Package: weaveCompare (via r-universe)

November 9, 2026

Type Package

Title Equivalence and Alignment of Periodic Binary Weaving Drawdowns

Version 0.1.0

Description Compares periodic binary weaving drawdowns using cyclic alignment, primitive rectangular repeats, exact canonical representations and Hamming distance on a common periodic domain. Optional reflections, axis exchange and face inversion define explicit equivalence rules. Supports collection distance matrices and duplicate grouping without requiring weaving design software. Comparisons describe binary interlacements, not fabric mechanics or cultural provenance.

License MIT + file LICENSE

Depends R (>= 3.6.0)

Imports stats

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Encoding UTF-8

Config/testthat/edition 3

RoxygenNote 7.3.1

NeedsCompilation no

Author Mark Joshua R. Secretario [aut, cre]

Maintainer Mark Joshua R. Secretario <markjoshuasecretario@gmail.com>

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

Date/Publication 2026-11-08 19:52:35 UTC

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

RemoteRef HEAD

RemoteSha 735501ebf9df35b9e30f220e579278b96258105f

Contents

as_drawdown 2
canonical_drawdown 2
compare_drawdowns 3
drawdown_dist 4
duplicate_drawdowns 5
primitive_repeat 6
shift_drawdown 6

Index **8**

`as_drawdown` *Validate a binary weaving drawdown*

Description

Validate a binary weaving drawdown

Usage

```
as_drawdown(x)
```

Arguments

`x`	A nonempty logical or numeric binary matrix. A weaveDraft weave_draft object is also accepted through its drawdown member.

Value

An integer matrix, with rows representing picks and columns ends.

Examples

```
as_drawdown(matrix(c(1, 0, 0, 1), 2))
```

`canonical_drawdown` *Canonical key for a periodic drawdown*

Description

Keys encode primitive dimensions and row-major bits. The lexicographically smallest representation over allowed transformations and all cyclic shifts is selected. Keys are lossless strings, not cryptographic hashes.

Usage

```
canonical_drawdown(
  x,
  reflect = FALSE,
  transpose = FALSE,
  invert = FALSE,
  max_work = 5e+07
)
```

Arguments

x	A binary drawdown.
reflect	Allow independent reversal of rows and columns.
transpose	Allow axis exchange. This does not invert bits automatically.
invert	Allow exchange of zero and one (face inversion).
max_work	Maximum estimated number of cell comparisons, guarding exhaustive search. Default 5e7. Memory and runtime still depend on input.

Value

A single character key. Equivalence is conditional on the options.

Examples

```
canonical_drawdown(matrix(c(1, 0, 0, 1), 2))
```

---

compare_drawdowns *Align two periodic drawdowns*

Description

Minimizes the fraction of unequal cells over allowed transformations of y and cyclic shifts. Primitive repeats are tiled to the least common multiple of each axis, so differently sized repeats can be compared without resizing. Ties use the first transformation, then smallest row and column offsets. Both matrices must use the same pick/end and zero/one conventions.

Usage

```
compare_drawdowns(
  x,
  y,
  reflect = FALSE,
  transpose = FALSE,
  invert = FALSE,
  max_work = 5e+07
)
```

Arguments

x, y	Binary drawdowns.
reflect, transpose, invert	Transformation flags; see [canonical_drawdown()].
max_work	Cell-comparison budget; see [canonical_drawdown()].

Value

A list with distance, similarity, equivalent (exact zero distance), transformation, offsets applied after transformation and repeat reduction, domain dimensions, and aligned matrices x and y on that domain.

Examples

```
a <- matrix(c(1, 0, 0, 1), 2)
compare_drawdowns(a, shift_drawdown(a, 1))$equivalent
```

---

drawdown_dist *Pairwise periodic Hamming distances*

Description

Pairwise periodic Hamming distances

Usage

```
drawdown_dist(
  drafts,
  reflect = FALSE,
  transpose = FALSE,
  invert = FALSE,
  max_work = 5e+07
)
```

Arguments

drafts	A nonempty list of binary drawdowns.
reflect, transpose, invert	Transformation flags.
max_work	Cell-comparison budget per pair.

Value

A stats dist object. Distances ignore phase and optionally other transformations. Distinct input matrices can have zero distance. No mechanical, perceptual or cultural meaning is implied. Clustering can use stats::hclust; Ward clustering is not recommended without Euclidean checks.

Examples

```
a <- matrix(c(1, 0, 0, 1), 2)
drawdown_dist(list(a, matrix(1, 2, 2)))
```

`duplicate_drawdowns` *Find exact duplicate structures in a collection*

Description

Find exact duplicate structures in a collection

Usage

```
duplicate_drawdowns(
  drafts,
  reflect = FALSE,
  transpose = FALSE,
  invert = FALSE,
  max_work = 5e+07
)
```

Arguments

`drafts`	A nonempty list of binary drawdowns, optionally uniquely named.
`reflect`, `transpose`, `invert`	Transformation flags.
`max_work`	Cell-comparison budget per canonicalization.

Value

A data frame of draft names, group numbers, first representative and canonical keys. Groups follow first occurrence, including singleton groups.

Examples

```
a <- matrix(c(1, 0, 0, 1), 2)
duplicate_drawdowns(list(first = a, shifted = shift_drawdown(a, 1)))
```

primitive_repeat *Reduce a drawdown to its primitive rectangular repeat*

Description

Only exact periods dividing the observed dimensions are considered. Incomplete repeats are not inferred. Constant matrices reduce to one cell.

Usage

```
primitive_repeat(x)
```

Arguments

x	A binary drawdown; see [as_drawdown()].

Value

An integer matrix containing the top-left primitive repeat.

Examples

```
primitive_repeat(matrix(rep(c(1, 0), 8), 4))
```

shift_drawdown *Shift a drawdown cyclically*

Description

Output cell (i,j) takes input cell (i + row_shift,j + col_shift), with indices wrapped around the matrix. Positive shifts move data up/left.

Usage

```
shift_drawdown(x, row_shift = 0, col_shift = 0)
```

Arguments

x	A binary drawdown.
row_shift, col_shift	Finite integer offsets; negative offsets allowed.

Value

An integer matrix of the original dimensions.

Examples

```
shift_drawdown(matrix(c(1, 0, 0, 0), 2), 1, 0)
```

Index

as_drawdown, 2

canonical_drawdown, 2
compare_drawdowns, 3

drawdown_dist, 4
duplicate_drawdowns, 5

primitive_repeat, 6

shift_drawdown, 6