NAG C Library Function Document

nag_binary_factor (g11sac)

1 Purpose

nag_binary_factor (g11sac) fits a latent variable model (with a single factor) to data consisting of a set of measurements on individuals in the form of binary-valued sequences (generally referred to as score patterns). Various measures of goodness-of-fit are calculated along with the factor (theta) scores.

2 Specification

```
void nag_binary_factor (Nag_OrderType order, Integer p, Integer n, Boolean gprob,
     Integer ns, Boolean x[], Integer pdx, Integer irl[], double a[], double c[],
     Integer iprint, const char *outfile, double cgetol, Integer maxit, Boolean chisqr,
     Integer *niter, double alpha[], double pigam[], double cm[], Integer pdcm,
     double g[], double expp[], Integer pde, double obs[], double exf[], double y[],
     Integer iob[], double *rlogl, double *chi, Integer *idf, double *siglev,
     NagError *fail)
```

3 Description

Given a set of p dichotomous variables $\tilde{x} = (x_1, x_2, \ldots, x_p)'$, where $'$ denotes vector or matrix transpose, the objective is to investigate whether the association between them can be adequately explained by a latent variable model of the form (see Bartholomew (1980) and Bartholomew (1987))

$$G\{\pi_i(\theta)\} = \alpha_{i0} + \alpha_{i1}\theta. \qquad (1)$$

The x_i are called item responses and take the value 0 or 1. θ denotes the latent variable assumed to have a standard Normal distribution over a population of individuals to be tested on p items. Call $\pi_i(\theta) = P(x_i = 1|\theta)$ the item response function: it represents the probability that an individual with latent ability θ will produce a positive response (1) to item i. α_{i0} and α_{i1} are item parameters which can assume any real values. The set of parameters, α_{i1}, for $i = 1, 2, \ldots, p$, being coefficients of the unobserved variable θ, can be interpreted as 'factor loadings'.

G is a function selected by the user as either Φ^{-1} or logit, mapping the interval (0,1) onto the whole real line. Data from a random sample of n individuals takes the form of the matrices X and R defined below:

$$X_{s\times p} = \begin{bmatrix} x_{11} & x_{12} & \ldots & x_{1p} \\ x_{21} & x_{22} & \ldots & x_{2p} \\ \vdots & \vdots & & \vdots \\ x_{s1} & x_{s2} & \ldots & x_{sp} \end{bmatrix} = \begin{bmatrix} \tilde{x}_1' \\ \tilde{x}_2' \\ \vdots \\ \tilde{x}_s' \end{bmatrix}, \quad R_{s\times 1} = \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_s \end{bmatrix}$$

where $\tilde{x}_l = (x_{l1}, x_{l2}, \ldots, x_{lp})'$ denotes the lth score pattern in the sample, r_l the frequency with which $\tilde{x}_l$ occurs and s the number of different score patterns observed. (Thus $\sum_{l=1}^{s} r_l = n$). It can be shown that the log likelihood function is proportional to

$$\sum_{l=1}^{s} r_l \log P_l,$$

where

$$P_l = P(\tilde{x} = \tilde{x}_l) = \int_{-\infty}^{\infty} P(\tilde{x} = \tilde{x}_l|\theta)\phi(\theta)\, d\theta \qquad (2)$$

($\phi(\theta)$ being the probability density function of a standard Normal random variable).

P_l denotes the unconditional probability of observing score pattern $\tilde{x}_l$. The integral in (2) is approximated using Gauss–Hermite quadrature. If we take $G(z) = \operatorname{logit} z = \log\left(\frac{z}{1-z}\right)$ in (1) and reparametrise as follows,

$$\begin{aligned} \alpha_i &= \alpha_{i1}, \\ \pi_i &= \operatorname{logit}^{-1} \alpha_{i0}, \end{aligned}$$

then (1) reduces to the logit model (see Bartholomew (1980))

$$\pi_i(\theta) = \frac{\pi_i}{\pi_i + (1 - \pi_i)\exp(-\alpha_i\theta)}.$$

If we take $G(z) = \Phi^{-1}(z)$ (where Φ is the cumulative distribution function of a standard Normal random variable) and reparametrise as follows,

$$\begin{aligned} \alpha_i &= \frac{\alpha_{i1}}{\sqrt{(1 + \alpha_{i1}^2)}} \\ \gamma_i &= \frac{-\alpha_{i0}}{\sqrt{(1 + \alpha_{i1}^2)}} \end{aligned},$$

then (1) reduces to the probit model (see Bock and Aitkin (1981))

$$\pi_i(\theta) = \phi\left(\frac{\alpha_i\theta - \gamma_i}{\sqrt{(1 - \alpha_i^2)}}\right).$$

An E-M algorithm (see Bock and Aitkin (1981)) is used to maximize the log likelihood function. The number of quadrature points used is set initially to 10 and once convergence is attained increased to 20.

The theta score of an individual responding in score pattern $\tilde{x}_l$ is computed as the posterior mean, i.e., $E(\theta|\tilde{x}_l)$. For the logit model the component score $X_l = \sum_{j=1}^{p} \alpha_j x_{lj}$ is also calculated. (Note that in calculating the theta scores and measures of goodness-of-fit nag_binary_factor (g11sac) automatically reverses the coding on item j if $\alpha_j < 0$; it is assumed in the model that a response at the one level is showing a higher measure of latent ability than a response at the zero level.)

The frequency distribution of score patterns is required as input data. If the user's data is in the form of individual score patterns (uncounted), then nag_binary_factor_service (g11sbc) may be used to calculate the frequency distribution.

4 References

Bartholomew D J (1980) Factor analysis for categorical data (with Discussion) *J. Roy. Statist. Soc. Ser. B* **42** 293–321

Bartholomew D J (1987) *Latent Variable Models and Factor Analysis* Griffin

Bock R D and Aitkin M (1981) Marginal maximum likelihood estimation of item parameters: Application of an E-M algorithm *Psychometrika* **46** 443–459

5 Parameters

1: **order** – Nag_OrderType *Input*

On entry: the **order** parameter specifies the two-dimensional storage scheme being used, i.e., row-major ordering or column-major ordering. C language defined storage is specified by **order** = **Nag_RowMajor**. See Section 2.2.1.4 of the Essential Introduction for a more detailed explanation of the use of this parameter.

Constraint: **order** = **Nag_RowMajor** or **Nag_ColMajor**.

2: **p** – Integer *Input*

On entry: the number of dichotomous variables, p.

Constraint: $\mathbf{p} \geq 3$.

3: **n** – Integer *Input*

On entry: the number of individuals in the sample, n.

Constraint: $\mathbf{n} \geq 7$.

4: **gprob** – Boolean *Input*

On entry: **gprob** must be set equal to **TRUE** if $G(z) = \Phi^{-1}(z)$ and **FALSE** if $G(z) = \operatorname{logit} z$.

5: **ns** – Integer *Input*

On entry: **ns** must be set equal to the number of different score patterns in the sample, s.

Constraint: $2 \times \mathbf{p} < \mathbf{ns} \leq \min(2^{\mathbf{p}}, \mathbf{n})$.

6: **x**[dim] – Boolean *Input/Output*

Note: the dimension, dim, of the array **x** must be at least $\max(1, \mathbf{pdx} \times \mathbf{p})$ when **order** = **Nag_ColMajor** and at least $\max(1, \mathbf{pdx} \times \mathbf{ns})$ when **order** = **Nag_RowMajor**.

Where $\mathbf{X}(i, j)$ appears in this document, it refers to the array element

if **order** = **Nag_ColMajor**, $\mathbf{x}[(j-1) \times \mathbf{pdx} + i - 1]$;

if **order** = **Nag_RowMajor**, $\mathbf{x}[(i-1) \times \mathbf{pdx} + j - 1]$.

On entry: the first s rows of **x** must contain the s different score patterns. The lth row of **x** must contain the lth score pattern with $\mathbf{X}(l, j)$ set equal to **TRUE** if $x_{lj} = 1$ and **FALSE** if $x_{lj} = 0$. All rows of **x** must be distinct.

On exit: given a valid parameter set then the first s rows of **x** still contain the s different score patterns. However, the following points should be noted:

(i) If the estimated factor loading for the jth item is negative then that item is re-coded, i.e., 0s and 1s (or **TRUE** and **FALSE**) in the jth column of **x** are interchanged.

(ii) The rows of **x** will be re-ordered so that the theta scores corresponding to rows of **x** are in increasing order of magnitude.

7: **pdx** – Integer *Input*

On entry: the stride separating matrix row or column elements (depending on the value of **order**) in the array **x**.

Constraints:

if **order** = **Nag_ColMajor**, $\mathbf{pdx} \geq \mathbf{ns}$;
if **order** = **Nag_RowMajor**, $\mathbf{pdx} \geq \mathbf{p}$.

8: **irl**[**ns**] – Integer *Input/Output*

On entry: the ith component of **irl** must be set equal to the frequency with which the ith row of **x** occurs.

Constraint:

$\mathbf{irl}[i] \geq 0$ for $i = 0, 1, \ldots, s-1$;

$$\sum_{i=0}^{s-1} \mathbf{irl}[i] = n.$$

On exit: given a valid parameter set then the first s components of **irl** are re-ordered as are the rows of **x**.

9: **a**[**p**] – double *Input/Output*

On entry: **a**[$j-1$] must be set equal to an initial estimate of α_{j1}. **In order to avoid divergence problems with the E-M algorithm the user is strongly advised to set all the a**[$j-1$] **to 0.5.**

On exit: **a**[$j-1$] contains the latest estimate of α_{j1}, for $j = 1, 2, \ldots, p$. (Because of possible recoding all elements of **a** will be positive.)

10: **c**[**p**] – double *Input/Output*

On entry: **c**[$j-1$] must be set equal to an initial estimate of α_{j0}. **In order to avoid divergence problems with the E-M algorithm the user is strongly advised to set all the c**[$j-1$] **to 0.0.**

On exit: **c**[$j-1$] contains the latest estimate of α_{j0}, for $j = 1, 2, \ldots, p$.

11: **iprint** – Integer *Input*

On entry: the frequency with which the maximum likelihood search routine is to be monitored.

If **iprint** > 0, the search is monitored once every **iprint** iterations, and when the number of quadrature points is increased, and again at the final solution point.

If **iprint** $= 0$, the search is monitored once at the final point.

If **iprint** < 0, the search is not monitored at all.

iprint should normally be set to a small positive number.

Suggested value: **iprint** $= 1$.

12: **outfile** – char * *Input*

On entry: the name of a file to which diagnostic output will be directed. If **outfile** is **NULL** the diagnostic output will be directed to standard output.

13: **cgetol** – double *Input*

On entry: the accuracy to which the solution is required. If **cgetol** is set to 10^{-l} and on exit **fail.code** = **NE_NOERROR** or **NE_ZERO_DF**, then all elements of the gradient vector will be smaller than 10^{-l} in absolute value. For most practical purposes the value 10^{-4} should suffice. The user should be wary of setting **cgetol** too small since the convergence criterion may then have become too strict for the machine to handle. If **cgetol** has been set to a value which is less than the square root of the ***machine precision***, ϵ, then nag_binary_factor (g11sac) will use the value $\sqrt{\epsilon}$ instead.

14: **maxit** – Integer *Input*

On entry: the maximum number of iterations to be made in the maximum likelihood search. There will be an error exit (see Section 6) if the search routine has not converged in **maxit** iterations.

Constraint: **maxit** ≥ 1.

Suggested value: **maxit** $= 1000$.

15: **chisqr** – Boolean *Input*

On entry: if **chisqr** is set equal to **TRUE**, then a likelihood ratio statistic will be calculated (see **chi**).

If **chisqr** is set equal to **FALSE**, no such statistic will be calculated.

16: **niter** – Integer * *Output*

On exit: given a valid parameter set then **niter** contains the number of iterations performed by the maximum likelihood search routine.

17: **alpha**[**p**] – double *Output*

On exit: given a valid parameter set then **alpha**$[j-1]$ contains the latest estimate of α_j. (Because of possible recoding all elements of **alpha** will be positive.)

18: **pigam**[**p**] – double *Output*

On exit: given a valid parameter set then **pigam**$[j-1]$ contains the latest estimate of either π_j if **gprob** = **FALSE** (logit model) or γ_j if **gprob** = **TRUE** (probit model).

19: **cm**[*dim*] – double *Output*

Note: the dimension, *dim*, of the array **cm** must be at least **pdcm** × 2 × **p**.

Where **CM**(i,j) appears in this document, it refers to the array element

if **order** = **Nag_ColMajor**, **cm**$[(j-1) \times$ **pdcm** $+ i - 1]$;

if **order** = **Nag_RowMajor**, **cm**$[(i-1) \times$ **pdcm** $+ j - 1]$.

On exit: given a valid parameter set then the strict lower triangle of **cm** contains the correlation matrix of the parameter estimates held in **alpha** and **pigam** on exit. The diagonal elements of **cm** contain the standard errors. Thus:

$$\begin{array}{lcl} \mathbf{CM}(2\times i-1, 2\times i-1) & = & \text{standard error } (\mathbf{alpha}[i-1]) \\ \mathbf{CM}(2\times i, 2\times i) & = & \text{standard error } (\mathbf{pigam}[i-1]) \\ \mathbf{CM}(2\times i, 2\times i-1) & = & \text{correlation } (\mathbf{pigam}[i-1], \mathbf{alpha}[i-1]), \end{array}$$

for $i = 1, 2, \ldots, p$;

$$\begin{array}{lcl} \mathbf{CM}(2\times i-1, 2\times j-1) & = & \text{correlation } (\mathbf{alpha}[i-1], \mathbf{alpha}[j-1]) \\ \mathbf{CM}(2\times i, 2\times j) & = & \text{correlation } (\mathbf{pigam}[i-1], \mathbf{pigam}[j-1]) \\ \mathbf{CM}(2\times i-1, 2\times j) & = & \text{correlation } (\mathbf{alpha}[i-1], \mathbf{pigam}[j-1]) \\ \mathbf{CM}(2\times i, 2\times j-1) & = & \text{correlation } (\mathbf{alpha}[j-1], \mathbf{pigam}[i-1]), \end{array}$$

for $j = 1, 2, \ldots, i-1$.

If the second derivative matrix cannot be computed then all the elements of **cm** are returned as zero.

20: **pdcm** – Integer *Input*

On entry: the stride separating row or column elements (depending on the value of **order**) of the matrix M in the array **cm**.

Constraint: **pdcm** ≥ 2 × **p**.

21: **g**[*dim*] – double *Output*

Note: the dimension, *dim*, of the array **g** must be at least 2 × **p**.

On exit: given a valid parameter set then **g** contains the estimated gradient vector corresponding to the final point held in the arrays **alpha** and **pigam**. **g**$[2\times j-2]$ contains the derivative of the log likelihood with respect to **alpha**$[j-1]$, for $j = 1, 2, \ldots, p$. **g**$[2\times j-1]$ contains the derivative of the log likelihood with respect to **pigam**$[j-1]$, for $j = 1, 2, \ldots, p$.

22: **expp**[*dim*] – double *Output*

Note: the dimension, *dim*, of the array **expp** must be at least **pde** × **p**.

Where **EXPP**(i,j) appears in this document, it refers to the array element

if **order** = **Nag_ColMajor**, **expp**$[(j-1) \times$ **pde** $+ i - 1]$;

if **order** = **Nag_RowMajor**, **expp**$[(i-1) \times$ **pde** $+ j - 1]$.

On exit: given a valid parameter set then **EXPP**(i,j) contains the expected percentage of individuals in the sample who respond positively to items i and j ($j \leq i$), corresponding to the final point held in the arrays **alpha** and **pigam**.

23: **pde** – Integer *Input*

On entry: the stride separating row or column elements (depending on the value of **order**) of the matrix E in the array **expp**.

Constraint: **pde** $\geq$ **p**.

24: **obs**[dim] – double *Output*

Note: the dimension, dim, of the array **obs** must be at least **pde** $\times$ **p**.

Where **OBS**(i, j) appears in this document, it refers to the array element

if **order** = **Nag_ColMajor**, **obs**$[(j-1) \times$ **pde** $+ i - 1]$;

if **order** = **Nag_RowMajor**, **obs**$[(i-1) \times$ **pde** $+ j - 1]$.

On exit: given a valid parameter set then **OBS**(i, j) contains the observed percentage of individuals in the sample who responded positively to items i and j ($j \leq i$).

25: **exf**[**ns**] – double *Output*

On exit: given a valid parameter set then **exf**$[l-1]$ contains the expected frequency of the lth score pattern (lth row of **x**), corresponding to the final point held in the arrays **alpha** and **pigam**.

26: **y**[**ns**] – double *Output*

On exit: given a valid parameter set then **y**$[l-1]$ contains the estimated theta score corresponding to the lth row of **x**, for the final point held in the arrays **alpha** and **pigam**.

27: **iob**[**ns**] – Integer *Output*

On exit: given a valid parameter set then **iob**$[l-1]$ contains the number of items in the lth row of **x** for which the response was positive (**TRUE**).

28: **rlogl** – double * *Output*

On exit: given a valid parameter set then **rlogl** contains the value of the log likelihood kernel corresponding to the final point held in the arrays **alpha** and **pigam**, namely

$$\sum_{l=0}^{s-1} \mathbf{irl}[l] \times \log(\mathbf{exf}[l]/n).$$

29: **chi** – double * *Output*

On exit: if **chisqr** was set equal to **TRUE** on entry, then given a valid parameter set, **chi** will contain the value of the likelihood ratio statistic corresponding to the final parameter estimates held in the arrays **alpha** and **pigam**, namely

$$2 \times \sum_{l=0}^{s-1} \mathbf{irl}[l] \times \log(\mathbf{exf}[l]/\mathbf{irl}[l]).$$

The summation is over those elements of **irl** which are positive. If **exf**$[l-1]$ is less than 5.0, then adjacent score patterns are pooled (the score patterns in **x** being first put in order of increasing theta score).

If **chisqr** has been set equal to **FALSE**, then **chi** is not used.

30: **idf** – Integer * *Output*

On exit: if **chisqr** was set equal to **TRUE** on entry, then given a valid parameter set, **idf** will contain the degrees of freedom associated with the likelihood ratio statistic, **chi**.

$$\begin{array}{ll} \mathbf{idf} = s_0 - 2 \times p & \text{if } s_0 < 2^p; \\ \mathbf{idf} = s_0 - 2 \times p - 1 & \text{if } s_0 = 2^p, \end{array}$$

where s_0 denotes the number of terms summed to calculate **chi** ($s_0 = s$ only if there is no pooling).

If **chisqr** has been set equal to **FALSE**, then **idf** is not used.

31: **siglev** – double * *Output*

On exit: if **chisqr** was set equal to **TRUE** on entry, then given a valid parameter set, **siglev** will contain the significance level of **chi** based on **idf** degrees of freedom. If **idf** is zero or negative then **siglev** is set to zero. If **chisqr** was set equal to **FALSE**, then **siglev** is not used.

32: **fail** – NagError * *Input/Output*

The NAG error parameter (see the Essential Introduction).

6 Error Indicators and Warnings

NE_INT

On entry, **p** = ⟨*value*⟩.
Constraint: **p** ≥ 3.

On entry, **pdx** = ⟨*value*⟩.
Constraint: **pdx** > 0.

On entry, **pdcm** = ⟨*value*⟩.
Constraint: **pdcm** > 0.

On entry, **pde** = ⟨*value*⟩.
Constraint: **pde** > 0.

On entry, **n** = ⟨*value*⟩.
Constraint: **n** ≥ 7.

On entry, **maxit** = ⟨*value*⟩.
Constraint: **maxit** ≥ 1.

NE_INT_2

On entry, **pdx** = ⟨*value*⟩, **ns** = ⟨*value*⟩.
Constraint: **pdx** $\geq$ **ns**.

On entry, **pdx** = ⟨*value*⟩, **p** = ⟨*value*⟩.
Constraint: **pdx** $\geq$ **p**.

On entry, **pdcm** = ⟨*value*⟩, **p** = ⟨*value*⟩.
Constraint: **pdcm** $\geq 2 \times$ **p**.

On entry, **pde** = ⟨*value*⟩, **p** = ⟨*value*⟩.
Constraint: **pde** $\geq$ **p**.

On entry, **ns** $> 2^{\mathbf{p}}$: **ns** = ⟨*value*⟩, **p** = ⟨*value*⟩.

On entry, **irl**$[0] + \cdots +$ **irl**[**ns** $- 1$] is not equal to **n**: **irl**$[0] + \cdots +$ **irl**[**ns** $- 1$] = ⟨*value*⟩, **n** = ⟨*value*⟩.

On entry, **ns** $>$ **n**: **ns** = ⟨*value*⟩, **n** = ⟨*value*⟩.

On entry, **irl**$[i-1] < 0$: i = ⟨*value*⟩, **irl**$[i-1]$ = ⟨*value*⟩.

On entry, rows i and j of **x** are identical: i = ⟨*value*⟩, j = ⟨*value*⟩.

On entry, **ns** $\leq 2 \times$ **p**: **ns** = ⟨*value*⟩, **p** = ⟨*value*⟩.

NE_INT_3

On entry, **p** = ⟨*value*⟩, **n** = ⟨*value*⟩, **ns** = ⟨*value*⟩.
Constraint: $2 \times$ **p** $<$ **ns** $\leq \min(2^{\mathbf{p}}, \mathbf{n})$.

NE_MAT_INV

Failure to invert Hessian matrix plus Heywood case encountered.

Failure to invert Hessian matrix and **maxit** iterations made: **maxit** = ⟨*value*⟩.

NE_REAL_ARRAY_ELEM_CONS

One of the elements of a has exceeded 10 in absolute value (Heywood case).

NE_RESPONSE_LEVEL

For at least one of the **p** items the responses are all at the same level.

NE_TOO_MANY_ITER

maxit iterations have been performed: **maxit** = ⟨*value*⟩.

NE_ZERO_DF

Chi-squared statistic has **idf** degrees of freedom: **idf** = ⟨*value*⟩.

NE_ALLOC_FAIL

Memory allocation failed.

NE_BAD_PARAM

On entry, parameter ⟨*value*⟩ had an illegal value.

NE_NOT_WRITE_FILE

Cannot open file ⟨*value*⟩ for writing.

NE_NOT_CLOSE_FILE

Cannot close file ⟨*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.

7 Accuracy

On exit from nag_binary_factor (g11sac) if **fail.code** = **NE_NOERROR** or **NE_ZERO_DF** then the following condition will be satisfied:

$$\max_{0 \leq i \leq 2 \times p-1} \{|\mathbf{g}[i]|\} < \mathbf{cgetol}.$$

If **fail.code** = **NE_TOO_MANY_ITER** or **NE_MAT_INV** on exit (i.e., **maxit** iterations have been performed but the above condition does not hold), then the elements in **a**, **c**, **alpha** and **pigam** may still be good approximations to the maximum likelihood estimates. The user is advised to inspect the elements of **g** to see whether this is confirmed.

8 Further Comments

8.1 Timing

The number of iterations required in the maximum likelihood search depends upon the number of observed variables, p, and the distance of the user-supplied starting point from the solution. The number of multiplications and divisions performed in an iteration is proportional to p.

8.2 Initial Estimates

The user is strongly advised to use the recommended starting values for the elements of **a** and **c**. Divergence may result from user-supplied values even if they are very close to the solution. Divergence may also occur when an item has nearly all its responses at one level.

8.3 Heywood Cases

As in normal factor analysis, Heywood cases can often occur, particularly when p is small and n not very big. To overcome this difficulty the maximum likelihood search routine is terminated when the absolute value of one of the α_{j1} exceeds 10.0. The user has the option of deciding whether to exit from nag_binary_factor (g11sac) (by setting **fail** = **NAGERR_DEFAULT** on entry) or to permit nag_binary_factor (g11sac) to proceed onwards as if it had exited normally from the maximum likelihood search routine (setting **fail.print** = **TRUE** or **FALSE** on entry). The elements in **a**, **c**, **alpha** and **pigam** may still be good approximations to the maximum likelihood estimates. The user is advised to inspect the elements **g** to see whether this is confirmed.

8.4 Goodness of Fit Statistic

When n is not very large compared to s a goodness-of-fit statistic should not be calculated as many of the expected frequencies will then be less than 5.

8.5 First and Second Order Margins

The observed and expected **percentages** of sample members responding to individual and pairs of items held in the arrays **obs** and **expp** on exit can be converted to observed and expected **numbers** by multiplying all elements of these two arrays by $n/100.0$.

9 Example

A program to fit the logit latent variable model to the following data:

Index	Score Pattern	Observed Frequency
1	0000	154
2	1000	11
3	0001	42
4	0100	49
5	1001	2
6	1100	10
7	0101	27
8	0010	84
9	1101	10
10	1010	25
11	0011	75
12	0110	129
13	1011	30
14	1110	50
15	0111	181
16	1111	121
Total		1000

9.1 Program Text

```
/* nag_binary_factor (g11sac) Example Program.
 *
 * Copyright 2002 Numerical Algorithms Group.
 *
 * Mark 7, 2002.
 */
```

```
#include <stdio.h>
#include <nag.h>
#include <nag_stdlib.h>
#include <nagg11.h>

int main(void)
{
  /* Scalars */
  double cgetol, chi, rlogl, siglev;
  Integer exit_status, i, pdcm, idf, p, iprint, is,
    j, maxit, n, niter, nrx, lw, pdx;
  NagError fail;
  Nag_OrderType order;
  Boolean chisqr, gprob;
  char flag;

  /* Arrays */
  double *a = 0, *alpha = 0, *c = 0, *cm = 0, *exf = 0, *expp = 0,
    *g = 0, *obs = 0, *pigam = 0, *xl = 0, *y = 0;
  Integer *iob = 0, *irl = 0;
  Boolean *x = 0;

#ifdef NAG_COLUMN_MAJOR
#define X(I,J) x[(J-1)*pdx + I - 1]
#define CM(I,J) cm[(J-1)*pdcm + I - 1]
  order = Nag_ColMajor;
#else
#define X(I,J) x[(I-1)*pdx + J - 1]
#define CM(I,J) cm[(I-1)*pdcm + J - 1]
  order = Nag_RowMajor;
#endif

  INIT_FAIL(fail);
  exit_status = 0;
  Vprintf("g11sac Example Program Results\n");

  /* Skip heading in data file */
  Vscanf("%*[^\n] ");

  Vscanf("%ld%ld%ld%*[^\n] ", &p, &n, &is);
  if (p > 0 && is >= 0)
    {
      /* Allocate arrays */
      pdcm = 2*p;
      nrx = is;
      lw = 4 * p * (p + 16);
      if ( !(a = NAG_ALLOC(p, double)) ||
           !(alpha = NAG_ALLOC(p, double)) ||
           !(c = NAG_ALLOC(p, double)) ||
           !(cm = NAG_ALLOC(pdcm * 2*p, double)) ||
           !(exf = NAG_ALLOC(is, double)) ||
           !(expp = NAG_ALLOC(p * p, double)) ||
           !(g = NAG_ALLOC(2*p, double)) ||
           !(obs = NAG_ALLOC(p * p, double)) ||
           !(pigam = NAG_ALLOC(p, double)) ||
           !(xl = NAG_ALLOC(is, double)) ||
           !(y = NAG_ALLOC(is, double)) ||
           !(iob = NAG_ALLOC(is, Integer)) ||
           !(irl = NAG_ALLOC(is, Integer)) ||
           !(x = NAG_ALLOC(nrx * p, Boolean)) )
        {
          Vprintf("Allocation failure\n");
          exit_status = -1;
          goto END;
        }

      if (order == Nag_ColMajor)
        pdx = nrx;
      else
        pdx = p;
```

```
      for (i = 1; i <= is; ++i)
        {
          Vscanf("%ld", &irl[i-1]);
          for (j = 1; j <= p; ++j)
            {
              Vscanf(" %c", &flag);
              X(i,j) = (flag == 'T');
            }
          Vscanf("%*[^\n] ");
        }
      gprob = FALSE;
      for (i = 1; i <= p; ++i)
        {
          a[i-1] = 0.5;
          c[i-1] = 0.0;
        }

      /* Set iprint > 0 to obtain intermediate output */
      iprint = -1;
      cgetol = 1e-4;
      maxit = 1000;
      chisqr = TRUE;

      g11sac(order, p, n, gprob, is, x, pdx, irl, a, c, iprint, 0,
             cgetol, maxit, chisqr, &niter, alpha, pigam, cm,
             pdcm, g, expp, p, obs, exf, y, iob, &rlogl, &chi,
             &idf, &siglev,  &fail);
      if (fail.code != NE_NOERROR)
        {
          Vprintf("Error from g11sac.\n%s\n",   fail.message);
          exit_status = 1;
          goto END;
        }

      Vprintf("\n");
      Vprintf("Item     Alpha      (s.e.)          Pi       (s.e.)\n");
      for (i=1; i<=p; i++)
         Vprintf("  %ld      %g  (%10g)   %g  (%10g)\n", i, alpha[i-1], CM(2*i-
1,2*i-1),
  pigam[i-1], CM(2*i,2*i));
      Vprintf("\n");
      Vprintf("Index   Observed   Expected     Theta    Pattern\n");
      Vprintf("        Frequency  Frequency    Score\n");
      for (i=1; i<=is; i++)
        {
          Vprintf("  %2ld       %3ld      %7g  %10g    ", i, irl[i-1], exf[i-1],
y[i-1]);
          for (j=1; j<=p; j++)
            Vprintf("%s",X(i,j)==1?"T":"F");
          Vprintf("\n");
        }
      Vprintf("\n");
      Vprintf("Chi-squared test statistic = %g\n", chi);
      Vprintf("Degrees of freedom =         %ld\n",idf);
      Vprintf("Significance =               %g\n",siglev);
    }

 END:
  if (a) NAG_FREE(a);
  if (alpha) NAG_FREE(alpha);
  if (c) NAG_FREE(c);
  if (cm) NAG_FREE(cm);
  if (exf) NAG_FREE(exf);
  if (expp) NAG_FREE(expp);
  if (g) NAG_FREE(g);
  if (obs) NAG_FREE(obs);
  if (pigam) NAG_FREE(pigam);
  if (xl) NAG_FREE(xl);
  if (y) NAG_FREE(y);
  if (iob) NAG_FREE(iob);
  if (irl) NAG_FREE(irl);
```

```
  if (x) NAG_FREE(x);

  return exit_status;
}
```

9.2 Program Data

```
g11sac Example Program Data
 4 1000 16
 154 F F F F
  11 T F F F
  42 F F F T
  49 F T F F
   2 T F F T
  10 T T F F
  27 F T F T
  84 F F T F
  10 T T F T
  25 T F T F
  75 F F T T
 129 F T T F
  30 T F T T
  50 T T T F
 181 F T T T
 121 T T T T
```

9.3 Program Results

```
g11sac Example Program Results

Item     Alpha      (s.e.)          Pi       (s.e.)
  1     1.04546  (  0.148087)   0.218165  ( 0.0173623)
  2     1.40938  (  0.178937)   0.604378  ( 0.0216392)
  3     2.65916  (  0.524787)   0.834117  ( 0.0357898)
  4     1.12169  (  0.139581)   0.484569  ( 0.0198529)

Index   Observed   Expected     Theta    Pattern
        Frequency  Frequency    Score
   1       154      147.061    -1.27348    FFFF
   2        11      13.4437   -0.873074    TFFF
   3        42      42.4201   -0.846239    FFFT
   4        49       54.818   -0.746856    FTFF
   5         2      5.88558   -0.494146    TFFT
   6        10      8.41022   -0.399461    TTFF
   7        27      27.5115   -0.374319    FTFT
   8        84      92.0619    -0.33196    FFTF
   9        10      6.23651  -0.0186861    TTFT
  10        25      21.8468   0.0272335    TFTF
  11        75      73.8352   0.0549022    FFTT
  12       129      123.766    0.161802    FTTF
  13        30      26.8989    0.465873    TFTT
  14        50      50.8813    0.591349    TTTF
  15       181      179.564    0.625634    FTTT
  16       121       125.36     1.14441    TTTT

Chi-squared test statistic = 9.02731
Degrees of freedom =         7
Significance =               0.250701
```