Package ‘rosario’

May 9, 2026

Title A Null Model Algorithm to Analyze Cyclical Data in Ecology

Version 0.1.1

Description Implements a null model analysis to quantify concurrent temporal niche overlap (i.e., activity or phenology) among biological identities (e.g., individuals, populations, species) using the Rosario randomization algorithm (Castro-Arellano et al. 2010) <doi:10.1111/j.2041-210X.2010.00031.x>.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.3

Imports broom, magrittr, purrr, furrr, future, rlang,

URL `https://alrobles.github.io/rosario/`

Suggests rmarkdown, ggplot2, spelling, knitr, lubridate, dplyr

VignetteBuilder knitr

Depends R (>= 4.1.0)

LazyData true

Language en-US

Config/Needs/website rmarkdown

NeedsCompilation no

Author Ángel L. Robles-Fernández [aut],
Maria A. Hurtado-Materon [aut],
Tatiana Velásquez-Roa [cre, aut],
Iván Castro-Arellano [ths]

Maintainer Tatiana Velásquez-Roa <`yxh11@txstate.edu`>

Repository CRAN

Date/Publication 2026-02-21 22:40:02 UTC

Contents

`czekanowski_index` *Czekanowski overlap index*

Description

Computes the Czekanowski index of overlap between two relative-frequency activity profiles p and q:

$$1 - \frac{1}{2}\sum_i |p_i - q_i|.$$

Usage

```
czekanowski_index(p, q)
```

Arguments

`p`	Numeric vector of non-negative relative frequencies (typically sums to 1) describing the first biological identity data (e.g. activity, population size, etc) across ordered time bins. Standardized use of time intervals through the manual.
`q`	Numeric vector of non-negative relative frequencies (same length as `p`) for the second biological identity across ordered time bins.

Details

Inputs should be on a *comparable scale*; if your data are raw counts, rescale rows to proportions first (see `rescale_matrix()`).

Value

A single numeric value in [0, 1] where 0 indicates no overlap and 1 indicates identical profiles.

Examples

```
set.seed(1)
n <- 6
p <- rmultinom(1, 20, rep(1, n))[,1]; p <- p / sum(p)
q <- rmultinom(1, 20, rep(1, n))[,1]; q <- q / sum(q)
czekanowski_index(p, q)
```

`ex1` *Example temporal activity dataset*

Description

An example dataset of 5 biological identities across 12 time intervals. Values represent counts of activity events (e.g., detections or captures) per interval. This dataset is provided for examples and vignettes.

Usage

```
ex1
```

Format

A numeric matrix with 5 rows (biological identities) and 12 columns (time intervals):

Rows Biological identities (`bioID1` … `bioID5`)

Columns Time intervals (`INT1` … `INT12`)

Values Counts of activity per identity × interval

Examples

```
ex1
rowSums(ex1)   # total activity per biological identity
colSums(ex1)   # total activity per time interval
```

`get_null_model` *Null-model test via ROSARIO algorithm randomization*

Description

Generates a null distribution of concurrent temporal niche overlap by repeatedly randomizing the input matrix with `rosario_sample()` and recomputing the mean pairwise overlap (see `temp_overlap()`).

Usage

```
get_null_model(mat, method, nsim = 100, parallel = FALSE)
```

Arguments

`mat`	Numeric matrix (rows = biological identities, columns = ordered time intervals). Time intervals are assumed to be circular (e.g., hours of the day, months of the year), so the last interval is treated as adjacent to the first. Biological identities can be individuals, populations, species or communities.
`method`	Character string naming the overlap index to use: `"pianka"` or `"czekanowski"`.
`nsim`	Integer number of randomizations to run (default `100`).
`parallel`	Logical. If `TRUE`, randomizations are executed in parallel using `furrr::future_map_dfr()` with a `multisession` plan set internally.

Details

For `"czekanowski"`, rows are rescaled to proportions internally to satisfy the index's assumptions. Randomization preserves each row's temporal autocorrelation by cyclically shifting (and optionally mirroring) its profile; see `rosario_sample()`.

When `parallel = TRUE`, the function calls `future::plan(multisession)` internally so that randomizations are distributed across available local sessions. This means the function overrides any previously set `future` plan. If you need custom control over parallelization (e.g. cluster backends), run the non-parallel mode (`parallel = FALSE`) and handle parallelism externally.

Value

A list with components:

observed_niche_overlap Mean from all possible pairwise comparisons among biological identities for `mat`.

p_value A one-sample `t.test` object comparing null means to the observed value (mu = observed).

null_niche_overlap A tibble/data.frame of simulated mean overlaps (one per randomization).

See Also

`temp_overlap()`, `rosario_sample()`, `temp_overlap_matrix()`

Examples

```
get_null_model(ex1, method = "pianka", nsim = 10, parallel = FALSE)
```

`pianka_index` *Pianka's niche-overlap index*

Description

Computes Pianka's symmetric index of overlap between two non-negative activity profiles p and q:

$$\frac{\sum_i p_i q_i}{\sqrt{\left(\sum_i p_i^2\right)\left(\sum_i q_i^2\right)}}.$$

Usage

```
pianka_index(p, q)
```

Arguments

`p`	Numeric vector of non-negative values (counts or relative frequencies) for the first biological identity data 1 (e.g. activity, population size, etc) across ordered time bins.
`q`	Numeric vector of non-negative values (same length as `p`) for the second biological identity (e.g. activity, population size, etc) across ordered time bins.

Details

Pianka's index does not require inputs to sum to 1, but both vectors must be non-negative and not all zero.

Value

A single numeric value in [0, 1]; larger values indicate greater overlap.

Examples

```
set.seed(1)
n <- 10
p <- rpois(n, 3); q <- rpois(n, 3)
pianka_index(p, q)
```

`plot_rosario` *Diagram of ROSARIO null-model randomizations*

Description

Visualizes the first 10 hypothetical time use distributions produced by `rosario()` for a single biological identity. Each panel displays one hypothetical time use distribution with its cyclic shift shown in dark gray and its mirror image shown in dark red.

Usage

```
plot_rosario(numvec, normalize = TRUE, cols = 4)
```

Arguments

`numvec`	Numeric vector representing a single biological identity' distribution across ordered time intervals.
`normalize`	Logical; if TRUE (default) scale each half to sum to 1 (compare shapes, not totals).
`cols`	Integer; number of panels (hypothetical distributions) per row.

Value

Invisibly, a list with:

- `variants` — the original list from `rosario(numvec)`
- `mat2k_plotted` — matrix of the plotted variants (min(10, n) × 2k)
- `k` — number of time bins
- `indices_plotted` — which variant indices (1..m) were drawn

Examples

```
one <- c(0,5,0,7,5,13,70,0)
plot_rosario(one, cols = 4)
```

`rescale_matrix` *Row-wise rescaling of a matrix to relative frequencies*

Description

Divides each row by its row sum so that every row sums to 1 (leaving dimnames intact).

Usage

```
rescale_matrix(m)
```

Arguments

`m`	Numeric matrix; rows are biological identities, columns are time bins (i.e., time resources).

Value

A numeric matrix of the same dimension with each row summing to 1. Rows with a zero sum are left unchanged (resulting in `NaN` if present).

Examples

```
ex1_rescale <- rescale_matrix(ex1)
rowSums(ex1_rescale)
```

`rosario` *Generate cyclic and mirrored permutations of a time series*

Description

For a numeric vector, creates the set of cyclic shifts and their mirror images (reverse order), preserving shape but changing location along the cycle. The suite of vectors and mirrors represent a complete set of possible distributions.

Usage

```
rosario(numvec)
```

Arguments

`numvec`	Numeric vector representing a single biological identity' distributions across ordered time intervals. Time intervals are assumed to be circular (e.g., hours of the day, months of the year), so the last interval is treated as adjacent to the first. Biological identities can be individuals, populations, species or communities.

Value

A list of numeric vectors with all the permutations in the time series, including the mirror patterns.

See Also

`vec_permutation()`, `rosario_sample()`

Examples

```
rosario(c(40, 25, 18, 10, 5, 2))
```

`rosario_sample` *ROSARIO randomization of an assemblage matrix*

Description

Randomly permutes each row by a uniform cyclic shift of its columns and, with probability 0.5, reverses the order (mirror image). This kind of permutations preserves each biological identity's temporal autocorrelation structure and niche breadth while randomizing location within the cycle.

Usage

```
rosario_sample(mat)
```

Arguments

`mat`	Numeric matrix with biological identities in rows and ordered time intervals in columns.

Value

A numeric matrix of the same dimension as `mat`, randomized row-wise.

See Also

`rosario()`, `vec_permutation()`

Examples

```
rosario_sample(ex1)
```

`temp_overlap` *Assemblage-wide temporal niche segregation*

Description

Assemblage-wide temporal niche segregation

Usage

```
temp_overlap(mat, method = c("pianka", "czekanowski"))
```

Arguments

`mat`	A matrix with species per time columns.
`method`	A measure of temporal niche overlap between pairs of species

Value

The mean temporal overlaping index from a pairwise distance matrix

Examples

```
temp_overlap(ex1, method = "pianka")
ex1_rescale <- rescale_matrix(ex1)
temp_overlap(ex1_rescale, method = "czekanowski")
```

`temp_overlap_df` *Convert a square overlap matrix to a tidy pairwise data frame*

Description

Tidies a symmetric overlap (or distance) matrix into a three-column tibble/data frame with pairs and values.

Usage

```
temp_overlap_df(mat)
```

Arguments

`mat`	Square numeric matrix (typically from `temp_overlap_matrix()`).

Value

A data frame with columns `item1`, `item2`, and `distance` (terminology follows `stats::as.dist()`).

Examples

```
d <- temp_overlap_matrix(ex1)
temp_overlap_df(d)
```

`temp_overlap_matrix` *Pairwise temporal niche-overlap matrix*

Description

Computes all pairwise overlaps among rows (biological identities) using the chosen index.

Usage

```
temp_overlap_matrix(mat, method = c("pianka", "czekanowski"))
```

Arguments

`mat`	Numeric matrix (rows = biological identities, columns = ordered time intervals).
`method`	Overlap index to use: `"pianka"` or `"czekanowski"`.

Details

For Czekanowski, supply a **row-rescaled** matrix (see `rescale_matrix()`) or use `temp_overlap()`, which handles rescaling.

Value

A square symmetric matrix of overlap values with row/colnames copied from `mat`. The first class of the object is set to the method name.

See Also

`temp_overlap()`, `rescale_matrix()`

Examples

```
temp_overlap_matrix(ex1, method = "pianka")
ex1_rescale <- rescale_matrix(ex1)
temp_overlap_matrix(ex1_rescale, method = "czekanowski")
```

`temp_overlap_plot` *Plot null-model results for temporal niche overlap*

Description

Creates a histogram of simulated mean niche overlap values from a null model (see `get_null_model()`) and overlays a dashed vertical line indicating the observed mean overlap.

Usage

```
temp_overlap_plot(results)
```

Arguments

`results`	A list object returned by `get_null_model()`, containing `null_niche_overlap` (data frame of simulated overlaps) and `observed_niche_overlap` (numeric observed value).

Value

A `ggplot2` object displaying the null distribution of overlap values with the observed overlap marked.

See Also

`get_null_model()`, `temp_overlap()`

Examples

```
mod <- get_null_model(ex1, method = "pianka", nsim = 100)
temp_overlap_plot(mod)
```

`vec_permutation` *Cyclic permutation (rotate) a numeric vector*

Description

Returns a cyclic shift of `numvec` so that position `x` becomes the first element and the order wraps around the end.

Usage

```
vec_permutation(numvec, x = 1)
```

Arguments

`numvec`	Numeric vector representing an ordered cycle.
`x`	Integer (1-based) index of the new starting position.

Value

A numeric vector of the same length as `numvec`, rotated so that `numvec[x]` is first.

Examples

```
vec_permutation(1:6, 4)  # 4 5 6 1 2 3
```

Index