NAG Toolbox for MATLAB

g13dn

1 Purpose

g13dn calculates the sample partial lag correlation matrices of a multivariate time series. A set of χ^2-statistics and their significance levels are also returned. A call to g13dm is usually made prior to calling this function in order to calculate the sample cross-correlation matrices.

2 Syntax

```
[maxlag, parlag, x, pvalue, ifail] = g13dn(n, m, r0, r, 'k', k)
```

3 Description

Let $W_t = (w_{1t}, w_{2t}, \ldots, w_{kt})^{\mathrm{T}}$, for $t = 1, 2, \ldots, n$, denote n observations of a vector of k time series. The partial lag correlation matrix at lag l, $P(l)$, is defined to be the correlation matrix between W_t and W_{t+l}, after removing the linear dependence on each of the intervening vectors $W_{t+1}, W_{t+2}, \ldots, W_{t+l-1}$. It is the correlation matrix between the residual vectors resulting from the regression of W_{t+l} on the carriers $W_{t+l-1}, \ldots, W_{t+1}$ and the regression of W_t on the same set of carriers; see Heyse and Wei 1985.

$P(l)$ has the following properties.

(i) If W_t follows a vector autoregressive model of order p, then $P(l) = 0$ for $l > p$;

(ii) When $k = 1$, $P(l)$ reduces to the univariate partial autocorrelation at lag l;

(iii) Each element of $P(l)$ is a properly normalized correlation coefficient;

(iv) When $l = 1$, $P(l)$ is equal to the cross-correlation matrix at lag 1 (a natural property which also holds for the univariate partial autocorrelation function).

Sample estimates of the partial lag correlation matrices may be obtained using the recursive algorithm described in Wei 1990. They are calculated up to lag m, which is usually taken to be at most $n/4$. Only the sample cross-correlation matrices ($\hat{R}(l)$, $l = 0, 1, \ldots, m$) and the standard deviations of the series are required as input to g13dn. These may be computed by g13dm. Under the hypothesis that W_t follows an autoregressive model of order $s - 1$, the elements of the sample partial lag matrix $\hat{P}(s)$, denoted by $\hat{P}_{ij}(s)$, are asymptotically Normally distributed with mean zero and variance $1/n$. In addition the statistic

$$X(s) = n\sum_{i=1}^{k}\sum_{j=1}^{k}\hat{P}_{ij}(s)^2$$

has an asymptotic χ^2-distribution with k^2 degrees of freedom. These quantities, $X(l)$, are useful as a diagnostic aid for determining whether the series follows an autoregressive model and, if so, of what order.

4 References

Heyse J F and Wei W W S 1985 The partial lag autocorrelation function *Technical Report No. 32* Department of Statistics, Temple University, Philadelphia

Wei W W S 1990 *Time Series Analysis: Univariate and Multivariate Methods* Addison–Wesley

5 Parameters

5.1 Compulsory Input Parameters

1: **n – int32 scalar**

n, the number of observations in each series.

Constraint: $\mathbf{n} \geq 2$.

2: **m – int32 scalar**

m, the number of partial lag correlation matrices to be computed. Note this also specifies the number of sample cross-correlation matrices that must be contained in the array **r**.

Constraint: $1 \leq \mathbf{m} < \mathbf{n}$.

3: **r0(kmax,k) – double array**

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

If $i \neq j$, then $\mathbf{r0}(i,j)$ must contain the (i,j)th element of the sample cross-correlation matrix at lag zero, $\hat{R}_{ij}(0)$. If $i = j$, then $\mathbf{r0}(i,i)$ must contain the standard deviation of the ith series.

4: **r(kmax,kmax,m) – double array**

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

$\mathbf{r}(i,j,l)$ must contain the (i,j)th element of the sample cross-correlation at lag l, $\hat{R}_{ij}(l)$, for $l = 1,2,\dots,m$; $i = 1,2,\dots,k$; $j = 1,2,\dots,k$, where series j leads series i (see Section 8).

5.2 Optional Input Parameters

1: **k – int32 scalar**

k, the dimension of the multivariate time series.

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

5.3 Input Parameters Omitted from the MATLAB Interface

kmax, work, lwork

5.4 Output Parameters

1: **maxlag – int32 scalar**

The maximum lag up to which partial lag correlation matrices (along with χ^2-statistics and their significance levels) have been successfully computed. On a successful exit **maxlag** will equal **m**. If **ifail** $= 2$ on exit, then **maxlag** will be less than **m**.

2: **parlag(kmax,kmax,m) – double array**

$\mathbf{parlag}(i,j,l)$ contains the (i,j)th element of the sample partial lag correlation matrix at lag l, $\hat{P}_{ij}(l)$, for $l = 1,2,\dots,\mathbf{maxlag}$; $i = 1,2,\dots,k$; $j = 1,2,\dots,k$.

3: **x(m) – double array**

$\mathbf{x}(l)$ contains the χ^2-statistic at lag l, for $l = 1,2,\dots,\mathbf{maxlag}$.

4: **pvalue(m) – double array**

$\mathbf{pvalue}(l)$ contains the significance level of the corresponding χ^2-statistic in **x** for $l = 1,2,\dots,\mathbf{maxlag}$.

5: **ifail – int32 scalar**

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

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= 1$

On entry, **k** < 1,
or **n** < 2,
or **m** < 1,
or **m** $\geq$ **n**,
or **kmax** $<$ **k**,
or **lwork** $< (5\mathbf{m} + 6)\mathbf{k}^2 + \mathbf{k}$.

ifail $= 2$

The recursive equations used to compute the sample partial lag correlation matrices have broken down at lag **maxlag** $+ 1$. All output quantities in the arrays **parlag**, **x** and **pvalue** up to and including lag **maxlag** will be correct.

7 Accuracy

The accuracy will depend upon the accuracy of the sample cross-correlations.

8 Further Comments

The time taken is roughly proportional to m^2k^3.

If you have calculated the sample cross-correlation matrices in the arrays **r0** and **r**, without calling g13dm, then care must be taken to ensure they are supplied as described in Section 5. In particular, for $l \geq 1$, $\hat{R}_{ij}(l)$ must contain the sample cross-correlation coefficient between $w_{i(t-l)}$ and w_{jt}.

The function g13db computes squared partial autocorrelations for a specified number of lags. It may also be used to estimate a sequence of partial autoregression matrices at lags $1, 2, \ldots$ by making repeated calls to the function with the parameter **nk** set to $1, 2, \ldots$. The (i,j)th element of the sample partial autoregression matrix at lag l is given by $W(i,j,l)$ when **nk** is set equal to l on entry to g13db. Note that this is the 'Yule–Walker' estimate. Unlike the partial lag correlation matrices computed by g13dn, when W_t follows an autoregressive model of order $s - 1$, the elements of the sample partial autoregressive matrix at lag s do not have variance $1/n$, making it very difficult to spot a possible cut-off point. The differences between these matrices are discussed further by Wei 1990.

Note that g13db takes the sample cross-covariance matrices as input whereas this function requires the sample cross-correlation matrices to be input.

9 Example

```
n = int32(48);
m = int32(10);
r0 = [2.817550272831091, 0.2493409556934405;
      0.2493409556934405, 2.815040887355392];
r = zeros(2, 2, 10);
r(1, 1, 1) = 0.7359386303134835;
r(1, 1, 2) = 0.4557429704863583;
r(1, 1, 3) = 0.3791683873221404;
r(1, 1, 4) = 0.3224043613297473;
r(1, 1, 5) = 0.3410663610398739;
r(1, 1, 6) = 0.3630532980536906;
```

```
r(1, 1, 7) = 0.279950963720059;
r(1, 1, 8) = 0.2479741466444122;
r(1, 1, 9) = 0.2397587770555531;
r(1, 1, 10) = 0.1619287942538523;
r(1, 2, 1) = 0.1743387857080356;
r(1, 2, 2) = 0.07648968471673315;
r(1, 2, 3) = 0.01385335286265662;
r(1, 2, 4) = 0.110017582948063;
r(1, 2, 5) = 0.2694726966973017;
r(1, 2, 6) = 0.3435921339678099;
r(1, 2, 7) = 0.4254153226333015;
r(1, 2, 8) = 0.5217552811465264;
r(1, 2, 9) = 0.266437286523482;
r(1, 2, 10) = -0.01971853544663743;
r(2, 1, 1) = 0.2113457408346849;
r(2, 1, 2) = 0.06928186692983163;
r(2, 1, 3) = 0.02598655726548196;
r(2, 1, 4) = 0.0932803134362757;
r(2, 1, 5) = 0.08722854902023115;
r(2, 1, 6) = 0.1322963090225139;
r(2, 1, 7) = 0.2069131464479435;
r(2, 1, 8) = 0.1970166844792453;
r(2, 1, 9) = 0.2536529873097416;
r(2, 1, 10) = 0.2666458165674043;
r(2, 2, 1) = 0.554565584968978;
r(2, 2, 2) = 0.2604546298918431;
r(2, 2, 3) = -0.03809776429573517;
r(2, 2, 4) = -0.2358548705865134;
r(2, 2, 5) = -0.250066742662944;
r(2, 2, 6) = -0.2265191429863443;
r(2, 2, 7) = -0.1284351528906316;
r(2, 2, 8) = -0.08463615282058312;
r(2, 2, 9) = 0.07457486793239025;
r(2, 2, 10) = 0.004727174709710211;
[maxlag, parlag, x, pvalue, ifail] = g13dn(n, m, r0, r)
```

```
maxlag =
          10
parlag =
(:,:,1) =
    0.7359    0.1743
    0.2113    0.5546
(:,:,2) =
   -0.1869   -0.0832
   -0.1805   -0.0724
(:,:,3) =
    0.2775   -0.0069
    0.0837   -0.2133
(:,:,4) =
   -0.0843    0.2268
    0.1284   -0.1763
(:,:,5) =
    0.2362    0.2384
   -0.0468   -0.0455
(:,:,6) =
   -0.0164    0.0873
    0.0996   -0.0810
(:,:,7) =
   -0.0355    0.2611
    0.1257    0.0121
(:,:,8) =
    0.0768    0.3815
    0.0268   -0.1492
(:,:,9) =
   -0.0651   -0.3867
    0.1887    0.0565
(:,:,10) =
   -0.0261   -0.2861
    0.0279   -0.1729
x =
```

```
   44.3621
    3.8239
    6.2189
    5.0941
    5.6094
    1.1698
    4.0983
    8.3707
    9.2440
    5.4353
pvalue =
    0.0000
    0.4304
    0.1834
    0.2778
    0.2303
    0.8830
    0.3929
    0.0789
    0.0553
    0.2455
ifail =
           0
```