NAG Toolbox for MATLAB

g13dc

1 Purpose

g13dc fits a vector autoregressive moving average (VARMA) model to an observed vector of time series using the method of Maximum Likelihood (ML). Standard errors of parameter estimates are computed along with their appropriate correlation matrix. The function also calculates estimates of the residual series.

2 Syntax

```
[par, qq, niter, rlogl, v, g, cm, ifail] = g13dc(ip, iq, mean, par, qq,
w, parhld, exact, iprint, cgetol, ishow, work, lwork, iw, liw, 'k', k,
'n', n, 'npar', npar, 'maxcal', maxcal)
```

3 Description

Let $W_t = (w_{1t}, w_{2t}, \ldots, w_{kt})^{\mathrm{T}}$, for $t = 1, 2, \ldots, n$, denote a vector of k time series which is assumed to follow a multivariate ARMA model of the form

$$\begin{aligned} W_t - \mu = \; & \phi_1(W_{t-1} - \mu) + \phi_2(W_{t-2} - \mu) + \cdots + \phi_p(W_{t-p} - \mu) \\ & + \epsilon_t - \theta_1\epsilon_{t-1} - \theta_2\epsilon_{t-2} - \cdots - \theta_q\epsilon_{t-q} \end{aligned} \qquad (1)$$

where $\epsilon_t = (\epsilon_{1t}, \epsilon_{2t}, \ldots, \epsilon_{kt})^{\mathrm{T}}$, for $t = 1, 2, \ldots, n$, is a vector of k residual series assumed to be Normally distributed with zero mean and positive-definite covariance matrix Σ. The components of ϵ_t are assumed to be uncorrelated at non-simultaneous lags. The ϕ_i and θ_j are k by k matrices of parameters. $\{\phi_i\}$, for $i = 1, 2, \ldots, p$, are called the autoregressive (AR) parameter matrices, and $\{\theta_i\}$, for $i = 1, 2, \ldots, q$, the moving average (MA) parameter matrices. The parameters in the model are thus the p (k by k) ϕ-matrices, the q (k by k) θ-matrices, the mean vector, μ, and the residual error covariance matrix Σ. Let

$$A(\phi) = \begin{bmatrix} \phi_1 & I & 0 & . & . & . & 0 \\ \phi_2 & 0 & I & 0 & . & . & 0 \\ . & & & . & & & \\ . & & & & . & & \\ . & & & & & . & \\ \phi_{p-1} & 0 & . & . & . & 0 & I \\ \phi_p & 0 & . & . & . & 0 & 0 \end{bmatrix}_{pk\times pk} \quad \text{and} \quad B(\theta) = \begin{bmatrix} \theta_1 & I & 0 & . & . & . & 0 \\ \theta_2 & 0 & I & 0 & . & . & 0 \\ . & & & . & & & \\ . & & & & . & & \\ . & & & & & . & \\ \theta_{q-1} & 0 & . & . & . & . & I \\ \theta_q & 0 & . & . & . & . & 0 \end{bmatrix}_{qk\times qk}$$

where I denotes the k by k identity matrix.

The ARMA model (1) is said to be stationary if the eigenvalues of $A(\phi)$ lie inside the unit circle. Similarly, the ARMA model (1) is said to be invertible if the eigenvalues of $B(\theta)$ lie inside the unit circle.

The method of computing the exact likelihood function (using a Kalman filter algorithm) is discussed in Shea 1987. A quasi-Newton algorithm (see Gill and Murray 1972) is then used to search for the maximum of the log-likelihood function. Stationarity and invertibility are enforced on the model using the reparameterisation discussed in Ansley and Kohn 1986. Conditional on the maximum likelihood estimates being equal to their true values the estimates of the residual series are uncorrelated with zero mean and constant variance Σ.

You have the option of setting a parameter (**exact** to **false**) so that g13dc calculates conditional maximum likelihood estimates (conditional on $W_0 = W_{-1} = \cdots = W_{1-p} = \epsilon_0 = \epsilon_{-1} = \cdots = \epsilon_{1-q} = 0$). This may be useful if the exact maximum likelihood estimates are close to the boundary of the invertibility region.

You also have the option (see Section 5) of requesting g13dc to constrain elements of the ϕ and θ matrices and μ vector to have pre-specified values.

4 References

Ansley C F and Kohn R 1986 A note on reparameterising a vector autoregressive moving average model to enforce stationarity *J. Statist. Comput. Simulation* **24** 99–106

Gill P E and Murray W 1972 Quasi-Newton methods for unconstrained optimization *J. Inst. Math. Appl.* **9** 91–108

Shea B L 1987 Estimation of multivariate time series *J. Time Ser. Anal.* **8** 95–110

5 Parameters

5.1 Compulsory Input Parameters

1: **ip – int32 scalar**

p, the number of AR parameter matrices.

Constraint: **ip** ≥ 0.

2: **iq – int32 scalar**

q, the number of MA parameter matrices.

Constraint: **iq** ≥ 0.

ip $=$ **iq** $= 0$ is **not permitted**.

3: **mean – logical scalar**

mean $=$ **true**, if components of μ have been estimated and **mean** $=$ **false**, if all elements of μ are to be taken as zero.

Constraint: **mean** $=$ **true** or **false**.

4: **par**(**npar**) **– double array**

Initial parameter estimates read in row by row in the order $\phi_1, \phi_2, \ldots, \phi_p$, $\theta_1, \theta_2, \ldots, \theta_q, \mu$.

Thus,

if **ip** > 0, **par**$((l-1) \times k \times k + (i-1) \times k + j)$ must be set equal to an initial estimate of the (i,j)th element of ϕ_l, for $l = 1, 2, \ldots, p$ and $i,j = 1, 2, \ldots, k$;
if **iq** > 0, **par**$(p \times k \times k + (l-1) \times k \times k + (i-1) \times k + j)$ must be set equal to an initial estimate of the (i,j)th element of θ_l, $l = 1, 2, \ldots, q$ and $i,j = 1, 2, \ldots, k$;
if **mean** $=$ 'TRUE', **par**$((p+q) \times k \times k + i)$ should be set equal to an initial estimate of the ith component of μ $(\mu(i))$. (If you set **par**$((p+q) \times k \times k + i)$ to 0.0 then g13dc will calculate the mean of the ith series and use this as an initial estimate of $\mu(i)$.)

The first $p \times k \times k$ elements of **par** must satisfy the stationarity condition and the next $q \times k \times k$ elements of **par** must satisfy the invertibility condition.

If in doubt set all elements of **par** to 0.0.

5: **qq**(**kmax,k**) **– double array**

kmax, the first dimension of the array, must be at least **k**.

qq(i,j) must be set equal to an initial estimate of the (i,j)th element of Σ. The lower triangle only is needed. **qq** must be positive-definite. It is strongly recommended that on entry the elements of **qq** are of the same order of magnitude as at the solution point. If you set **qq**$(i,j) = 0.0$, for $i = 1, 2, \ldots, k$ and $j = 1, 2, \ldots, i$, then g13dc will calculate the covariance matrix between the k time series and use this as an initial estimate of Σ.

6: **w**(**kmax,n**) **– double array**

kmax, the first dimension of the array, must be at least **k**.

w(i, t) must be set equal to the ith component of W_t, for $i = 1, 2, \ldots, k$ and $t = 1, 2, \ldots, n$.

7: **parhld**(**npar**) – **logical array**

parhld(i) must be set to **true**, if **par**(i) is to be held constant at its input value and **false**, if **par**(i) is a free parameter, for $i = 1, 2, \ldots,$ **npar**.

If in doubt try setting all elements of **parhld** to **false**.

8: **exact – logical scalar**

Must be set equal to **true** if you wish the function to compute exact maximum likelihood estimates. **exact** must be set equal to **false** if only conditional likelihood estimates are required.

9: **iprint – int32 scalar**

The frequency with which the automatic monitoring function is to be called. See Section missing entity g13dcexample for an example of the printed output.

iprint > 0

The ML search procedure is monitored once every **iprint** iterations and just before exit from the search function.

iprint $= 0$

The search function is monitored once at the final point.

iprint < 0

The search function is not monitored at all.

10: **cgetol – double scalar**

The accuracy to which the solution in **par** and **qq** is required.

If **cgetol** is set to 10^{-l} and on exit **ifail** $= 0$ or **ifail** ≥ 6, then all the elements in **par** and **qq** should be accurate to approximately l decimal places. For most practical purposes the value 10^{-4} should suffice. You should be wary of setting **cgetol** too small since the convergence criteria may then have become too strict for the machine to handle.

If **cgetol** has been set to a value which is less than the ***machine precision***, ϵ, then g13dc will use the value $10.0 \times \sqrt{\epsilon}$ instead.

11: **ishow – int32 scalar**

Specifies which of the following two quantities are to be printed.

(i) table of maximum likelihood estimates and their standard errors (as returned in the output arrays **par**, **qq** and **cm**);

(ii) table of residual series (as returned in the output array **v**).

ishow $= 0$

None of the above are printed.

ishow $= 1$

only is printed.

ishow $= 2$

and are printed.

Constraint: $0 \leq$ **ishow** ≤ 2.

12: **work(1) – double array**

13: **lwork – int32 scalar**

14: **iw(1) – int32 array**

15: **liw – int32 scalar**

These parameters are no longer accessed by g13dc. Workspace is provided internally by dynamic allocation instead.

5.2 Optional Input Parameters

1: **k – int32 scalar**

Default: The dimension of the array **qq**.

k, the number of observed time series.

Constraint: **k** ≥ 1.

2: **n – int32 scalar**

Default: The dimension of the arrays **w**, **v**. (An error is raised if these dimensions are not equal.)

n, the number of observations in each time series.

Constraint: **n** ≥ 3.

3: **npar – int32 scalar**

Default: The dimension of the arrays **par**, **parhld**, **g** and the second dimension of the array **cm**. (An error is raised if these dimensions are not equal.)

npar is the number of initial parameter estimates. The total number of observations $(n \times k)$ must exceed the total number of parameters in the model ($\mathbf{npar} + k(k+1)/2$).

Constraints:

if **mean** = **true**, **npar** must be set equal to $k_6 = (p+q) \times k \times k$;
if **mean** = **false**, **npar** must be set equal to $k_6 = (p+q) \times k \times k + k$.

4: **maxcal – int32 scalar**

The maximum number of likelihood evaluations to be permitted by the search procedure.

Constraint: **maxcal** ≥ 1.

Suggested value: $\mathbf{maxcal} = 40 \times \mathbf{npar} \times (\mathbf{npar} + 5)$.

Default: $40 \times \mathbf{npar} \times (\mathbf{npar} + 5)$

5.3 Input Parameters Omitted from the MATLAB Interface

kmax, ldcm

5.4 Output Parameters

1: **par(npar) – double array**

If **ifail** $= 0$ or **ifail** ≥ 4 then all the elements of **par** will be overwritten by the latest estimates of the corresponding ARMA parameters.

2: **qq(kmax,k) – double array**

If **ifail** $= 0$ or **ifail** ≥ 4 then **qq**(i,j) will contain the latest estimate of the (i,j)th element of Σ. The lower triangle only is returned.

3: **niter – int32 scalar**

If **ifail** $= 0$ or **ifail** ≥ 4 then **niter** contains the number of iterations performed by the search function.

4: **rlogl – double scalar**

If **ifail** $= 0$ or **ifail** ≥ 4 then **rlogl** contains the value of the log-likelihood function corresponding to the final point held in **par** and **qq**.

5: **v(kmax,n) – double array**

If **ifail** $= 0$ or **ifail** ≥ 4 then $\mathbf{v}(i,t)$ will contain an estimate of the ith component of ϵ_t, for $i = 1, 2, \ldots, k$ and $t = 1, 2, \ldots, n$, corresponding to the final point held in **par** and **qq**.

6: **g(npar) – double array**

If **ifail** $= 0$ or **ifail** ≥ 4 then $\mathbf{g}(i)$ will contain the estimated first derivative of the log-likelihood function with respect to the ith element in the array **par**. If the gradient cannot be computed then all the elements of **g** are returned as zero.

7: **cm(ldcm,npar) – double array**

If **ifail** $= 0$ or **ifail** ≥ 4 then $\mathbf{cm}(i,j)$ will contain an estimate of the correlation coefficient between the ith and jth elements in the **par** array for $1 \leq i \leq \mathbf{npar}$, $1 \leq j \leq \mathbf{npar}$. If $i = j$, then $\mathbf{cm}(i,j)$ will contain the estimated standard error of $\mathbf{par}(i)$. If the lth component of **par** has been held constant, i.e., $\mathbf{parhld}(l)$ was set to **true**, then the lth row and column of **cm** will be set to zero. If the second derivative matrix cannot be computed then all the elements of **cm** are returned as zero.

8: **ifail – int32 scalar**

0 unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Note: g13dc may return useful information for one or more of the following detected errors or warnings.

ifail $= 1$

On entry, $\mathbf{k} < 1$,
or $\mathbf{ip} < 0$,
or $\mathbf{iq} < 0$,
or $\mathbf{ip} = \mathbf{iq} = 0$,
or $\mathbf{npar} \neq (\mathbf{ip} + \mathbf{iq}) \times \mathbf{k} \times \mathbf{k} + \Delta \times \mathbf{k}$, where $\Delta = 1$ if **mean** $=$ **true** or $\Delta = 0$ if **mean** $=$ **false**,
or $\mathbf{n} \times \mathbf{k} \leq \mathbf{npar} + \mathbf{k} \times (\mathbf{k} + 1)/2$,
or $\mathbf{kmax} < \mathbf{k}$,
or $\mathbf{maxcal} < 1$,
or $\mathbf{ishow} < 0$,
or $\mathbf{ishow} > 2$,
or $\mathbf{ldcm} < \mathbf{npar}$,

ifail $= 2$

On entry, either the initial estimate of Σ is not positive-definite, or the initial estimates of the AR parameters are such that the model is non-stationary, or the initial estimates of the MA parameters are such that the model is non-invertible. To proceed, you must try a different starting point.

ifail $= 3$

The function cannot compute a sufficiently accurate estimate of the gradient vector at the user-supplied starting point. This usually occurs if either the initial parameter estimates are very close to the ML parameter estimates, or you have supplied a very poor estimate of Σ or the starting point is very close to the boundary of the stationarity or invertibility region. To proceed, you must try a different starting point.

ifail $= 4$

There have been **maxcal** log-likelihood evaluations made in the function. If steady increases in the log-likelihood function were monitored up to the point where this exit occurred, then the exit probably simply occurred because **maxcal** was set too small, so the calculations should be restarted from the final point held in **par** and **qq**. This type of exit may also indicate that there is no maximum to the likelihood surface. Output quantities (as described in Section 5) are computed at the final point held in **par** and **qq**, except that if **g** or **cm** cannot be computed, in which case they are set to zero.

ifail $= 5$

The conditions for a solution have not all been met, but a point at which the log-likelihood took a larger value could not be found.

Provided that the estimated first derivatives are sufficiently small, and that the estimated condition number of the second derivative (Hessian) matrix, as printed when **iprint** ≥ 0, is not too large, this error exit may simply mean that, although it has not been possible to satisfy the specified requirements, the algorithm has in fact found the solution as far as the accuracy of the machine permits.

Such a condition can arise, for instance, if **cgetol** has been set so small that rounding error in evaluating the likelihood function makes attainment of the convergence conditions impossible.

If the estimated condition number at the final point is large, it could be that the final point is a solution but that the smallest eigenvalue of the Hessian matrix is so close to zero at the solution that it is not possible to recognize it as a solution. Output quantities (as described in Section 5) are computed at the final point held in **par** and **qq**, except that if **g** or **cm** cannot be computed, in which case they are set to zero.

ifail $= 6$

The ML solution is so close to the boundary of either the stationarity region or the invertibility region that g13dc cannot evaluate the Hessian matrix. The elements of **cm** will then be set to zero on exit. The elements of **g** will also be set to zero. All other output quantities will be correct.

ifail $= 7$

This is an unlikely exit, which could occur in e04xa, which computes an estimate of the second derivative matrix and the gradient vector at the solution point. Either the Hessian matrix was found to be too ill-conditioned to be evaluated accurately or the gradient vector could not be computed to an acceptable degree of accuracy. In this case the elements of **cm** will be set to zero on exit as will the elements of **g**. All other output quantities will be correct.

ifail $= 8$

The second derivative matrix at the solution point is not positive-definite. In this case the elements of **cm** will be set to zero on exit. All other output quantities will be correct.

ifail $= -999$

Internal memory allocation failed.

7 Accuracy

On exit from g13dc, if **ifail** $= 0$ or **ifail** ≥ 6 and **cgetol** has been set to 10^{-l}, then all the parameters should be accurate to approximately l decimal places. If **cgetol** was set equal to a value less than the ***machine precision***, ϵ, then all the parameters should be accurate to approximately $10.0 \times \sqrt{\epsilon}$.

If **ifail** $= 4$ on exit (i.e., **maxcal** likelihood evaluations have been made but the convergence conditions of the search function have not been satisfied), then the elements in **par** and **qq** may still be good approximations to the ML estimates. You are advised to inspect the elements of **g** to see whether this is likely to be so.

8 Further Comments

8.1 Memory Usage

Let $r = \max(\mathbf{ip}, \mathbf{iq})$ and $s = \mathbf{npar} + \mathbf{k} \times (\mathbf{k} + 1)/2$. Local workspace arrays of fixed lengths are allocated internally by g13dc. The total size of these arrays amounts to $s + \mathbf{k} \times r + 52$ integer elements and