NAG Toolbox

nag_tsa_multi_autocorr_part (g13db)

1 Purpose

nag_tsa_multi_autocorr_part (g13db) calculates the multivariate partial autocorrelation function of a multivariate time series.

2 Syntax

```
[p, v0, v, d, db, w, wb, nvp, ifail] = nag_tsa_multi_autocorr_part(c0, c, nl, nk,
'ns', ns)
```

```
[p, v0, v, d, db, w, wb, nvp, ifail] = g13db(c0, c, nl, nk, 'ns', ns)
```

3 Description

The input is a set of lagged autocovariance matrices $C_0, C_1, C_2, \ldots, C_m$. These will generally be sample values such as are obtained from a multivariate time series using nag_tsa_multi_corrmat_cross (g13dm).

The main calculation is the recursive determination of the coefficients in the finite lag (forward) prediction equation

$$x_t = \Phi_{l,1} x_{t-1} + \cdots + \Phi_{l,l} x_{t-l} + e_{l,t}$$

and the associated backward prediction equation

$$x_{t-l-1} = \Psi_{l,1} x_{t-l} + \cdots + \Psi_{l,l} x_{t-1} + f_{l,t}$$

together with the covariance matrices D_l of $e_{l,t}$ and G_l of $f_{l,t}$.

The recursive cycle, by which the order of the prediction equation is extended from l to $l+1$, is to calculate

$$M_{l+1} = C_{l+1}^{\mathrm{T}} - \Phi_{l,1} C_l^{\mathrm{T}} - \cdots - \Phi_{l,l} C_1^{\mathrm{T}} \quad (1)$$

then $\Phi_{l+1,l+1} = M_{l+1} D_l^{-1}, \quad \Psi_{l+1,l+1} = M^{\mathrm{T}}{}_{l+1} G_l^{-1}$

from which

$$\Phi_{l+1,j} = \Phi_{l,j} - \Phi_{l+1,l+1} \Psi_{l,l+1-j}, \quad j = 1,2,\ldots,l \quad (2)$$

and

$$\Psi_{l+1,j} = \Psi_{l,j} - \Psi_{l+1,l+1} \Phi_{l,l+1-j}, \quad j = 1,2,\ldots,l. \quad (3)$$

Finally, $D_{l+1} = D_l - M_{l+1} \Phi^{\mathrm{T}}{}_{l+1,l+1}$ and $G_{l+1} = G_l - M^{\mathrm{T}}{}_{l+1} \Psi^{\mathrm{T}}{}_{l+1,l+1}$.

(Here T denotes the transpose of a matrix.)

The cycle is initialized by taking (for $l = 0$)

$$D_0 = G_0 = C_0.$$

In the step from $l = 0$ to 1, the above equations contain redundant terms and simplify. Thus (1) becomes $M_1 = C^{\mathrm{T}}{}_1$ and neither (2) or (3) are needed.

Quantities useful in assessing the effectiveness of the prediction equation are generalized variance ratios

$$v_l = \det D_l / \det C_0, \quad l = 1,2,\ldots$$

and multiple squared partial autocorrelations

$$p_l^2 = 1 - v_l / v_{l-1}.$$

4 References

Akaike H (1971) Autoregressive model fitting for control *Ann. Inst. Statist. Math.* **23** 163–180

Whittle P (1963) On the fitting of multivariate autoregressions and the approximate canonical factorization of a spectral density matrix *Biometrika* **50** 129–134

5 Parameters

5.1 Compulsory Input Parameters

1: **c0**($ldc0$, **ns**) – REAL (KIND=nag_wp) array

$ldc0$, the first dimension of the array, must satisfy the constraint $ldc0 \geq \max(\mathbf{ns}, 1)$.

Contains the zero lag cross-covariances between the **ns** series as returned by nag_tsa_multi_corrmat_cross (g13dm). (**c0** is assumed to be symmetric, upper triangle only is used.)

2: **c**($ldc0, ldc0$, **nl**) – REAL (KIND=nag_wp) array

$ldc0$, the first dimension of the array, must satisfy the constraint $ldc0 \geq \max(\mathbf{ns}, 1)$.

Contains the cross-covariances at lags 1 to **nl**. **c**(i, j, k) must contain the cross-covariance, c_{ijk}, of series i and series j at lag k. Series j leads series i.

3: **nl** – INTEGER

m, the maximum lag for which cross-covariances are supplied in **c**.

Constraint: $\mathbf{nl} \geq 1$.

4: **nk** – INTEGER

The number of lags to which partial auto-correlations are to be calculated.

Constraint: $1 \leq \mathbf{nk} \leq \mathbf{nl}$.

5.2 Optional Input Parameters

1: **ns** – INTEGER

Default: the second dimension of the array **c0**.

k, the number of time series whose cross-covariances are supplied in **c** and **c0**.

Constraint: $\mathbf{ns} \geq 1$.

5.3 Output Parameters

1: **p**(**nk**) – REAL (KIND=nag_wp) array

The multiple squared partial autocorrelations from lags 1 to **nvp**; that is, **p**(l) contains p_l^2, for $l = 1, 2, \ldots, \mathbf{nvp}$. For lags $\mathbf{nvp} + 1$ to **nk** the elements of **p** are set to zero.

2: **v0** – REAL (KIND=nag_wp)

The lag zero prediction error variance (equal to the determinant of **c0**).

3: **v**(**nk**) – REAL (KIND=nag_wp) array

The prediction error variance ratios from lags 1 to **nvp**; that is, **v**(l) contains v_l, for $l = 1, 2, \ldots, \mathbf{nvp}$. For lags $\mathbf{nvp} + 1$ to **nk** the elements of **v** are set to zero.

4: **d**($ldc0, ldc0$, **nk**) – REAL (KIND=nag_wp) array

$ldc0 = \max(\mathbf{ns}, 1)$.

The prediction error variance matrices at lags 1 to **nvp**.

Element (i, j, k) of **d** contains the prediction error covariance of series i and series j at lag k, for $k = 1, 2, \ldots,$ **nvp**. Series j leads series i; that is, the (i, j)th element of D_k. For lags **nvp** $+ 1$ to **nk** the elements of **d** are set to zero.

5: **db**($ldc0$, **ns**) – REAL (KIND=nag_wp) array

$ldc0 = \max($**ns**$, 1)$.

The backward prediction error variance matrix at lag **nvp**.

db(i, j) contains the backward prediction error covariance of series i and series j; that is, the (i, j)th element of the G_k, where $k =$ **nvp**.

6: **w**($ldc0$, $ldc0$, **nk**) – REAL (KIND=nag_wp) array

$ldc0 = \max($**ns**$, 1)$.

The prediction coefficient matrices at lags 1 to **nvp**.

w(i, j, l) contains the jth prediction coefficient of series i at lag l; that is, the (i, j)th element of Φ_{kl}, where $k =$ **nvp**, for $l = 1, 2, \ldots,$ **nvp**. For lags **nvp** $+ 1$ to **nk** the elements of **w** are set to zero.

7: **wb**($ldc0$, $ldc0$, **nk**) – REAL (KIND=nag_wp) array

$ldc0 = \max($**ns**$, 1)$.

The backward prediction coefficient matrices at lags 1 to **nvp**.

wb(i, j, l) contains the jth backward prediction coefficient of series i at lag l; that is, the (i, j)th element of Ψ_{kl}, where $k =$ **nvp**, for $l = 1, 2, \ldots,$ **nvp**. For lags **nvp** $+ 1$ to **nk** the elements of **wb** are set to zero.

8: **nvp** – INTEGER

The maximum lag, L, for which calculation of **p**, **v**, **d**, **db**, **w** and **wb** was successful. If the function completes successfully **nvp** will equal **nk**.

9: **ifail** – INTEGER

ifail $= 0$ unless the function detects an error (see Section 5).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= 1$

On entry, $ldc0 < 1$,
or **ns** < 1,
or **ns** $> ldc0$,
or **nl** < 1,
or **nk** < 1,
or **nk** $>$ **nl**,
or $iwa < (2 \times$ **ns** $+ 1) \times$ **ns**.

ifail $= 2$

c0 is not positive definite.

v0, **v**, **p**, **d**, **db**, **w**, **wb** and **nvp** are set to zero.

ifail $= 3$ (*warning*)

At lag $k = \textbf{nvp} + 1 \leq \textbf{nk}$, D_k was found not to be positive definite. Up to lag **nvp**, **v0**, **v**, **p**, **d**, **w** and **wb** contain the values calculated so far and from lag $\textbf{nvp} + 1$ to lag **nk** the matrices contain zero. **db** contains the backward prediction coefficients for lag **nvp**.

ifail $= -99$

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

ifail $= -399$

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

ifail $= -999$

Dynamic memory allocation failed.

7 Accuracy

The conditioning of the problem depends on the prediction error variance ratios. Very small values of these may indicate loss of accuracy in the computations.

8 Further Comments

The time taken by nag_tsa_multi_autocorr_part (g13db) is roughly proportional to $\textbf{nk}^2 \times \textbf{ns}^3$.

If sample autocorrelation matrices are used as input, then the output will be relevant to the original series scaled by their standard deviations. If these autocorrelation matrices are produced by nag_tsa_multi_corrmat_cross (g13dm), you must replace the diagonal elements of C_0 (otherwise used to hold the series variances) by 1.

9 Example

This example reads the autocovariance matrices for four series from lag 0 to 5. It calls nag_tsa_multi_autocorr_part (g13db) to calculate the multivariate partial autocorrelation function and other related matrices of statistics up to lag 3. It prints the results.

9.1 Program Text

```
     function g13db_example

fprintf('g13db example results\n\n');

% autocovariances
c0 = [0.0109,     -0.0077917,  0.0013004,  0.0012654;
     -0.0077917,  0.05704,     0.002418,   0.014409;
      0.0013004,  0.002418,    0.04396,   -0.021421;
      0.0012654,  0.014409,   -0.021421,   0.072289];
c(:,:,1) = ...
     [0.0045889,  0.0004651, -0.00013275, 0.0077531;
     -0.0024419, -0.011667,  -0.021956,  -0.0045803;
      0.001108,  -0.0080479,  0.013621,  -0.0085868;
     -0.00050614, 0.014045,  -0.0010087,  0.012269];
c(:,:,2) = ...
     [0.0018652, -0.0064389,  0.0088307, -0.0024808;
     -0.011865,   0.0072367, -0.019802,   0.0059069;
     -0.0080307,  0.014306,   0.014546,   0.01351;
     -0.0021791, -0.029528,  -0.015887,   0.00088308];
c(:,:,3) = ...
     [-8.055e-005,-0.0037759, 0.0075463, -0.0042276;
      0.0041447, -0.0037987,  0.0019332, -0.017564;
     -0.010582,   0.0067733,  0.0069832,  0.0061747;
      0.0041352, -0.016013,   0.017043,  -0.013412];
c(:,:,4) = ...
```

```
     [0.00076079,-0.0010134,  0.01187,    -0.0041651;
      0.0036014, -0.0036375, -0.025571,    0.0050218;
     -0.013924,   0.011718,  -0.0059088,  0.0059297;
      0.010739,  -0.014571,   0.013816,  -0.012588];
c(:,:,5) = ...
     [-0.00064365,-0.0044556, 0.0051334,  0.00071587;
      0.0063617,  0.00015217, 0.002727,  -0.0022261;
     -0.0085855,  0.0014468, -0.0028698,  0.0044384;
      0.0068339, -0.002179,   0.013759,   0.00028217];

nl = nag_int(5);
nk = nag_int(3);
ns = size(c0,1);

% Calculate multivariate partial autocorrelation function
[p, v0, v, d, db, w, wb, nvp, ifail] = ...
  g13db( ...
  c0, c, nl, nk);

%   Display results
fprintf('Number of valid parameters = %10d\n\n', nvp);
fprintf('Multivariate partial autocorrelations\n');
for j = 1:5:nk
  fprintf('%12.5f', p(j:min(j+4,nk)));
  fprintf('\n');
end
fprintf('\nZero lag predictor error variance determinant\n');
fprintf('followed by error variance ratios\n');
fprintf('%12.5f\n', v0);
for j = 1:5:nk
  fprintf('%12.5f', v(j:min(j+4,nk)));
  fprintf('\n');
end
fprintf('\nPrediction error variances\n');
for k = 1:nk
  fprintf('\nLag = %4d\n', k);
  disp(d(1:ns,1:ns,k));
end
fprintf('\nLast backward prediction error variances\n\n');
fprintf('Lag = %4d\n', nvp);
disp(db(1:ns,1:ns));
fprintf('\nPrediction coefficients\n');
for k = 1:nk
  fprintf('\nLag = %4d\n', k);
  disp(w(1:ns,1:ns,k));
end
fprintf('\nBackward prediction coefficients\n');
for k = 1:nk
  fprintf('\nLag = %4d\n', k);
  disp(wb(1:ns,1:ns,k));
end
```

9.2 Program Results

```
     g13db example results

Number of valid parameters =          3

Multivariate partial autocorrelations
     0.64498     0.92669     0.84300

Zero lag predictor error variance determinant
followed by error variance ratios
     0.00000
     0.35502     0.02603     0.00409

Prediction error variances

Lag =    1
    0.0081   -0.0051    0.0016   -0.0003
```

```
   -0.0051    0.0409    0.0076    0.0184
    0.0016    0.0076    0.0383   -0.0189
   -0.0003    0.0184   -0.0189    0.0676

Lag =    2
    0.0035   -0.0009   -0.0007   -0.0011
   -0.0009    0.0195    0.0053    0.0057
   -0.0007    0.0053    0.0190   -0.0107
   -0.0011    0.0057   -0.0107    0.0406

Lag =    3
    0.0030   -0.0009   -0.0005    0.0007
   -0.0009    0.0182    0.0087    0.0025
   -0.0005    0.0087    0.0093   -0.0022
    0.0007    0.0025   -0.0022    0.0225

Last backward prediction error variances

Lag =    3
    0.0033   -0.0039   -0.0011    0.0059
   -0.0039    0.0189    0.0035   -0.0033
   -0.0011    0.0035    0.0100   -0.0105
    0.0059   -0.0033   -0.0105    0.0334

Prediction coefficients

Lag =    1
    0.8186    0.2340   -0.1710    0.0926
    0.0674   -0.4872   -0.1406    0.0429
    0.1504    0.1192   -0.3672   -0.4209
   -0.7097    0.0300    0.5978    0.3461

Lag =    2
   -0.3405   -0.1337    0.4061   -0.0218
   -1.2757   -0.1359   -0.6578   -0.1127
   -0.4544    0.1938    0.6342    0.3392
   -0.4324   -0.5485   -0.6290    0.1667

Lag =    3
    0.1644    0.1386    0.0129    0.0346
    0.3929    0.0741   -0.0880   -0.1536
   -1.2924   -0.2449    0.3023    0.3944
    0.8977   -0.3904    0.2515   -0.2830

Backward prediction coefficients

Lag =    1
    0.4154    0.0615    0.1532    0.0508
    0.1237   -0.2647   -0.2272    0.4850
   -0.8693   -0.4737    0.3792    0.1381
    1.3078   -0.0918   -1.4540   -0.2197

Lag =    2
   -0.0674   -0.1226   -0.1367   -0.0973
   -1.2480    0.0309    0.5171   -0.2892
    0.9804   -0.2019    0.1631   -0.1087
   -1.6839   -0.7459    0.5290    0.4158

Lag =    3
    0.0379    0.1049   -0.2164    0.0801
    0.7539    0.2260   -0.2566   -0.4745
   -0.0034    0.0564   -0.0882    0.1272
    0.5502   -0.4123    0.7165   -0.1457
```