Package ‘TSsmoothing’

October 12, 2022

Type Package

Title Trend Estimation of Univariate and Bivariate Time Series with Controlled Smoothness

Version 0.1.0

Description It performs the smoothing approach provided by penalized least squares for univariate and bivariate time series, as proposed by Guerrero (2007) and Gerrero et al. (2017). This allows to estimate the time series trend by controlling the amount of resulting (joint) smoothness.

---


Depends R (>= 3.5.0)

Imports ggplot2 (>= 3.2.0), MASS (>= 7.3.0), gridExtra (>= 2.3.0), Matrix (>= 1.2.0)

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

NeedsCompilation no

Author L. Leticia Ramirez-Ramirez [aut, cre], Alejandro Islas-Camargo [aut], Victor M. Guerrero [aut]

Maintainer L. Leticia Ramirez-Ramirez <leticia.ramirez@cimat.mx>

Repository CRAN

Date/Publication 2019-07-15 10:50:03 UTC

R topics documented:

cormvc ................................................................. 2
emp_agr .............................................................. 2
### `corrmvc`

**Correlation from a 2d covariance matrix.**

**Description**

Computes the correlation given a covariance matrix of a bivariate variable.

**Usage**

```r
corrmvc(mat)
```

**Arguments**

- `mat` is a 2x2 covariance matrix

**Value**

The empirical correlation for the two series

---

### `emp_agr`

**Employment in agriculture**

**Description**

Dataset of the

**Usage**

```r
emp_agr
```

**Format**

A ts vector a length of 25 observations from 1991 to 2015.
**graph_trend**

Source

https://databank.worldbank.org/source/jobs#

---

**graph_trend**

*Plot of original and smoothed time series.*

---

**Description**

It plots the univariate or bivariate. This function is not intended for users but to be called by `trend_estimate`.

**Usage**

```r
graph_trend(dat, N, tau, dvar, label = NULL, jump = 1:N,
             bands = TRUE, las, bivariate = TRUE)
```

**Arguments**

- `dat` is a 2x2 covariance matrix
- `N` the number of observations
- `tau` the smoothed time series
- `dvar` the estimated variance for tau
- `label` the vectors of characters associated to the time points to appear in the axis
- `jump` if label is too long, jump thin them on the axis
- `bands` is TRUE to draw the approximately 95% confidence bands around tau
- `las` is 1 and 2 if the asis should appear vertical and horizontal, respectively
- `bivariate` is FALSE if dat is a univariate time series

**Value**

The empirical correlation for the two series
**lambda_value**  
*Calculation of the lambda value.*

**Description**
Obtains the lambda value for specific values of the smoothing level, correlation and length.

**Usage**

\[
\text{lambda_value}(s, \rho, N)
\]

**Arguments**
- \(s\) is a scalar that specifies the smoothing level.
- \(\rho\) is the estimated correlation of the two time series. If the time series is univariate, \(\rho\) should be 0.
- \(N\) the length of the bivariate time series.

**Value**
The value of lambda \(\text{lambda_value}\) that corresponds to a smoothing level \(s\)

A flag to indicate if the lambda value was read from \(\text{ltable}\)

**ltable**  
*Lambda values table.*

**Description**
An array that presents the lambda values according to time series (N), the smoothing value (s), and the ts correlation (rho).

**Usage**

\[
ltable
\]

**Format**
A 3d array with dimension 393 x 12 x 11, where dimensions are:

- \(N\) with values from 8 to 400
- \(s\) with smoothing values \(c(0.5, 0.6, 0.7, 0.75, 0.8, 0.825, 0.85, 0.875, 0.9, 0.925, 0.95, 0.975)\)
- \(\rho\) with values \(c(0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 0.95)\)
plot_trend

Details

3d array of float number that correspond to the lambda values that correspond to the specified values of the length of the

Source

http://www.diamondse.info/

---

plot_trend

Plot fo the time series and its smoothed version in ggplot

Description

It plots the univariate or bivariate time series and its smoothed version (trend) using ggplot. It directly uses the output of trend_estimate.

Usage

plot_trend(smoothedTS, title = NULL, xlab = "Time", ylab = names(dat)[1:2])

Arguments

- smoothedTS: Is an object generated by the function trend_estimate
- title: Main title of the graph
- xlab: Common x label
- ylab: A 2-length vector of characters.

Value

The ggplot of the original time series, their trend and its approximated 95

---

positive_definite

Checks if a squared matrix is positive definite and turn it to positive definied if necessary

Description

Checks if a squared matrix is positive definite and turn it to positive definied if necessary

Usage

positive_definite(m, c = NULL)
Arguments

\( m \)  
Is a 2x2 matrix.

\( c \)  
Is a small nonegative number.

Value

The same matrix (if positive definite) or its modification that is positive definite.

---

**preliminar**  
**Preliminar smoothing**

Description

Obtains the preliminar smoothed series, based on the preliminar lambda value and empirical estimates for Sigma_eta and Sigma_epsilon. This function is called by trend_estimate as part of the smoothing process.

Usage

```r
preliminar(dat)
```

Arguments

dat  
is a 2-column matrix with the observations of a bivariate time series. Each row correspond to the values at a given time.

Value

The preliminary smoothen series ptau.

The final estimate for sigma.eta

The time series correlation given by sigma.eta, rho.eta.

The preliminatory estimation for sigma_epsilon, sigma.epsilon,

A suggested value for lambda given by the empirical estimations.

The empirical time series correlation (preliminar to rho.eta), emp_rho.

The time series length \( N \).
psigma_estimates

Description
It computes the preliminary estimates of Sigma_epsilon and Sigma_eta

Usage
```
psigma_estimates(dat)
```

Arguments
- `dat`: is a 2 column matrix with for the bivariate time series observations. Each column correspond to the values at a given time.

Value
- Sigma epsilon
- Sigma eta

sigma_zf

Description
Function that calculates the empirical cross-covariance of order h for a bivariate time series.

Usage
```
sigma_zf(h, vec1, vec2, N)
```

Arguments
- `h`: the lag value.
- `vec1`: observations for the first variable of the bivariate time series.
- `vec2`: observations for the second variable of the bivariate time series.
- `N`: the common length of vec1 and vec2.

Value
The value of lambda that corresponds to a smoothing level s.
**smoothing_level**  
*Smoothing value*

**Description**  
Function that reports the smoothing level for a given value of lambda, N and rho (=0 if univariate).

**Usage**  

```r  
smoothing_level(lambda, rho, N)  
```

**Arguments**  
- `lambda`  
a nonegative number.
- `rho`  
the correlation of the time series.
- `N`  
the length of the observations.

**Value**  

S

---

**trade**  
*Annual Trade for USA and Mexico*

**Description**  
A dataset (matrix) containing the annual trade (  

**Usage**  

```r  
trade  
```

**Format**  
An object of class matrix with 49 rows and 2 columns.

**Source**  
**trend_estimate**  
*Trend estimation with controlled smoothing.*

**Description**

This is the main function that estimates the trend for univariate or bivariate time series for a specified smoothing level.

**Usage**

```r
trend_estimate(dat, smoothing_level = NULL, lambda = NULL,  
    plot = TRUE, label = time(dat), jump = NULL, las = 2,  
    bands = TRUE)
```

**Arguments**

- `dat` is a 2x2 matrix with the two time series. Each column correspond to the values at a given time.
- `smoothing_level` is a scalar between 0 and 1 that specifies the smoothing of the resulting time series tau.
- `lambda` is an alternative, the function directly accepts the lambda value that corresponds to the desired smoothing level.
- `plot` is TRUE when we want to plot of the original against the resulting series.
- `label` is a vector of characters that corresponds to the labels for each time point in the serie.
- `jump` is a vector of integers that specifies which values of labels should appear in the x labels.
- `las` is 1(2) if the x labels should be vertical (horizontal).
- `bands` is TRUE to include 95% confidence bands in the plots.

**Value**

- The smoothed series tau.
- The original data `dat`.
- The estimation for `sigma_eta`, `sigma.eta`
- The length of the time series `N`.
- The lambda value corresponding to the smoothing level.
- The diagonal values of the estimated variance of `tau`, `diag.var.tau`
- A flag that indicates if data is a bivariate time series.
Examples

# Employment in agriculture (% of total employment) (modeled ILO estimate) in OCDE members
data(emp_agr) #It is a ts object with one single time series
sts<-trend_estimate(emp_agr,0.70)
plot_trend(sts, title="Employment in agriculture in OCDE members", xlab = "Years")

# Data Trade (% of GDP) for USA and Mexico downloaded from
data(trade) #It is a numeric matrix with two columns
sts<-trend_estimate(trade,0.7)
plot_trend(sts, title="Trade in% of GDP",xlab="years")

ts_trade<-ts(trade, start=1969,end=2017) #We transform tade to a ts object
sts<-trend_estimate(ts_trade,0.7)
plot_trend(sts, title="Trade in% of GDP",xlab="years")
Index

* datasets
  emp_agr, 2
  ltable, 4
  trade, 8

corrmvc, 2

emp_agr, 2

graph_trend, 3

lambda_value, 4
ltable, 4

plot_trend, 5
positive_definite, 5
preliminar, 6
psigma_estimates, 7

sigma_zf, 7
smoothing_level, 8

trade, 8
trend_estimate, 9