Package 'covercorr'

August 25, 2025

Title Coverage Correlation Coefficient and Testing for Independence

Version 1.0.0

Maintainer Tengyao Wang <t.wang59@lse.ac.uk>

Description Computes the coverage correlation coefficient introduced in <doi:10.48550/arXiv.2508.06402> , a statistical measure that quantifies dependence between two random vectors by computing the union volume of data-centered hypercubes in a uniform space.

License GPL-3

Imports transport

Encoding UTF-8

RoxygenNote 7.3.2

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation yes

Author Tengyao Wang [aut, cre],
Mona Azadkia [aut, ctb],
Xuzhi Yang [aut, ctb]

Depends R (>= 3.5.0)

Repository CRAN

Date/Publication 2025-08-25 09:50:15 UTC

Contents

`CD8T` *Dataset: CD8+ T cell gene expression data*

Description

The CD8T dataset provides the gene expression data of fetal CD8+ T cells obtained in a single-cell RNA-seq experiment.

Usage

```
data(CD8T)
```

Format

A data frame with 9369 rows (cells) and 1000 columns (genes).

Source

Suo et al., Science (2022).

References

Suo, C., Dann, E., Goh, I., Jardine, L., Kleshchevnikov, V., Park, J.-E., Botting, R. A., et al. "Mapping the developing human immune system across organs." Science 376(6597), eabo0510 (2022).

`coverage_correlation` *Coverage-based Dependence Measure with Optional Visualisation*

Description

Computes the coverage correlation coefficient between input `x` and `y`, as introduced in the arXiv preprint. This coefficient measures the dependence between two random variables or vectors.

Usage

```
coverage_correlation(
  x,
  y,
  visualise = FALSE,
  method = c("auto", "exact", "approx"),
  M = NULL,
  na.rm = TRUE
)
```

Arguments

x	Numeric vector or matrix.
y	Numeric vector or matrix with the same number of rows as x.
visualise	Logical; if TRUE, displays a scatter plot of the rank-transformed points with overlaid rectangles to illustrate the coverage calculation. The default is FALSE (no plot). If set to TRUE but either x or y has more than one column, a warning is issued and visualise is reset to FALSE.
method	Character string specifying the computation method. Options are "auto", "exact", or "approx". See Details.
M	Integer; Number of Monte Carlo integration sample points (used when method = "approx"). Optional.
na.rm	Logical; if TRUE, remove NA values before computation.

Details

The procedure is as follows:

1. Calculate the rank transformations (r_x, r_y) of the inputs x and y.
2. Construct small cubes (in 2D, squares) of volume n^{-1} centered at each rank-transformed point.
3. Compute the total area of the union of these cubes, intersected with $[0, 1]^d$ where $d = d_x + d_y$.

The coverage correlation coefficient is then calculated based on this union area.

For more details, please refer to the original paper: the arXiv preprint.

The method argument controls how the computation is performed:

- "exact": Computes the exact value.
- "approx": Uses a Monte Carlo approximation with M sample points.
- "auto": Automatically selects a method based on the total number of columns in x and y: if more than 6, "approx" is used (with M = nrow(x)^{1.5} if M is not provided); otherwise, "exact" is used.

Value

A list with four elements:

- stat – The numeric value of the coverage correlation coefficient.
- pval – The p-value, calculated using the exact variance under the null hypothesis of independence between x and y.
- method – A character string indicating the computation method used.
- mc_se – A numeric value. If method "approx" was used mc_se is the standard error of the Monte Carlo approximation, otherwise it is 0.

Examples

```
set.seed(1)
n <- 100
x <- runif(n)
y <- sin(3*x) + runif(n) * 0.01
coverage_correlation(x, y, visualise = TRUE)
```

`covered_volume` *Total volume of union of rectangles*

Description

Total volume of union of rectangles

Usage

```
covered_volume(zmin, zmax)
```

Arguments

`zmin`	n x d matrix of bottomleft coordinates, one row per rectangle
`zmax`	n x d matrix of topright coordinates, one row per rectangle

Details

This is a wrapper of the C_covered_volume_partitioned function in C

Value

a numeric value of the volume of the union

`covered_volume_mc` *Total volume of union of rectangles using Monte Carlo integration*

Description

Total volume of union of rectangles using Monte Carlo integration

Usage

```
covered_volume_mc(zmin_s, zmax_s, M)
```

Arguments

zmin_s	n x d matrix of bottomleft coordinates, one row per rectangle
zmax_s	n x d matrix of topright coordinates, one row per rectangle
M	number of Monte Carlo integration sample points

Details

This is a wrapper of the C_covered_volume_mc function in C

Value

a list of the estimated volume of the union and its standard error

---

covered_volume_partitioned

Total volume of union of rectangles using volume hashing

Description

Total volume of union of rectangles using volume hashing

Usage

```
covered_volume_partitioned(zmin, zmax)
```

Arguments

zmin	n x d matrix of bottomleft coordinates, one row per rectangle
zmax	n x d matrix of topright coordinates, one row per rectangle

Details

This is a wrapper of the C_covered_volume_partitioned function in C

Value

a numeric value of the volume of the union

MK_rank *Monge–Kantorovich ranks (uniform OT via squared distances)*

Description

Computes the optimal matching that maps each observation in `X` to a reference point in `U` using uniform weights and squared Euclidean cost. Internally uses `transport::transport(method = "networkflow", p = 2)`. In 1D, this reduces to a rank-based matching `sort(U)[rank(X, ties.method = "random")]`.

Usage

```
MK_rank(X, U)
```

Arguments

`X`	Numeric vector of length n, or numeric matrix with n rows and d columns. If not a matrix, it is coerced with `as.matrix()`.
`U`	Numeric vector of length n, or numeric matrix with n rows and d columns. If not a matrix, it is coerced with `as.matrix()`. Must have the same number of rows as `X`.

Details

- Rows must match: `nrow(X) == nrow(U)` (otherwise an error is thrown).
- Columns must match: `ncol(X) == ncol(U)` (otherwise an error is thrown).
- Weights are uniform $(1/n)$ and the cost matrix is the sum of squared coordinate differences across columns.
- In 1D, ties in `X` are broken at random via `ties.method = "random"`; use `set.seed()` for reproducibility.

Value

If `ncol(X) == 1`, a numeric vector of length n containing the entries of `U` reordered to match the ranks of `X`. Otherwise, a numeric $n \times d$ matrix whose i-th row is the matched row of `U` corresponding to the i-th row of `X`.

Dependencies

Requires the **transport** package.

Examples

```
# 1D example (set seed for reproducible tie-breaking)
set.seed(1)
x <- rnorm(10)
u <- seq(0, 1, length.out = 10)
MK_rank(x, u)

# 2D example
set.seed(42)
X <- matrix(rnorm(200), ncol = 2)   # 100 x 2
U <- matrix(runif(200),  ncol = 2)  # 100 x 2
R <- MK_rank(X, U)
dim(R)  # 100 2
```

`plot_rectangles` *Plot a collection of axis-aligned rectangles in the unit square*

Description

Draws rectangles specified by their `xmin`, `xmax`, `ymin`, and `ymax`, optionally adding them to an existing plot. When `add = FALSE`, a fresh $[0,1] \times [0,1]$ plot with a grid and equal aspect ratio is created.

Usage

```
plot_rectangles(xmin, xmax, ymin, ymax, add = FALSE)
```

Arguments

`xmin`	Numeric vector of left x-coordinates.
`xmax`	Numeric vector of right x-coordinates (same length as `xmin`).
`ymin`	Numeric vector of bottom y-coordinates (same length as `xmin`).
`ymax`	Numeric vector of top y-coordinates (same length as `xmin`).
`add`	Logical; if `TRUE`, add to an existing plot. Default `FALSE`.

Value

Invisibly returns `NULL`. Use this function for its plotting output, not for a returned value.

split_rectangles *Split rectangles by wrapping them around edges of* $[0,1]\hat{}d$

Description

Split rectangles by wrapping them around edges of $[0,1]^d$

Usage

```
split_rectangles(zmin, zmax)
```

Arguments

zmin	n x d matrix of bottom-left coordinates, one row per rectangle
zmax	n x d matrix of top-right coordinates, one row per rectangle

Details

This is a wrapper of the C_split_rectangles function implemented in C

Value

a list of zmin and zmax, describing the bottom-left and top-right coordinates of splitted rectangles

variance_formula *Variance of the the excess vacancy*

Description

Exact formula for n times the variance of the excess vacancy. For independent X and Y, the variance of the coverage correlation coefficient is obtained by dividing the returned value by $n(1-e^{-1})^2$. check the arXiv preprint for more details

Usage

```
variance_formula(n, d)
```

Arguments

n	sample size
d	dimension (X, Y)

Value

variance formula in paper

Index