE.1) WRITE (NOUT,99999) NSTOR(I),
     +       ' Points     Answer = ', ANS
   80 CONTINUE
      STOP
*
99999 FORMAT (1X,I5,A,F10.5)
99998 FORMAT (1X,A,I2)
      END
*
      real FUNCTION FUN1(X)
*     .. Scalar Arguments ..
      real                X
*     .. Executable Statements ..
      FUN1 = 4.0e0/(1.0e0+X*X)
      RETURN
      END
*
      real FUNCTION FUN2(X)
*     .. Scalar Arguments ..
      real                X
*     .. Intrinsic Functions ..
      INTRINSIC           LOG
*     .. Executable Statements ..
      FUN2 = 1.0e0/(X*X*LOG(X))
      RETURN
      END
*
      real FUNCTION FUN3(X)
```

```
*     .. Scalar Arguments ..
      real              X
*     .. Intrinsic Functions ..
      INTRINSIC         EXP
*     .. Executable Statements ..
      FUN3 = EXP(-X)/X
      RETURN
      END
*
      real FUNCTION FUN4(X)
*     .. Scalar Arguments ..
      real              X
*     .. Intrinsic Functions ..
      INTRINSIC         EXP
*     .. Executable Statements ..
      FUN4 = EXP(-3.0e0*X*X-4.0e0*X-1.0e0)
      RETURN
      END
```

9.2 Program Data

None.

9.3 Program Results

```
 D01BAF Example Program Results

 Gauss-Legendre example
     4 Points     Answer =    3.14161
     8 Points     Answer =    3.14159
    16 Points     Answer =    3.14159


 Gauss-Rational example
     4 Points     Answer =    0.37910
     8 Points     Answer =    0.37876
    16 Points     Answer =    0.37869


 Gauss-Laguerre example
     4 Points     Answer =    0.04887
     8 Points     Answer =    0.04890
    16 Points     Answer =    0.04890


 Gauss-Hermite  example
     4 Points     Answer =    1.42803
     8 Points     Answer =    1.42817
    16 Points     Answer =    1.42817
```