

NAG Library Function Document

nag_prob_students_t (g01ebc)

1 Purpose

nag_prob_students_t (g01ebc) returns the lower tail, upper tail or two tail probability for the Student's t -distribution with real degrees of freedom.

2 Specification

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

double nag_prob_students_t (Nag_TailProbability tail, double t, double df,
                             NagError *fail)
```

3 Description

The lower tail probability for the Student's t -distribution with ν degrees of freedom, $P(T \leq t : \nu)$ is defined by:

$$P(T \leq t : \nu) = \frac{\Gamma((\nu + 1)/2)}{\sqrt{\pi\nu}\Gamma(\nu/2)} \int_{-\infty}^t \left[1 + \frac{T^2}{\nu}\right]^{-(\nu+1)/2} dT, \quad \nu \geq 1.$$

Computationally, there are two situations:

(i) when $\nu < 20$, a transformation of the beta distribution, $P_\beta(B \leq \beta : a, b)$ is used

$$P(T \leq t : \nu) = \frac{1}{2}P_\beta\left(B \leq \frac{\nu}{\nu + t^2} : \nu/2, \frac{1}{2}\right) \quad \text{when } t < 0.0$$

or

$$P(T \leq t : \nu) = \frac{1}{2} + \frac{1}{2}P_\beta\left(B \geq \frac{\nu}{\nu + t^2} : \nu/2, \frac{1}{2}\right) \quad \text{when } t > 0.0;$$

(ii) when $\nu \geq 20$, an asymptotic normalizing expansion of the Cornish–Fisher type is used to evaluate the probability, see Hill (1970).

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

Hill G W (1970) Student's t -distribution *Comm. ACM* **13(10)** 617–619

5 Arguments

1: **tail** – Nag_TailProbability *Input*

On entry: indicates which tail the returned probability should represent.

tail = Nag_UpperTail

The upper tail probability is returned, i.e., $P(T \geq t : \nu)$.

tail = Nag_TwoTailSignif

The two tail (significance level) probability is returned, i.e., $P(T \geq |t| : \nu) + P(T \leq -|t| : \nu)$.

tail = Nag_TwoTailConfid

The two tail (confidence interval) probability is returned, i.e., $P(T \leq |t| : \nu) - P(T \leq -|t| : \nu)$.

tail = Nag_LowerTail

The lower tail probability is returned, i.e., $P(T \leq t : \nu)$.

Constraint: **tail** = Nag_UpperTail, Nag_TwoTailSignif, Nag_TwoTailConfid or Nag_LowerTail.

- 2: **t** – double *Input*
On entry: t , the value of the Student's t variate.
- 3: **df** – double *Input*
On entry: ν , the degrees of freedom of the Student's t -distribution.
Constraint: **df** ≥ 1.0 .
- 4: **fail** – NagError * *Input/Output*
 The NAG error argument (see Section 3.6 in the Essential Introduction).

6 Error Indicators and Warnings

NE_ALLOC_FAIL

Dynamic memory allocation failed.

NE_BAD_PARAM

On entry, argument $\langle value \rangle$ had an illegal value.

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_REAL_ARG_LT

On entry, **df** = $\langle value \rangle$.
Constraint: **df** ≥ 1.0 .

7 Accuracy

The computed probability should be accurate to five significant places for reasonable probabilities but there will be some loss of accuracy for very low probabilities (less than 10^{-10}), see Hastings and Peacock (1975).

8 Further Comments

The probabilities could also be obtained by using the appropriate transformation to a beta distribution (see Abramowitz and Stegun (1972)) and using nag_prob_beta_dist (g01eec). This function allows you to set the required accuracy.

9 Example

This example reads values from, and degrees of freedom for Student's t -distributions along with the required tail. The probabilities are calculated and printed until the end of data is reached.

9.1 Program Text

```

/* nag_prob_students_t (g01ebc) Example Program.
 *
 * Copyright 1996 Numerical Algorithms Group.
 *
 * Mark 4, 1996.
 * Mark 5 revised, 1998.
 * Mark 7 revised, 2001.
 *
 */

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

int main(void)
{
    Integer          exit_status = 0;
    double           df, prob, t;
    int              i;
    static Nag_TailProbability tail[4] = { Nag_LowerTail, Nag_UpperTail,
                                          Nag_TwoTailSignif, Nag_TwoTailConfid };
    static const char *tailmess[] = { "Nag_LowerTail", "Nag_UpperTail",
                                       "Nag_TwoTailSignif",
                                       "Nag_TwoTailConfid" };
    NagError         fail;

    INIT_FAIL(fail);

    printf("nag_prob_students_t (g01ebc) Example Program Results\n\n");
    /* Skip heading in data file */
    scanf("%*[\n]");
    printf("      t      df      prob      tail\n\n");
    while (scanf("%lf %lf %d\n", &t, &df, &i) != EOF)
    {
        /* nag_prob_students_t (g01ebc).
         * Probabilities for Student's t-distribution
         */
        prob = nag_prob_students_t(tail[i], t, df, &fail);
        if (fail.code != NE_NOERROR)
        {
            printf("Error from nag_prob_students_t (g01ebc).\n%s\n",
                  fail.message);
            exit_status = 1;
            goto END;
        }
        printf(" %6.3f%8.3f%8.4f  %s\n", t, df, prob, tailmess[i]);
    }

    END:
    return exit_status;
}

```

9.2 Program Data

```

nag_prob_students_t (g01ebc) Example Program Data
0.85  20.0  0
0.85  20.0  2
0.85  20.0  3
0.85  20.0  1

```

9.3 Program Results

nag_prob_students_t (g01ebc) Example Program Results

t	df	prob	tail
0.850	20.000	0.7973	Nag_LowerTail
0.850	20.000	0.4054	Nag_TwoTailSignif
0.850	20.000	0.5946	Nag_TwoTailConfid
0.850	20.000	0.2027	Nag_UpperTail
