nag_mann_whitney (g08amc) performs the Mann–Whitney U test on two independent samples of possibly unequal size and calculates the exact probability for the Mann–Whitney rank sum test statistic for the case where there are either ties or no ties in the samples pooled together.

2 Specification

```c
#include <nag.h>
#include <nagg08.h>
void nag_mann_whitney (Integer n1, const double x[], Integer n2, const double y[], Nag_TailProbability tail, Nag_CompProb exact, double *u, double *z, double *p, NagError *fail)
```

3 Description

The Mann–Whitney U test investigates the difference between two populations defined by the distribution functions \( F(x) \) and \( G(y) \) respectively. The data consist of two independent samples of size \( n_1 \) and \( n_2 \), denoted by \( x_1, x_2, \ldots, x_{n_1} \) and \( y_1, y_2, \ldots, y_{n_2} \), taken from the two populations.

The hypothesis under test, \( H_0 \), often called the null hypothesis, is that the two distributions are the same, that is \( F(x) = G(y) \), and this is to be tested against an alternative hypothesis \( H_1 \) which is

- \( H_1 : F(x) \neq G(y) \); or
- \( H_1 : F(x) < G(y) \), i.e., the \( x \)'s tend to be greater than the \( y \)'s; or
- \( H_1 : F(x) > G(y) \), i.e., the \( x \)'s tend to be less than the \( y \)'s,

using a two tailed, upper tailed or lower tailed probability respectively. You select the alternative hypothesis by choosing the appropriate tail probability to be computed (see the description of argument `tail` in Section 5).

Note that when using this test to test for differences in the distributions one is primarily detecting differences in the location of the two distributions. That is to say, if we reject the null hypothesis \( H_0 \) in favour of the alternative hypothesis \( H_1 : F(x) > G(y) \) we have evidence to suggest that the location, of the distribution defined by \( F(x) \), is less than the location, of the distribution defined by \( G(y) \).

The Mann–Whitney U test differs from the Median test (see nag_median_test (g08acc)) in that the ranking of the individual scores within the pooled sample is taken into account, rather than simply the position of a score relative to the median of the pooled sample. It is therefore a more powerful test if score differences are meaningful.

The test procedure involves ranking the pooled sample, average ranks being used for ties. Let \( r_{1i} \) be the rank assigned to \( x_i \), for \( i = 1, 2, \ldots, n_1 \) and \( r_{2j} \) the rank assigned to \( y_j \), for \( j = 1, 2, \ldots, n_2 \). Then the test statistic \( U \) is defined as follows;

\[
U = \sum_{i=1}^{n_1} r_{1i} - \frac{n_1(n_1 + 1)}{2}
\]

\( U \) is also the number of times a score in the second sample precedes a score in the first sample (where we only count a half if a score in the second sample actually equals a score in the first sample).

nag_mann_whitney (g08amc) returns:
(a) the test statistic $U$;
(b) the approximate Normal test statistic,

$$
z = \frac{U - \text{mean}(U) \pm \frac{1}{2}}{\sqrt{\text{var}(U)}}
$$

where

$$\text{mean}(U) = \frac{n_1 n_2}{2}
$$

and

$$\text{var}(U) = \frac{n_1 n_2 (n_1 + n_2 + 1)}{12} - \frac{n_1 n_2}{(n_1 + n_2)(n_1 + n_2 - 1)} \times TS
$$

where

$$TS = \sum_{j=1}^{\tau} \frac{(t_j)(t_j - 1)(t_j + 1)}{12}
$$

$\tau$ is the number of groups of ties in the sample and $t_j$ is the number of ties in the $j$th group.

Note that if no ties are present the variance of $U$ reduces to $\frac{n_1 n_2}{12}(n_1 + n_2 + 1)$.

(c) An indicator as to whether ties were present in the pooled sample or not.

(d) The tail probability, $p$, corresponding to $U$, depending on the choice of tail, i.e., the choice of alternative hypothesis, $H_1$. An exact probability or a normal approximation may be selected using the exact option. For large values of $n_1$ and $n_2$ the normal approximation should be adequate while the exact computation may require extensive calculation, particularly in cases where there are ties in the observations. For small samples the exact probability should be used.

The value of $p$ can be used to perform a significance test on the null hypothesis $H_0$ against the alternative hypothesis $H_1$. Let $\alpha$ be the size of the significance test (that is, $\alpha$ is the probability of rejecting $H_0$ when $H_0$ is true). If $p < \alpha$ then the null hypothesis is rejected. Typically $\alpha$ might be 0.05 or 0.01.

4 References


5 Arguments

1: n1 – Integer

On entry: the number of non-tied pairs, $n_1$.

Constraint: n1 $\geq 1$.

2: x[n1] – const double

On entry: the first vector of observations. $x_1, x_2, \ldots, x_{n_1}$.

3: n2 – Integer

On entry: the size of the second sample, $n_2$.

Constraint: n2 $\geq 1$. 
4: y[n2] – const double
   
   On entry: the second vector of observations. y_1, y_2, \ldots, y_{n_2}.

5: tail – Nag_TailProbability
   
   On entry: indicates the choice of tail probability, and hence the alternative hypothesis.

   tail = Nag_TwoTail
   A two tailed probability is calculated and the alternative hypothesis is \( H_1: F(x) \neq G(y) \).

   tail = Nag_UpperTail
   An upper tailed probability is calculated and the alternative hypothesis \( H_1: F(x) < G(y) \), i.e., the \( x \)'s tend to be greater than the \( y \)'s.

   tail = Nag_LowerTail
   A lower tailed probability is calculated and the alternative hypothesis \( H_1: F(x) > G(y) \), i.e., the \( x \)'s tend to be less than the \( y \)'s.

   Constraint: tail = Nag_TwoTail, Nag_UpperTail or Nag_LowerTail.

6: exact – Nag_CompProb
   
   On entry: indicates if exact probability, \( p \), is to be computed.

   exact = Nag_CompProbExact
   Compute the exact probability.

   exact = Nag_CompProbApprox
   Compute the approximate probability.

   Constraint: exact = Nag_CompProbExact or Nag_CompProbApprox.

7: u – double *
   
   On exit: the Mann–Whitney rank sum statistic, \( U \).

8: z – double *
   
   On exit: the approximate Normal test statistic, \( z \), as described in Section 3.

9: p – double *
   
   On exit: the exact tail probability, \( p \), as specified by the argument tail.

10: fail – NagError *
   
   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 exact had an illegal value.
   On entry, argument tail had an illegal value.

NE_G08AH_SAMP_IDEN
   The pooled samples are all the same, that is the variance of \( U = 0.0 \).
NE_INT_ARG_LT
On entry, n1 must not be less than 1: n1 = (value).
On entry, n2 must not be less than 1: n2 = (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_MANN_WHITNEY_STAT
The computed Mann–Whitney statistic, \( U = \text{(value)} \). If the exact tail probability is to be computed, then \( U \geq 0 \).

7 Accuracy
The approximate tail probability, \( p \), returned by nag_mann_whitney (g08amc) is a good approximation to the exact probability for cases where \( \max(n_1, n_2) \geq 30 \) and \( (n_1 + n_2) \geq 40 \). The relative error of the approximation should be less than 10 percent, for most cases falling in this range.

8 Parallelism and Performance
Not applicable.

9 Further Comments
The time taken by nag_mann_whitney (g08amc) increases with \( n_1 \) and \( n_2 \).

10 Example
The example program performs the Mann–Whitney test on two independent samples of sizes 16 and 23 respectively. This is used to test the null hypothesis that the distributions of the two populations from which the samples were taken are the same against the alternative hypothesis that the distributions are different. The test statistic, the approximate Normal statistic and the approximate two-tail probability are printed. An exact tail probability is also calculated and printed depending on whether ties were found in the pooled sample or not.

10.1 Program Text
/* nag_mann_whitney (g08amc) Example Program. *
 * Copyright 2014 Numerical Algorithms Group. *
 * Mark 6a revised, 2001. *
*/
#include <stdio.h>
#include <nag.h>
#include <nag_stdlib.h>
#include <nagg08.h>

int main(void)
{
    Integer exit_status = 0, i, n1, n2;
    NagError fail;
    double p, u, *x = 0, *y = 0, z;
    INIT_FAIL(fail);
    printf("nag_mann_whitney (g08amc) Example Program Results\n\n");

/* Skip heading in data file */

#ifdef _WIN32
    scanf_s("%*\[^
\]" );
#else
    scanf("%*\[`\n\]" );
#endif

#ifdef _WIN32
    scanf_s("%"NAG_IFMT" %"NAG_IFMT" ", &n1, &n2);
#else
    scanf("%"NAG_IFMT" %"NAG_IFMT" ", &n1, &n2);
#endif

printf("%s%5"NAG_IFMT"
", "Sample size of group 1 = ", n1);
printf("%s%5"NAG_IFMT"
", "Sample size of group 2 = ", n2);

if (!(x = NAG_ALLOC(n1, double))
    || !(y = NAG_ALLOC(n2, double)))
{
    printf("Allocation failure\n");
    exit_status = -1;
    goto END;
}

printf("\n");
for (i = 1; i <= n1; ++i)
#ifdef _WIN32
    scanf_s("%lf", &x[i - 1]);
#else
    scanf("%lf", &x[i - 1]);
#endif

printf("\n");
#ifdef _WIN32
    scanf_s("%lf", &y[i - 1]);
#else
    scanf("%lf", &y[i - 1]);
#endif

nag_mann_whitney(n1, x, n2, y, Nag_LowerTail, Nag_CompProbApprox, &u, &z, &p, &fail);
if (fail.code != NE_NOERROR)
{
    printf("Error from nag_mann_whitney (g08amc)\n%s\n", fail.message);
    exit_status = 1;
    goto END;
}

nag_mann_whitney(n1, x, n2, y, Nag_LowerTail, Nag_CompProbExact, &u, &z, &p, &fail);
if (fail.code != NE_NOERROR)
{
    printf("Error from nag_mann_whitney (g08amc)\n%s\n", fail.message);
    exit_status = 1;
    goto END;
}
g08amc

NAG Library Manual

goto END;
}
printf("%s%8.4f\n", "Exact tail probability = ", p);
END:
NAG_FREE(x);
NAG_FREE(y);
return exit_status;
}

10.2 Program Data

nag_mann_whitney (g08amc) Example Program Data
16 23
13.0 6.0 12.0 7.0 12.0 7.0 10.0 7.0
10.0 7.0 16.0 7.0 10.0 8.0 9.0 8.0
17.0 6.0 10.0 8.0 15.0 8.0 15.0 10.0
15.0 10.0 14.0 10.0 14.0 11.0 14.0
13.0 12.0 13.0 12.0 13.0 12.0 12.0

10.3 Program Results

nag_mann_whitney (g08amc) Example Program Results

Sample size of group 1 = 16
Sample size of group 2 = 23

Mann-Whitney U test

Data values

<table>
<thead>
<tr>
<th>Group 1</th>
<th>13.0 6.0 12.0 7.0 12.0 7.0 10.0 7.0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10.0 7.0 16.0 7.0 10.0 8.0 9.0 8.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Group 2</th>
<th>17.0 6.0 10.0 8.0 15.0 8.0 15.0 10.0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>15.0 10.0 14.0 10.0 14.0 11.0 14.0</td>
</tr>
<tr>
<td></td>
<td>13.0 12.0 13.0 12.0 13.0 12.0 12.0</td>
</tr>
</tbody>
</table>

Test statistic = 86.0000
Normal Statistic = -2.8039
Approximate tail probability = 0.0025
Exact tail probability = 0.0020