nag_2_sample_t_test (g07cac)

1. Purpose

nag_2_sample_t_test (g07cac) computes a t-test statistic to test for a difference in means between two Normal populations, together with a confidence interval for the difference between the means.

2. Specification

```
#include <nag.h>
#include <nagg07.h>

void nag_2_sample_t.test(Nag_TailProbability tail, Nag_PopVar equal,
             Integer nx, Integer ny, double xmean, double ymean, double xstd,
             double ystd, double clevel, double *t, double *df,
             double *prob, double *dl, double *du,
             NagError *fail)
```

3. Description

Consider two independent samples, denoted by X and Y, of size n_x and n_y drawn from two Normal populations with means μ_x and μ_y, and variances σ_x^2 and σ_y^2 respectively. Denote the sample means by $\bar{x}$ and $\bar{y}$ and the sample variances by s_x^2 and s_y^2 respectively.

nag_2_sample_t_test calculates a test statistic and its significance level to test the null hypothesis $H_0 : \mu_x = \mu_y$, together with upper and lower confidence limits for $\mu_x - \mu_y$. The test used depends on whether or not the two population variances are assumed to be equal.

(1) It is assumed that the two variances are equal, that is $\sigma_x^2 = \sigma_y^2$.

The test used is the two sample t-test. The test statistic t is defined by;

$$t_{\text{obs}} = \frac{\bar{x} - \bar{y}}{s\sqrt{(1/n_x) + (1/n_y)}}$$

where $s^2 = \dfrac{(n_x - 1)s_x^2 + (n_y - 1)s_y^2}{n_x + n_y - 2}$ is the pooled variance of the two samples.

Under the null hypothesis H_0 this test statistic has a t-distribution with $(n_x + n_y - 2)$ degrees of freedom.

The test of H_0 is carried out against one of three possible alternatives;

$H_1 : \mu_x \neq \mu_y$; the significance level, $p = P(t \geq |t_{\text{obs}}|)$, i.e., a two-tailed probability.
$H_1 : \mu_x > \mu_y$; the significance level, $p = P(t \geq t_{\text{obs}})$, i.e., an upper tail probability.
$H_1 : \mu_x < \mu_y$; the significance level, $p = P(t \leq t_{\text{obs}})$, i.e., a lower tail probability.

Upper and lower $100(1 - \alpha)\%$ confidence limits for $\mu_x - \mu_y$ are calculated as:

$$(\bar{x} - \bar{y}) \pm t_{1-\alpha/2}s\sqrt{(1/n_x) + (1/n_y)},$$

where $t_{1-\alpha/2}$ is the $100(1 - \alpha/2)$ percentage point of the t-distribution with $(n_x + n_y - 2)$ degrees of freedom.

(2) It is not assumed that the two variances are equal.

If the population variances are not equal the usual two sample t-statistic no longer has a t-distribution and an approximate test is used.

This problem is often referred to as the Behrens-Fisher problem, see Kendall and Stuart (1979). The test used here is based on Satterthwaites procedure. To test the null hypothesis the test statistic t' is used where

$$t'_{\text{obs}} = \frac{\bar{x} - \bar{y}}{\text{se}(\bar{x} - \bar{y})}$$

where $\text{se}(\bar{x}-\bar{y}) = \sqrt{\dfrac{s_x^2}{n_x} + \dfrac{s_y^2}{n_y}}$.

A t-distribution with f degrees of freedom is used to approximate the distribution of t' where

$$f = \frac{\text{se}(\bar{x}-\bar{y})^4}{\dfrac{s_x^2/n_x^2}{(n_x-1)} + \dfrac{s_y^2/n_y^2}{(n_y-1)}}.$$

The test of H_0 is carried out against one of the three alternative hypotheses described above, replacing t by t' and t_{obs} by t'_{obs}.

Upper and lower $100(1-\alpha)\%$ confidence limits for $\mu_x - \mu_y$ are calculated as:

$$(\bar{x}-\bar{y}) \pm t_{1-\alpha/2}\text{se}(x-\bar{y}).$$

where $t_{1-\alpha/2}$ is the $100(1-\alpha/2)$ percentage point of the t-distribution with f degrees of freedom.

4. Parameters

tail

Input: indicates which tail probability is to be calculated, and thus which alternative hypothesis is to be used.
If **tail** = **Nag_TwoTail**, the two tail probability, i.e., $H_1 : \mu_x \neq \mu_y$.
If **tail** = **Nag_UpperTail**, the upper tail probability, i.e., $H_1 : \mu_x > \mu_y$.
If **tail** = **Nag_LowerTail**, the lower tail probability, i.e., $H_1 : \mu_x < \mu_y$.
Constraint: **tail** = **Nag_UpperTail**, **Nag_LowerTail**, or **Nag_TwoTail**.

equal

Input: indicates whether the population variances are assumed to be equal or not.
If **equal** = **Nag_PopVarEqual**, the population variances are assumed to be equal, that is $\sigma_x^2 = \sigma_y^2$.
If **equal** = **Nag_PopVarNotEqual**, the population variances are not assumed to be equal.
Constraint: **equal** = **Nag_PopVarEqual** or **Nag_PopVarNotEqual**.

nx

Input: the size of the X sample, n_x.
Constraint: **nx** ≥ 2.

ny

Input: the size of the Y sample, n_y.
Constraint: **ny** ≥ 2.

xmean

Input: the mean of the X sample, $\bar{x}$.

ymean

Input: the mean of the Y sample, $\bar{y}$.

xstd

Input: the standard deviation of the X sample, s_x.
Constraint: **xstd** > 0.0.

ystd

Input: the standard deviation of the Y sample, s_y.
Constraint: **ystd** > 0.0.

clevel

Input: the confidence level, $1-\alpha$, for the specified tail. For example **clevel** $= 0.95$ will give a 95% confidence interval.
Constraint: $0.0 <$ **clevel** < 1.0.

t
: Output: contains the test statistic, t_{obs} or t'_{obs}.

df
: Output: contains the degrees of freedom for the test statistic.

prob
: Output: contains the significance level, that is the tail probability, p, as defined by **tail**.

dl
: Output: contains the lower confidence limit for $\mu_x - \mu_y$.

du
: Output: contains the upper confidence limit for $\mu_x - \mu_y$.

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

5. Error Indications and Warnings

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

NE_INT_ARG_LT
On entry, **nx** must not be less than 2: **nx** = ⟨*value*⟩.
On entry, **ny** must not be less than 2: **ny** = ⟨*value*⟩.

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

NE_REAL_ARG_GE
On entry, **clevel** must not be greater than or equal to 1.0: **clevel** = ⟨*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 consult NAG for assistance.

6. Further Comments

The sample means and standard deviations can be computed using nag_summary_stats_1var (g01aac).

6.1. Accuracy

The computed probability and the confidence limits should be accurate to approximately 5 significant figures.

6.2. References

Johnson M G and Kotz A (1969) *The Encyclopedia of Statistics.* **2** Griffin.
Kendall M G and Stuart A (1979) *The Advanced Theory of Statistics. (Volume 2)* Griffin (4th Edition).
Snedecor G W and Cochran W G (1967) *Statistical Methods.* Iowa State University Press.

7. See Also

None.

8. Example

The following example program reads the two sample sizes and the sample means and standard deviations for two independent samples. The data is taken from Snedecor and Cochran, page 116, from a test to compare two methods of estimating the concentration of a chemical in a vat. A test of the equality of the means is carried out first assuming that the two population variances are equal and then making no assumption about the equality of the population variances.

8.1. Program Text

```
/* nag_2_sample_t_test(g07cac) Example Program.
 *
 * Copyright 1996 Numerical Algorithms Group.
 *
 * Mark 4, 1996.
 *
 */

#include <nag.h>
#include <stdio.h>
#include <nag_stdlib.h>
#include <nagg07.h>

main()
{
  /* Local variables */
  double prob, xstd, ystd;
  double t;
  double xmean, ymean, df, dl, du;
  double clevel;

  Integer ifail;
  Integer nx, ny;

  Vprintf("g07cac Example Program Results\n");
  /*     Skip heading in data file */
  Vscanf("%*[^\n]");
  Vscanf("%ld %ld", &nx, &ny);
  Vscanf("%lf %lf %lf %lf", &xmean, &ymean,&xstd, &ystd);
  Vscanf("%lf",&clevel);

  g07cac(Nag_TwoTail, Nag_PopVarEqual, nx, ny, xmean, ymean, xstd, ystd,
         clevel, &t, &df, &prob, &dl, &du, NAGERR_DEFAULT);

  Vprintf("\nAssuming population variances are equal.\n\n");
  Vprintf("t test statistic = %10.4f\n",t);
  Vprintf("Degrees of freedom = %8.1f\n",df);
  Vprintf("Significance level = %8.4f\n", prob);
  Vprintf("Lower confidence limit for difference in means = %10.4f\n", dl);
  Vprintf("Upper confidence limit for difference in means = %10.4f\n\n",du);

  g07cac(Nag_TwoTail, Nag_PopVarNotEqual, nx, ny, xmean, ymean, xstd, ystd,
         clevel, &t, &df, &prob, &dl, &du, NAGERR_DEFAULT);

  Vprintf("\nNo assumptions about population variances.\n\n");
  Vprintf("t test statistic = %10.4f\n",t);
  Vprintf("Degrees of freedom = %8.4f\n",df);
  Vprintf("Significance level = %8.4f\n", prob);
  Vprintf("Lower confidence limit for difference in means = %10.4f\n",dl);
  Vprintf("Upper confidence limit for difference in means = %10.4f\n",du);
  exit(EXIT_SUCCESS);
}
```

8.2. Program Data

```
g07cac Example Program Data
4  8
25.0 21.0
0.8185  4.2083
0.95
```

8.3. Program Results

```
g07cac Example Program Results

Assuming population variances are equal.

t test statistic =     1.8403
Degrees of freedom =     10.0
Significance level =   0.0955
Lower confidence limit for difference in means =    -0.8429
Upper confidence limit for difference in means =     8.8429


No assumptions about population variances.

t test statistic =     2.5922
Degrees of freedom =   7.9925
Significance level =   0.0320
Lower confidence limit for difference in means =     0.4410
Upper confidence limit for difference in means =     7.5590
```