NAG Library Function Document

nag_bivariate_students_t (g01hcc)

1 Purpose

nag_bivariate_students_t (g01hcc) returns probabilities for the bivariate Student’s t-distribution.

2 Specification

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

double nag_bivariate_students_t (Nag_TailProbability tail, const double a[],
const double b[], Integer df, double rho, NagError *fail)
```

3 Description

Let the vector random variable \( X = (X_1, X_2)^T \) follow a bivariate Student’s t-distribution with degrees of freedom \( \nu \) and correlation \( \rho \), then the probability density function is given by

\[
f(X : \nu, \rho) = \frac{1}{2\pi\sqrt{1 - \rho^2}} \left(1 + \frac{X_1^2 + X_2^2 - 2\rho X_1 X_2}{\nu(1 - \rho^2)}\right)^{-\nu/2-1}.
\]

The lower tail probability is defined by:

\[
P(X_1 \leq b_1, X_2 \leq b_2 : \nu, \rho) = \int_{-\infty}^{b_1} \int_{-\infty}^{b_2} f(X : \nu, \rho) dX_2 dX_1.
\]

The upper tail probability is defined by:

\[
P(X_1 \geq a_1, X_2 \geq a_2 : \nu, \rho) = \int_{a_1}^{\infty} \int_{a_2}^{\infty} f(X : \nu, \rho) dX_2 dX_1.
\]

The central probability is defined by:

\[
P(a_1 \leq X_1 \leq b_1, a_2 \leq X_2 \leq b_2 : \nu, \rho) = \int_{a_1}^{b_1} \int_{a_2}^{b_2} f(X : \nu, \rho) dX_2 dX_1.
\]

Calculations use the Dunnet and Sobel (1954) method, as described by Genz (2004).

4 References

Dunnet C W and Sobel M (1954) A bivariate generalization of Student’s t-distribution, with tables for certain special cases *Biometrika* 41 153–169


5 Arguments

1: \( \text{tail} \) – Nag_TailProbability

*Input*

On entry: indicates which probability is to be returned.

- \text{tail} = Nag_LowerTail
  - The lower tail probability is returned.

- \text{tail} = Nag_UpperTail
  - The upper tail probability is returned.
tail = Nag_Central
The central probability is returned.

Constraint: tail = Nag_LowerTail, Nag_UpperTail or Nag_Central.

   Input
   On entry: if tail = Nag_Central or Nag_UpperTail, the lower bounds $a_1$ and $a_2$.
   If tail = Nag_LowerTail, a is not referenced.

   Input
   On entry: if tail = Nag_Central or Nag_LowerTail, the upper bounds $b_1$ and $b_2$.
   If tail = Nag_UpperTail, b is not referenced.
   Constraint: if tail = Nag_Central, $a_i < b_i$, for $i = 1, 2$.

4: df – Integer
   Input
   On entry: $\nu$, the degrees of freedom of the bivariate Student’s $t$-distribution.
   Constraint: df $\geq 1$.

5: rho – double
   Input
   On entry: $\rho$, the correlation of the bivariate Student’s $t$-distribution.
   Constraint: $-1.0 \leq \rho \leq 1.0$.

6: 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.
See Section 3.2.1.2 in the Essential Introduction for further information.

NE_BAD_PARAM
On entry, argument value had an illegal value.

NE_INT
On entry, df = (value).
Constraint: df $\geq 1$.

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.

An unexpected error has been triggered by this function. Please contact NAG.
See Section 3.6.6 in the Essential Introduction for further information.

NE_NO_LICENCE
Your licence key may have expired or may not have been installed correctly.
See Section 3.6.5 in the Essential Introduction for further information.
NE_REAL

On entry, \( \rho = \langle \text{value} \rangle \).
Constraint: \(-1.0 \leq \rho \leq 1.0 \).

NE_REAL_2

On entry, \( b[i-1] \leq a[i-1] \) for central probability, for some \( i = 1, 2 \).

7 Accurary

Accuracy of the algorithm implemented here is discussed in comparison with algorithms based on a
generalized Plackett formula by Genz (2004), who recommends the Dunnet and Sobel method. This
implementation should give a maximum absolute error of the order of \( 10^{-16} \).

8 Parallelism and Performance

Not applicable.

9 Further Comments

None.

10 Example

This example calculates the bivariate Student's \( t \) probability given the choice of tail and degrees of
freedom, correlation and bounds.

10.1 Program Text

```c
/* nag_bivariate_students_t (g01hcc) Example Program. */
/* Copyright 2014 Numerical Algorithms Group. */
/* Mark 23, 2011. */
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <nag.h>
#include <nag_stdlib.h>
#include <nagg01.h>

int main(void)
{
    /* Scalars */
    Integer df, exit_status = 0, ierr;
    double prob, rho;
    /* Arrays */
    char nag_enum_arg[14];
    double a[2], b[2];
    /* NAG types */
    NAG_TailProbability tail;
    NagError fail;

    printf("%s\n\n",
        "nag_bivariate_students_t (g01hcc) Example Program Results");

    /* Skip heading in data file */
    #ifdef _WIN32
    scanf_s("%*[\n]");
    #else
    scanf("%*[\n]");
    #endif
```
/* Display headers */
printf("%-8s%-8s%-8s%-8s%-8s%-8s%-8s%-8s
", "a1", "b1", "a2", "b2", "df", "rho", "Tail", "p");

while (1)
{
    #ifdef _WIN32
        ierr = scanf_s("%13s", nag_enum_arg, _countof(nag_enum_arg));
    #else
        ierr = scanf("%13s", nag_enum_arg);
    #endif
    if (ierr == EOF || ierr < 1)
    {
        break;
    }

    /* Initialize limits */
a[0] = a[1] = b[0] = b[1] = 0.0;
    /* nag_enum_name_to_value (x04nac).
     * Converts NAG enum member name to value */
tail = (Nag_TailProbability)nag_enum_name_to_value(nag_enum_arg);
    /* Read parameter values*/
    switch (tail)
    {
        case Nag_LowerTail:
            #ifdef _WIN32
                scanf_s("%lf%lf%lf", &df, &rho, b, b+1);
            #else
                scanf("%lf%lf%lf", &df, &rho, b, b+1);
            #endif
            break;
        case Nag_Central:
            #ifdef _WIN32
                scanf_s("%lf%lf%lf%lf%lf", &df, &rho, a, b, a+1, b+1);
            #else
                scanf("%lf%lf%lf%lf%lf", &df, &rho, a, b, a+1, b+1);
            #endif
            break;
        case Nag_UpperTail:
            #ifdef _WIN32
                scanf_s("%lf%lf%lf", &df, &rho, a, a+1);
            #else
                scanf("%lf%lf%lf", &df, &rho, a, a+1);
            #endif
            break;
        default:
            printf(" Invalid tail specification in data file\n");
            exit_status = -1;
            goto END;
    }
    #ifdef _WIN32
    scanf_s("%*[\n]");
    #else
    scanf("%*[\n]");
    #endif

/* Calculate probability for the bivariate Student’s t-distribution */
INIT_FAIL(fail);
/* nag_bivariate_students_t (g01hcc) */
prob = nag_bivariate_students_t(tail, a, b, df, rho, &fail);

/* Display results */
switch (tail)
{
    case Nag_LowerTail:
        printf("%-8s%-8g%-8s%-8s%-8s%-8g", "a1", "df", "b1", "rho", "Tail", "p");
        break;
    case Nag_Central:
        printf("%-8s%-8g%-8s%-8s%-8s%-8g", "a1", "df", "b1", "rho", "Tail", "p");
        break;
    case Nag_UpperTail:
        printf("%-8s%-8g%-8s%-8s%-8s%-8g", "a1", "df", "b1", "rho", "Tail", "p");
        break;
    default:
        printf(" Invalid tail specification in data file\n");
        exit_status = -1;
        goto END;
}
"-Inf", " ", b[0], " ", "-Inf", " ", b[1]);
break;
case Nag_Central:
    printf("%-8g%2s%-8g%2s%-8g%2s%-8g",
           a[0], " ", b[0], " ", a[1], " ", b[1]);
break;
case Nag_UpperTail:
    printf("%-8g%2s%-8s%2s%-8g%2s%-8s",
           a[0], " ", "Inf", " ", a[1], " ", "Inf"));
break;
default:
{
    printf("Invalid tail specification.\n");
    exit_status = -1;
goto END;
}
}
printf("%2s%-4"NAG_IFMT"%2s%-8g%2s%-14s%2s%-8.4f\n",
       " ", df, " ", rho, " ", nag_enum_arg, " ", prob);
}
END:
return exit_status;
}

10.2 Program Data

nag_bivariate_students_t (g01hcc) Example Program Data
Nag_LowerTail 8 0.6 4.0 0.8 : tail df rho b[i], i=0,1
Nag_Central 12 -0.2 -40.0 2.0 0.0 4.0 : tail df rho (a, b)[i], i=0,1
Nag_UpperTail 2 0.3 -2.0 8.0 : tail df rho a[i], i=0,1

10.3 Program Results

nag_bivariate_students_t (g01hcc) Example Program Results

<table>
<thead>
<tr>
<th>a1</th>
<th>b1</th>
<th>a2</th>
<th>b2</th>
<th>df</th>
<th>rho</th>
<th>Tail</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>-Inf</td>
<td>4</td>
<td>-Inf</td>
<td>0.8</td>
<td>8</td>
<td>0.6</td>
<td>Nag_LowerTail</td>
<td>0.7764</td>
</tr>
<tr>
<td>-40</td>
<td>2</td>
<td>0</td>
<td>4</td>
<td>12</td>
<td>-0.2</td>
<td>Nag_Central</td>
<td>0.4876</td>
</tr>
<tr>
<td>-2</td>
<td>Inf</td>
<td>8</td>
<td>Inf</td>
<td>2</td>
<td>0.3</td>
<td>Nag_UpperTail</td>
<td>0.0059</td>
</tr>
</tbody>
</table>