Package ‘LSWPlib’

May 7, 2026

Type Package

Title Simulation and Spectral Estimation of Locally Stationary Wavelet Packet Processes

Version 0.1.0

Description Library of functions for the statistical analysis and simulation of Locally Stationary Wavelet Packet (LSWP) processes. The methods implemented by this library are described in Cardinali and Nason (2017) <doi:10.1111/jtsa.12230>.

Depends R (>= 2.10)

Imports stats, graphics, waveslim, wavethresh

License GPL-3

LazyData TRUE

Encoding UTF-8

RoxygenNote 7.1.2

NeedsCompilation no

Author Alessandro Cardinali [aut, cre],
Guy Nason [aut]

Maintainer Alessandro Cardinali <alessandro.cardinali@plymouth.ac.uk>

Repository CRAN

Date/Publication 2022-03-09 08:30:02 UTC

Contents

autoconv 2
best.basis 3
get.flat.basis 4
get.wavelet.basis 5
LSWPbasis 6
LSWPlib 7
LSWPsim 7
LSWPspec 8
marg.spec 10

plot.LSWPspec 11
sp500 12
summary.LSWPbasis 12

Index **14**

`autoconv` *Auto Convolution*

Description

`autoconv` computes the linear convolution of a numeric vector with itself. It is based on the `fft` function and is twicked to achieve maximum performance.

Usage

```
autoconv(x)
```

Arguments

`x`	a real or complex vector

Details

The speed of calculation for the linear convolution depends upon the number of factors in the number representing the vector length. This implementation maximizes calculation speed for vectors of dyadic length, or lengths with a single factor.

Value

The linear auto convolution of a given vector with itself, which is equivalent with its inner product.

Author(s)

Alessandro Cardinali

References

A. Cardinali and G.P. Nason (2017). Locally Stationary Wavelet Packet Processes: Basis Selection and Model Fitting. Journal of Time Series Analysis, 38:2, 151-174.

See Also

`fft`, `convolve`.

Examples

```
 v <- rnorm(n = 64)
vv <- autoconv(x = v)
```

`best.basis` *Best basis selection from a dyadic tree*

Description

`best.basis` returns a selection of packets from a dyadic tree where the selection is made by minimizing the cost functionals associated with each packet.

Usage

```
best.basis(wpc)
```

Arguments

`wpc`	this is a list containing the cost functionals associated with each packet. Each element in the list corresponds to a level of the dyadic tree.

Details

The function implements best basis selection from an arbitrary tree. Typically this tree is produced by other functions in this package and this function is also used to produce a selected basis. Since best basis methods are of general interest this function has been exported for possible other uses.

Value

A matrix of two columns where each row refers to a different selected packet. The first index refers to the tree level, the second index refers to the packet within that level.

Author(s)

Alessandro Cardinali

References

A. Cardinali and G.P. Nason (2017). Locally Stationary Wavelet Packet Processes: Basis Selection and Model Fitting. Journal of Time Series Analysis, 38:2, 151-174.

See Also

`LSWPbasis`, `LSWPspec`.

Examples

```
costs <- vector(mode = 'list', length = 4)
for(i in 1:4) costs[[i]] <- rnorm(2^i)^2
best.basis(wpc = costs)
```

`get.flat.basis` *Wavelet Packet Basis for a single scale*

Description

`get.flat.basis` returns the full set of packet indices relative to a basis for a single scale from the wavelet packet table.

Usage

```
get.flat.basis(scale)
```

Arguments

`scale`	The scale for which the indices of wavelet packet basis are returned. Typically this is a positive integer.

Details

This function is used internally by other routines but it might be useful when the wavelet packet spectral estimation over a fixed scale is of interest. The function returns an object of class `lswpbb`, whose structure is the same to the object produced by `best.basis`.

Value

A matrix of two columns where each row refers to a different selected packet. The first index is the argument `scale`, the second index refers to the packet within this level.

Author(s)

Alessandro Cardinali

References

A. Cardinali and G.P. Nason (2017). Locally Stationary Wavelet Packet Processes: Basis Selection and Model Fitting. Journal of Time Series Analysis, 38:2, 151-174.

See Also

`best.basis`, `get.wavelet.basis`

Examples

```
get.flat.basis(scale = 4)
```

`get.wavelet.basis` *Wavelet Packet indices for a Wavelet Basis*

Description

`get.wavelet.basis` returns the full set of packet indices relative to a wavelet basis selected from the wavelet packet table.

Usage

```
get.wavelet.basis(scale)
```

Arguments

`scale`	The maximum scale for which the indices of the wavelet basis are returned. Tipically this is a positive integer.

Details

This function is used internally by other routines but it might be useful when the 'classical' wavelet spectral estimation is of interest. The function returns an object of class `LSWPbasis`, whose structure is the same to the object produced by `best.basis`.

Value

A matrix of two columns where each row refers to a different selected packet. The first index refers to each given scale, the second index refers to the packet within this level.

Author(s)

Alessandro Cardinali

References

A. Cardinali and G.P. Nason (2017). Locally Stationary Wavelet Packet Processes: Basis Selection and Model Fitting. Journal of Time Series Analysis, 38:2, 151-174.

See Also

`best.basis`, `get.flat.basis`

Examples

```
get.wavelet.basis(scale = 4)
```

`LSWPbasis` *Estimate an LSWP basis by penalised least squares*

Description

`LSWPbasis` returns a matrix containing the wavelet packet basis indices.

Usage

```
LSWPbasis(x, wavelet, lev.max, smooth, spa, penalty = 0.976)
```

Arguments

`x`	a (locally stationary) time series of dyadic length.
`wavelet`	the wavelet used to estimate the wavelet packet spectra.
`lev.max`	the maximum scale for which the basis is fitted.
`smooth`	should the penasised least squares cost functionals be smoothed? Default value is `TRUE`.
`spa`	parameter for the local polynomial smothing implemented through `lowess`
`penalty`	implemets increasing penalty for increasing scales.

Details

This function fits a wavelet packet basis to data using a penalised least square method.

This function implements a data-driven basis selection of locally stationary time series. The wavelet argument is specified as in other functions of this package. Therefore, the current implementation allows for three discrete wavelets: Haar (`"haar"`), Daubechies Extremal Phase linear filters of length 4 (`"d4"`) and Least Asymmetric linear filters of length 8 (`"la8"`). Smoothing is controlled through the argument spa.

Value

A matrix of dimensions |b| x 2, where |b| is the number of packets in the basis. The first column contains the scale indices of each packet in the basis, the second column contains the packet index within each scale.

Author(s)

Alessandro Cardinali

References

A. Cardinali and G.P. Nason (2017). Locally Stationary Wavelet Packet Processes: Basis Selection and Model Fitting. Journal of Time Series Analysis, 38:2, 151-174.

See Also

LSWPspec, LSWPsim

Examples

```
wpb <- LSWPbasis(x = sp500, wavelet = 'la8', lev.max = 4, smooth = TRUE, spa = 0.35)
```

LSWPlib *LSWPlib Package*

Description

Library of functions for the statistical analysis and simulation of LSWP processes.

Author(s)

Alessandro Cardinali <alessandro.cardinali@plymouth.ac.uk>

LSWPsim *Simulation of LSWP processes*

Description

LSWPsim returns simulated time series from a specified LSWP specification.

Usage

```
LSWPsim(bb, spec, lev, wavelet, N)
```

Arguments

bb	a wavelet packet basis for the simulated process.
spec	a (locally stationary) spectra corresponding to the wavelet packet basis.
lev	the maximum level of the basis that is considered for simulation. Usually this is set as the maximum level in bb.
wavelet	the Daubechies wavelet used to build wavelet packets to simulate the process. See also Details.
N	the number of realizations to be simulated.

Details

This function produces one or multiple realizations of an LSWP process that is specified in terms of a wavelet packet basis and its corresponding spectra.

The function simulates realizations accordingly to the specified arguments. The wavelet argument is specified as in other functions of this package. Therefore, the current implementation allows for three discrete wavelets: Haar (`"haar"`), Daubechies Extremal Phase linear filters of length 4 (`"d4"`) and Least Asymmetric linear filters of length 8 (`"la8"`).

Value

If `N = 1` the function returns a vector containing the simulated time series. If `N > 1` the function returns a matrix with `N` columns each containing a different simulated series.

Author(s)

Alessandro Cardinali

References

A. Cardinali and G.P. Nason (2017). Locally Stationary Wavelet Packet Processes: Basis Selection and Model Fitting. Journal of Time Series Analysis, 38:2, 151-174.

See Also

`LSWPspec`, `best.basis`, `get.flat.basis`, `get.wavelet.basis`.

Examples

```
wwb <- get.flat.basis(scale = 4)
wwp <- matrix(2^{-(1:4)}, nrow = 4, ncol = 512, byrow = FALSE)
xt <- LSWPsim(bb = wwb, spec = wwp, lev = 4, wavelet = 'la8', N = 10)
```

`LSWPspec` *Locally Stationary Wavelet Packet Spectral Estimation*

Description

`LSWPspec` returns the spectral estimate of a locally stationary time series characterized by a wavelet packet basis.

Usage

```
LSWPspec(x, lev, bb, wavelet, smooth, spa, correct = TRUE, AA = NULL)
```

Arguments

`x`	a real valued numeric vector containing a time series of dyadic length.
`lev`	the maximum level for which the spectra should be estimated.
`bb`	a wavelet packet basis for which the spectra is estimated.
`wavelet`	wavelet used to estimate the wavelet packet spectra. Possible values are `"haar"`, `"d4"` and `"la8"`. See also Details.
`smooth`	logical. If `FALSE` the returned spectral estimate is not smoothed. Default value is `FALSE`. See also Details.
`spa`	window length for spectral smoothing. Increasing values increase the smoothing.
`correct`	logical. Should the returned spectral estimate be unbiased? Default is `TRUE`.
`AA`	this argument is for internal use only and should be left alone. See also Details.

Details

The current implementation allow the use of these three well known Daubechies discrete wavelets for spectral estimation. Default choice is the `"la8"` wavelet which has decent control over frequency leakage characterizing compactly supported filters. In this initial implementation smoothing is provided by local polynomials through the `lowess` function and the smoothing parameter `spa` is passed to `lowess`. Future package versions will allow for different smoothing methods. The argument `AA` is tipically used by other functions to provide the inner product matrix when running simulations. For a direct usage on a single time series the matrix is calculated internally usig the default settings.

Value

A matrix containing the time-frequency spectral estimate where each column corresponds to a different time point and ech row corresponds to a different packet from the given basis.

Author(s)

Alessandro Cardinali

References

A. Cardinali and G.P. Nason (2017). Locally Stationary Wavelet Packet Processes: Basis Selection and Model Fitting. Journal of Time Series Analysis, 38:2, 151-174.

See Also

`get.wavelet.basis`, `LSWPsim`, `best.basis`.

Examples

```
 wb <- get.wavelet.basis(4)
wpp <- LSWPspec(x = sp500, lev = 4, bb = wb, wavelet = 'la8', smooth = TRUE, spa = 0.35)
```

`marg.spec` *Wavelet Packet Marginal Spectra*

Description

`marg.spec` returns the time-average spectra of LSWP processes for each packet.

Usage

```
marg.spec(bas, spec, plot = TRUE)
```

Arguments

`bas`	is a (|b| x 2) matrix containing indices of a wavelet packet basis, where |b| is the number of packets in the basis.
`spec`	is a (|b| x T) matrix containing, in each row, the values of the time-varying spectra for each packet.
`plot`	should a plot of the marginal spectra vs frequency intervals be returned?

Details

This function computes the frequency intervals corresponding to each packet, along with the (time) average spectra for each packet.

This function is used to compute, and eventually plot, the time averaged spectra (or spectral estimate) vs packet frequencies. The arguments bas and spec shuld be provided as matrices.

Value

A (|b| x 2) matrix. In the first column the lower frequency relative to each packet is displayed. The second column contains the (time) average spectra.

Author(s)

Alessandro Cardinali

References

A. Cardinali and G.P. Nason (2017). Locally Stationary Wavelet Packet Processes: Basis Selection and Model Fitting. Journal of Time Series Analysis, 38:2, 151-174.

See Also

`best.basis`, `get.wavelet.basis`

Examples

```
wwb <- get.wavelet.basis(scale = 4)
wwp <- matrix(2^{-(1:4)}, nrow = 4, ncol = 512, byrow = FALSE)
msp <- marg.spec(bas = wwb, spec = wwp, plot = TRUE)
```

`plot.LSWPspec` *Plot for Objects of Class LSWPspec*

Description

`plot.LSWPspec` returns the plot for objects of class "`LSWPspec`", typically (but not exclusively) a wavelet packet spectral estimate.

Usage

```
## S3 method for class 'LSWPspec'
plot(x, y, ...)
```

Arguments

`x`	an object of class lswpspec.
`y`	not used, is set to `NULL`.
`...`	not currently used.

Details

This function implements the `plot` method for objects of class "`LSWPspec`". It is mainly used to plot spectral estimates returned by `LSWPspec`. The label of the vertical axis uses the wavelet packet basis index notation `p = 1,2,...,|b|`, where |b| is the number of packets in a wavelet packet basis as defined in Cardinali and Nason (2017). The label of the horizontal axis is the time index.

Value

A plot of the time-varying spectral estimates.

Author(s)

Alessandro Cardinali

References

A. Cardinali and G.P. Nason (2017). Locally Stationary Wavelet Packet Processes: Basis Selection and Model Fitting. Journal of Time Series Analysis, 38:2, 151-174.

See Also

`LSWPspec`, `LSWPbasis`

Examples

```
 wb <- get.wavelet.basis(4)
wpp <- LSWPspec(x = sp500, lev = 4, bb = wb, wavelet = 'la8', smooth = TRUE, spa = 0.35)
       plot(wpp)
```

`sp500` *Daily log-returns for the S&P 500 stock index.*

Description

Log-returns from daily quotes of the Standard & Poors 500 Index.

Usage

```
sp500
```

Format

A vector containing 1024 daily log-returns.

Period Covered The data cover the period April 1996 - December 1999.

`summary.LSWPbasis` *Summary method for Objects of Class LSWPbasis*

Description

`summary.LSWPbasis` returns a table containing the wavelet packet basis and its wavelet packet bases index notation.

Usage

```
## S3 method for class 'LSWPbasis'
summary(object, ...)
```

Arguments

`object`	an object of class "`LSWPbasis`", typically (but not exclusively) returned by the function `LSWPbasis`.
`...`	not currently used.

Details

This function is used to print a wavelet packet basis with the wavelet packet basis index notation p = 1,2,...,|b|, where |b| is the number of packets in a wavelet packet basis as defined in Cardinali and Nason (2017). The doublets "j_p, i_p" refer, to the scale and packet index within each scale, respectively.

Value

Print an object of class LSWPbasis with LSWP basis notation.

Author(s)

Alessandro Cardinali

References

A. Cardinali and G.P. Nason (2017). Locally Stationary Wavelet Packet Processes: Basis Selection and Model Fitting. Journal of Time Series Analysis, 38:2, 151-174.

See Also

LSWPbasis, best.basis

Examples

```
wpb <- LSWPbasis(x = sp500, wavelet = 'la8', lev.max = 4, smooth = TRUE, spa = 0.35)
summary(wpb)
```

Index

∗ **datasets**
    sp500, 12

autoconv, 2

best.basis, 3, *4, 5*, *8–10*, *13*

convolve, *2*

fft, *2*

get.flat.basis, 4, *5*, *8*
get.wavelet.basis, *4*, 5, *8–10*

LSWPbasis, *3*, 6, *11*, *13*
LSWPlib, 7
LSWPsim, *7*, 7, *9*
LSWPspec, *3*, *7, 8*, 8, *11*

marg.spec, 10

plot.LSWPspec, 11

sp500, 12
summary.LSWPbasis, 12