NAG Library Routine Document

G13BAF

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of ***bold italicised*** terms and other implementation-dependent details.

1 Purpose

G13BAF filters a time series by an ARIMA model.

2 Specification

```
SUBROUTINE G13BAF (Y, NY, MR, NMR, PAR, NPAR, CY, WA, NWA, B, NB, IFAIL)

INTEGER             NY, MR(NMR), NMR, NPAR, NWA, NB, IFAIL
REAL (KIND=nag_wp) Y(NY), PAR(NPAR), CY, WA(NWA), B(NB)
```

3 Description

From a given series $y_1, y_2, \ldots, y_n$, a new series $b_1, b_2, \ldots, b_n$ is calculated using a supplied (filtering) ARIMA model. This model will be one which has previously been fitted to a series x_t with residuals a_t. The equations defining b_t in terms of y_t are very similar to those by which a_t is obtained from x_t. The only dissimilarity is that no constant correction is applied after differencing. This is because the series y_t is generally distinct from the series x_t with which the model is associated, though y_t may be related to x_t. Whilst it is appropriate to apply the ARIMA model to y_t so as to preserve the same relationship between b_t and a_t as exists between y_t and x_t, the constant term in the ARIMA model is inappropriate for y_t. The consequence is that b_t will not necessarily have zero mean.

The equations are precisely:

$$w_t = \nabla^d \nabla_s^D y_t, \tag{1}$$

the appropriate differencing of y_t; both the seasonal and non-seasonal inverted autoregressive operations are then applied,

$$u_t = w_t - \Phi_1 w_{t-s} - \cdots - \Phi_P w_{t-s\times P} \tag{2}$$

$$v_t = u_t - \phi_1 u_{t-1} - \cdots - \phi_p u_{t-p} \tag{3}$$

followed by the inverted moving average operations

$$z_t = v_t + \Theta_1 z_{t-s} + \cdots + \Theta_Q z_{t-s\times Q} \tag{4}$$

$$b_t = z_t + \theta_1 b_{t-1} + \cdots + \theta_q b_{t-q}. \tag{5}$$

Because the filtered series value b_t depends on present and past values $y_t, y_{t-1}, \ldots$, there is a problem arising from ignorance of $y_0, y_{-1}, \ldots$ which particularly affects calculation of the early values $b_1, b_2, \ldots$, causing 'transient errors'. The routine allows two possibilities.

(i) The equations (1), (2) and (3) are applied from successively later time points so that all terms on their right-hand sides are known, with v_t being defined for $t = (1 + d + s \times D + s \times P), \ldots, n$. Equations (4) and (5) are then applied over the same range, taking any values on the right-hand side associated with previous time points to be zero.

This procedure may still however result in unacceptably large transient errors in early values of b_t.

(ii) The unknown values $y_0, y_{-1}, \ldots$ are estimated by backforecasting. This requires that an ARIMA model distinct from that which has been supplied for filtering, should have been previously fitted to y_t.

For efficiency, you are asked to supply both this ARIMA model for y_t and a limited number of backforecasts which are prefixed to the known values of y_t. Within the routine further backforecasts of y_t, and the series w_t, u_t, v_t in (1), (2) and (3) are then easily calculated, and a set of linear equations solved for backforecasts of z_t, b_t for use in (4) and (5) in the case that $q + Q > 0$.

Even if the best model for y_t is not available, a very approximate guess such as

$$y_t = c + e_t$$

or

$$\nabla y_t = e_t$$

can help to reduce the transients substantially.

The backforecasts which need to be prefixed to y_t are of length $Q'_y = q_y + s_y \times Q_y$, where q_y and Q_y are the non-seasonal and seasonal moving average orders and s_y the seasonal period for the ARIMA model of y_t. Thus you need not carry out the backforecasting exercise if $Q'_y = 0$. Otherwise, the series $y_1, y_2, \ldots, y_n$ should be reversed to obtain $y_n, y_{n-1}, \ldots, y_1$ and G13AJF should be used to forecast Q'_y values, $\hat{y}_0, \ldots, \hat{y}_{1-Q'_y}$. The ARIMA model used is that fitted to y_t (as a forward series) except that, if $d_y + D_y$ is odd, the constant should be changed in sign (to allow, for example, for the fact that a forward upward trend is a reversed downward trend). The ARIMA model for y_t supplied to the filtering routine must however have the appropriate constant for the forward series.

The series $\hat{y}_{1-Q'_y}, \ldots, \hat{y}_0, y_1, \ldots, y_n$ is then supplied to the routine, and a corresponding set of values returned for b_t.

4 References

Box G E P and Jenkins G M (1976) *Time Series Analysis: Forecasting and Control* (Revised Edition) Holden–Day

5 Parameters

1: Y(NY) – REAL (KIND=nag_wp) array *Input*

On entry: the Q'_y backforecasts, starting with backforecast at time $1 - Q'_y$ to backforecast at time 0, followed by the time series starting at time 1, where $Q'_y = \mathrm{MR}(10) + \mathrm{MR}(13) \times \mathrm{MR}(14)$. If there are no backforecasts, either because the ARIMA model for the time series is not known, or because it is known but has no moving average terms, then the time series starts at the beginning of Y.

2: NY – INTEGER *Input*

On entry: the total number of backforecasts and time series data points in array Y.

Constraint: $\mathrm{NY} \geq \max\left(1 + Q'_y, \mathrm{NPAR}\right)$.

3: MR(NMR) – INTEGER array *Input*

On entry: the orders vector for the filtering model, followed by the orders vector for the ARIMA model for the time series if the latter is known. The orders appear in the standard sequence (p, d, q, P, D, Q, s) as given in the G13 Chapter Introduction. If the ARIMA model for the time series is supplied, then the routine will assume that the first Q'_y values of the array Y are backforecasts.

Constraints:

the filtering model is restricted in the following ways:

$\mathrm{MR}(1) + \mathrm{MR}(3) + \mathrm{MR}(4) + \mathrm{MR}(6) > 0$, i.e., filtering by a model which contains only differencing terms is not permitted;
$\mathrm{MR}(k) \geq 0$, for $k = 1, 2, \ldots, 7$;
if $\mathrm{MR}(7) = 0$, $\mathrm{MR}(4) + \mathrm{MR}(5) + \mathrm{MR}(6) = 0$;
if $\mathrm{MR}(7) \neq 0$, $\mathrm{MR}(4) + \mathrm{MR}(5) + \mathrm{MR}(6) \neq 0$;
$\mathrm{MR}(7) \neq 1$.

the ARIMA model for the time series is restricted in the following ways:

$\mathrm{MR}(k) \geq 0$, for $k = 8, 9, \ldots, 14$;
if $\mathrm{MR}(14) = 0$, $\mathrm{MR}(11) + \mathrm{MR}(12) + \mathrm{MR}(13) = 0$;
if $\mathrm{MR}(14) \neq 0$, $\mathrm{MR}(11) + \mathrm{MR}(12) + \mathrm{MR}(13) \neq 0$;
$\mathrm{MR}(14) \neq 1$.

4: NMR – INTEGER *Input*

On entry: the number of values specified in the array MR. It takes the value 7 if no ARIMA model for the time series is supplied but otherwise it takes the value 14. Thus NMR acts as an indicator as to whether backforecasting can be carried out.

Constraint: $\mathrm{NMR} = 7$ or 14.

5: PAR(NPAR) – REAL (KIND=nag_wp) array *Input*

On entry: the parameters of the filtering model, followed by the parameters of the ARIMA model for the time series, if supplied. Within each model the parameters are in the standard order of non-seasonal AR and MA followed by seasonal AR and MA.

6: NPAR – INTEGER *Input*

On entry: the total number of parameters held in array PAR.

Constraints:

if $\mathrm{NMR} = 7$, $\mathrm{NPAR} = \mathrm{MR}(1) + \mathrm{MR}(3) + \mathrm{MR}(4) + \mathrm{MR}(6)$;
if $\mathrm{NMR} = 14$, $\mathrm{NPAR} = \mathrm{MR}(1) + \mathrm{MR}(3) + \mathrm{MR}(4) + \mathrm{MR}(6) +$
$\mathrm{MR}(8) + \mathrm{MR}(10) + \mathrm{MR}(11) + \mathrm{MR}(13)$.

Note: the first constraint (i.e., $\mathrm{MR}(1) + \mathrm{MR}(3) + \mathrm{MR}(4) + \mathrm{MR}(6) > 0$) on the orders of the filtering model, in parameter MR, ensures that $\mathrm{NPAR} > 0$.

7: CY – REAL (KIND=nag_wp) *Input*

On entry: if the ARIMA model is known (i.e., $\mathrm{NMR} = 14$), CY must specify the constant term of the ARIMA model for the time series. If this model is not known (i.e., $\mathrm{NMR} = 7$), then CY is not used.

8: WA(NWA) – REAL (KIND=nag_wp) array *Workspace*
9: NWA – INTEGER *Input*

On entry: the dimension of the array WA as declared in the (sub)program from which G13BAF is called. Workspace is only required if the ARIMA model for the time series is known.

Constraints:

let $K = \mathrm{MR}(3) + \mathrm{MR}(6) \times \mathrm{MR}(7) + \mathrm{MR}(8) + \mathrm{MR}(9) + (\mathrm{MR}(11) + \mathrm{MR}(12)) \times \mathrm{MR}(14)$, then

if $\mathrm{NMR} = 14$, $\mathrm{NWA} \geq K \times (K + 2)$;
if $\mathrm{NMR} = 7$, $\mathrm{NWA} \geq 1$.

10: B(NB) – REAL (KIND=nag_wp) array *Output*

On exit: the filtered output series. If the ARIMA model for the time series was known, and hence Q_y' backforecasts were supplied in Y, then B contains Q_y' 'filtered' backforecasts followed by the filtered series. Otherwise, the filtered series begins at the start of B just as the original series began at the start of Y. In either case, if the value of the series at time t is held in $\mathrm{Y}(t)$, then the filtered value at time t is held in $\mathrm{B}(t)$.

11: NB – INTEGER *Input*

On entry: the dimension of the array B as declared in the (sub)program from which G13BAF is called. In addition to holding the returned filtered series, B is also used as an intermediate work array if the ARIMA model for the time series was known.

Constraints:

if $\mathrm{NMR} = 14$, $\mathrm{NB} \geq \mathrm{NY} + \max(K_3, K_1 + K_2)$;
if $\mathrm{NMR} = 7$, $\mathrm{NB} \geq \mathrm{NY}$.

Where

$K_1 = \mathrm{MR}(1) + \mathrm{MR}(4) \times \mathrm{MR}(7)$;
$K_2 = \mathrm{MR}(2) + \mathrm{MR}(5) \times \mathrm{MR}(7)$;
$K_3 = \mathrm{MR}(3) + \mathrm{MR}(6) \times \mathrm{MR}(7)$.

12: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction for details.

For environments where it might be inappropriate to halt program execution when an error is detected, the value -1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, if you are not familiar with this parameter, the recommended value is 0. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

On exit: $\mathrm{IFAIL} = 0$ unless the routine detects an error or a warning has been flagged (see Section 6).

6 Error Indicators and Warnings

If on entry $\mathrm{IFAIL} = 0$ or -1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

$\mathrm{IFAIL} = 1$

On entry, $\mathrm{NMR} \neq 7$ and $\mathrm{NMR} \neq 14$.

$\mathrm{IFAIL} = 2$

On entry, the orders vector MR does not satisfy the constraints given in Section 5.

$\mathrm{IFAIL} = 3$

On entry, NPAR is inconsistent with the contents of MR (see Section 5).

$\mathrm{IFAIL} = 4$

On entry, NY is too small to successfully carry out the requested filtering, (see Section 5).

$\mathrm{IFAIL} = 5$

On entry, the work array WA is too small.

$\mathrm{IFAIL} = 6$

On entry, the array B is too small.

$\mathrm{IFAIL} = 7$

The orders vector for the filtering model is invalid.

IFAIL = 8

The orders vector for the ARIMA model is invalid. (Only occurs if NMR = 14.)

IFAIL = 9

The initial values of the filtered series are indeterminate for the given models.

IFAIL = −999

Internal memory allocation failed.

7 Accuracy

Accuracy and stability are high except when the MA parameters are close to the invertibility boundary.

8 Further Comments

If an ARIMA model is supplied, a local workspace array of fixed length is allocated internally by G13BAF. The total size of this array amounts to K integer elements, where K is the expression defined in the description of the parameter WA.

The time taken by G13BAF is approximately proportional to

$$\mathrm{NY} \times (\mathrm{MR}(1) + \mathrm{MR}(3) + \mathrm{MR}(4) + \mathrm{MR}(6)),$$

with an appreciable fixed increase if an ARIMA model is supplied for the time series.

9 Example

This example reads a time series of length 296. It reads the univariate ARIMA $(4, 0, 2, 0, 0, 0, 0)$ model and the ARIMA filtering $(3, 0, 0, 0, 0, 0, 0)$ model for the series. Two initial backforecasts are required and these are calculated by a call to G13AJF . The backforecasts are inserted at the start of the series and G13BAF is called to perform the calculations.

9.1 Program Text

```
    Program g13bafe

!     G13BAF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: g13ajf, g13baf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: cx, cy, rms
      Integer                          :: i, idd, ifail, ifv, ii, ij, ipar,    &
                                          iqxd, ist, iw, nb, nmr, npar, nparx, &
                                          nst, nwa, nx, ny, pp, qp, sy
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: b(:), fsd(:), fva(:), par(:),        &
                                          parx(:), st(:), w(:), wa(:), x(:),   &
                                          y(:)
      Integer                          :: isf(4), mrx(7)
      Integer, Allocatable             :: mr(:)
!     .. Intrinsic Procedures ..
      Intrinsic                        :: max, min, mod
!     .. Executable Statements ..
      Write (nout,*) 'G13BAF Example Program Results'
      Write (nout,*)
```

```
!     Skip heading in data file
      Read (nin,*)

!     Read in the problem size
      Read (nin,*) nx

!     Read univariate ARIMA for series
      Read (nin,*) mrx(1:7)
      Read (nin,*) cx

!     Calculate number of backforecasts required
      iqxd = mrx(3) + mrx(6)*mrx(7)
      If (iqxd/=0) Then
        nmr = 14
      Else
        nmr = 7
      End If

!     Back forecasts will be stored in first IQXD elements
!     of Y, the series will be stored in last NX elements of
!     Y, so calculate start point for the series
      sy = iqxd + 1

!     Calculate length of series with back forecasts
      ny = nx + iqxd

      Allocate (y(ny),mr(nmr))

!     Read in the series into the end of Y
      Read (nin,*) y(sy:ny)

!     Get back forecasts if required
      If (iqxd/=0) Then

!       Calculate number of parameters in ARIMA model
        nparx = mrx(1) + mrx(3) + mrx(4) + mrx(6)

        ist = mrx(4) + mrx(7) + mrx(2) + mrx(5) + mrx(3) + &
          max(mrx(1),mrx(6)*mrx(7))
        ifv = max(1,iqxd)
        qp = mrx(6)*mrx(7) + mrx(3)
        pp = mrx(4)*mrx(7) + mrx(1)
        iw = 6*nx + 5*nparx + qp*(qp+11) + 3*pp + 7
        Allocate (parx(nparx),x(nx),st(ist),fva(ifv),fsd(ifv),w(iw))

!       Read in initial values
        Read (nin,*) parx(1:nparx)

!       Reverse series
        x(nx:1:-1) = y(sy:ny)

!       Possible sign reversal for ARIMA constant
        idd = mrx(2) + mrx(5)
        If (mod(idd,2)/=0) Then
          cx = -cx
        End If

!       Calculate back forecasts
        ifail = 0
        Call g13ajf(mrx,parx,nparx,cx,1,x,nx,rms,st,ist,nst,iqxd,fva,fsd,ifv, &
          isf,w,iw,ifail)

!       Move back forecasts into Y, in reverse order
        y(1:iqxd) = fva(iqxd:1:-1)

!       Reverse sign for ARIMA constant back again
        If (mod(idd,2)/=0) Then
          cx = -cx
        End If
      End If
```

```
!     Read model by which to filter series
      Read (nin,*) mr(1:7)

!     Calculate NPAR
      ipar = mr(1) + mr(3) + mr(4) + mr(6)
      npar = ipar + nparx

      Allocate (par(npar))

!     Read in initial parameter values
      Read (nin,*) par(1:ipar)

      If (iqxd/=0) Then
!       Move ARIMA series into MR
        mr(8:14) = mrx(1:7)

!       Move parameters of ARIMA for Y into PAR
        par((ipar+1):(ipar+nparx)) = parx(1:nparx)
      End If

!     Move constant
      cy = cx

!     Set parameters for call to filter routine G13BAF
      If (nmr==14) Then
        nwa = mr(3) + mr(6)*mr(7) + mr(8) + mr(9) + (mr(11)+mr(12))*mr(14)
        nwa = nwa*(nwa+2)
        nb = ny + max(mr(3)+mr(6)*mr(7),mr(1)+mr(2)+(mr(4)+mr(5))*mr(7))
      Else
        nwa = 1
        nb = ny
      End If
      Allocate (wa(nwa),b(nb))

!     Filter series by call to G13BAF
      ifail = 0
      Call g13baf(y,ny,mr,nmr,par,npar,cy,wa,nwa,b,nb,ifail)

!     Display results
      If (iqxd/=0) Then
        Write (nout,*) '                   Original        Filtered'
        Write (nout,*) 'Backforecasts    y-series          series'
        ij = -iqxd
        Do i = 1, iqxd
          Write (nout,99999) ij, y(i), b(i)
          ij = ij + 1
        End Do
        Write (nout,*)
      End If
      Write (nout,*) &
        '       Filtered        Filtered        Filtered        Filtered'
      Write (nout,*) &
        '        series          series          series          series'
      Do i = iqxd + 1, ny, 4
        Write (nout,99998)(ii-iqxd,b(ii),ii=i,min(ny,i+3))
      End Do

99999 Format (1X,I8,F17.4,F15.4)
99998 Format (1X,I5,F9.4,I7,F9.4,I7,F9.4,I7,F9.4)
    End Program g13bafe
```

9.2 Program Data

```
G13BAF Example Program Data
 296                                                        :: NX
 4   0   2   0   0   0   0                                  :: MRX
 0.000                                                      :: CX
 53.8 53.6 53.5 53.5 53.4 53.1 52.7 52.4 52.2 52.0 52.0
 52.4 53.0 54.0 54.9 56.0 56.8 56.8 56.4 55.7 55.0 54.3
 53.2 52.3 51.6 51.2 50.8 50.5 50.0 49.2 48.4 47.9 47.6
```

```
47.5 47.5 47.6 48.1 49.0 50.0 51.1 51.8 51.9 51.7 51.2
50.0 48.3 47.0 45.8 45.6 46.0 46.9 47.8 48.2 48.3 47.9
47.2 47.2 48.1 49.4 50.6 51.5 51.6 51.2 50.5 50.1 49.8
49.6 49.4 49.3 49.2 49.3 49.7 50.3 51.3 52.8 54.4 56.0
56.9 57.5 57.3 56.6 56.0 55.4 55.4 56.4 57.2 58.0 58.4
58.4 58.1 57.7 57.0 56.0 54.7 53.2 52.1 51.6 51.0 50.5
50.4 51.0 51.8 52.4 53.0 53.4 53.6 53.7 53.8 53.8 53.8
53.3 53.0 52.9 53.4 54.6 56.4 58.0 59.4 60.2 60.0 59.4
58.4 57.6 56.9 56.4 56.0 55.7 55.3 55.0 54.4 53.7 52.8
51.6 50.6 49.4 48.8 48.5 48.7 49.2 49.8 50.4 50.7 50.9
50.7 50.5 50.4 50.2 50.4 51.2 52.3 53.2 53.9 54.1 54.0
53.6 53.2 53.0 52.8 52.3 51.9 51.6 51.6 51.4 51.2 50.7
50.0 49.4 49.3 49.7 50.6 51.8 53.0 54.0 55.3 55.9 55.9
54.6 53.5 52.4 52.1 52.3 53.0 53.8 54.6 55.4 55.9 55.9
55.2 54.4 53.7 53.6 53.6 53.2 52.5 52.0 51.4 51.0 50.9
52.4 53.5 55.6 58.0 59.5 60.0 60.4 60.5 60.2 59.7 59.0
57.6 56.4 55.2 54.5 54.1 54.1 54.4 55.5 56.2 57.0 57.3
57.4 57.0 56.4 55.9 55.5 55.3 55.2 55.4 56.0 56.5 57.1
57.3 56.8 55.6 55.0 54.1 54.3 55.3 56.4 57.2 57.8 58.3
58.6 58.8 58.8 58.6 58.0 57.4 57.0 56.4 56.3 56.4 56.4
56.0 55.2 54.0 53.0 52.0 51.6 51.6 51.1 50.4 50.0 50.0
52.0 54.0 55.1 54.5 52.8 51.4 50.8 51.2 52.0 52.8 53.8
54.5 54.9 54.9 54.8 54.4 53.7 53.3 52.8 52.6 52.6 53.0
54.3 56.0 57.0 58.0 58.6 58.5 58.3 57.8 57.3 57.0        :: End of Y
2.420  -2.380   1.160  -0.230   0.310  -0.470           :: PARX
3   0   0   0   0   0   0                               :: MR
1.970  -1.370   0.340                                   :: PAR
```

9.3 Program Results

```
G13BAF Example Program Results

                  Original        Filtered
Backforecasts     y-series         series
      -2           49.9807         3.4222
      -1           52.6714         3.0809

       Filtered          Filtered          Filtered          Filtered
        series            series            series            series
   1    2.9813      2    2.7803      3    3.7057      4    3.2450
   5    3.0760      6    3.0070      7    3.0610      8    3.1720
   9    3.1170     10    3.0360     11    3.2580     12    3.4520
  13    3.3320     14    3.6980     15    3.3140     16    3.8070
  17    3.3330     18    2.9580     19    3.2800     20    3.0960
  21    3.2270     22    3.0830     23    2.6410     24    3.1870
  25    2.9910     26    3.1110     27    2.8460     28    3.0240
  29    2.7030     30    2.6130     31    2.8060     32    2.9560
  33    2.8170     34    2.8950     35    2.8510     36    2.9160
  37    3.2530     38    3.3050     39    3.1830     40    3.3760
  41    2.9730     42    2.8610     43    3.0490     44    2.8420
  45    2.3190     46    2.3660     47    2.9410     48    2.3810
  49    3.3420     50    2.9340     51    3.1800     52    2.9230
  53    2.6470     54    2.8860     55    2.5310     56    2.6200
  57    3.4170     58    3.4940     59    3.2590     60    3.1310
  61    3.1420     62    2.6710     63    2.8990     64    2.8180
  65    3.2150     66    2.8800     67    2.9610     68    2.8800
  69    3.0020     70    2.8930     71    3.1210     72    3.2210
  73    3.2040     74    3.5360     75    3.7520     76    3.5630
  77    3.7260     78    3.1560     79    3.6310     80    2.9380
  81    3.1480     82    3.4490     83    3.1400     84    3.7380
  85    4.1200     86    3.1540     87    3.7480     88    3.3280
  89    3.3640     90    3.3400     91    3.3950     92    3.0720
  93    3.0050     94    2.8520     95    2.7810     96    3.1950
  97    3.2490     98    2.6370     99    3.0080    100    3.2410
 101    3.5570    102    3.2080    103    3.0880    104    3.3980
 105    3.1660    106    3.1960    107    3.2460    108    3.2870
 109    3.1590    110    3.2620    111    2.7280    112    3.4130
 113    3.2190    114    3.6750    115    3.8550    116    4.0100
 117    3.5380    118    3.8440    119    3.4660    120    3.0640
 121    3.4780    122    3.1140    123    3.5300    124    3.2400
```

125	3.3630	126	3.2610	127	3.3020	128	3.1150
129	3.3280	130	2.8730	131	3.0800	132	2.8390
133	2.6570	134	3.0260	135	2.4580	136	3.2600
137	2.8380	138	3.2150	139	3.1140	140	3.1050
141	3.1400	142	2.9100	143	3.1370	144	2.7500
145	3.1160	146	3.0680	147	2.8590	148	3.3840
149	3.5500	150	3.4160	151	3.1770	152	3.3390
153	3.0190	154	3.1780	155	3.0110	156	3.1940
157	3.2680	158	3.0500	159	2.8060	160	3.1850
161	3.0560	162	3.2690	163	2.7940	164	3.0900
165	2.7100	166	2.7890	167	2.9510	168	3.2440
169	3.2570	170	3.4360	171	3.4450	172	3.3780
173	3.3520	174	3.9180	175	2.9190	176	3.1780
177	2.2580	178	3.5150	179	2.8010	180	3.6030
181	3.2610	182	3.5300	183	3.3270	184	3.4420
185	3.5240	186	3.2720	187	3.1110	188	2.8240
189	3.2330	190	3.1500	191	3.5710	192	3.0810
193	2.7820	194	2.9040	195	3.2350	196	2.7970
197	3.1320	198	3.1680	199	4.5210	200	2.6650
201	4.6870	202	3.9470	203	3.2220	204	3.3410
205	3.9950	206	3.4820	207	3.3630	208	3.4550
209	3.2950	210	2.6910	211	3.4600	212	2.9440
213	3.4400	214	3.1830	215	3.4200	216	3.4100
217	4.0550	218	2.9990	219	3.8250	220	3.1340
221	3.5010	222	3.0430	223	3.2660	224	3.3660
225	3.2650	226	3.3720	227	3.2880	228	3.5470
229	3.6840	230	3.3100	231	3.6790	232	3.1780
233	2.9360	234	2.7910	235	3.8020	236	2.6100
237	4.1690	238	3.7460	239	3.4560	240	3.3910
241	3.5820	242	3.6220	243	3.4870	244	3.5770
245	3.4240	246	3.3960	247	3.1220	248	3.4300
249	3.4580	250	3.0280	251	3.7660	252	3.3770
253	3.2470	254	3.0180	255	2.9720	256	2.8000
257	3.2040	258	2.8020	259	3.4100	260	3.1680
261	2.4600	262	2.8810	263	3.1750	264	3.1740
265	4.8640	266	3.0600	267	2.9600	268	2.2530
269	2.5620	270	3.3150	271	3.3480	272	3.5900
273	3.2560	274	3.2320	275	3.6160	276	3.1700
277	3.2890	278	3.1200	279	3.3300	280	2.9910
281	2.9420	282	3.4070	283	2.8720	284	3.3470
285	3.1920	286	3.4880	287	4.0680	288	3.7550
289	3.0510	290	3.9680	291	3.3900	292	3.1380
293	3.6170	294	3.1700	295	3.4150	296	3.4830