# NAG Library Routine Document

# G12ABF

**Note:** before using this routine, please read the Users' Note for your implementation to check the interpretation of **bold italicised** terms and other implementation-dependent details.

## 1    Purpose

G12ABF calculates the rank statistics, which can include the logrank test, for comparing survival curves.

## 2    Specification

```
SUBROUTINE G12ABF (N, T, IC, GRP, NGRP, FREQ, IFREQ, WEIGHT, WT, TS, DF, P,   &
                   OBSD, EXPT, ND, DI, NI, LDN, IFAIL)

INTEGER           N, IC(N), GRP(N), NGRP, IFREQ(*), DF, ND, DI(LDN),          &
                  NI(LDN), LDN, IFAIL
REAL (KIND=nag_wp) T(N), WT(*), TS, P, OBSD(NGRP), EXPT(NGRP)
CHARACTER(1)      FREQ, WEIGHT
```

## 3    Description

A survivor function, $S(t)$, is the probability of surviving to at least time $t$. Given a series of $n$ failure or right-censored times from $g$ groups G12ABF calculates a rank statistic for testing the null hypothesis

$$H_0 : S_1(t) = S_2(t) = \cdots = S_g(t), t \leq \tau$$

where $\tau$ is the largest observed time, against the alternative hypothesis

$$H_1 : \text{at least one of the } S_i(t) \text{ differ, for some } t \leq \tau.$$

Let $t_i$, for $i = 1, 2, \ldots, n_d$, denote the list of distinct failure times across all $g$ groups and $w_i$ a series of $n_d$ weights. Let $d_{ij}$ denote the number of failures at time $t_i$ in group $j$ and $n_{ij}$ denote the number of observations in the group $j$ that are known to have not failed prior to time $t_i$, i.e., the size of the risk set for group $j$ at time $t_i$. If a censored observation occurs at time $t_i$ then that observation is treated as if the censoring had occurred slightly after $t_i$ and therefore the observation is counted as being part of the risk set at time $t_i$. Finally let

$$d_i = \sum_{j=1}^{g} d_{ij} \qquad \text{and} \qquad n_i = \sum_{j=1}^{g} n_{ij}.$$

The (weighted) number of observed failures in the $j$th group, $O_j$, is therefore given by

$$O_j = \sum_{i=1}^{n_d} w_i d_{ij}$$

and the (weighted) number of expected failures in the $j$th group, $E_j$, by

$$E_j = \sum_{i=1}^{n_d} w_i \frac{n_{ij} d_i}{n_i}.$$

If $x$ denotes the vector of differences $x = (O_1 - E_1, O_2 - E_2, \ldots, O_g - E_g)$ and

$$V_{jk} = \sum_{i=1}^{n_d} w_i^2 \left( \frac{d_i(n_i - d_i)(n_i n_{ik} I_{jk} - n_{ij} n_{ik})}{n_i^2(n_i - 1)} \right)$$

where $I_{jk} = 1$ if $j = k$ and $0$ otherwise, then the rank statistic, $T$, is calculated as

$$T = xV^- x^T$$

where $V^-$ denotes a generalized inverse of the matrix $V$. Under the null hypothesis, $T \sim \chi_\nu^2$ where the degrees of freedom, $\nu$, is taken as the rank of the matrix $V$.

## 4    References

Gross A J and Clark V A (1975) *Survival Distributions: Reliability Applications in the Biomedical Sciences* Wiley

Kalbfleisch J D and Prentice R L (1980) *The Statistical Analysis of Failure Time Data* Wiley

Rostomily R C, Duong D, McCormick K, Bland M and Berger M S (1994) Multimodality management of recurrent adult malignant gliomas: results of a phase II multiagent chemotherapy study and analysis of cytoreductive surgery *Neurosurgery* **35** 378

## 5    Parameters

1:    N – INTEGER                                                                                                  *Input*

*On entry*: $n$, the number of failure and censored times.

*Constraint*: $N \geq 2$.

2:    T(N) – REAL (KIND=nag_wp) array                                                       *Input*

*On entry*: the observed failure and censored times; these need not be ordered.

*Constraint*: $T(i) \neq T(j)$ for at least one $i \neq j$, for $i = 1, 2, \ldots, N$ and $j = 1, 2, \ldots, N$.

3:    IC(N) – INTEGER array                                                                          *Input*

*On entry*: $IC(i)$ contains the censoring code of the $i$th observation, for $i = 1, 2, \ldots, N$.

$IC(i) = 0$
        the $i$th observation is a failure time.

$IC(i) = 1$
        the $i$th observation is right-censored.

*Constraints*:

        $IC(i) = 0$ or 1, for $i = 1, 2, \ldots, N$;
        $IC(i) = 0$ for at least one $i$.

4:    GRP(N) – INTEGER array                                                                     *Input*

*On entry*: $GRP(i)$ contains a flag indicating which group the $i$th observation belongs in, for $i = 1, 2, \ldots, N$.

*Constraints*:

        $1 \leq GRP(i) \leq NGRP$, for $i = 1, 2, \ldots, N$;
        each group must have at least one observation.

5:    NGRP – INTEGER                                                                                    *Input*

*On entry*: $g$, the number of groups.

*Constraint*: $2 \leq NGRP \leq N$.

6:    FREQ – CHARACTER(1)                                                                          *Input*

*On entry*: indicates whether frequencies are provided for each time point.

FREQ = 'F'
        Frequencies are provided for each failure and censored time.

FREQ = 'S'
> The failure and censored times are considered as single observations, i.e., a frequency of 1 is assumed.

*Constraint*: FREQ = 'F' or 'S'.

7:     IFREQ(∗) – INTEGER array                                                               *Input*

**Note**: the dimension of the array IFREQ must be at least N if FREQ = 'F'.

*On entry*: if FREQ = 'F', IFREQ($i$) must contain the frequency (number of observations) to which each entry in T corresponds.

If FREQ = 'S', each entry in T is assumed to correspond to a single observation, i.e., a frequency of 1 is assumed, and IFREQ is not referenced.

*Constraint*: if FREQ = 'F', IFREQ($i$) $\geq 0$, for $i = 1, 2, \ldots, N$.

8:     WEIGHT – CHARACTER(1)                                                                  *Input*

*On entry*: indicates if weights are to be used.

WEIGHT = 'U'
> All weights are assumed to be 1.

WEIGHT = 'W'
> The weights, $w_i$ are supplied in WT.

*Constraint*: WEIGHT = 'U' or 'W'.

9:     WT(∗) – REAL (KIND=nag_wp) array                                                       *Input*

**Note**: the dimension of the array WT must be at least LDN if WEIGHT = 'W'.

*On entry*: if WEIGHT = 'W', WT must contain the $n_d$ weights, $w_i$, where $n_d$ is the number of distinct failure times.

If WEIGHT = 'U', WT is not referenced and $w_i = 1$ for all $i$.

*Constraint*: if WEIGHT = 'W', WT($i$) $\geq 0.0$, for $i = 1, 2, \ldots, n_d$.

10:    TS – REAL (KIND=nag_wp)                                                                *Output*

*On exit*: $T$, the test statistic.

11:    DF – INTEGER                                                                          *Output*

*On exit*: $\nu$, the degrees of freedom.

12:    P – REAL (KIND=nag_wp)                                                                *Output*

*On exit*: $P(X \geq T)$, when $X \sim \chi_\nu^2$, i.e., the probability associated with TS.

13:    OBSD(NGRP) – REAL (KIND=nag_wp) array                                                  *Output*

*On exit*: $O_i$, the observed number of failures in each group.

14:    EXPT(NGRP) – REAL (KIND=nag_wp) array                                                  *Output*

*On exit*: $E_i$, the expected number of failures in each group.

15:    ND – INTEGER                                                                          *Output*

*On exit*: $n_d$, the number of distinct failure times.

16:    DI(LDN) – INTEGER array                                                                *Output*

*On exit*: the first ND elements of DI contain $d_i$, the number of failures, across all groups, at time $t_i$.

17:    NI(LDN) – INTEGER array                                                                                        *Output*

On exit: the first ND elements of NI contain $n_i$, the size of the risk set, across all groups, at time $t_i$.

18:    LDN – INTEGER                                                                                                  *Input*

On entry: the size of arrays DI and NI.  As $n_d \leq n$, if $n_d$ is not known *a priori* then a value of N can safely be used for LDN.

Constraint: LDN $\geq n_d$, the number of unique failure times.

19:    IFAIL – INTEGER                                                                                            *Input/Output*

On entry: IFAIL must be set to 0, $-1$ or 1.  If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction for details.

For environments where it might be inappropriate to halt program execution when an error is detected, the value $-1$ or 1 is recommended.  If the output of error messages is undesirable, then the value 1 is recommended.  Otherwise, if you are not familiar with this parameter, the recommended value is 0.  **When the value $-1$ or 1 is used it is essential to test the value of IFAIL on exit.**

On exit: IFAIL $= 0$ unless the routine detects an error or a warning has been flagged (see Section 6).

# 6    Error Indicators and Warnings

If on entry IFAIL $= 0$ or $-1$, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL $= 1$

On entry, N $< 2$.

IFAIL $= 2$

On entry, all the times in T are the same.

IFAIL $= 3$

On entry, IC$(i) \neq 0$ or 1, for at least one $i = 1, 2, \ldots, $ N.

IFAIL $= 31$

On entry, all observations are censored.

IFAIL $= 4$

On entry, GRP$(i) < 1$ or GRP$(i) >$ NGRP, for at least one $i = 1, 2, \ldots, $ N.

IFAIL $= 41$

On entry, at least one group has no observations.

IFAIL $= 5$

On entry, NGRP $< 2$ or NGRP $>$ N.

IFAIL $= 6$

On entry, FREQ $\neq$ 'F' or 'S'.

IFAIL = 7

On entry, FREQ = 'F' and IFREQ$(i) < 0$, for at least one $i = 1, 2, \ldots, N$.

IFAIL = 8

On entry, WEIGHT $\neq$ 'U' or 'W'.

IFAIL = 9

On entry, WEIGHT = 'W' and WT$(i) < 0.0$, for some $i = 1, 2, \ldots, N$.

IFAIL = 11

The degrees of freedom are zero.

IFAIL = 18

On entry, LDN $< n_d$.

## 7 Accuracy

Not applicable.

## 8 Further Comments

The use of different weights in the formula given in Section 3 leads to different rank statistics being calculated. The logrank test has $w_i = 1$, for all $i$, which is the equivalent of calling G12ABF when WEIGHT = 'U' . Other rank statistics include Wilcoxon ($w_i = n_i$), Tarone–Ware ($w_i = \sqrt{n_i}$) and Peto–Peto ($w_i = \tilde{S}(t_i)$ where $\tilde{S}(t_i) = \prod_{t_j \leq t_i} \frac{n_j - d_j + 1}{n_j + 1}$) amongst others.

Calculation of any test, other than the logrank test, will probably require G12ABF to be called twice, once to calculate the values of $n_i$ and $d_i$ to facilitate in the computation of the required weights, and once to calculate the test statistic itself.

## 9 Example

This example compares the time to death for 51 adults with two different types of recurrent gliomas (brain tumour), astrocytoma and glioblastoma, using a logrank test. For further details on the data see Rostomily *et al.* (1994).

### 9.1 Program Text

```
    Program g12abfe

!     G12ABF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: g12abf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter                 :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)                 :: p, ts
      Integer                            :: df, i, ifail, ldn, lfreq, lwt, n,    &
                                            nd, ngrp
      Character (1)                      :: freq, weight
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable    :: expt(:), obsd(:), t(:), wt(:)
      Integer, Allocatable               :: di(:), grp(:), ic(:), ifreq(:), ni(:)
```

```
!     .. Executable Statements ..
      Write (nout,*) 'G12ABF Example Program Results'
      Write (nout,*)

!     Skip heading in data file
      Read (nin,*)

!     Read in the problem size
      Read (nin,*) n, ngrp, freq, weight

      If (freq=='F' .Or. freq=='f') Then
        lfreq = n
      Else
        lfreq = 0
      End If
      If (weight=='W' .Or. weight=='w') Then
        lwt = n
      Else
        lwt = 0
      End If
      ldn = n
      Allocate (t(n),ic(n),grp(n),ifreq(lfreq),obsd(ngrp),expt(ngrp),di(ldn), &
        ni(ldn),wt(lwt))

!     Read in data
      If (lfreq==0) Then
        Read (nin,*)(t(i),ic(i),grp(i),i=1,n)
      Else
        Read (nin,*)(t(i),ic(i),grp(i),ifreq(i),i=1,n)
      End If

!     Calculate the statistic
      ifail = 0
      Call g12abf(n,t,ic,grp,ngrp,freq,ifreq,weight,wt,ts,df,p,obsd,expt,nd, &
        di,ni,ldn,ifail)

!     Display results
      Write (nout,99999) 'Observed', 'Expected'
      Write (nout,99998)('Group',i,obsd(i),expt(i),i=1,ngrp)
      Write (nout,*)
      Write (nout,99996) 'No. Unique Failure Times = ', nd
      Write (nout,*)
      Write (nout,99997) 'Test Statistic           = ', ts
      Write (nout,99996) 'Degrees of Freedom       = ', df
      Write (nout,99997) 'p-value                  = ', p

99999 Format (11X,A,2X,A)
99998 Format (1X,A,1X,I1,1X,F8.2,2X,F8.2)
99997 Format (1X,A,1X,F8.4)
99996 Format (1X,A,1X,I3)
    End Program g12abfe
```

## 9.2  Program Data

```
G12ABF Example Program Data
51 2 'S' 'U'  :: N, NGRP, FREQ, WEIGHT
  6.0  0  1
 13.0  0  1
 21.0  0  1
 30.0  0  1
 31.0  1  1
 37.0  0  1
 38.0  0  1
 47.0  1  1
 49.0  0  1
 50.0  0  1
 63.0  0  1
 79.0  0  1
 80.0  1  1
 82.0  1  1
```

```
 82.0  1  1
 86.0  0  1
 98.0  0  1
149.0  1  1
202.0  0  1
219.0  0  1
 10.0  0  2
 10.0  0  2
 12.0  0  2
 13.0  0  2
 14.0  0  2
 15.0  0  2
 16.0  0  2
 17.0  0  2
 18.0  0  2
 20.0  0  2
 24.0  0  2
 24.0  0  2
 25.0  0  2
 28.0  0  2
 30.0  0  2
 33.0  0  2
 34.0  1  2
 35.0  0  2
 37.0  0  2
 40.0  0  2
 40.0  0  2
 40.0  1  2
 46.0  0  2
 48.0  0  2
 70.0  1  2
 76.0  0  2
 81.0  0  2
 82.0  0  2
 91.0  0  2
112.0  0  2
181.0  0  2    :: T,IC,GRP
```

## 9.3   Program Results

```
G12ABF Example Program Results

          Observed   Expected
Group 1     14.00      22.48
Group 2     28.00      19.52

No. Unique Failure Times =   36

Test Statistic         =     7.4966
Degrees of Freedom     =     1
p-value                =     0.0062
```

_____