Package: ringSeg (via r-universe)

June 25, 2026

Type Package

Title Asymptotic Distribution-Free Change-Point Detection via a New Ranking Scheme (RING)

Version 0.1.0

Description Rank-based, asymptotic distribution-free change-point detection for modern (high-dimensional, non-Euclidean) data, based on the graph-induced ranking scheme of Zhou and Chen (2025) <doi:10.1109/TIT.2025.3575858>. Given a rank matrix built from a pairwise similarity, the method scans for a single change-point or a changed interval using three statistics (weighted 'WR', max-type 'MR', and generalized 'TR') and returns analytic distribution-free p-value approximations (with an optional skewness correction) as well as optional permutation p-values.

License GPL (>= 2)

Encoding UTF-8

Imports stats

NeedsCompilation no

Author Doudou Zhou [aut], Hao Chen [aut, cre]

Maintainer Hao Chen <hxchen@ucdavis.edu>

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

Date/Publication 2026-06-25 11:30:02 UTC

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

RemoteRef HEAD

RemoteSha 7d5a7198f3ab66811291b454bb95d07723052d31

Contents

`ringSeg-package` *ringSeg: rank-based asymptotic distribution-free change-point detection*

Description

Implements the RING change-point detector (Zhou & Chen, 2025, IEEE TIT): a rank-based, asymptotic distribution-free scan for a single change-point or a changed interval in a sequence of (possibly high-dimensional / non-Euclidean) observations. The method operates on a rank matrix built from a pairwise similarity and returns the weighted (WR), max-type (MR), and generalized (TR) statistics with analytic distribution-free p-value approximations (optionally skewness-corrected) and optional permutation p-values.

Details

Main entry points: `ring_cpd` (data / distance / rank-matrix in), `rcpd` (rank matrix in), `ring_graph` (build the RING k-NN rank graph), `Rise_Rank` (low-level ranker).

Author(s)

Doudou Zhou, Hao Chen (maintainer)

References

Zhou, D. and Chen, H. (2025). Asymptotic Distribution-Free Change-Point Detection for Modern Data Based on a New Ranking Scheme. *IEEE Transactions on Information Theory*, **71**(8), 6183–6197. doi:10.1109/TIT.2025.3575858.

`rcpd` *RING rank-based change-point detection (rank matrix in)*

Description

Core RING detector operating directly on a rank matrix. The analytic p-value approximations are distribution-free and require no permutation; they assume the sparse RING k-NN graph, so build `R` with `ring_graph`, or call `ring_cpd` to build it for you.

Usage

```
rcpd(R, n0 = NULL, n1 = NULL, pval.appr = TRUE, skew.corr = TRUE,
     B = 0, alternative = c("single", "interval"))
```

Arguments

R	an $n \times n$ rank matrix with zero diagonal, as produced by ring_graph (the RING k-NN rank graph). Observations (nodes) are indexed in time order.
n0, n1	scan range; defaults ceiling(0.05*n) and floor(0.95*n), clamped to [2, n-2]. For alternative = "interval" these are the minimum (l0) and maximum (l1) interval lengths.
pval.appr	logical; compute analytic distribution-free p-value approximations for the WR, MR and TR statistics.
skew.corr	logical; additionally compute the skewness-corrected approximation (WR.cor, MR.cor). The skewness-corrected p-values are better calibrated at finite n; the uncorrected Gaussian/field approximation can be anti-conservative for small n.
B	number of permutations for permutation p-values (0 = none).
alternative	"single" for a single change-point, or "interval" for a changed interval.

Value

A list containing

scanZ	the scan statistics. For alternative = "single", each of TR (generalized, $Z_w^2 + Z_{\text{diff}}^2$), MR (max-type, $\max(\lvert Z_{\text{diff}}\rvert, Z_w)$) and WR (weighted, Z_w) holds the full scan curve, the estimated change-point tau, and the maximum.
pval.appr	(if pval.appr) analytic p-values TR, MR, WR, and (if skew.corr) the skewness-corrected WR.cor, MR.cor.
pval.perm	(if B > 0) permutation p-values for TR, MR, WR.

References

Zhou, D. and Chen, H. (2025). Asymptotic Distribution-Free Change-Point Detection for Modern Data Based on a New Ranking Scheme. *IEEE Transactions on Information Theory*, **71**(8), 6183–6197. doi:10.1109/TIT.2025.3575858.

See Also

ring_cpd, Rise_Rank

Examples

```
set.seed(1)
n <- 150; d <- 8
X <- matrix(rnorm(n * d), n, d)
X[76:n, ] <- X[76:n, ] + 0.8
R <- ring_graph(as.matrix(dist(X)), k = 13)
res <- rcpd(R, B = 0)
res$scanZ$TR$tau
res$pval.appr$TR
```

`ring_cpd` *RING change-point detection from data or a distance matrix*

Description

Convenience wrapper: builds the RING rank graph from the input (unless a rank matrix is supplied) and calls `rcpd`. Observations must be in time order.

Usage

```
ring_cpd(x, is.distance = FALSE, is.rank = FALSE, k = NULL,
         dist.method = "euclidean",
         n0 = NULL, n1 = NULL, pval.appr = TRUE, skew.corr = TRUE,
         B = 0, alternative = c("single", "interval"))
```

Arguments

`x`	One of: an $n \times d$ data matrix (rows = time-ordered observations; the default), an $n \times n$ distance matrix (`is.distance = TRUE`), or an $n \times n$ rank matrix (`is.rank = TRUE`).
`is.distance`	logical; if `TRUE`, `x` is a distance matrix.
`is.rank`	logical; if `TRUE`, `x` is already a rank matrix (e.g. from `ring_graph`) and is used as-is.
`k`	number of nearest neighbours for the RING k-NN graph (default `round(n^0.65)`).
`dist.method`	distance passed to `dist` when `x` is a data matrix (default `"euclidean"`).
`n0, n1`	scan range (defaults `ceiling(0.05*n)` / `floor(0.95*n)`; for `alternative = "interval"` these are the minimum / maximum interval lengths).
`pval.appr`	logical; compute analytic p-value approximations.
`skew.corr`	logical; also compute the skewness-corrected approximation.
`B`	number of permutations for permutation p-values (`0` = none).
`alternative`	`"single"` (a single change-point) or `"interval"` (a changed interval).

Value

A list with `scanZ` (the WR / MR / TR statistics and estimated change-point(s)) and, as requested, `pval.appr` and `pval.perm`. See `rcpd`.

References

Zhou, D. and Chen, H. (2025). Asymptotic Distribution-Free Change-Point Detection for Modern Data Based on a New Ranking Scheme. *IEEE Transactions on Information Theory*, **71**(8), 6183–6197. doi:10.1109/TIT.2025.3575858.

See Also

`ring_graph`, `rcpd`, `Rise_Rank`

Examples

```
set.seed(1)
n <- 200; d <- 10; tau <- 100
X <- matrix(rnorm(n * d), n, d)
X[(tau + 1):n, ] <- X[(tau + 1):n, ] * 1.6   # a dispersion change at t = 100
res <- ring_cpd(X, skew.corr = TRUE)
res$scanZ$TR$tau
res$pval.appr$TR
```

`ring_graph` *Build a RING rank graph from a distance matrix*

Description

Constructs the sparse, rank-based k-nearest-neighbour graph that the RING statistics operate on – RING's ranking scheme: rank each observation's neighbours within its row of the similarity $S = \max(D) - D$, keep the k nearest (weights k..1 from nearest to k-th), and symmetrize.

Usage

```
ring_graph(D, k = NULL)
```

Arguments

`D`	an $n \times n$ distance matrix.
`k`	number of nearest neighbours (default `round(n^0.65)`, a value in the k-range used by Zhou & Chen).

Value

An $n \times n$ rank matrix with zero diagonal, suitable as the `R` argument of `rcpd`.

See Also

`ring_cpd`, `rcpd`, `Rise_Rank`

Examples

```
set.seed(1)
X <- matrix(rnorm(60 * 5), 60, 5)
R <- ring_graph(as.matrix(dist(X)), k = 8)
dim(R)
```

Rise_Rank *Build a rank matrix from a similarity matrix*

Description

Low-level ranker for the RING statistics: turns a similarity matrix into a rank matrix. `method = "row"` is the per-row ranking used to build the sparse RING k-NN graph. For change-point detection, build the `R` argument of `rcpd` with `ring_graph` (the validated sparse graph); the analytic p-value theory assumes that graph rather than a dense `"overall"` ranking.

Usage

```
Rise_Rank(S, method = c("overall", "row"))
```

Arguments

`S`	an $n \times n$ similarity matrix (larger = more similar; e.g. the negative of a distance matrix).
`method`	`"overall"` ranks all distinct pairs jointly (the symmetric rank matrix is returned); `"row"` ranks within each row.

Value

An $n \times n$ rank matrix with zero diagonal.

See Also

`rcpd`, `ring_cpd`

Examples

```
set.seed(1)
X <- matrix(rnorm(50 * 5), 50, 5)
R <- Rise_Rank(-as.matrix(dist(X)), method = "overall")
dim(R)
```

Index