Package: topcc (via r-universe)

October 22, 2026

Type Package

Title Topological Correlation Coefficient

Version 1.2

Description Topological correlation coefficient is used to identify dependencies between Time-Dependent Objects and is applicable to objects such as time series, chaotic systems, and dynamic networks.

License GPL-3

Encoding UTF-8

LazyData true

Depends R (>= 2.10)

Imports parallel

RoxygenNote 7.3.2

NeedsCompilation no

Author Chun-Xiao Nie [aut, cre] (ORCID: <https://orcid.org/0000-0002-7790-0803>)

Maintainer Chun-Xiao Nie <niechunxiao2009@163.com>

Repository https://niechunxiao.r-universe.dev

Date/Publication 2026-05-08 15:56:06 UTC

RemoteUrl https://github.com/cran/topcc

RemoteRef HEAD

RemoteSha 46ab7a81d3f2e90c470086cad6dbcc30f4bc92ed

Contents

autotopcor 2
autotopcortest 3
indextest 4
local_shuffle 5
paralleltopcortest 5

sliding_window_tcc_test 6
tcc_local_shuffle 7
tcc_local_shuffle_dist_parallel 8
tdoshuffling 9
topcor 10
topcortest 10

Index **12**

`autotopcor` *Auto Topological Correlation Coefficient (ATCC)*

Description

Computes the auto topological correlation coefficient for a time series or a time-dependent object (TDO) at given lags. For each lag, the original input is split into two overlapping segments shifted by the lag, and the topological correlation coefficient (TCC) between the two segments is returned.

Usage

```
autotopcor(x, lags = 1, k1, type = c("series", "tdo"))
```

Arguments

`x`	A numeric vector (when ‘type = "series"‘) or a square distance matrix (when ‘type = "tdo"‘). If ‘type = "series"‘, the Euclidean distance matrix is computed internally.
`lags`	Integer vector of lag orders (default: ‘1‘). All values must be strictly less than the length (or dimension) of ‘x‘.
`k1`	Numeric scalar giving the step size for the k-nearest neighbour filtration. Internally, for a given lag and original size ‘n‘, the vector ‘k‘ passed to [‘topcor()‘] is constructed as ‘c(1, n - lag - 1, k1)‘.
`type`	Character string, either ‘"series"‘ (default) or ‘"tdo"‘.

Value

A numeric vector of ATCC values, one for each element of ‘lags‘.

References

Nie, Chun-Xiao. "Unveiling complex nonlinear dynamics in stock markets through topological data analysis." Physica A: Statistical Mechanics and its Applications 680 (2025): 131025. Nie, Chun-Xiao. "Persistence of return distribution sequence in financial markets." Communications in Nonlinear Science and Numerical Simulation 131 (2024): 107856.

See Also

[‘topcor()‘] for the underlying topological correlation computation.

Examples

```
set.seed(123)
x <- rnorm(50)
atcc <- autotopcor(x, lags = 1:3, k1 = 2)

# Using a pre-computed distance matrix (TDO)
D <- as.matrix(dist(x))
atcc_tdo <- autotopcor(D, lags = 1:3, k1 = 2, type = "tdo")
```

`autotopcortest` *Significance Test for Auto Topological Correlation Coefficient*

Description

Tests the significance of ATCC values at given lags using random shuffling. For time-series input (‘type = "series"‘) the observations are permuted; for TDO input (‘type = "tdo"‘) the time labels of the distance matrix are permuted via [‘tdoshuffling()‘].

Usage

```
autotopcortest(x, lags = 1, k1, nrand = 300, type = c("series", "tdo"))
```

Arguments

`x`	A numeric vector (when ‘type = "series"‘) or a square distance matrix (when ‘type = "tdo"‘). If ‘type = "series"‘, the Euclidean distance matrix is computed internally.
`lags`	Integer vector of lag orders (default: ‘1‘).
`k1`	Numeric scalar giving the step size for the k-nearest neighbour filtration. See [‘autotopcor()‘] for details.
`nrand`	Integer, number of random shuffles used to build the null distribution (default: ‘300‘).
`type`	Character string, either ‘"series"‘ (default) or ‘"tdo"‘.

Value

A list with the following components:

`tcc`	Numeric vector of observed ATCC values.
`z`	Z-scores of the observed ATCC values relative to the null distribution.
`tcc_rand`	Numeric matrix of null ATCC values (rows = shuffles, columns = lags).
`p_value`	Numeric matrix with two rows: ‘"normal"‘ (p-value based on normal approximation) and ‘"empirical"‘ (fraction of null values exceeding the observed value). Each column corresponds to a lag.

References

Nie, Chun-Xiao. "Unveiling complex nonlinear dynamics in stock markets through topological data analysis." Physica A: Statistical Mechanics and its Applications 680 (2025): 131025. Nie, Chun-Xiao. "Persistence of return distribution sequence in financial markets." Communications in Nonlinear Science and Numerical Simulation 131 (2024): 107856.

See Also

['autotopcor()'], ['topcor()'], ['tdoshuffling()'].

Examples

```
set.seed(123)
x <- rnorm(50)
test_res <- autotopcortest(x, lags = 1, k1 = 2, nrand = 20,type="series")
```

`indextest` *indextest*

Description

This dataset contains example data for demonstrating package functions.

Usage

```
indextest
```

Format

A data frame with 521 rows and 3 variables:

Time Time

Price Numeric,index price

Volume Numeric,index price

Source

Simulated data for package examples.

local_shuffle *Generate local random permutation surrogates for a time series*

Description

This function randomly shuffles the observations within a specified window of a time series, leaving the rest unchanged. This is useful for creating surrogates that preserve local structure outside the window while destroying temporal order within it.

Usage

```
local_shuffle(s, w)
```

Arguments

s	A numeric vector representing the time series.
w	A numeric vector of length 2 giving the start and end indices of the window to be shuffled. The indices are inclusive, i.e., s[w[1]:w[2]] will be permuted.

Value

A numeric vector of the same length as s, with the values in the specified window randomly permuted.

Examples

```
s <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
# Shuffle indices 3 to 7
s_shuffled <- local_shuffle(s, c(3, 7))
print(s_shuffled)
```

paralleltopcortest *Significance test for TCC using shuffling (parallel version, with optional seed)*

Description

Significance test for TCC using shuffling (parallel version, with optional seed)

Usage

```
paralleltopcortest(d1, d2, k, nrand, ncores = NULL, seed = NULL)
```

Arguments

d1	Distance matrix of the first TDO.
d2	Distance matrix of the second TDO.
k	Vector of parameters for generating kNN networks.
nrand	Number of random shufflings.
ncores	Number of CPU cores to use. If NULL, uses all but one core.
seed	Optional random seed for reproducibility. If NULL (default), no seed is set.

Value

A list containing observed TCC, Z-score, random TCC values, and p-values.

Examples

```
x=rnorm(40,0,1)
y=x+rnorm(40,0,1)
result <- paralleltopcortest(as.matrix(dist(x)), as.matrix(dist(y)), c(1,39,1),20, ncores = 2)
# Run with seed for reproducibility
result_repro <- paralleltopcortest(as.matrix(dist(x)), as.matrix(dist(y)),
c(1,39,6), 20,ncores = 2, seed = 123)
```

---

sliding_window_tcc_test

Sliding window TCC significance test using local random permutation surrogates

---

Description

This function slides a window of fixed length over two time series, and for each window position it generates a distribution of TCC values under local random permutations (shuffling values within that window). The observed TCC (based on the original full series) is then compared to each window-specific distribution to compute a z-score.

Usage

```
sliding_window_tcc_test(s1, s2, l1, l2, k, nrand, ncores = NULL, seed = NULL)
```

Arguments

s1	Numeric vector, first time series.
s2	Numeric vector, second time series.
l1	Window length (number of consecutive observations to shuffle).
l2	Step size (shift) between consecutive windows.
k	Vector of parameters for TCC calculation (see tdotcc).

nrand	Number of random shufflings per window.
ncores	Number of CPU cores to use for parallel computation within each window. Passed to tcc_local_shuffle_dist_parallel.
seed	Optional base random seed for reproducibility. If provided, each window will use a distinct seed = seed + window_index. If NULL, no seed is set (results will vary between runs).

Value

A list with the following components:

windows	A matrix with two columns: start and end indices of each window.
tcc_obs	The observed TCC value for the full unshuffled series.
tcc_dist	A list of length equal to number of windows, each element being a numeric vector of length nrand containing TCC values from local shuffling for that window.
z_scores	A numeric vector of z-scores for each window, computed as (tcc_obs - mean(tcc_dist[[i]])) / sd(tcc_dist[[i]]).

Examples

```
s1 <- rnorm(40)
s2 <- rnorm(40)
result <- sliding_window_tcc_test(s1, s2, l1 = 10,
l2 = 5,k = c(1, 39, 1), nrand = 20, ncores = 2, seed = 123)
```

---

tcc_local_shuffle *Calculate TCC after local random permutation of two time series within a window*

Description

This function applies local shuffling to two time series within a specified window, computes Euclidean distance matrices for the shuffled series, and calculates the TCC between them using the tdotcc function.

Usage

```
tcc_local_shuffle(s1, s2, w, k)
```

Arguments

s1	Numeric vector, first time series.
s2	Numeric vector, second time series.
w	Numeric vector of length 2 giving the start and end indices of the window to shuffle.
k	Vector of parameters for TCC calculation (see tdotcc).

Value

The TCC value computed from the locally shuffled series.

Examples

```
s1 <- rnorm(100)
s2 <- rnorm(100)
tcc_val <- tcc_local_shuffle(s1, s2, w = c(30, 70), k = c(1, 99, 2))
```

---

tcc_local_shuffle_dist_parallel

Generate distribution of TCC values by repeatedly applying local shuffling (parallel)

Description

This function repeatedly applies local random permutation to two time series within a specified window, computes the TCC for each shuffled pair using `tcc_local_shuffle`, and returns the distribution of TCC values over many repetitions. Parallel computing is employed to accelerate the process.

Usage

```
tcc_local_shuffle_dist_parallel(
  s1,
  s2,
  w,
  k,
  nrand,
  ncores = NULL,
  seed = NULL
)
```

Arguments

`s1`	Numeric vector, first time series.
`s2`	Numeric vector, second time series.
`w`	Numeric vector of length 2 giving the start and end indices of the window to shuffle.
`k`	Vector of parameters for TCC calculation (see `tdotcc`).
`nrand`	Number of random shufflings to perform.
`ncores`	Number of CPU cores to use. If `NULL`, uses all but one core.
`seed`	Optional random seed for reproducibility. If `NULL`, no seed is set.

Value

A numeric vector of length nrand containing the TCC values from each shuffling.

Examples

```
s1 <- rnorm(100)
s2 <- rnorm(100)
tcc_dist <- tcc_local_shuffle_dist_parallel(s1, s2,
w = c(30, 70), k = c(1, 99, 2), nrand = 20, ncores = 2)
```

---

tdoshuffling *Generate surrogates by shuffling (random permutation) of rows and columns.*

Description

Generate surrogates by shuffling (random permutation) of rows and columns.

Usage

```
tdoshuffling(d)
```

Arguments

d	A distance matrix (square, symmetric).

Value

The shuffled distance matrix (same dimensions, permuted rows and columns).

Examples

```
data("indextest")
r=diff(log(as.matrix(indextest[,c(2,3)])))
d_shuf <- tdoshuffling(as.matrix(dist(r[,1])))
```

topcor — *Calculate Topological correlation Coefficient (TCC) between two TDOs*

Description

Calculate Topological correlation Coefficient (TCC) between two TDOs

Usage

```
topcor(d1, d2, k)
```

Arguments

d1	Distance matrix of the first TDO.
d2	Distance matrix of the second TDO.
k	A vector of three values: start, end, and step for the k parameter. e.g., k = c(1, 999, 6) generates k = 1, 7, 13, ...

Value

TCC value (mean Jaccard similarity over the k range).

References

Nie, Chun-Xiao. "Nonlinear correlation analysis of time series based on complex network similarity." International Journal of Bifurcation and Chaos 30.15 (2020): 2050225. Nie, Chun-Xiao. "Topological similarity of time-dependent objects." Nonlinear Dynamics 111.1 (2023): 481-492.

Examples

```
x=rnorm(100,0,1)
y=x+rnorm(100,0,1)
tcc <- topcor(as.matrix(dist(x)), as.matrix(dist(y)), c(1,99,3))
```

topcortest — *Significance test for TCC using shuffling (serial version)*

Description

Significance test for TCC using shuffling (serial version)

Usage

```
topcortest(d1, d2, k, nrand)
```

Arguments

d1	Distance matrix of the first TDO.
d2	Distance matrix of the second TDO.
k	Vector of parameters for generating kNN networks (see tdotcc).
nrand	Number of random shufflings to generate null distribution.

Value

A list containing:

tcc	Observed TCC value.
z	Z-score of observed TCC relative to null distribution.
tcc_rand	Vector of TCC values from random shufflings.
p_value	Two p-values: parametric (based on normal approximation) and empirical.

Examples

```
x=rnorm(40,0,1)
y=rnorm(40,0,1)
tcctest <- topcortest(as.matrix(dist(x)), as.matrix(dist(y)), c(1,39,2),20)
```

Index

∗ **datasets**
indextest, 4

autotopcor, 2
autotopcortest, 3

indextest, 4

local_shuffle, 5

paralleltopcortest, 5

sliding_window_tcc_test, 6

tcc_local_shuffle, 7
tcc_local_shuffle_dist_parallel, 8
tdoshuffling, 9
topcor, 10
topcortest, 10