NAG Toolbox

nag_tsa_inhom_iema_all (g13mf)

1 Purpose

nag_tsa_inhom_iema_all (g13mf) calculates the iterated exponential moving average for an inhomogeneous time series, returning the intermediate results.

2 Syntax

```
[iema, p, pn, rcomm, ifail] = nag_tsa_inhom_iema_all(z, t, tau, m1, m2, sinit,
inter, ftype, p, x, 'sorder', sorder, 'nb', nb, 'pn', pn, 'rcomm', rcomm)
```

```
[iema, p, pn, rcomm, ifail] = g13mf(z, t, tau, m1, m2, sinit, inter, ftype, p,
x, 'sorder', sorder, 'nb', nb, 'pn', pn, 'rcomm', rcomm)
```

3 Description

nag_tsa_inhom_iema_all (g13mf) calculates the iterated exponential moving average for an inhomogeneous time series. The time series is represented by two vectors of length n: a vector of times, t; and a vector of values, z. Each element of the time series is therefore composed of the pair of scalar values (t_i, z_i), for $i = 1, 2, \ldots, n$. Time can be measured in any arbitrary units, as long as all elements of t use the same units.

The exponential moving average (EMA), with parameter τ, is an average operator, with the exponentially decaying kernel given by

$$\frac{e^{-t_i/\tau}}{\tau}.$$

The exponential form of this kernel gives rise to the following iterative formula (Zumbach and MÏller (2001)) for the EMA operator:

$$\mathrm{EMA}[\tau; y](t_i) = \mu \mathrm{EMA}[\tau; y](t_{i-1}) + (\nu - \mu) y_{i-1} + (1 - \nu) y_i$$

where

$$\mu = e^{-\alpha} \quad \text{and} \quad \alpha = \frac{t_i - t_{i-1}}{\tau}.$$

The value of ν depends on the method of interpolation chosen and the relationship between y and the input series z depends on the transformation function chosen. nag_tsa_inhom_iema_all (g13mf) gives the option of three interpolation methods:

1. Previous point: $\nu = 1$;
2. Linear: $\nu = (1 - \mu)/\alpha$;
3. Next point: $\nu = \mu$.

and three transformation functions:

1. Identity: $y_i = z_i^{[p]}$;
2. Absolute value: $y_i = |z_i|^p$;
3. Absolute difference: $y_i = |z_i - x_i|^p$;

where the notation $[p]$ is used to denote the integer nearest to p. In the case of the absolute difference x is a user-supplied vector of length n and therefore each element of the time series is composed of the triplet of scalar values, (t_i, z_i, x_i).

The m-iterated exponential moving average, $\mathrm{EMA}[\tau, m; y](t_i)$, is defined using the recursive formula:

$$\mathrm{EMA}[\tau, m; y](t_i) = \mathrm{EMA}[\tau; \mathrm{EMA}[\tau, m-1; y](t_i)](t_i)$$

with

$$\mathrm{EMA}[\tau, 1; y](t_i) = \mathrm{EMA}[\tau; y](t_i).$$

For large datasets or where all the data is not available at the same time, z, t and, where required, x can be split into arbitrary sized blocks and nag_tsa_inhom_iema_all (g13mf) called multiple times.

4 References

Dacorogna M M, Gencay R, MÏller U, Olsen R B and Pictet O V (2001) *An Introduction to High-frequency Finance* Academic Press

Zumbach G O and MÏller U A (2001) Operators on inhomogeneous time series *International Journal of Theoretical and Applied Finance* **4(1)** 147–178

5 Parameters

5.1 Compulsory Input Parameters

1: **z**(**nb**) – REAL (KIND=nag_wp) array

z_i, the current block of observations, for $i = k+1, \ldots, k+b$, where k is the number of observations processed so far, i.e., the value supplied in **pn** on entry.

Constraint: if **ftype** $= 1$ or 2 and **p** < 0.0, **z**$(i) \neq 0$, for $i = 1, 2, \ldots,$ **nb**.

2: **t**(**nb**) – REAL (KIND=nag_wp) array

t_i, the times for the current block of observations, for $i = k+1, \ldots, k+b$, where k is the number of observations processed so far, i.e., the value supplied in **pn** on entry.

If $t_i \leq t_{i-1}$, **ifail** $= 61$ will be returned, but nag_tsa_inhom_iema_all (g13mf) will continue as if t was strictly increasing by using the absolute value.

3: **tau** – REAL (KIND=nag_wp)

τ, the parameter controlling the rate of decay. τ must be sufficiently large that $e^{-\alpha}$, $\alpha = (t_i - t_{i-1})/\tau$ can be calculated without overflowing, for all i.

Constraint: **tau** > 0.0.

4: **m1** – INTEGER

The minimum number of times the EMA operator is to be iterated.

Constraint: **m1** ≥ 1.

5: **m2** – INTEGER

The maximum number of times the EMA operator is to be iterated. Therefore nag_tsa_inhom_iema_all (g13mf) returns $\mathrm{EMA}[\tau, m; y]$, for $m =$ **m1**, **m1** $+ 1, \ldots,$ **m2**.

Constraint: **m2** $\geq$ **m1**.

6: **sinit**(**m2** $+$ **2**) – REAL (KIND=nag_wp) array

If **pn** $= 0$, the values used to start the iterative process, with

sinit$(1) = t_0$,

sinit$(2) = y_0$,

sinit$(j+2) = \mathrm{EMA}[\tau, j; y](t_0)$, $j = 1, 2, \ldots,$ **m2**.

If **pn** $\neq 0$ then **sinit** is not referenced.

Constraint: if **ftype** $\neq 1$, **sinit**$(j) \geq 0$, for $j = 2, 3, \ldots, \mathbf{m2} + 2$.

7: **inter(2)** – INTEGER array

The type of interpolation used with **inter**(1) indicating the interpolation method to use when calculating $\mathrm{EMA}[\tau, 1; z]$ and **inter**(2) the interpolation method to use when calculating $\mathrm{EMA}[\tau, j; z]$, $j > 1$.

Three types of interpolation are possible:

inter$(i) = 1$
Previous point, with $\nu = 1$.

inter$(i) = 2$
Linear, with $\nu = (1 - \mu)/\alpha$.

inter$(i) = 3$
Next point, $\nu = \mu$.

Zumbach and MÏller (2001) recommend that linear interpolation is used in second and subsequent iterations, i.e., **inter**$(2) = 2$, irrespective of the interpolation method used at the first iteration, i.e., the value of **inter**(1).

Constraint: **inter**$(i) = 1$, 2 or 3, for $i = 1, 2$.

8: **ftype** – INTEGER

The function type used to define the relationship between y and z when calculating $\mathrm{EMA}[\tau, 1; y]$. Three functions are provided:

ftype $= 1$
The identity function, with $y_i = z_i^{[p]}$.

ftype $= 2$
The absolute value, with $y_i = |z_i|^p$.

ftype $= 3$
The absolute difference, with $y_i = |z_i - x_i|^p$, where the vector x is supplied in **x**.

Constraint: **ftype** $= 1$, 2 or 3.

9: **p** – REAL (KIND=nag_wp)

p, the power used in the transformation function.

Constraint: **p** $\neq 0$.

10: **x**(:) – REAL (KIND=nag_wp) array

The dimension of the array **x** must be at least **nb** if **ftype** $= 3$

If **ftype** $= 3$, x_i, the vector used to shift the current block of observations, for $i = k + 1, \ldots, k + b$, where k is the number of observations processed so far, i.e., the value supplied in **pn** on entry.

If **ftype** $\neq 3$ then **x** is not referenced.

Constraint: if **ftype** $= 3$ and **p** < 0, **x**$(i) \neq$ **z**(i), for $i = 1, 2, \ldots, \mathbf{nb}$.

5.2 Optional Input Parameters

1: **sorder** – INTEGER

Default: 1

Determines the storage order of output returned in **iema**.

Constraint: **sorder** $= 1$ or 2.

2: **nb** – INTEGER

Default: the dimension of the arrays **z**, **t**, **x**. (An error is raised if these dimensions are not equal.)

b, the number of observations in the current block of data. At each call the size of the block of data supplied in **z**, **t** and **x** can vary; therefore **nb** can change between calls to nag_tsa_inhom_iema_all (g13mf).

Constraint: **nb** ≥ 0.

3: **pn** – INTEGER

Default: 0

k, the number of observations processed so far. On the first call to nag_tsa_inhom_iema_all (g13mf), or when starting to summarise a new dataset, **pn** must be set to 0. On subsequent calls it must be the same value as returned by the last call to nag_tsa_inhom_iema_all (g13mf).

Constraint: **pn** ≥ 0.

4: **rcomm**($lrcomm$) – REAL (KIND=nag_wp) array

Communication array, used to store information between calls to nag_tsa_inhom_iema_all (g13mf). On the first call to nag_tsa_inhom_iema_all (g13mf), or if all the data is provided in one go, **rcomm** need not be provided.

5.3 Output Parameters

1: **iema**($ldiema, *$) – REAL (KIND=nag_wp) array

Note: the second dimension of the array **iema** must be at least **m2** $-$ **m1** $+ 1$ if **sorder** $= 1$, otherwise at least **nb**.

The iterated exponential moving average.

If **sorder** $= 1$, **iema**$(i, j) = \mathrm{EMA}[\tau, j +$ **m1** $- 1; y](t_{i+k})$.

If **sorder** $= 2$, **iema**$(j, i) = \mathrm{EMA}[\tau, j +$ **m1** $- 1; y](t_{i+k})$.

For $i = 1, 2, \ldots,$ **nb**, $j = 1, 2, \ldots,$ **m2** $-$ **m1** $+ 1$ and k is the number of observations processed so far, i.e., the value supplied in **pn** on entry.

2: **p** – REAL (KIND=nag_wp)

If **ftype** $= 1$, then $[p]$, the actual power used in the transformation function is returned, otherwise **p** is unchanged.

3: **pn** – INTEGER

Default: 0

$k + b$, the updated number of observations processed so far.

4: **rcomm**($lrcomm$) – REAL (KIND=nag_wp) array

Communication array, used to store information between calls to nag_tsa_inhom_iema_all (g13mf).

5: **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 $= 11$

Constraint: **sorder** $= 1$ or 2.

ifail $= 21$

Constraint: **nb** ≥ 0.

ifail $= 51$

Constraint: $ldiema \geq$ **m2** $-$ **m1** $+ 1$.

Constraint: $ldiema \geq$ **nb**.

ifail $= 61$ (*warning*)

Constraint: **t** should be strictly increasing.

ifail $= 62$

Constraint: **t**$(i) \neq$ **t**$(i-1)$ if linear interpolation is being used.

ifail $= 71$

Constraint: **tau** > 0.0.

ifail $= 72$

Constraint: if **pn** > 0 then **tau** must be unchanged since previous call.

ifail $= 81$

Constraint: **m1** ≥ 1.

ifail $= 82$

Constraint: if **pn** > 0 then **m1** must be unchanged since previous call.

ifail $= 91$

Constraint: **m2** $\geq$ **m1**.

ifail $= 92$

Constraint: if **pn** > 0 then **m2** must be unchanged since previous call.

ifail $= 101$

Constraint: if **ftype** $\neq 1$, **sinit**$(j) \geq 0.0$, for $j = 2, 3, \ldots,$ **m2** $+ 2$.

ifail $= 111$

Constraint: **inter**$(1) = 1$, 2 or 3.

ifail $= 112$

Constraint: **inter**$(2) = 1$, 2 or 3.

ifail $= 113$

Constraint: if **pn** $\neq 0$, **inter** must be unchanged since the last call.

ifail $= 121$

Constraint: **ftype** $= 1,\ 2$ or 3.

ifail $= 122$

Constraint: if **pn** $\neq 0$, **ftype** must be unchanged since the previous call.

ifail $= 131$

Constraint: absolute value of **p** must be representable as an integer.

ifail $= 132$

Constraint: if **ftype** $\neq 1$, **p** $\neq 0.0$. If **ftype** $= 1$, the nearest integer to **p** must not be 0.

ifail $= 133$

Constraint: if **ftype** $= 1$ or 2 and $\mathbf{z}(i) = 0$ for any i then **p** > 0.0.

ifail $= 134$

Constraint: if **ftype** $= 3$ and $\mathbf{z}(i) = \mathbf{x}(i)$ for any i then **p** > 0.0.

ifail $= 135$

Constraint: if **pn** > 0 then **p** must be unchanged since previous call.

ifail $= 151$

Constraint: **pn** ≥ 0.

ifail $= 152$

Constraint: if **pn** > 0 then **pn** must be unchanged since previous call.

ifail $= 161$

rcomm has been corrupted between calls.

ifail $= 171$

Constraint: if **pn** $= 0$, $lrcomm = 0$ or $lrcomm \geq$ **m2** $+ 20$.

ifail $= 172$

Constraint: if **pn** $\neq 0$ then $lrcomm \geq$ **m2** $+ 20$.

ifail $= 301$ (*warning*)

Truncation occurred to avoid overflow, check for extreme values in **t**, **z**, **x** or for **tau**. Results are returned using the truncated values.

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

Not applicable.

8 Further Comments

Approximately $4 \times$ **m2** real elements are internally allocated by nag_tsa_inhom_iema_all (g13mf).

The more data you supply to nag_tsa_inhom_iema_all (g13mf) in one call, i.e., the larger **nb** is, the more efficient the routine will be, particularly if the function is being run using more than one thread.

Checks are made during the calculation of α and y_i to avoid overflow. If a potential overflow is detected the offending value is replaced with a large positive or negative value, as appropriate, and the calculations performed based on the replacement values. In such cases **ifail** $= 301$ is returned. This should not occur in standard usage and will only occur if extreme values of **z**, **t**, **x** or **tau** are supplied.

9 Example

This example reads in three blocks of simulated data from an inhomogeneous time series, then calculates and prints the iterated EMA for m between 2 and 6.

9.1 Program Text

```
     function g13mf_example

fprintf('g13mf example results\n\n');

m1    = nag_int(2);
m2    = nag_int(6);
ftype = nag_int(1);
p     = 1;
inter = [nag_int(3); 2];
tau   = 2;
sinit = zeros(8, 1);
nb    = [5, 10, 15];
rcomm = zeros(20+m2, 1);
x     = [];
t     = cell(3, 1);
z     = cell(3, 1);

t{1} = [ 7.5;  8.2; 18.1; 22.8; 25.8];
z{1} = [ 0.6;  0.6;  0.8;  0.1;  0.2];
t{2} = [26.8; 31.1; 38.4; 45.9; 48.2; 48.9; 57.9; 58.5; 63.9; 65.2];
z{2} = [0.2;   0.5;  0.7;  0.1;  0.4;  0.7;  0.8;  0.3;  0.2;  0.5];
t{3} = [66.6; 67.4; 69.3; 69.9; 73.0; 75.6; 77.0; 84.7; 86.8; 88.0; ...
        88.5; 91.0; 93.0; 93.7; 94.0];
z{3} = [ 0.2;  0.3;  0.8;  0.6;  0.1;  0.7;  0.9;  0.6;  0.3;  0.1;  ...
         0.1;  0.4;  1.0;  1.0;  0.1];

fprintf('%41s\n%17s', 'Iteration', 'Time');
fprintf('%10d', [2:6]);
fprintf('\n');

% Loop over each block of data.
miema = m2-m1+1;
iema = cell(numel(nb), 1);
fmt = '%3d%14.1f  %10.3f%10.3f%10.3f%10.3f%10.3f\n';
for i = 1:numel(nb)
  if i == 1
    % Initialise the iterated EMA
    [iema{i}, p, pn, rcomm, ifail] = ...
    g13mf( ...
           z{i}, t{i}, tau, m1, m2, sinit, inter, ftype, p, x, 'rcomm', rcomm);
  else
    % Update the iterated EMA for this block of data
    [iema{i}, p, pn, rcomm, ifail] = ...
    g13mf( ...
```

```
          z{i}, t{i}, tau, m1, m2, sinit, inter, ftype, p, x, ...
          'pn', pn, 'rcomm', rcomm);
  end

  % Display the results for this block of data
  for j=1:nb(i)
    fprintf(fmt, pn-nb(i)+j, t{i}(j), iema{i}(j, 1:miema));
  end
  fprintf('\n');
end
```

9.2 Program Results

```
     g13mf example results

                                     Iteration
             Time          2          3          4          5          6
  1           7.5      0.433      0.320      0.237      0.175      0.130
  2           8.2      0.479      0.361      0.268      0.198      0.147
  3          18.1      0.756      0.700      0.631      0.558      0.485
  4          22.8      0.406      0.535      0.592      0.600      0.577
  5          25.8      0.232      0.351      0.459      0.530      0.561

  6          26.8      0.217      0.301      0.406      0.491      0.540
  7          31.1      0.357      0.309      0.318      0.364      0.422
  8          38.4      0.630      0.556      0.490      0.445      0.425
  9          45.9      0.263      0.357      0.407      0.428      0.432
 10          48.2      0.241      0.284      0.343      0.388      0.413
 11          48.9      0.279      0.277      0.325      0.372      0.403
 12          57.9      0.713      0.617      0.543      0.496      0.469
 13          58.5      0.717      0.643      0.566      0.511      0.478
 14          63.9      0.385      0.495      0.541      0.546      0.531
 15          65.2      0.346      0.432      0.502      0.533      0.535

 16          66.6      0.330      0.384      0.453      0.504      0.526
 17          67.4      0.315      0.364      0.427      0.483      0.515
 18          69.3      0.409      0.367      0.389      0.435      0.478
 19          69.9      0.459      0.385      0.386      0.423      0.465
 20          73.0      0.377      0.403      0.394      0.398      0.419
 21          75.6      0.411      0.399      0.399      0.397      0.403
 22          77.0      0.536      0.440      0.410      0.401      0.401
 23          84.7      0.632      0.606      0.563      0.524      0.493
 24          86.8      0.538      0.587      0.583      0.557      0.526
 25          88.0      0.444      0.542      0.574      0.567      0.542
 26          88.5      0.401      0.515      0.564      0.567      0.548
 27          91.0      0.331      0.404      0.481      0.529      0.545
 28          93.0      0.495      0.418      0.438      0.483      0.518
 29          93.7      0.585      0.455      0.438      0.469      0.506
 30          94.0      0.612      0.475      0.441      0.465      0.500
```