NAG Library Routine Document

G12ZAF

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

G12ZAF creates the risk sets associated with the Cox proportional hazards model for fixed covariates.

2 Specification

```
SUBROUTINE G12ZAF (N, M, NS, Z, LDZ, ISZ, IP, T, IC, ISI, NUM, IXS, NXS,    &
                   X, MXN, ID, ND, TP, IRS, IFAIL)
INTEGER            N, M, NS, LDZ, ISZ(M), IP, IC(N), ISI(*), NUM,           &
                   IXS(MXN), NXS, MXN, ID(MXN), ND, IRS(N), IFAIL
REAL (KIND=nag_wp) Z(LDZ,M), T(N), X(MXN,IP), TP(N)
```

3 Description

The Cox proportional hazards model (see Cox (1972)) relates the time to an event, usually death or failure, to a number of explanatory variables known as covariates. Some of the observations may be right-censored, that is, the exact time to failure is not known, only that it is greater than a known time.

Let t_i, for $i = 1, 2, \ldots, n$, be the failure time or censored time for the ith observation with the vector of p covariates z_i. It is assumed that censoring and failure mechanisms are independent. The hazard function, $\lambda(t, z)$, is the probability that an individual with covariates z fails at time t given that the individual survived up to time t. In the Cox proportional hazards model, $\lambda(t, z)$ is of the form

$$\lambda(t, z) = \lambda_0(t) \exp\left(z^{\mathrm{T}}\beta\right),$$

where λ_0 is the base-line hazard function, an unspecified function of time, and β is a vector of unknown parameters. As λ_0 is unknown, the parameters β are estimated using the conditional or marginal likelihood. This involves considering the covariate values of all subjects that are at risk at the time when a failure occurs. The probability that the subject that failed had their observed set of covariate values is computed.

The risk set at a failure time consists of those subjects that fail or are censored at that time and those who survive beyond that time. As risk sets are computed for every distinct failure time, it should be noted that the combined risk sets may be considerably larger than the original data. If the data can be considered as coming from different strata such that λ_0 varies from strata to strata but β remains constant, then G12ZAF will return a factor that indicates to which risk set/strata each member of the risk sets belongs rather than just to which risk set.

Given the risk sets the Cox proportional hazards model can then be fitted using a Poisson generalized linear model (G02GCF with G04EAF to compute dummy variables) using Breslow's approximation for ties (see Breslow (1974)). This will give the same fit as G12BAF. If the exact treatment of ties in discrete time is required, as given by Cox (1972), then the model is fitted as a conditional logistic model using G11CAF.

4 References

Breslow N E (1974) Covariate analysis of censored survival data *Biometrics* **30** 89–99

Cox D R (1972) Regression models in life tables (with discussion) *J. Roy. Statist. Soc. Ser. B* **34** 187–220

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

5 Parameters

1: N – INTEGER *Input*

On entry: n, the number of data points.

Constraint: N ≥ 2.

2: M – INTEGER *Input*

On entry: the number of covariates in array Z.

Constraint: M ≥ 1.

3: NS – INTEGER *Input*

On entry: the number of strata. If NS > 0 then the stratum for each observation must be supplied in ISI.

Constraint: NS ≥ 0.

4: Z(LDZ, M) – REAL (KIND=nag_wp) array *Input*

On entry: the ith row must contain the covariates which are associated with the ith failure time given in T.

5: LDZ – INTEGER *Input*

On entry: the first dimension of the array Z as declared in the (sub)program from which G12ZAF is called.

Constraint: LDZ $\geq$ N.

6: ISZ(M) – INTEGER array *Input*

On entry: indicates which subset of covariates are to be included in the model.

ISZ(j) ≥ 1
The jth covariate is included in the model.

ISZ(j) $= 0$
The jth covariate is excluded from the model and not referenced.

Constraint: ISZ(j) ≥ 0 and at least one value must be nonzero.

7: IP – INTEGER *Input*

On entry: p, the number of covariates included in the model as indicated by ISZ.

Constraint: IP $=$ the number of nonzero values of ISZ.

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

On entry: the vector of n failure censoring times.

9: IC(N) – INTEGER array *Input*

On entry: the status of the individual at time t given in T.

IC(i) $= 0$
Indicates that the ith individual has failed at time T(i).

IC(i) $= 1$
Indicates that the ith individual has been censored at time T(i).

Constraint: IC(i) $= 0$ or 1, for $i = 1, 2, \ldots,$ N.

10: ISI($*$) – INTEGER array *Input*

Note: the dimension of the array ISI must be at least N if NS > 0, and at least 1 otherwise.

On entry: if NS > 0, the stratum indicators which also allow data points to be excluded from the analysis.

If NS $= 0$, ISI is not referenced.

ISI(i) $= k$
Indicates that the ith data point is in the kth stratum, where $k = 1, 2, \ldots,$ NS.

ISI(i) $= 0$
Indicates that the ith data point is omitted from the analysis.

Constraint: if NS > 0, $0 \leq$ ISI(i) $\leq$ NS, for $i = 1, 2, \ldots,$ N.

11: NUM – INTEGER *Output*

On exit: the number of values in the combined risk sets.

12: IXS(MXN) – INTEGER array *Output*

On exit: the factor giving the risk sets/strata for the data in X and ID.

If NS $= 0$ or 1, IXS(i) $= l$ for members of the lth risk set.

If NS > 1, IXS(i) $= (j - 1) \times$ ND $+ l$ for the observations in the lth risk set for the jth strata.

13: NXS – INTEGER *Output*

On exit: the number of levels for the risk sets/strata factor given in IXS.

14: X(MXN, IP) – REAL (KIND=nag_wp) array *Output*

On exit: the first NUM rows contain the values of the covariates for the members of the risk sets.

15: MXN – INTEGER *Input*

On entry: the first dimension of the array X and the dimension of the arrays IXS and ID as declared in the (sub)program from which G12ZAF is called.

Constraint: MXN must be sufficiently large for the arrays to contain the expanded risk sets. The size will depend on the pattern of failures times and censored times. The minimum value will be returned in NUM unless the routine exits with IFAIL $= 1$ or 2.

16: ID(MXN) – INTEGER array *Output*

On exit: indicates if the member of the risk set given in X failed.

ID(i) $= 1$ if the member of the risk set failed at the time defining the risk set and ID(i) $= 0$ otherwise.

17: ND – INTEGER *Output*

On exit: the number of distinct failure times, i.e., the number of risk sets.

18: TP(N) – REAL (KIND=nag_wp) array *Output*

On exit: TP(i) contains the ith distinct failure time, for $i = 1, 2, \ldots,$ ND.

19: IRS(N) – INTEGER array *Output*

On exit: indicates rows in X and elements in IXS and ID corresponding to the risk sets. The first risk set corresponding to failure time TP(1) is given by rows 1 to IRS(1). The lth risk set is given by rows IRS($l - 1$) $+ 1$ to IRS(l), for $l = 1, 2, \ldots,$ ND.

20: 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, M < 1,
or NS < 0,
or LDZ $<$ N.

IFAIL $= 2$

On entry, ISZ$(i) < 0$ for some i,
or the value of IP is incompatible with ISZ,
or IC$(i) \neq 1$ or 0.
or NS > 0 and ISI$(i) < 0$,
or NS > 1 and ISI$(i) >$ NS.

IFAIL $= 3$

MXN is too small, the minimum value is returned in NUM.

IFAIL $= -99$

An unexpected error has been triggered by this routine. Please contact NAG.

See Section 3.8 in the Essential Introduction for further information.

IFAIL $= -399$

Your licence key may have expired or may not have been installed correctly.

See Section 3.7 in the Essential Introduction for further information.

IFAIL $= -999$

Dynamic memory allocation failed.

See Section 3.6 in the Essential Introduction for further information.

7 Accuracy

Not applicable.

8 Parallelism and Performance

Not applicable.

9 Further Comments

When there are strata present, i.e., NS $>$ 1, not all the NXS groups may be present.

10 Example

The data are the remission times for two groups of leukemia patients (see page 242 of Gross and Clark (1975)). A dummy variable indicates which group they come from. The risk sets are computed using G12ZAF and the Cox's proportional hazard model is fitted using G11CAF.

10.1 Program Text

```
    Program g12zafe

!     G12ZAF Example Program Text.

!     Mark 25 Release. NAG Copyright 2014.

!     .. Use Statements ..
      Use nag_library, Only: g11caf, g12zaf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: dev, tol
      Integer                          :: cm, i, ifail, ip, iprint, ldz, lisi, &
                                          lwk, m, maxit, mxn, n, nd, ns, num,  &
                                          nxs
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: b(:), cov(:), sc(:), se(:), t(:),    &
                                          tp(:), wk(:), x(:,:), z(:,:)
      Integer, Allocatable             :: cnt(:), ic(:), id(:), irs(:),        &
                                          isi(:), isz(:), ixs(:), nca(:), nct(:)
!     .. Intrinsic Procedures ..
      Intrinsic                        :: count, maxval
!     .. Executable Statements ..
      Write (nout,*) 'G12ZAF Example Program Results'
      Write (nout,*)

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

!     Read in problem size
      Read (nin,*) n, m, ns, maxit, iprint

      If (ns>0) Then
        lisi = n
      Else
        lisi = 0
      End If

      ldz = n
      Allocate (z(ldz,m),isz(m),t(n),ic(n),isi(lisi),tp(n),irs(n))

!     Read in the data
      If (ns>0) Then
        Read (nin,*)(t(i),z(i,1:m),ic(i),isi(i),i=1,n)
      Else
        Read (nin,*)(t(i),z(i,1:m),ic(i),i=1,n)
      End If

!     Read in the variable indicator
```

```
      Read (nin,*) isz(1:m)

!     Calculate number of parameters in the model
      ip = count(isz(1:m)>0)

!     Call the routine once to calculate size of MXN ...

!     Dummy allocation
      mxn = 0
      Allocate (x(mxn,ip),id(mxn),ixs(mxn))

!     Call G12ZAF to calculate MXN
      ifail = 1
      Call g12zaf(n,m,ns,z,ldz,isz,ip,t,ic,isi,num,ixs,nxs,x,mxn,id,nd,tp,irs, &
        ifail)
      If (ifail/=0 .And. ifail/=3) Then
        Go To 100
      End If

!     Required size for MXN is returned in NUM, so reallocate memory
      mxn = num
      Deallocate (x,id,ixs)
      Allocate (x(mxn,ip),id(mxn),ixs(mxn))

!     Create risk set
      ifail = 0
      Call g12zaf(n,m,ns,z,ldz,isz,ip,t,ic,isi,num,ixs,nxs,x,mxn,id,nd,tp,irs, &
        ifail)

      Allocate (cnt(nxs),b(ip),se(ip),sc(ip),nca(nxs),nct(nxs),cov(ip*(ip+ &
        1)/2))

!     Set tolerance
      tol = 1.0E-5_nag_wp

!     Read in initial parameter estimates
      Read (nin,*) b(1:ip)

!     Count the number of observations in each stratum
      cnt(1:nxs) = 0
      Do i = 1, num
        cnt(ixs(i)) = cnt(ixs(i)) + 1
      End Do
      cm = maxval(cnt(1:nxs))

      lwk = ip*num + (cm+1)*(ip+1)*(ip+2)/2 + cm
      Allocate (wk(lwk))

!     Get parameter estimates from conditional logistic analysis
      ifail = 0
      Call g11caf(num,ip,nxs,x,mxn,isz,ip,id,ixs,dev,b,se,sc,cov,nca,nct,tol, &
        maxit,iprint,wk,lwk,ifail)

!     Display results
      Write (nout,*) ' Parameter      Estimate', '       Standard Error'
      Write (nout,*)
      Write (nout,99999)(i,b(i),se(i),i=1,ip)

100   Continue

99999 Format (I6,10X,F8.4,10X,F8.4)
    End Program g12zafe
```

10.2 Program Data

```
G12ZAF Example Program Data
42 1 0 20 0       : N,M,NS,MAXIT,IPRINT
 1 0 0
 1 0 0
 2 0 0
 2 0 0
 3 0 0
 4 0 0
 4 0 0
 5 0 0
 5 0 0
 8 0 0
 8 0 0
 8 0 0
 8 0 0
11 0 0
11 0 0
12 0 0
12 0 0
15 0 0
17 0 0
22 0 0
23 0 0
 6 1 0
 6 1 0
 6 1 0
 7 1 0
10 1 0
13 1 0
16 1 0
22 1 0
23 1 0
 6 1 1
 9 1 1
10 1 1
11 1 1
17 1 1
19 1 1
20 1 1
25 1 1
32 1 1
32 1 1
34 1 1
35 1 1            : T,Z,IC
1                 : ISZ
0.0 0.0           : B (G11CAF)
```

10.3 Program Results

```
 G12ZAF Example Program Results

  Parameter      Estimate       Standard Error

     1            1.6282            0.4331
```