Package ‘rhosa’

August 10, 2020

Title Higher-Order Spectral Analysis
Date 2020-08-04
Version 0.1.0

Description Higher-order spectra or polyspectra of time series, such as bispectrum and bicoherence, have been investigated in abundant literature and applied to problems of signal detection in a wide range of fields. This package aims to provide a simple API to estimate and analyze them. The current implementation is based on Brillinger and Irizarry (1998) <doi:10.1016/S0165-1684(97)00217-X> for estimating bispectrum, with Lii and Helland (1981) <doi:10.1145/355958.355961> for cross-bispectrum.

License GPL-3
Encoding UTF-8

URL https://github.com/tabe/rhosa

BugReports https://github.com/tabe/rhosa/issues

RoxygenNote 7.1.1

Suggests ggplot2, knitr, rgl, rmarkdown, testthat (>= 2.1.0)

VignetteBuilder knitr

NeedsCompilation no

Author Takeshi Abe [aut, cre] (<https://orcid.org/0000-0002-7074-4561>)

Maintainer Takeshi Abe <tabe@fixedpoint.jp>

Repository CRAN

Date/Publication 2020-08-10 13:00:02 UTC

R topics documented:

bicoherence ................................................................................. 2
bispectrum .................................................................................. 3
cross_bicoherence ........................................................................ 4
cross_bispectrum .......................................................................... 5

Index 7
bicoherence

**Estimate bicoherence from given time series data.**

**Description**

Estimate magnitude-squared bicoherence from given real- or complex-valued time series data.

**Usage**

```r
bicoherence(data, window_function = NULL, alpha = 0.05, p_adjust_method = "BH")
```

**Arguments**

data
- **data**
  - Given time series, as a data frame or matrix with which columns correspond to sampled stretches.

window_function
- **window_function**
  - A window function's name for tapering. Defaults to NULL ("no tapering"). Currently the following window functions are available: Hamming window ("hamming"), Hann window ("hann"), and Blackman window ("blackman").

alpha
- **alpha**
  - The alpha level of the hypothesis test. Defaults to 0.05.

p_adjust_method
- **p_adjust_method**
  - The correction method for p-values, given to `p.adjust()`. Defaults to "BH" (Benjamini and Hochberg). No correction if a non-character is given.

**Value**

A data frame including the following columns:

- **f1**: The first elements of frequency pairs.
- **f2**: The second elements of frequency pairs.
- **value**: The estimate of magnitude-squared bicoherence at the respective frequency pair.
- **p_value**: The (corrected, if requested) p-value for hypothesis testing under null hypothesis that bicoherence is 0.
- **significance**: TRUE if the null hypothesis of the above hypothesis test is rejected with given alpha level.

**References**

Examples

```r
f <- function(x) {
    sin(2 * x) + sin(3 * x + 1) + sin(2 * x) * sin(3 * x + 1)
}
v <- sapply(seq_len(1280), f) + rnorm(1280)
m <- matrix(v, nrow = 128)
bicoherence(m)
bicoherence(m, "hamming")
```

bispectrum

```
| bispectrum  | Estimate bispectrum from time series data. |
```

Description

Estimate bispectrum from real- or complex-valued time series data.

Usage

```r
bispectrum(data, window_function = NULL)
```

Arguments

data Given time series, as a data frame or matrix with which columns correspond to sampled stretches.

window_function A window function’s name for tapering. Defaults to NULL ("no tapering").

Currently the following window functions are available: Hamming window ("hamming"), Hann window ("hann"), and Blackman window ("blackman").

Value

A data frame including the following columns:

- **f1**: The first elements of frequency pairs.
- **f2**: The second elements of frequency pairs.
- **value**: The estimated bispectrum at each frequency pair.

References

Examples

```r
def <- function(x) {
  sin(2 * x) + sin(3 * x + 1) + sin(2 * x) * sin(3 * x + 1)
}
v <- sapply(seq_len(1280), f) + rnorm(1280)
m <- matrix(v, nrow = 128)
bispectrum(m)
bispectrum(m, "hamming")
```

cross_bicoherence  

Estimate cross-coherence from time series data.

Description

Estimate cross-coherence from three real-valued time series data.

Usage

```r
cross_bicoherence(x, y, z = y, dft_given = FALSE)
```

Arguments

- `x`: Given 1st time series, as a data frame or matrix with which columns correspond to sampled stretches.
- `y`: Given 2nd time series, with the same dimension as `x`.
- `z`: Optional 3rd time series, with the same dimension as `x` (and thus as `y`). If omitted, `y` is used instead.
- `dft_given`: If `TRUE`, suppose that DFTs is given instead of time series data and skip the fast fourier transform. Default: `FALSE`.

Value

A data frame including the following columns:

- `f1`: The first elements of frequency pairs.
- `f2`: The second elements of frequency pairs.
- `value`: The estimated value of magnitude-squared cross-bicoherence at the respective frequency pair.

References

**Examples**

```r
x <- seq_len(1280)
v1 <- sapply(x, function(x) {sin(2 * x)}) + rnorm(1280)
v2 <- sapply(x, function(x) {sin(3 * x + 1)}) + rnorm(1280)
v3 <- sapply(x, function(x) {cos(2 * x) * cos(3 * x + 1)}) + rnorm(1280)
m1 <- matrix(v1, nrow = 128)
m2 <- matrix(v2, nrow = 128)
m3 <- matrix(v3, nrow = 128)
cross_bicoherence(m1, m2, m3)

d1 <- stats::mvfft(m1)
d2 <- stats::mvfft(m2)
d3 <- stats::mvfft(m3)
cross_bicoherence(d1, d2, d3, dft_given = TRUE)
```

---

**cross_bispectrum**

Estimate cross-bispectrum from time series data.

**Description**

Estimate cross-bispectrum from three real-valued time series data.

**Usage**

```r
cross_bispectrum(x, y, z = y, dft_given = FALSE)
```

**Arguments**

- `x`: Given 1st time series, as a data frame or matrix with which columns correspond to sampled stretches.
- `y`: Given 2nd time series, with the same dimension as `x`.
- `z`: Optional 3rd time series, with the same dimension as `x` (and thus as `y`). If omitted, `y` is used instead.
- `dft_given`: If `TRUE`, suppose that DFTs is given instead of time series data and skip the fast fourier transform. Default: `FALSE`.

**Value**

A data frame including the following columns:

- `f1`: The first elements of frequency pairs.
- `f2`: The second elements of frequency pairs.
- `value`: The estimated cross-bispectrum at each frequency pair.
References


Examples

```r
x <- seq_len(1280)
v1 <- sapply(x, function(x) {sin(2 * x)}) + rnorm(1280)
v2 <- sapply(x, function(x) {sin(3 * x + 1)}) + rnorm(1280)
v3 <- sapply(x, function(x) {cos(2 * x) * cos(3 * x + 1)}) + rnorm(1280)
m1 <- matrix(v1, nrow = 128)
m2 <- matrix(v2, nrow = 128)
m3 <- matrix(v3, nrow = 128)
cross_bispectrum(m1, m2, m3)

d1 <- stats::mvfft(m1)
d2 <- stats::mvfft(m2)
d3 <- stats::mvfft(m3)
cross_bispectrum(d1, d2, d3, dft_given = TRUE)
```
Index

bicoherence, 2
bispectrum, 3

cross_bicoherence, 4
cross_bispectrum, 5

p.adjust, 2