

Package ‘tsdisagg2’

July 22, 2025

Type Package

Title Time Series Disaggregation

Version 0.1.0

Author Jorge Vieira <jorgealexandre Vieira@gmail.com>

Maintainer Jorge Vieira <jorgealexandre Vieira@gmail.com>

Description Disaggregates low frequency time series data to higher frequency series. Implements the following methods for temporal disaggregation: Boot, Feibes and Lisman (1967) <[DOI:10.2307/2985238](#)>, Chow and Lin (1971) <[DOI:10.2307/1928739](#)>, Fernandez (1981) <[DOI:10.2307/1924371](#)> and Litterman (1983) <[DOI:10.2307/1391858](#)>.

Depends R (>= 3.1)

License GPL (>= 2)

RoxygenNote 5.0.1

Suggests R.rsp

VignetteBuilder R.rsp

NeedsCompilation no

Repository CRAN

Date/Publication 2016-11-29 08:10:07

Contents

tsdisagg2	2
Index	4

tsdisagg2

*Time Series Disaggregation.***Description**

The "tsdisagg2" function performs temporal disaggregation or interpolation of low frequency to high frequency time series.

Usage

```
tsdisagg2(y, x, c = 0, method = "cl1", s = 4, type = "sum", ML = 0,
          rho, neg = 0, da, dz, plots = 0)
```

Arguments

y	A data.frame, matrix, list or vector with low frequency data.
x	A data.frame, matrix, list or vector with high frequency data.
c	Constant; If c=1, the model will be estimated with a constant; If c=0, the opposite case. (Default: c=0)
method	Set disaggregation method; Available methods are Boot, Feibes and Lisman (method="bfl1" or method="bfl2"), Chow and Lin (method="cl1" or method="cl2"), Fernandez (method="f") and Litterman (method="l"). Default: method="cl1"
s	Frequency of observations; Available frequencies are 3, 4 or 12. For example, if s=4, we have quaterly observations. (Default: s=4)
type	Type of restriction; Could be "last", "first", "sum" or "average". (Default: type="sum")
ML	Maximum Likelihood (ML=1) or Generalised Least Squares (ML=0) "rho" estimation. (Default: ML=0)
rho	Sets a value for "rho" (Default: rho=0)
neg	If neg=1, will be tested negative for "rho"; If neg=0, only positive values will be tested. (Default: neg=0)
da	First year considered on low frequency data.
dz	Last year considered on low frequency data.
plots	If plots=1, generates the plot of the estimated series and the plot with the Objective function values; If plots=0, the opposite case. (Default: plots=0)

Details

The function is used to disaggregate a low frequency to a higher frequency time series, while either the sum, the average, the first or the last value of the resulting high-frequency series is consistent with the low frequency series. Implements the following methods for temporal disaggregation: Boot, Feibes and Lisman (first and second differences), Chow and Lin (independent and AR(1) errors), Fernandez and Litterman. For Boot, Feibes and Lisman methods, the disaggregation can not be performed with help of indicator series. For the remaining methods, disaggregation can be performed with or without the help of one or more indicator series. If the high-frequency indicator(s) cover(s) a longer time span than the low-frequency series, an extrapolation is performed, using the same model as for interpolation.

Value

The function prints details of the disaggregation (smooth, loglik, ...), estimated parameters (sigma_ols, sigma_gls, model coefficients, ...) and the disaggregated series. The function also returns an invisible list containing all the numeric results.

Examples

```

anual <- runif( 19, 300, 455 )
indicators <- data.frame( runif( 76, 500, 700 ), runif( 76, 800, 980 ) )

### Constant ###

tsdisagg2( y=anual, x=indicators, c=1, da=1995, dz=2013, plots=1 )
# Estimate model with constant

### Method selection ###

tsdisagg2( y=anual, x=indicators, method="f", da=1995, dz=2013, plots=1 )
# Use option method

### "rho" value ###

tsdisagg2( y=anual, x=indicators, method="cl2", da=1995, dz=2013, plots=1 )
# Search for positive optimal "rho" is enabled (if method="cl2" or method="l")

tsdisagg2( y=anual, x=indicators, method="cl2", rho=0.35, da=1995, dz=2013, plots=1 )
# Set "rho" value manually (the grid search is not performed)

### Interpolation or distribution ###

tsdisagg2( y=anual, x=indicators, da=1995, dz=2013, method="f", type="last" )
# Performs disaggregation by interpolation with type="last" or type="first"

tsdisagg2( y=anual, x=indicators, da=1995, dz=2013, method="f", type="average" )
# Performs disaggregation by distribution with type="sum" or type="average"

### Use returned objects ###
td <- tsdisagg2( y=anual, x=indicators, da=1995, dz=2013, method="f", type="average" )
names(td)
td$BETA_ESTIMATION

```

Index

tsdisagg2, [2](#)