# A MATLAB LIBRARY OF TEMPORAL DISAGGREGATION METHODS: SUMMARY

Enrique M. Quilis[1]

Instituto Nacional de Estadística
Paseo de la Castellana, 183
28046 - Madrid (SPAIN)

First version: December, 2002
This version: November, 2003

1 The author thanks Ana Abad, Juan Bógalo and Silvia Relloso for their help.

## 1. INTRODUCTION

The library includes a set of function to perform temporal disaggregation (distribution, averaging and interpolation), according to the following structure:

Adjustment or quadratic programming methods:

- bfl
- denton_uni
- sw (Stram-Wei method)

served by: tduni_print (ASCII output), tduni_plot (graphic output)

Model-based (or BLUE) methods:

- chowlin
- fernandez
- litterman
- ssc (Santos Silva-Cardoso method: a dynamic version of Chow-Lin)

served by: td_print (ASCII output), td_plot (graphic output)

Multivariate methods that include a transversal restriction:

- denton
- difonzo

served by: mtd_print (ASCII output), mtd_plot (graphic output)

Extrapolation is feasible using chowlin, fernandez, litterman, ssc and difonzo. Constrained extrapolation can be performed also by means of difonzo.

The presentation of the functions is self-contained: help text, script to run the function and output (ASCII file and plots).

Matlab source code is freely available if requested at: www.ine.es/info

## 2. BOOT-FEIBES-LISMAN

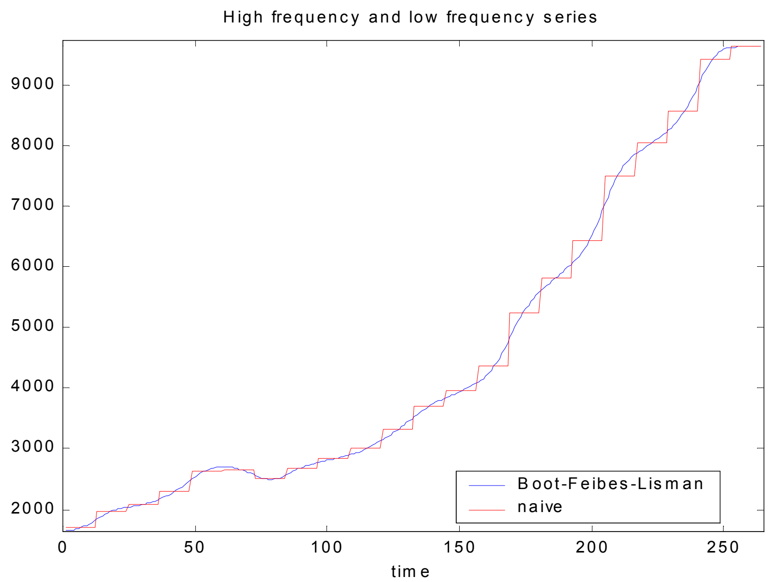
```
PURPOSE: Temporal disaggregation using the Boot-Feibes-Lisman method
---------------------------------------------------------------------
SYNTAX: res=bfl(Y,ta,d,s);
---------------------------------------------------------------------
OUTPUT: res: a structure
      res.meth  = 'Boot-Feibes-Lisman';
      res.N     = Number of low frequency data
      res.ta    = Type of disaggregation
      res.s     = Frequency conversion
      res.d     = Degree of differencing
      res.y     = High frequency estimate
      res.et    = Elapsed time
---------------------------------------------------------------------
INPUT: Y: Nx1 ---> vector of low frequency data
      ta: type of disaggregation
         ta=1 ---> sum (flow)
         ta=2 ---> average (index)
         ta=3 ---> last element (stock) ---> interpolation
      d: objective function to be minimized: volatility of ...
         d=0 ---> levels
         d=1 ---> first differences
         d=2 ---> second differences
      s: number of high frequency data points for each low frequency data point
         s= 4 ---> annual to quarterly
         s=12 ---> annual to monthly
         s= 3 ---> quarterly to monthly
---------------------------------------------------------------------
LIBRARY: sw
---------------------------------------------------------------------
SEE ALSO: tduni_print, tduni_plot
---------------------------------------------------------------------
REFERENCE: Boot, J.C.G., Feibes, W. y Lisman, J.H.C. (1967)
"Further methods of derivation of quarterly figures from annual data",
Applied Statistics, vol. 16, n. 1, p. 65-75.
```

Application:

```
Y=load('c:\x\td\data\Y.anu');
res=bfl(Y,1,1,12);
tduni_print(res,'td.sal');
tduni_plot(res);
edit td.sal
```

ASCII file containing detailed output:

```
 ****************************************************
 TEMPORAL DISAGGREGATION METHOD: Boot-Feibes-Lisman
 ****************************************************
----------------------------------------------------
 Number of low-frequency observations  :   22
 Frequency conversion                  :   12
 Number of high-frequency observations :  264
 ----------------------------------------------------
 Degree of differencing                :    1
 Type of disaggregation: sum (flow).
 ----------------------------------------------------
 High frequency series (columnwise):
 ----------------------------------------------------
 4972.2800
 4971.1389
 .........
 .........
 .........
 7898.7692
 7899.3631
 7899.6600
 ----------------------------------------------------
 Elapsed time:   0.3200
```



**3.** STRAM-WEI

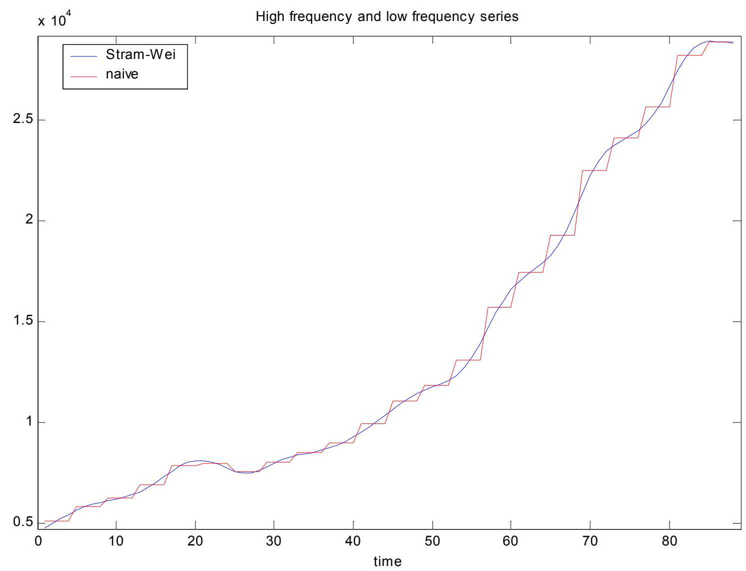
```
% PURPOSE: Temporal disaggregation using the Stram-Wei method.
% ------------------------------------------------------------------
% SYNTAX: res = sw(Y,ta,d,s,v);
% ------------------------------------------------------------------
% OUTPUT: res: a structure
%        res.meth  = 'Stram-Wei';
%        res.N     = Number of low frequency data
%        res.ta    = Type of disaggregation
%        res.d     = Degree of differencing
%        res.s     = Frequency conversion
%        res.H     = nxN temporal disaggregation matrix
%        res.y     = High frequency estimate
%        res.et    = Elapsed time
% ------------------------------------------------------------------
% INPUT: Y: Nx1 ---> vector of low frequency data
%       ta: type of disaggregation
%           ta=1 ---> sum (flow)
%           ta=2 ---> average (index)
%           ta=3 ---> last element (stock) ---> interpolation
%       d: number of unit roots
%       s: number of high frequency data points for each low frequency data point
%           s= 4 ---> annual to quarterly
%           s=12 ---> annual to monthly
%           s= 3 ---> quarterly to monthly
%       v: (n-d)x(n-d) VCV matrix of high frequency stationary series
% ------------------------------------------------------------------
% LIBRARY: aggreg, aggreg_v, dif, movingsum
% ------------------------------------------------------------------
% SEE ALSO: bfl, tduni_print, tduni_plot
% ------------------------------------------------------------------
% REFERENCE: Stram, D.O. & Wei, W.W.S. (1986) "A methodological note on the
% disaggregation of time series totals", Journal of Time Series Analysis,
% vol. 7, n. 4, p. 293-302.
```

Application:

```
Y=load('c:\x\td\data\Y.anu');
N = length(Y);  n = s*N;
% Defining the VCV matrix of stationary high-frequency time series
% Assumption of the example: IMA(d,2)
th1 =  0.9552; th2 = -0.0015; va = 0.87242 * ((223.5965)^2);
acf0 = va * (1+th1^2+th2^2); acf1 = -va * th1 * (1-th2); acf2 = -va * th2;
a0(1:n-d)=acf0; a1(1:n-d-1)=acf1; a2(1:n-d-2)=acf2;
v=diag(a0)+diag(a1,-1)+diag(a2,-2); v=v+tril(v)';
res = sw(Y,1,1,4,v);
tduni_print(res,'sw.sal');
tduni_plot(res);
edit sw.sal
```

ASCII file containing detailed output:

```
*****************************************************
TEMPORAL DISAGGREGATION METHOD: Stram-Wei
*****************************************************
-----------------------------------------------------
Number of low-frequency observations  :   22
Frequency conversion                  :    4
Number of high-frequency observations :   88
-----------------------------------------------------
Degree of differencing                :    1
Type of disaggregation: sum (flow).
-----------------------------------------------------
High frequency series (columnwise):
-----------------------------------------------------
4792.4658
5015.8665
 ........
 ........
 ........
28880.7153
28822.8148
-----------------------------------------------------
Elapsed time:   0.1100
```



## 4. DENTON

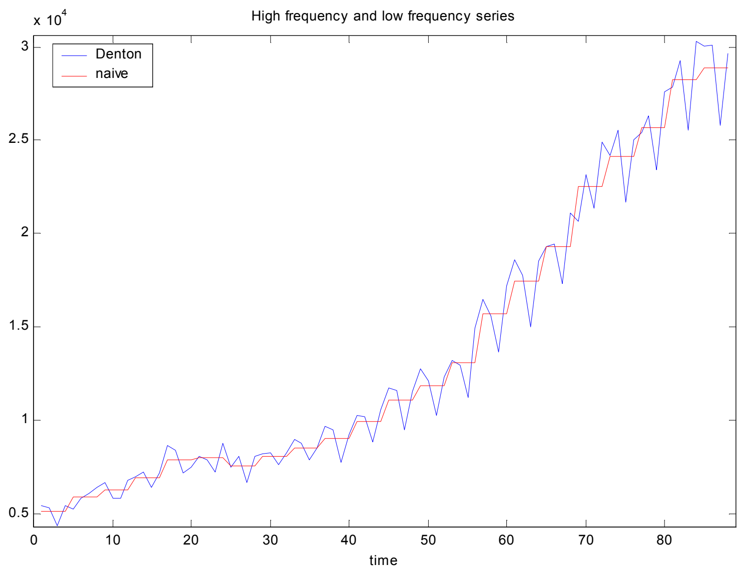
```
PURPOSE: Temporal disaggregation using the Denton method
--------------------------------------------------------------------
SYNTAX: res=denton_uni(Y,x,ta,d,s);
--------------------------------------------------------------------
OUTPUT: res: a structure
        res.meth  = 'Denton';
        res.N     = Number of low frequency data
        res.ta    = Type of disaggregation
        res.s     = Frequency conversion
        res.d     = Degree of differencing
        res.y     = High frequency estimate
        res.U     = Low frequency residuals
        res.u     = High frequency residuals
        res.et    = Elapsed time
--------------------------------------------------------------------
INPUT: Y: Nx1 ---> vector of low frequency data
       x: nx1 ---> vector of low frequency data
       ta: type of disaggregation
          ta=1 ---> sum (flow)
          ta=2 ---> average (index)
          ta=3 ---> last element (stock) ---> interpolation
       d: objective function to be minimized: volatility of ...
          d=0 ---> levels
          d=1 ---> first differences
          d=2 ---> second differences
       s: number of high frequency data points for each low frequency data point
          s= 4 ---> annual to quarterly
          s=12 ---> annual to monthly
          s= 3 ---> quarterly to monthly
--------------------------------------------------------------------
LIBRARY: aggreg, bfl
--------------------------------------------------------------------
SEE ALSO: tduni_plot, tduni_print
--------------------------------------------------------------------
REFERENCE: Denton, F.T. (1971) "Adjustment of monthly or quarterly
series to annual totals: an approach based on quadratic minimization",
Journal of the American Statistical Society, vol. 66, n. 333, p. 99-102.
```

Application:

```
Y=load('c:\x\td\data\Y.prn');
x=load('c:\x\td\data\x.ind');
res=denton_uni(Y,x,1,1,4);
tduni_print(res,'td.sal');
tduni_plot(res);
edit td.sal
```

ASCII file containing detailed output:

```
****************************************************
TEMPORAL DISAGGREGATION METHOD: Denton
****************************************************
----------------------------------------------------
Number of low-frequency observations  :   22
Frequency conversion                  :    4
Number of high-frequency observations :   88
----------------------------------------------------
Degree of differencing                :    1
Type of disaggregation: sum (flow).
----------------------------------------------------
High frequency series (columnwise):
----------------------------------------------------
15374.9285
15169.7571
..........
..........
..........
24883.3098
20609.0705
24415.4509
----------------------------------------------------
Elapsed time:   0.0500
```



## 5. CHOW-LIN

```
PURPOSE: Temporal disaggregation using the Chow-Lin method
------------------------------------------------------------
SYNTAX: res=chowlin(Y,x,ta,s,type);
------------------------------------------------------------
OUTPUT: res: a structure
        res.meth    ='Chow-Lin';
        res.ta      = type of disaggregation
        res.type    = method of estimation
        res.N       = nobs. of low frequency data
        res.n       = nobs. of high-frequency data
        res.pred    = number of extrapolations
        res.s       = frequency conversion between low and high freq.
        res.p       = number of regressors (including intercept)
        res.Y       = low frequency data
        res.x       = high frequency indicators
        res.y       = high frequency estimate
        res.y_dt    = high frequency estimate: standard deviation
        res.y_lo    = high frequency estimate: sd - sigma
        res.y_up    = high frequency estimate: sd + sigma
        res.u       = high frequency residuals
        res.U       = low frequency residuals
        res.beta    = estimated model parameters
        res.beta_sd = estimated model parameters: standard deviation
        res.beta_t  = estimated model parameters: t ratios
        res.rho     = innovational parameter
        res.aic     = Information criterion: AIC
        res.bic     = Information criterion: BIC
        res.val     = Objective function used by the estimation method
        res.r       = grid of innovational parameters used by the estimation method
------------------------------------------------------------
INPUT: Y: Nx1 ---> vector of low frequency data
      x: nxp ---> matrix of high frequency indicators (without intercept)
      ta: type of disaggregation
         ta=1 ---> sum (flow)
         ta=2 ---> average (index)
         ta=3 ---> last element (stock) ---> interpolation
      s: number of high frequency data points for each low frequency data points
         s= 4 ---> annual to quarterly
         s=12 ---> annual to monthly
         s= 3 ---> quarterly to monthly
      type: estimation method:
         type=0 ---> weighted least squares
         type=1 ---> maximum likelihood
------------------------------------------------------------
LIBRARY: aggreg
------------------------------------------------------------
SEE ALSO: litterman, fernandez, td_plot, td_print
------------------------------------------------------------
REFERENCE: Chow, G. y Lin, A.L. (1971) "Best linear unbiased
distribution and extrapolation of economic time series by related
series", Review of Economic and Statistics, vol. 53, n. 4, p. 372-375.
```

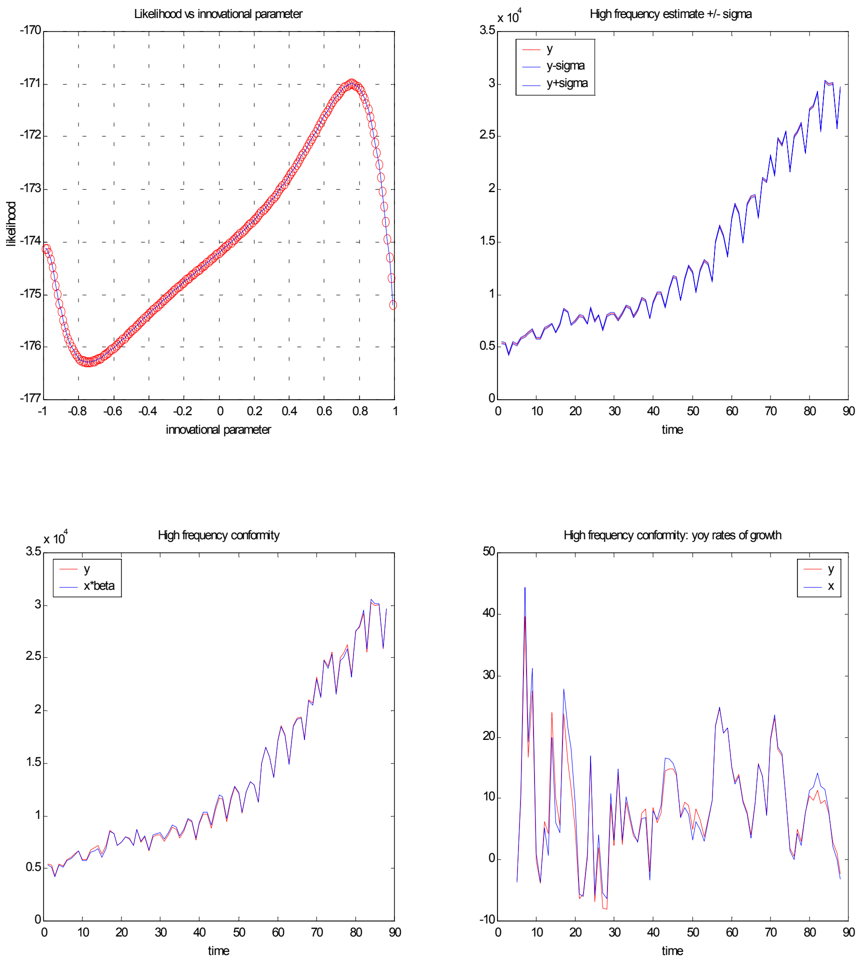
Application:

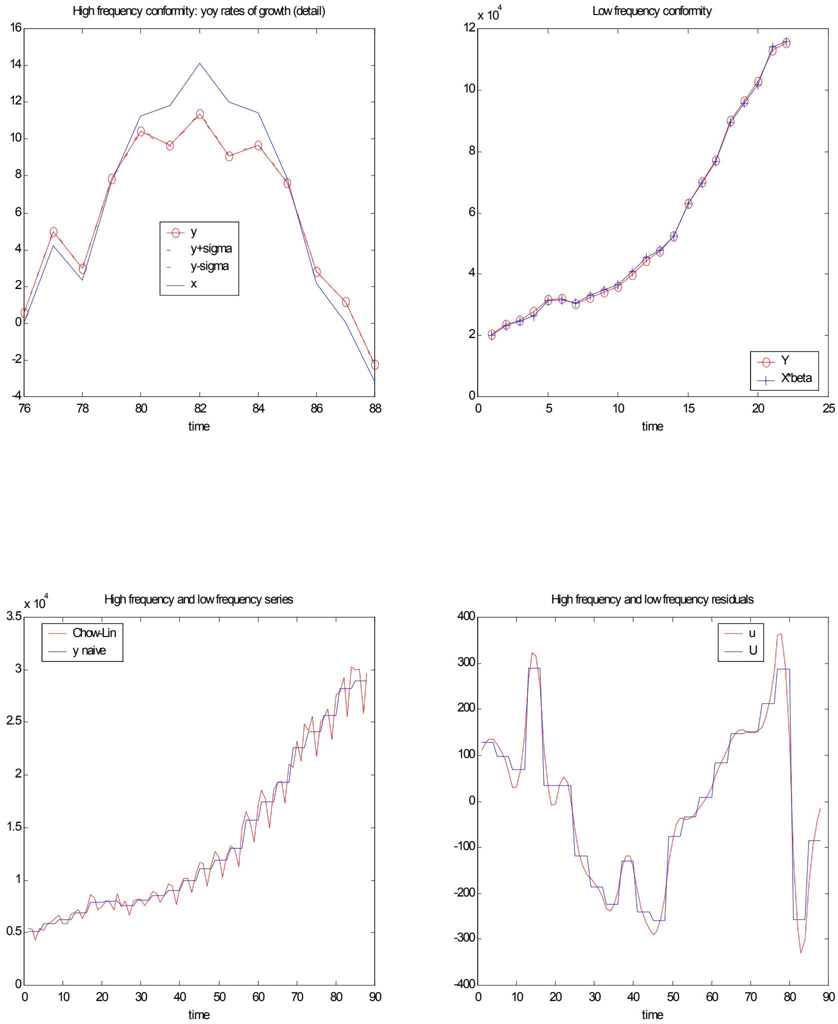
```
Y=load('c:\x\td\data\Y.prn');
x=load('c:\x\td\data\x.ind');
res=chowlin(Y,x,1,4,1);
td_print(res,'td.sal',1);     % op1=1: series are printed in ASCII file
td_plot(res);
edit td.sal
```

ASCII file containing detailed output:

```
****************************************************
TEMPORAL DISAGGREGATION METHOD: Chow-Lin
****************************************************
----------------------------------------------------
Number of low-frequency observations :   22
Frequency conversion                 :    4
Number of high-frequency observations:   88
Number of extrapolations             :    0
Number of indicators (+ constant)    :    2
----------------------------------------------------
Type of disaggregation: sum (flow).
----------------------------------------------------
Estimation method: Maximum likelihood.
----------------------------------------------------
Beta parameters (columnwise):
  * Estimate
  * Std. deviation
  * t-ratios
----------------------------------------------------
 215.4518       111.7079         1.9287
   0.9828         0.0069       142.0272
----------------------------------------------------
Innovational parameter:   0.7600
----------------------------------------------------
AIC:  10.0340
BIC:  10.1828
----------------------------------------------------
Low-frequency correlation
 - levels     : 0.9998
 - yoy rates  : 0.9617
----------------------------------------------------
High-frequency correlation
 - levels     : 0.9998
 - yoy rates  : 0.9812
----------------------------------------------------
High-frequency volatility of yoy rates
 - estimate   : 8.4282
 - indicator  : 9.0226
 - ratio      : 0.9341
----------------------------------------------------
```

```
High frequency series (columnwise):
   * Estimate
   * Std. deviation
   * 1 sigma lower limit
   * 1 sigma upper limit
   * Residuals
 ---------------------------------------------------
  5400.9896        114.8247       5286.1649       5515.8143       112.3095
  5311.2409         83.7296       5227.5112       5394.9705       128.7034
  .........          .......      .........       .........       ........
  .........          .......      .........       .........       ........
  .........          .......      .........       .........       ........
 30079.6885         86.7557      29992.9328      30166.4443       -97.4913
 25874.7702         86.2867      25788.4835      25961.0569       -43.9249
 29614.4998        116.3242      29498.1756      29730.8240       -16.2417
 ---------------------------------------------------
 ---------------------------------------------------
 Elapsed time:   1.8100
```






A variant to be applied with a fixed innovational parameter:

```
PURPOSE: Temporal disaggregation using the Chow-Lin method
         rho parameter is fixed (supplied by the user)
---------------------------------------------------------------
SYNTAX: res=chowlin_fix(Y,x,ta,s,type,rho);
---------------------------------------------------------------
```

## 6. FERNÁNDEZ

```
PURPOSE: Temporal disaggregation using the Fernandez method
---------------------------------------------------------
SYNTAX: res=fernandez(Y,x,ta,s);
---------------------------------------------------------
OUTPUT: res: a structure
        res.meth    ='Fernandez';
        res.ta      = type of disaggregation
        res.type    = method of estimation
        res.N       = nobs. of low frequency data
        res.n       = nobs. of high-frequency data
        res.pred    = number of extrapolations
        res.s       = frequency conversion between low and high freq.
        res.p       = number of regressors (including intercept)
        res.Y       = low frequency data
        res.x       = high frequency indicators
        res.y       = high frequency estimate
        res.y_dt    = high frequency estimate: standard deviation
        res.y_lo    = high frequency estimate: sd - sigma
        res.y_up    = high frequency estimate: sd + sigma
        res.u       = high frequency residuals
        res.U       = low frequency residuals
        res.beta    = estimated model parameters
        res.beta_sd = estimated model parameters: standard deviation
        res.beta_t  = estimated model parameters: t ratios
        res.aic     = Information criterion: AIC
        res.bic     = Information criterion: BIC
---------------------------------------------------------
INPUT: Y: Nx1 ---> vector of low frequency data
     x: nxp ---> matrix of high frequency indicators (without intercept)
     ta: type of disaggregation
        ta=1 ---> sum (flow)
        ta=2 ---> average (index)
        ta=3 ---> last element (stock) ---> interpolation
     s: number of high frequency data points for each low frequency data points
        s= 4 ---> annual to quarterly
        s=12 ---> annual to monthly
        s= 3 ---> quarterly to monthly
---------------------------------------------------------
LIBRARY: aggreg
---------------------------------------------------------
SEE ALSO: chowlin, litterman, td_plot, td_print
---------------------------------------------------------
REFERENCE: Fernández, R.B.(1981)"Methodological note on the
estimation of time series", Review of Economic and Statistics,
vol. 63, n. 3, p. 471-478.
```

Application:

```
Y=load('c:\x\td\data\Y.prn');
x=load('c:\x\td\data\x.tri');
res=fernandez(Y,x,1,4);
td_print(res,'td.sal',1);     % op1=1: series are printed in ASCII file
td_plot(res);
edit td.sal
```

ASCII file containing detailed output:

```
****************************************************
TEMPORAL DISAGGREGATION METHOD: Fernandez
****************************************************
----------------------------------------------------
Number of low-frequency observations :   22
Frequency conversion                 :    4
Number of high-frequency observations:   90
Number of extrapolations             :    2
Number of indicators (+ constant)    :    2
----------------------------------------------------
Type of disaggregation: sum (flow).
----------------------------------------------------
Estimation method: Maximum likelihood.
----------------------------------------------------
Beta parameters (columnwise):
  * Estimate
  * Std. deviation
  * t-ratios
----------------------------------------------------
 564.9834       195.9404         2.8834
   0.9360         0.0292        32.0284
----------------------------------------------------
Innovational parameter:   1.0000
----------------------------------------------------
AIC:   9.6079
BIC:   9.7567
----------------------------------------------------
Low-frequency correlation
 - levels     : 0.9998
 - yoy rates  : 0.9617
----------------------------------------------------
High-frequency correlation
 - levels     : 0.9997
 - yoy rates  : 0.9817
----------------------------------------------------
High-frequency volatility of yoy rates
 - estimate   : 8.3477
 - indicator  : 9.1506
 - ratio      : 0.9123
----------------------------------------------------
```

```
High frequency series (columnwise):
  * Estimate
  * Std. deviation
  * 1 sigma lower limit
  * 1 sigma upper limit
  * Residuals
----------------------------------------------------
 5396.6742        91.6250       5305.0492      5488.2992        -0.0000
 5297.9198        60.8871       5237.0327      5358.8069         2.3349
 .........        .......       .........      .........         ......
 .........        .......       .........      .........         ......
 .........        .......       .........      .........         ......
30021.1833        73.6977      29947.4856     30094.8810       920.9566
26022.3844       108.3992      25913.9852     26130.7837       977.8951
29586.1687        92.9937      29493.1750     29679.1625      1006.3644
----------------------------------------------------
28366.5459       140.8431      28225.7028     28507.3889      1006.3644
29461.6792       176.5235      29285.1557     29638.2027      1006.3644
----------------------------------------------------
Elapsed time:   0.0500
```

Graphs are the same than in the Chow-Lin case, except that the first one (objective function vs innovational parameter) is not generated.

## 7. LITTERMAN

```
PURPOSE: Temporal disaggregation using the Litterman method
---------------------------------------------------------
SYNTAX: res=litterman(Y,x,ta,s,type);
---------------------------------------------------------
OUTPUT: res: a structure
          res.meth    ='Litterman';
          res.ta      = type of disaggregation
          res.type    = method of estimation
          res.N       = nobs. of low frequency data
          res.n       = nobs. of high-frequency data
          res.pred    = number of extrapolations
          res.s       = frequency conversion between low and high freq.
          res.p       = number of regressors (including intercept)
          res.Y       = low frequency data
          res.x       = high frequency indicators
          res.y       = high frequency estimate
          res.y_dt    = high frequency estimate: standard deviation
          res.y_lo    = high frequency estimate: sd - sigma
          res.y_up    = high frequency estimate: sd + sigma
          res.u       = high frequency residuals
          res.U       = low frequency residuals
          res.beta    = estimated model parameters
          res.beta_sd = estimated model parameters: standard deviation
          res.beta_t  = estimated model parameters: t ratios
          res.rho     = innovational parameter
          res.aic     = Information criterion: AIC
          res.bic     = Information criterion: BIC
          res.val     = Objective function used by the estimation method
          res.r       = grid of innovational parameters used by the estimation method
---------------------------------------------------------
INPUT: Y: Nx1 ---> vector of low frequency data
       x: nxp ---> matrix of high frequency indicators (without intercept)
       ta: type of disaggregation
           ta=1 ---> sum (flow)
           ta=2 ---> average (index)
           ta=3 ---> last element (stock) ---> interpolation
       s: number of high frequency data points for each low frequency data points
           s= 4 ---> annual to quarterly
           s=12 ---> annual to monthly
           s= 3 ---> quarterly to monthly
       type: estimation method:
           type=0 ---> weighted least squares
           type=1 ---> maximum likelihood
---------------------------------------------------------
LIBRARY: aggreg
---------------------------------------------------------
SEE ALSO: chowlin, fernandez, td_plot, td_print
---------------------------------------------------------
REFERENCE: Litterman, R.B. (1983a) "A random walk, Markov model
for the distribution of time series", Journal of Business and
Economic Statistics, vol. 1, n. 2, p. 169-173.
```

Application:

```
Y=load('c:\x\td\data\Y.prn');
x=load('c:\x\td\data\x.tri');
res=litterman(Y,x,1,4,0);
td_print(res,'td.sal',0); % op1=0: series are not printed in ASCII file
td_plot(res);
edit td.sal
```

ASCII file containing detailed output:

```
****************************************************
TEMPORAL DISAGGREGATION METHOD: Litterman
****************************************************
----------------------------------------------------
Number of low-frequency observations :   22
Frequency conversion                 :    4
Number of high-frequency observations:   90
Number of extrapolations             :    2
Number of indicators (+ constant)    :    2
----------------------------------------------------
Type of disaggregation: sum (flow).
----------------------------------------------------
Estimation method: Weighted least squares.
----------------------------------------------------
Beta parameters (columnwise):
  * Estimate
  * Std. deviation
  * t-ratios
----------------------------------------------------
 1205.4851        233.5241          5.1621
    0.7910          0.0480         16.4821
----------------------------------------------------
Innovational parameter:   0.9700
----------------------------------------------------
AIC:   7.9478
BIC:   8.0966
----------------------------------------------------
Low-frequency correlation
 - levels     : 0.9998
 - yoy rates  : 0.9617
----------------------------------------------------
High-frequency correlation
 - levels     : 0.9994
 - yoy rates  : 0.9735
----------------------------------------------------
High-frequency volatility of yoy rates
 - estimate   : 7.6249
 - indicator  : 9.1506
 - ratio      : 0.8333
----------------------------------------------------
Elapsed time:   2.5300
```

A variant to be applied with a fixed innovational parameter:

PURPOSE: Temporal disaggregation using the Litterman method
mu parameter is fixed (supplied by the user)
-----------------------------------------------------------
SYNTAX: res=litterman(Y,x,ta,s,type,mu);

Graphical output contains the same information than in the Chow-Lin case.

## 8. SANTOS SILVA-CARDOSO (ssc)

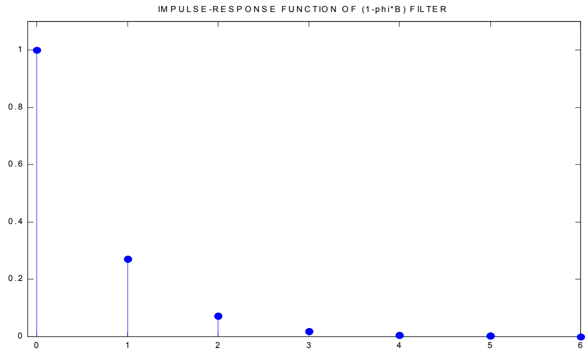
```
function res=ssc(Y,x,ta,s,type)
% PURPOSE: Temporal disaggregation using the dynamic Chow-Lin method
%              proposed by Santos Silva-Cardoso (2001).
% ---------------------------------------------------------
% SYNTAX: res=ssc(Y,x,ta,s,type);
% ---------------------------------------------------------
% OUTPUT: res: a structure
%       res.meth              ='Santos Silva-Cardoso';
%       res.ta                = type of disaggregation
%       res.type              = method of estimation
%       res.N                 = nobs. of low frequency data
%       res.n                 = nobs. of high-frequency data
%       res.pred              = number of extrapolations
%       res.s                 = frequency conversion between low and high freq.
%       res.p                 = number of regressors (+ intercept)
%       res.Y                 = low frequency data
%       res.x                 = high frequency indicators
%       res.y                 = high frequency estimate
%       res.y_dt              = high frequency estimate: standard deviation
%       res.y_lo              = high frequency estimate: sd - sigma
%       res.y_up              = high frequency estimate: sd + sigma
%       res.u                 = high frequency residuals
%       res.U                 = low frequency residuals
%       res.gamma             = estimated model parameters (including y(0))
%       res.gamma_sd          = estimated model parameters: standard deviation
%       res.gamma_t             = estimated model parameters: t ratios
%       res.rho             = dynamic parameter phi
%       res.beta              = estimated model parameters (excluding y(0))
%       res.beta_sd           = estimated model parameters: standard deviation
%       res.beta_t            = estimated model parameters: t ratios
%       res.aic             = Information criterion: AIC
%       res.bic             = Information criterion: BIC
%       res.val             = Objective function used by the estimation method
%       res.r               = grid of dynamic parameters used by the estimation method
%       res.et              = elapsed time
% ---------------------------------------------------------
% INPUT: Y: Nx1 ---> vector of low frequency data
%       x: nxp ---> matrix of high frequency indicators (without intercept)
%       ta: type of disaggregation
%          ta=1 ---> sum (flow)
%          ta=2 ---> average (index)
%          ta=3 ---> last element (stock) ---> interpolation
%       s: number of high frequency data points for each low frequency data points
%          s= 4 ---> annual to quarterly
%          s=12 ---> annual to monthly
%          s= 3 ---> quarterly to monthly
%       type: estimation method:
%          type=0 ---> weighted least squares
%          type=1 ---> maximum likelihood
% ---------------------------------------------------------
% LIBRARY: aggreg
```

% -----------------------------------------------------------
% SEE ALSO: chowlin, litterman, fernandez, td_plot, td_print
% -----------------------------------------------------------
% REFERENCE: Santos, J.M.C. y Cardoso, F.(2001) "The Chow-Lin method
% using dynamic models",Economic Modelling, vol. 18, p. 269-280.

```
****************************************************
 TEMPORAL DISAGGREGATION METHOD: Santos Silva-Cardoso
 ****************************************************

 ----------------------------------------------------
 Number of low-frequency observations :   32
 Frequency conversion                 :    4
 Number of high-frequency observations:  128
 Number of extrapolations             :    0
 Number of indicators (+ constant)    :    2
 ----------------------------------------------------
 Type of disaggregation: sum (flow).
 ----------------------------------------------------
 Estimation method: Maximum likelihood.
 ----------------------------------------------------
 Beta parameters (columnwise):
   * Estimate
   * Std. deviation
   * t-ratios
 ----------------------------------------------------
    1.0946         3.7817          0.2895
    0.6718         0.0049        136.9983
 ----------------------------------------------------
 Dynamic parameter:   0.2600
 ----------------------------------------------------
 Long-run beta parameters (columnwise):
     1.4792
     0.9078
 ----------------------------------------------------
 Truncation remainder: expected y(0):
   * Estimate
   * Std. deviation
   * t-ratios
 ----------------------------------------------------
  310.3328        90.5351          3.4278
 ----------------------------------------------------
 AIC:   5.2524
 BIC:   5.3898
 ----------------------------------------------------
 Low-frequency correlation
  - levels     : 0.9994
  - yoy rates  : 0.8561
 ----------------------------------------------------
 High-frequency correlation
  - levels     : 0.9993
  - yoy rates  : 0.8881
 ----------------------------------------------------
 High-frequency volatility of yoy rates
  - estimate   : 2.0592
  - indicator  : 2.3430
  - ratio      : 0.8789
 ----------------------------------------------------
```

Graphical output contains the same information than in the Chow-Lin case and includes a plot of the implied impulse-response function:




A variant to be applied with a fixed innovational parameter:

```
PURPOSE: Temporal disaggregation using the Santos Silva-Cardoso method
        Phi parameter is fixed (supplied by the user)
------------------------------------------------------------
SYNTAX: res=ssc(Y,x,ta,s,type,phi);
```

## 9. MULTIVARIATE DENTON

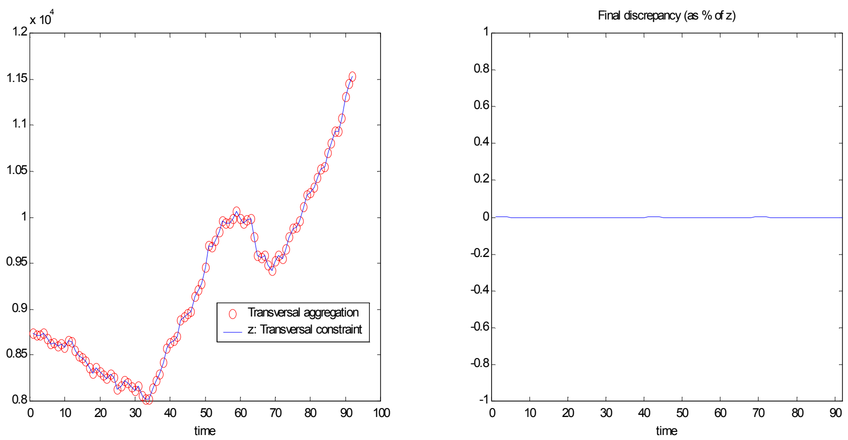
```
function res = denton(Y,x,z,ta,s,d);
% PURPOSE: Multivariate temporal disaggregation with transversal constraint
% ------------------------------------------------------------------------
% SYNTAX: res = denton(Y,x,z,ta,s,d);
% ------------------------------------------------------------------------
% OUTPUT: res: a structure
%         res.meth  = 'Multivariate Denton';
%         res.N     = Number of low frequency data
%         res.n     = Number of high frequency data
%         res.pred  = Number of extrapolations (=0 in this case)
%         res.ta    = Type of disaggregation
%         res.s     = Frequency conversion
%         res.d     = Degree of differencing
%         res.y     = High frequency estimate
%         res.et    = Elapsed time
% ------------------------------------------------------------------
% INPUT: Y: NxM ---> M series of low frequency data with N observations
%        x: nxM ---> M series of high frequency data with n observations
%        z: nzx1 ---> high frequency transversal constraint
%        ta: type of disaggregation
%            ta=1 ---> sum (flow)
%            ta=2 ---> average (index)
%            ta=3 ---> last element (stock) ---> interpolation
%        s: number of high frequency data points for each low frequency data points
%            s= 4 ---> annual to quarterly
%            s=12 ---> annual to monthly
%            s= 3 ---> quarterly to monthly
%        d: objective function to be minimized: volatility of ...
%            d=0 ---> levels
%            d=1 ---> first differences
%            d=2 ---> second differences
% ------------------------------------------------------------------
% LIBRARY: aggreg, aggreg_v, dif, vec, desvec
% ------------------------------------------------------------------
% SEE ALSO: difonzo, mtd_print, mtd_plot
% ------------------------------------------------------------------
% REFERENCE: di Fonzo, T. (1994) "Temporal disaggregation of a system of
% time series when the aggregate is known: optimal vs. adjustment methods",
% INSEE-Eurostat Workshop on Quarterly National Accounts, Paris, december
```

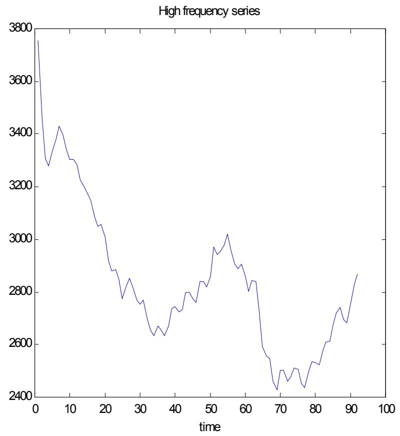
Application:

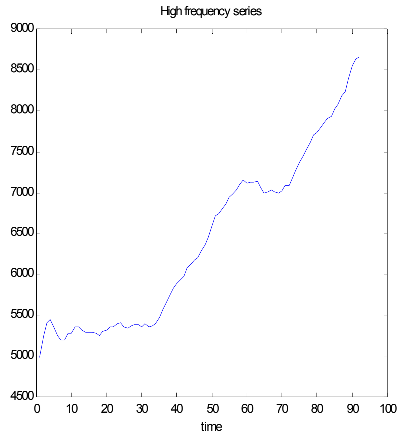
```
Y=load('YY.anu'); % Loading low frequency data
x=load('x.tri'); % Loading high frequency data
z=load('z.prn'); % Loading high frequency transversal restriction
res=denton(Y,x,z,2,4,1);
mtd_print(res,'mtd.sal');
edit mtd.sal;
mtd_plot(res,z);
```

ASCII file containing detailed output:

```
****************************************************
TEMPORAL DISAGGREGATION METHOD: Multivariate Denton
****************************************************
-----------------------------------------------------
Number of low-frequency observations  :   23
Frequency conversion                  :    4
Number of high-frequency observations :   92
Number of extrapolations              :    0
-----------------------------------------------------
Degree of differencing                :    1
Type of disaggregation: average (index).
-----------------------------------------------------
High frequency series (columnwise):
  * Point estimate
-----------------------------------------------------
3752.9096  4982.6505
3459.3681  5257.1693
.........  .........
.........  .........
.........  .........
2757.8458  8545.8074
2825.1411  8624.4561
2867.5816  8657.9733
-----------------------------------------------------
Elapsed time:   0.2800
```




High frequency series
3800
3600
3400
3200
3000
2800
2600
2400
0 10 20 30 40 50 60 70 80 90 100
time


High frequency series
9000
8500
8000
7500
7000
6500
6000
5500
5000
4500
0 10 20 30 40 50 60 70 80 90 100
time

## 10. DI FONZO

```
function res = difonzo(Y,x,z,ta,s,type,f);
% PURPOSE: Multivariate temporal disaggregation with transversal constraint
% ------------------------------------------------------------------------------
% SYNTAX: res = difonzo(Y,x,z,ta,s,type,f);
% ------------------------------------------------------------------------------
% OUTPUT: res: a structure
%         res.meth  = 'Multivariate di Fonzo';
%         res.N     = Number of low frequency data
%         res.n     = Number of high frequency data
%         res.pred  = Number of extrapolations
%         res.ta    = Type of disaggregation
%         res.s     = Frequency conversion
%         res.type  = Model for high frequency innovations
%         res.beta  = Model parameters
%         res.y     = High frequency estimate
%         res.d_y   = High frequency estimate: std. deviation
%         res.et    = Elapsed time
% ------------------------------------------------------------------------------
% INPUT: Y: NxM  ---> M series of low frequency data with N observations
%    x: nxm  ---> m series of high frequency data with n observations, m>=M see (*)
%       z: nzx1 ---> high frequency transversal constraint with nz obs.
%       ta: type of disaggregation
%          ta=1 ---> sum (flow)
%          ta=2 ---> average (index)
%          ta=3 ---> last element (stock) ---> interpolation
%       s: number of high frequency data points for each low frequency data points
%          s= 4 ---> annual to quarterly
%          s=12 ---> annual to monthly
%          s= 3 ---> quarterly to monthly
%       type: model for the high frequency innvations
%          type=0 ---> multivariate white noise
%          type=1 ---> multivariate random walk
% (*) Optional:
%       f: 1xM ---> Set the number of high frequency indicators linked to
%                   each low frequency variable. If f is explicitly included,
%                   the high frequency indicators should be placed in
%                   consecutive columns
% ------------------------------------------------------------------------------
% NOTE: Extrapolation is automatically performed when n>sN.
%      If n=nz>sN restricted extrapolation is applied.
%      Finally, if n>nz>sN extrapolation is perfomed in constrained
%      form in the first nz-sN observatons and in free form in
%      the last n-nz observations.
% ------------------------------------------------------------------------------
% LIBRARY: aggreg, dif, vec, desvec
% ------------------------------------------------------------------------------
% SEE ALSO: denton, mtd_print, mtd_plot
% ------------------------------------------------------------------------------
% REFERENCE: di Fonzo, T. (1990) "The estimation of M disaggregate time
% series when contemporaneous and temporal aggregates are known", Review
% of Economics and Statistics, vol. 72, n. 1, p. 178-182.
```

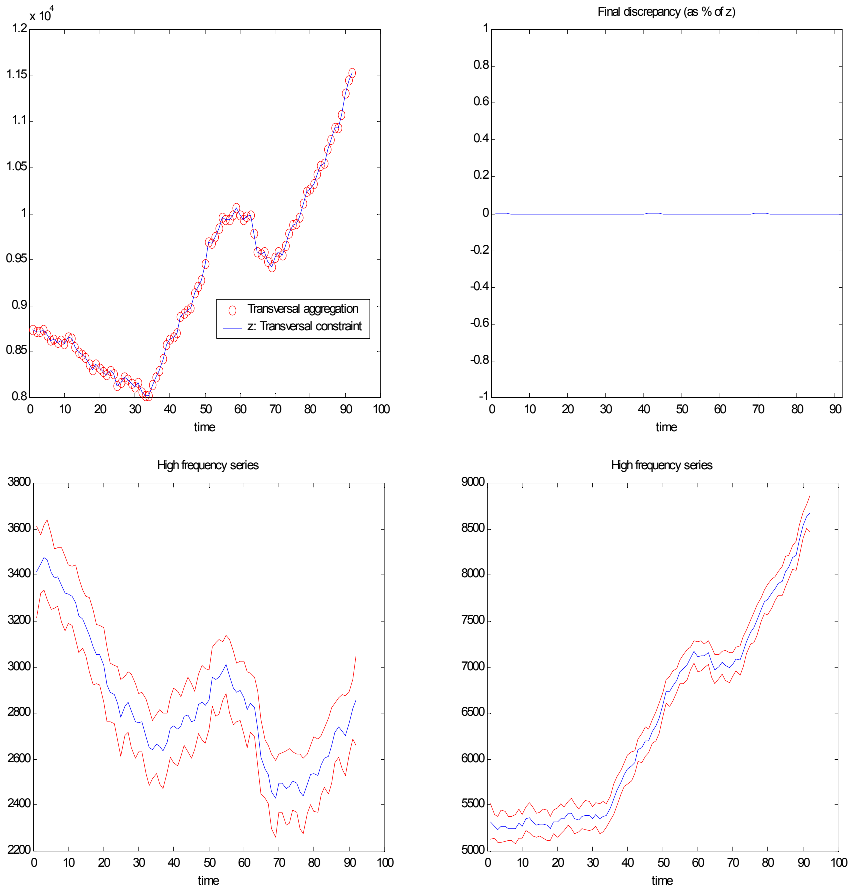
Application:

```
Y=load('YY.anu'); % Loading low frequency data
x=load('x.tri'); % Loading high frequency data
z=load('z.prn'); % Loading high frequency transversal restriction
res = difonzo(Y,x,z,2,4,1);
mtd_print(res,'mtd.sal');
edit mtd.sal;
mtd_plot(res,z);
```

ASCII file containing detailed output:

```
*******************************************************
TEMPORAL DISAGGREGATION METHOD: Multivariate di Fonzo
*******************************************************
-------------------------------------------------------
Number of low-frequency observations  :   23
Frequency conversion                  :    4
Number of high-frequency observations :   92
Number of extrapolations            :     0
-------------------------------------------------------
Model for the innovations: random walk.
Type of disaggregation: average (index).
-------------------------------------------------------
High frequency series (columnwise):
  * Point estimate
-------------------------------------------------------
3413.3839  5322.1762
3447.4092  5269.1282
.........  .........
.........  .........
.........  .........
2758.4657  8545.1875
2817.9882  8631.6090
2856.1605  8669.3944
```

```
--------------------------------------------------------
 High frequency series (columnwise):
   * Std. desviation
 --------------------------------------------------------
  197.8732   197.8732
  127.3900   127.3900
  .........  .........
  .........  .........
  .........  .........
  137.9397   137.9397
  128.1006   128.1006
  194.9112   194.9112
--------------------------------------------------------
Elapsed time:   0.3300
```



## APPENDIX: RELATIONSHIPS AMONG FUNCTIONS IN THE LIBRARY

The “X → Y” notation means “X function calls Y function”.

- bfl → sw
- denton_uni → aggreg, bfl
- sw → aggreg, aggreg_v, dif, movingsum

- chowlin → aggreg
- fernandez → aggreg
- litterman → aggreg
- ssc → aggreg

- denton → aggreg, aggreg_v, dif, vec, desvec
- difonzo → aggreg, dif, vec, desvec

- bal → vec, desvec
- td_print → tasa, aggreg
- td_plot → tasa
- tduni_plot → temporal_agg