nag_deviates_students_t (g01fbc)

1. Purpose

nag_deviates_students_t (g01fbc) returns the deviate associated with the given tail probability of Student's t-distribution with real degrees of freedom.

2. Specification

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

double nag_deviates_students_t(Nag_TailProbability tail, double p, double df,
       NagError *fail)
```

3. Description

The deviate, t_p associated with the lower tail probability, p, of the Student's t-distribution with ν degrees of freedom is defined as the solution to

$$P(T \leq t_p : \nu) = p = \frac{\Gamma((\nu+1)/2)}{\sqrt{\nu\pi}\Gamma(\nu/2)} \int_{-\infty}^{t_p} \left(1 + \frac{T^2}{\nu}\right)^{-(\nu+1)/2} dT \qquad \nu \geq 1;\ -\infty < t_p < \infty.$$

For $\nu = 1$ or 2 the integral equation is easily solved for t_p.

For other values of $\nu < 3$ a transformation to the beta distribution is used and the result obtained from nag_deviates_beta (g01fec).

For $\nu \geq 3$ an inverse asymptotic expansion of Cornish–Fisher type is used. The algorithm is described by Hill (1970).

4. Parameters

tail

Input: indicates which tail the supplied probability represents.
If **tail** = **Nag_UpperTail**, the upper tail probability, i.e., $P(T \geq t_p : \nu)$.
If **tail** = **Nag_LowerTail**, the lower tail probability, i.e., $P(T \leq t_p : \nu)$.
If **tail** = **Nag_TwoTailSignif**, the two tail (significance level) probability, i.e., $P(T \geq |t_p| : \nu) + P(T \leq -|t_p| : \nu)$.
If **tail** = **Nag_TwoTailConfid**, the two tail (confidence interval) probability, i.e., $P(T \leq |t_p| : \nu) - P(T \leq -|t_p| : \nu)$.
Constraint: **tail** = **Nag_UpperTail** or **Nag_LowerTail** or **Nag_TwoTailSignif** or **Nag_TwoTailConfid**.

p

Input: the probability, p, from the required Student's t-distribution as defined by **tail**.
Constraint: $0.0 < \mathbf{p} < 1.0$.

df

Input: the degrees of freedom, ν, of the Student's t-distribution.
Constraint: **df** ≥ 1.0.

fail

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

5. Error Indications and Warnings

On any of the error conditions listed below except **NE_SOL_NOT_CONV** nag_deviates_students_t returns 0.0.

NE_BAD_PARAM

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

NE_REAL_ARG_LE
On entry, **p** must not be less than or equal to 0.0: **p** = ⟨*value*⟩.

NE_REAL_ARG_GE
On entry, **p** must not be greater than or equal to 1.0: **p** = ⟨*value*⟩.

NE_REAL_ARG_LT
On entry, **df** must not be less than 1.0: **df** = ⟨*value*⟩.

NE_SOL_CLOSE_TO_ZERO
The solution is too close to zero to be determined accurately.
This error will only occur when **df** = 1.0. The returned value of zero will be a good approximation in terms of absolute value but will have a poor relative precision.

NE_SOL_NOT_CONV
The solution has failed to converge.
However, the result should be a reasonable approximation.

6. Further Comments

The value t_p may be calculated by using the transformation described in Section 3 and using nag_deviates_beta (g01fec). This function allows the user to set the required accuracy.

6.1. Accuracy

The results should be accurate to 5 significant digits, for most parameter values. The error behaviour for various parameter values is discussed in Hill (1970).

6.2. References

Hastings N A J and Peacock J B (1975) *Statistical Distributions* Butterworth.
Hill G W (1970) Student's t quantiles *Commun. ACM* **13** (10) Algorithm 396 619–620.

7. See Also

nag_deviates_beta (g01fec)

8. Example

Lower tail probabilities are read for several t-distributions, and the corresponding deviates calculated and printed, until the end of data is reached.

8.1. Program Text

```
/* nag_deviates_students_t(g01fbc) Example Program
 *
 * Copyright 1996 Numerical Algorithms Group.
 *
 * Mark 4, 1996.
 */

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

main()
{
  double df, p, t;
  int i;
  static Nag_TailProbability tail[] = { Nag_LowerTail, Nag_UpperTail,
                                        Nag_TwoTailSignif, Nag_TwoTailConfid};
  static char *tailmess[] = { "Nag_LowerTail", "Nag_UpperTail",
                              "Nag_TwoTailSignif", "Nag_TwoTailConfid"};
  static NagError fail;

  Vprintf("g01fbc Example Program Results\n\n");
  /*  Skip heading in data file */
```

```
  Vscanf("%*[^\n]");
  Vprintf("    p      df          tail                  t\n\n");
  while (scanf("%lf %lf %ld", &p, &df, &i) != EOF)
    {
      t = g01fbc(tail[i], p, df, &fail);
      if (fail.code==NE_NOERROR)
        Vprintf("%8.3f%8.3f   %-19s  %8.3f\n", p, df, tailmess[i], t);
      else
        Vprintf("%8.3f%8.3f   %-19s  %8.3f\n Note: %s\n",p,df,
                tailmess[i], t, fail.message);
    }
  exit(EXIT_SUCCESS);
}
```

8.2. Program Data

```
g01fbc Example Program Data
0.0100  20.0  2
0.01     7.5  0
0.99    45.0  3
```

8.3. Program Results

```
g01fbc Example Program Results

    p      df          tail                  t

   0.010  20.000   Nag_TwoTailSignif       2.845
   0.010   7.500   Nag_LowerTail          -2.943
   0.990  45.000   Nag_TwoTailConfid       2.690
```