NAG Library Function Document

nag_deviates_f_dist (g01fdc)

1 Purpose

nag_deviates_f_dist (g01fdc) returns the deviate associated with the given lower tail probability of the F or variance-ratio distribution with real degrees of freedom.

2 Specification

```
#include <nag.h>
#include <nagg01.h>
double nag_deviates_f_dist (double p, double df1, double df2, NagError *fail)
```

3 Description

The deviate, f_p, associated with the lower tail probability, p, of the F-distribution with degrees of freedom ν_1 and ν_2 is defined as the solution to

$$P\left(F \leq f_p : \nu_1, \nu_2\right) = p = \frac{\nu_1^{\frac{1}{2}\nu_1}\nu_2^{\frac{1}{2}\nu_2}\Gamma\left(\frac{\nu_1+\nu_2}{2}\right)}{\Gamma\left(\frac{\nu_1}{2}\right)\Gamma\left(\frac{\nu_2}{2}\right)}\int_0^{f_p} F^{\frac{1}{2}(\nu_1-2)}\left(\nu_2+\nu_1 F\right)^{-\frac{1}{2}(\nu_1+\nu_2)}\,dF,$$

where $\nu_1, \nu_2 > 0$; $0 \leq f_p < \infty$.

The value of f_p is computed by means of a transformation to a beta distribution, $P_\beta(B \leq \beta : a, b)$:

$$P(F \leq f : \nu_1, \nu_2) = P_\beta\left(B \leq \frac{\nu_1 f}{\nu_1 f + \nu_2} : \nu_1/2, \nu_2/2\right)$$

and using a call to nag_deviates_beta (g01fec).

For very large values of both ν_1 and ν_2, greater than 10^5, a normal approximation is used. If only one of ν_1 or ν_2 is greater than 10^5 then a χ^2 approximation is used; see Abramowitz and Stegun (1972).

4 References

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

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

5 Arguments

1: **p** – double *Input*

On entry: p, the lower tail probability from the required F-distribution.

Constraint: $0.0 \leq \mathbf{p} < 1.0$.

2: **df1** – double *Input*

On entry: the degrees of freedom of the numerator variance, ν_1.

Constraint: **df1** > 0.0.

3: **df2** – double *Input*

On entry: the degrees of freedom of the denominator variance, ν_2.

Constraint: **df2** > 0.0.

4: **fail** – NagError * *Input/Output*

The NAG error argument (see Section 3.6 in the Essential Introduction).

6 Error Indicators and Warnings

On any of the error conditions listed below except **fail.code** = NE_SOL_NOT_CONV nag_deviates_f_dist (g01fdc) returns 0.0.

NE_ALLOC_FAIL

Dynamic memory allocation failed.

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.

NE_PROBAB_CLOSE_TO_TAIL

The probability is too close to 0.0 or 1.0. The value of f_p cannot be computed. This will only occur when the large sample approximations are used.

NE_REAL_ARG_GE

On entry, **p** = ⟨*value*⟩.
Constraint: **p** < 1.0.

NE_REAL_ARG_LE

On entry, **df1** = ⟨*value*⟩ and **df2** = ⟨*value*⟩.
Constraint: **df1** > 0.0 and **df2** > 0.0.

NE_REAL_ARG_LT

On entry, **p** = ⟨*value*⟩.
Constraint: **p** ≥ 0.0.

NE_SOL_NOT_CONV

The solution has failed to converge. However, the result should be a reasonable approximation. Alternatively, nag_deviates_beta (g01fec) can be used with a suitable setting of the argument **tol**.

7 Accuracy

The result should be accurate to five significant digits.

8 Parallelism and Performance

Not applicable.

9 Further Comments

For higher accuracy nag_deviates_beta (g01fec) can be used along with the transformations given in Section 3.

10 Example

This example reads the lower tail probabilities for several F-distributions, and calculates and prints the corresponding deviates until the end of data is reached.

10.1 Program Text

```
/* nag_deviates_f_dist (g01fdc) Example Program.
 *
 * Copyright 1990 Numerical Algorithms Group.
 *
 * Mark 1, 1990.
 */

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

int main(void)
{
  Integer  exit_status = 0;
  double   df1, df2, f, p;
  NagError fail;

  INIT_FAIL(fail);

  /* Skip heading in data file */
  scanf("%*[^\n]");
  printf("nag_deviates_f_dist (g01fdc) Example Program Results\n");
  printf("     p      df1     df2      f\n\n");
  while (scanf("%lf %lf %lf", &p, &df1, &df2) != EOF)
    {
      /* nag_deviates_f_dist (g01fdc).
       * Deviates for the F-distribution
       */
      f = nag_deviates_f_dist(p, df1, df2, &fail);
      if (fail.code != NE_NOERROR)
        {
          printf("Error from nag_deviates_f_dist (g01fdc).\n%s\n",
                  fail.message);
          exit_status = 1;
          goto END;
        }
      printf("%8.3f%8.3f%8.3f%8.3f\n", p, df1, df2, f);
    }

 END:
  return exit_status;
}
```

10.2 Program Data

```
nag_deviates_f_dist (g01fdc) Example Program Data
0.9837  10.0   25.5
0.9000  1.0    1.0
0.5342  20.25  1.0
```

10.3 Program Results

```
nag_deviates_f_dist (g01fdc) Example Program Results
     p      df1     df2      f

   0.984  10.000  25.500   2.837
   0.900   1.000   1.000  39.863
   0.534  20.250   1.000   2.500
```