Package ‘arkhaia’

May 7, 2026

Title Archaeological and Historical Analysis

Version 0.5.5

Date 2026-04-11

Description Tools for quantitative analysis related to archaeological and historical problems for irregularly spaced time indexed observations, toward evaluating linear dependence and homogeneity over time. Methods include effect sizes for measuring homogeneity, simulation from a truncated Poisson distribution for random right-censoring of count data, and least-squares spectral analysis by lowest frequency iteration for model fitting. Collins-Elliott (2026) <`https://volweb.utk.edu/~scolli46/sce_aqysuppl2026.pdf`>.

License GPL (>= 3)

Imports Rcpp (>= 1.1.0), Rdpack, stats

RdMacros Rdpack

Encoding UTF-8

LinkingTo Rcpp, RcppArmadillo

RoxygenNote 7.3.2

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

NeedsCompilation yes

Author Stephen A. Collins-Elliott [aut, cre] (ORCID: <`https://orcid.org/0000-0002-5642-6903`>)

Maintainer Stephen A. Collins-Elliott <`sce@utk.edu`>

Repository CRAN

Date/Publication 2026-05-05 18:00:02 UTC

Contents

`CR` *Cressie-Read Power-Divergence Statistic*

Description

For a matrix of cross-tabulated counts of observations, computes the Cressie-Read power-divergence statistic according to the selection of a parameter λ (Cressie and Read 1984; Read and Cressie 1988).

Usage

```
CR(x, lambda = 2/3)

## S3 method for class 'matrix'
CR(x, lambda = 2/3)

## S3 method for class 'data.frame'
CR(x, lambda = 2/3)

## S3 method for class 'xtabs'
CR(x, lambda = 2/3)

## S3 method for class 'table'
CR(x, lambda = 2/3)
```

Arguments

`x`	A matrix of cross-tabulated counts.
`lambda`	The parameter of the Cressie-Read power-divergence statistic. Default is the recommended value of 2/3. To use Pearson's method, set `lambda = 1`.

Value

The Cressie-Read power-divergence statistic.

References

Cressie NAC, Read TRC (1984). "Multinomial Goodness-of-Fit Tests." *Journal of the Royal Statistical Society. Series B (Methodological)*, **46**, 440–464. doi:10.1111/j.25176161.1984.tb01318.x.

Read TRC, Cressie NAC (1988). *Goodness-of-Fit Statistics for Discrete Multivariate Data*. Springer, New York.

Examples

```
x1 <- c(2, 0, 10, 11, 5)
x2 <- c(1, 1, 17, 23, 3)
x3 <- c(0, 0, 2, 81, 11)

x <- matrix(c(x1, x2, x3), ncol = 3)

CR(x)
CR(x, lambda = 1)
```

`homogeneity` *Homogeneity of Related Assemblages via Effect Size*

Description

Given a contingency table of cross-tabulated counts, with contexts along the columns and types along rows, this function estimates the distribution of effect sizes between pairs of "related" assemblages (determined *a priori*), as compared against a distribtion of "unrelated" assemblages (if not specified, are supplied as all pairs which are not included in the "related" set). Hence, the practical significance of the level of homogeneity between related deposits is evaluated against that of unrelated deposits.

Usage

```
homogeneity(x, related = NULL, unrelated = NULL, direction = "UW")

## S3 method for class 'effect_sizes'
homogeneity(x, related = NULL, unrelated = NULL, direction = "UW")
```

Arguments

`x`	An effect_sizes object as returned by `VB_pair`.
`related`	The related pairs of contexts as a two-column matrix or data frame (contexts between which one anticipates a meaningful relationship). Names must match `colnames(x)`.
`unrelated`	The unrelated pairs of contexts as a two-column matrix or data frame (contexts between which one does not anticipate a meaningful relationship). Names must match `colnames(x)`. May be left `NULL`, in which event all pairs not expressed in `related` are created
`direction`	Whether the related or unrelated effect size should come first. Default is `"UW"`; alternative is `"WU"`.

Value

A list of results:

- `n` - A vector of the number of related and unrelated pairs of contexts, respectively n_W and n_U.
- `U` - The effect sizes between unrelated pairs of contexts.
- `W` - The effect sizes between related pairs of contexts.
- `Q` - The quantile indicating the proportion of related contexts more homogenous than unrelated contexts (if direction is `"UW"`); less homogenous if direction is set to `"UW"`.
- `D` - The distribution of differences, $D_{ij} = U_j - W_i$, if direction is set to `"UW"`. The proportion of $D > 0$ is equivalent to the mean of `Q`.

Examples

```
x1 <- c(2, 0, 10, 11, 5)
x2 <- c(1, 1, 17, 23, 3)
x3 <- c(0, 0, 2, 81, 11)
x4 <- c(3, 18, 9, 0, 23)
x <- matrix(c(x1, x2, x3, x4), ncol = 4)
colnames(x) <- c("surface1", "subsurface1", "surface2", "subsurface2")
rownames(x) <- LETTERS[1:5]

# related pairs
W_contexts <- matrix(c("surface1", "surface2", "subsurface1", "subsurface2"), ncol = 2)

# unrelated pairs (will be automatically created if left NULL)
U_contexts <- matrix(c("surface1", "surface1", "surface2", "subsurface1",
                       "surface2","subsurface2", "surface1", "subsurface2"), ncol = 2)

effect_sizes <- VB_pair(x)
homogeneity(effect_sizes, related = W_contexts, unrelated = U_contexts)
homogeneity(effect_sizes, related = W_contexts)
```

homogeneity_LOO *Leave-One-Out Routine for Homogeneity of Related Assemblages*

Description

To evaluate the stability of estimates of effect sizes between archaeological contexts in light of the inclusion or exclusion of types and contexts, this routine computes the homogeneity of related assemblages iteratively in two ways, first by leaving out a type on each iteration and second by leaving out a context on each iteration. For details on the arguments, see also `homogeneity` and `VB`).

Usage

```
homogeneity_LOO(
  x,
  related = NULL,
  unrelated = NULL,
  direction = "UW",
  lambda = 2/3
)

## S3 method for class 'matrix'
homogeneity_LOO(
  x,
  related = NULL,
  unrelated = NULL,
  direction = "UW",
  lambda = 2/3
)

## S3 method for class 'data.frame'
homogeneity_LOO(
  x,
  related = NULL,
  unrelated = NULL,
  direction = "UW",
  lambda = 2/3
)

## S3 method for class 'table'
homogeneity_LOO(
  x,
  related = NULL,
  unrelated = NULL,
  direction = "UW",
  lambda = 2/3
)
```

```
## S3 method for class 'xtabs'
homogeneity_LOO(
  x,
  related = NULL,
  unrelated = NULL,
  direction = "UW",
  lambda = 2/3
)
```

Arguments

`x`	A data frame or matrix representing a contingency table of counts, with contexts along the columns and types along rows.
`related`	The related pairs of contexts as a two-column matrix or data frame (contexts between which one anticipates a meaningful relationship). Names must match `colnames(x)`.
`unrelated`	The unrelated pairs of contexts as a two-column matrix or data frame (contexts between which one does not anticipate a meaningful relationship). Names must match `colnames(x)`.
`direction`	Whether the related or unrelated effect size should come first. Default is `"UW"`; alternative is `"WU"`.
`lambda`	Parameter of the Cressie-Read power-divergence statistic. Default is the recommended value of 2/3.

Value

A list containing:

- EQ - The mean quantile expressing the degree of homogeneity among related contexts.
- EQ_T_mean - The mean quantile upon iterating over the omission of each type (of the LOO samples).
- EQ_T_var - The variance of the LOO type samples.
- EQ_C_mean - The mean quantile upon iterating over the omission of each context.
- EQ_C_var - The variance of the LOO context samples.

Examples

```
x1 <- c(2, 0, 10, 11, 5)
x2 <- c(1, 1, 17, 23, 3)
x3 <- c(0, 0, 2, 81, 11)
x4 <- c(3, 18, 9, 0, 23)
x <- matrix(c(x1, x2, x3, x4), ncol = 4)
colnames(x) <- c("surface1", "subsurface1", "surface2", "subsurface2")
rownames(x) <- LETTERS[1:5]

# related pairs
W_contexts <- matrix(c("surface1", "surface2", "subsurface1", "subsurface2"), ncol = 2)
```

```
# unrelated pairs (will be automatically created if left NULL)
U_contexts <- matrix(c("surface1", "surface1", "surface2", "subsurface1",
                       "surface2","subsurface2", "surface1", "subsurface2"), ncol = 2)

homogeneity_LOO(x, related = W_contexts, unrelated = U_contexts)
```

`log_OR_pair` *Log Odds Ratio with Haldane-Anscombe's Correction Pairwise between Columns*

Description

For a contingency table with columns representing contexts and rows representing types, computes the log odds ratio of a 2 x 2 contingency table of the presence-absences across each pair of columns. The addition of 0.5 to cells of the contingency table is performed (Anscombe 1956; Haldane 1956).

Usage

```
log_OR_pair(x)

## S3 method for class 'matrix'
log_OR_pair(x)

## S3 method for class 'data.frame'
log_OR_pair(x)

## S3 method for class 'xtabs'
log_OR_pair(x)

## S3 method for class 'table'
log_OR_pair(x)
```

Arguments

`x`	A matrix or data frame with contexts along columns and types along rows.

Value

A matrix giving the log odds ratio between all columns of the input.

Examples

```
x1 <- c(2, 0, 0, 11, 5, 0, 2, 0, 4)
x2 <- c(1, 1, 0, 23, 3, 3, 0, 0, 0)
x3 <- c(1, 0, 0, 0, 10, 0, 4, 0, 1)

x <- data.frame(S1 = x1, S2 = x2, S3 = x3)
```

```
rownames(x) <- LETTERS[1:nrow(x)]

log_OR_pair(x)
```

`LSSA` *Least Squares Spectral Analysis (LSSA)*

Description

Peforms a simple least squares fitting to time indexed data of the form $x(t) = \beta_0 + \sum_{i=1}^{n} \beta_{1i} \cos(2\pi f t) + \beta_{2i} \sin(2\pi f t)$, using a range of potential frequencies (Vaníček 1969, 1971). Intercept may be ommitted.

Usage

```
LSSA(
  x,
  freqs = seq(0.001, 0.5, by = 5e-04),
  intercept = TRUE,
  type = "frequency"
)

## S3 method for class 'matrix'
LSSA(
  x,
  freqs = seq(0.001, 0.5, by = 5e-04),
  intercept = TRUE,
  type = "frequency"
)

## S3 method for class 'data.frame'
LSSA(
  x,
  freqs = seq(0.001, 0.5, by = 5e-04),
  intercept = TRUE,
  type = "frequency"
)
```

Arguments

`x`	A data frame of two columns, the first containing time indices and the second containing values.
`freqs`	A vector of frequencies to evaluate. By default a grid from 0.001 to 0.5 is tested at an interval of 0.005.
`intercept`	Whether to include the intercept. Default is `TRUE`.
`type`	Type of output. Can be either `"frequency"` (the default) or `"period"`.

Value

The a data frame containing the power and residual sum of squares for each frequency, as well as coefficients.

References

Vaníček P (1969). "Approximate Spectral Analysis by Least-Squares Fit." *Astrophysics and Space Science*, **4**, 387–391. doi:10.1007/BF00651344.

Vaníček P (1971). "Further development and properties of the spectral analysis by least-squares fit." *Astrophysics and Space Science*, **12**, 10–33. doi:10.1007/BF00656134.

`LSSA_LFI` *Least Squares Spectral Analysis via Lowest Frequency Iteration (LSSA-LFI)*

Description

Peforms a simple least squares fitting to time indexed data of the form y = SIGMA B cos(2 pi f t) + C sin(2 pi f t) + D, using an input of frequencies; the lowest frequency peak (not the highest power frequency) is chosen for regression, up to a chosen number of iterations. Intercept may be ommitted. The lowest frequency is equivalent to the longest period.

Usage

```
LSSA_LFI(x, n_iter = 1, intercept = TRUE, AIC = FALSE)

## S3 method for class 'matrix'
LSSA_LFI(x, n_iter = 1, intercept = TRUE, AIC = FALSE)

## S3 method for class 'data.frame'
LSSA_LFI(x, n_iter = 1, intercept = TRUE, AIC = FALSE)
```

Arguments

`x`	A data frame of two columns, with the first column containing time indices and the second containing values.
`n_iter`	The number of iterations to run. Default is 1.
`intercept`	Whether to include the intercept. Default is `TRUE`.
`AIC`	If `TRUE`, only the result that has yeilded the lowest AIC (Aikake Information Criterion) is given. Default is `FALSE`.

Value

A list containing: * A list of the coefficients for each iteration (the intercept is included in the first iteration). * A vector of the frequencies. * The residual sum of squares (RSS) after each iteration (decreasing). * The AIC upon each iteration. (If the paraemter AIC is `TRUE`, this will stop at the lowest AIC value produced by the frequencies tested).

`LSSA_LFI_candidates` *Linear Dependence of LSSA-LFI Candidate Models via AIC*

Description

For a set of time series (namely partitions of set of series) contained in a `list`, will compute the Akaike Information Criterion (AIC) for each candidate set.

Usage

```
LSSA_LFI_candidates(x, sets = NULL, n_iter = 1, intercept = TRUE)

## S3 method for class 'list'
LSSA_LFI_candidates(x, sets = NULL, n_iter = 1, intercept = TRUE)
```

Arguments

`x`	A `list` containing the time series, each of which should be a matrix or data frame with time index in the first column and value in the second.
`sets`	Candidate sets to evaluate; must be a `list` of `lists` containing the indices of the sets in `x`. If left `NULL`, two sets are evaluated: all series pooled together [1] and all series kept separate [2].
`n_iter`	The number of iterations to run for the least squares spectral analysis via lowest frequency iteration (LSSA-LFI). Default is 1.
`intercept`	Whether to include the intercept in the least squares spectral analysis via lowest frequency iteration (LSSA-LFI). Default is `TRUE`.

Value

A `list` containing the AIC for each candidate set, for each iteration.

`LSSA_LFI_epoch` *Period-Variable Pairwise Selection of Linearlly Dependent LSSA-LFI Models*

Description

Evaluate pairwise linear dependence between two time series using a LSSA-LFI valdiated model selection (see `LSSA_LFI_validated`)), with variable length time period.

Usage

```
LSSA_LFI_epoch(
  x,
  pair = NULL,
  n_iter = 1,
  intercept = TRUE,
  t_range = NULL,
  h_range = NULL,
  t_grid = 1,
  h_grid = 1
)

LSSA_LFI_epoch(
  x,
  pair = NULL,
  n_iter = 1,
  intercept = TRUE,
  t_range = NULL,
  h_range = NULL,
  t_grid = 1,
  h_grid = 1
)
```

Arguments

`x`	A list of data frames.
`pair`	The pair of commodities to evaluate in the list `x`, either names or indices.
`n_iter`	The number of iterations to run. Default is 1.
`intercept`	Whether to include the intercept in the least squares spectral analysis via lowest frequency iteration (LSSA-LFI). Default is `TRUE`.
`t_range`	The range of the time period. Default are the minimum and maximum dates spanned by the data in `x`.
`h_range`	The range of the potential windows of h. Default is from 0 to entire span of `t_range`.
`t_grid`	The length of interval along which to sample t. Default is 1.
`h_grid`	The length of interval along which to sample h. Default is 1.

Value

A matrix giving the probability of linear dependence, predicated upon time t and window length h.

`LSSA_LFI_model` *LSSA-LFI Model*

Description

Generates a data frame of values $f(t)$ of the model generated by LSSA-LFI (see `LSSA_LFI`)).

Usage

```
LSSA_LFI_model(x, t_ = NULL, n_iter = 1, intercept = TRUE, label = "model")

## S3 method for class 'matrix'
LSSA_LFI_model(x, t_ = NULL, n_iter = 1, intercept = TRUE, label = "model")

## S3 method for class 'data.frame'
LSSA_LFI_model(x, t_ = NULL, n_iter = 1, intercept = TRUE, label = "model")
```

Arguments

`x`	A data frame where time indices are in the first column and values are in the second.
`t_`	A vector giving samples range of t for computing $f(t)$. Default is from the minimum to maximum time index sampled at 0.01 intervals.
`n_iter`	The number of iterations to run. Default is 1.
`intercept`	Whether to include the intercept in the least squares spectral analysis via lowest frequency iteration (LSSA-LFI). Default is `TRUE`.
`label`	Default is `"model"`.

Value

A data frame containing t, $f(t)$.

`LSSA_LFI_pairwise` *Pairwise Selection of Linearlly Dependent LSSA-LFI Models*

Description

Evaluate pairwise linear dependence between observations using a LSSA-LFI valdiated model selection (see `LSSA_LFI_validated`)).

Usage

```
LSSA_LFI_pairwise(x, n_iter = 1, intercept = TRUE)

## S3 method for class 'list'
LSSA_LFI_pairwise(x, n_iter = 1, intercept = TRUE)
```

Arguments

`x`	A list of data frames.
`n_iter`	The number of iterations to run. Default is 1.
`intercept`	Whether to include the intercept in the least squares spectral analysis via lowest frequency iteration (LSSA-LFI). Default is `TRUE`.

Value

An upper-triangular matrix, containing the probability of linear dependence between series.

`LSSA_LFI_validated` *Validated Linear Dependence via LSSA-LFI*

Description

Probability of linear dependence between two groups of time series observations using a LSSA-LFI model selection (see `LSSA_LFI_candidates`)), given a list of at least three time series. Confounding variate is selected from the remaining time series in the list.

Usage

```
LSSA_LFI_validated(x, pair = NULL, n_iter = 1, intercept = TRUE)

## S3 method for class 'list'
LSSA_LFI_validated(x, pair = NULL, n_iter = 1, intercept = TRUE)
```

Arguments

`x`	A list of data frames.
`pair`	The pair of series to evaluate in the list `x`, either names or indices.
`n_iter`	The number of iterations to run. Default is 1.
`intercept`	Whether to include the intercept in the least squares spectral analysis via lowest frequency iteration (LSSA-LFI). Default is `TRUE`.

Value

The probability of linear dependence between two time series, in which 1 indicates linear dependence almost surely and 0 indicates independence.

`model_select` *Model Selection of LSSA-LFI Candidates*

Description

For an `LSSA_LFI_AIC` object see `LSSA_LFI_candidates`), returns the index of the candidate model. If `sets` is `NULL` in the `LSSA_LFI_candidates` function, an index of [1] refers to the model of a pooled (joint) grouping, [2] refers to the model of discrete, separate groupings.

Usage

```
model_select(x)

## S3 method for class 'LSSA_LFI_AIC'
model_select(x)
```

Arguments

`x`	An `LSSA_LFI_AIC` object.

Value

The index of the grouping which contains the lowest AIC score.

`pa_matrix` *Presence-Absence Matrix*

Description

Create a 2 x 2 contingency table of the presence/absence of a given type

Usage

```
pa_matrix(x)

## S3 method for class 'matrix'
pa_matrix(x)

## S3 method for class 'data.frame'
pa_matrix(x)

## S3 method for class 'table'
pa_matrix(x)

## S3 method for class 'xtabs'
pa_matrix(x)
```

Arguments

`x`	A matrix or data frame where the two columns indicate contexts and the rows indicate types, with each cell containing counts of types in a context.

Value

A 2 x 2 contingency table of the counts of types present in both, either, or neither context.

Examples

```
x1 <- c(2, 0, 10, 11, 5)
x2 <- c(1, 1, 17, 23, 3)

x <- data.frame(S1 = x1, S2 = x2)
rownames(x) <- LETTERS[1:5]

pa_matrix(x)
```

`pois_rcens` *Least Squares Fit of Poisson Distribution for Random Right-Censored Data*

Description

For a matrix of cross-tabulated counts of observations which constitute a minimum threshold, this function esimates the rate parameter column-wise, either retaining or omitting zeros, by a least-squares approach.

Usage

```
pois_rcens(x, lambda_grid = seq(0.01, 100, by = 0.01), omit_zero = TRUE)

## S3 method for class 'matrix'
pois_rcens(x, lambda_grid = seq(0.01, 100, by = 0.01), omit_zero = TRUE)

## S3 method for class 'data.frame'
pois_rcens(x, lambda_grid = seq(0.01, 100, by = 0.01), omit_zero = TRUE)

## S3 method for class 'xtabs'
pois_rcens(x, lambda_grid = seq(0.01, 100, by = 0.01), omit_zero = TRUE)

## S3 method for class 'table'
pois_rcens(x, lambda_grid = seq(0.01, 100, by = 0.01), omit_zero = TRUE)
```

Arguments

`x`	A matrix of cross-tabulated counts.
`lambda_grid`	The resolutation at which to sample for the rate parameter. Default is `seq(0.1, 100, by = 0.01)`.
`omit_zero`	Whether to omit zeros. Default is `TRUE`.

Value

A vector of the rate parameters for each column.

Examples

```
x1 <- c(1,2,2,5,7,0,0)
x2 <- c(9,2,5,15,7,90,0)
x <- matrix(c(x1,x2), ncol = 2)

pois_rcens(x)
pois_rcens(x, omit_zero = FALSE)
```

`trace_pair` *Trace Coefficient Pairwise between Columns*

Description

For a contingency table with columns representing contexts and rows representing types, computes the trace of a 2 x 2 contingency table of the presence-absences across each pair of columns.

Usage

```
trace_pair(x)

## S3 method for class 'matrix'
trace_pair(x)

## S3 method for class 'data.frame'
trace_pair(x)

## S3 method for class 'xtabs'
trace_pair(x)

## S3 method for class 'table'
trace_pair(x)
```

Arguments

`x`	A matrix or data frame with contexts along columns and types along rows.

Value

A matrix giving the trace between all columns of the input.

Examples

```
x1 <- c(2, 0, 0, 11, 5, 0, 2, 0, 4)
x2 <- c(1, 1, 0, 23, 3, 3, 1, 1, 1)
x3 <- c(1, 0, 0, 0, 10, 0, 4, 0, 0)

x <- data.frame(S1 = x1, S2 = x2, S3 = x3)
rownames(x) <- LETTERS[1:nrow(x)]

trace_pair(x)
```

`trim_epoch` *Trim to Epoch*

Description

For a data frame in which the first column contains a time index and the second colum observations, trim the data frame to include observations only with a given epoch (time period).

Usage

```
trim_epoch(x, epoch = NULL)

## S3 method for class 'matrix'
trim_epoch(x, epoch = NULL)

## S3 method for class 'data.frame'
trim_epoch(x, epoch = NULL)
```

Arguments

`x`	A data frame or matrix containing time-indexed data, with time index in the first column and value in the second.
`epoch`	A numeric vector given the start and end time indices of the epoch.

Value

A data frame containing only those observations with time indices within the epoch.

`trunc_pois` *Resampled Contingency Table via a Truncated Poisson for Random Right-Censored Data*

Description

For a matrix of cross-tabulated counts of observations which constitute a minimum threshold, returns a contingency table whose counts are sampled according to a truncated Poisson distribution, whose rate parameter is determined column-wise (see `pois_rcens`).

Usage

```
trunc_pois(x, lambda_grid = seq(0.01, 100, by = 0.01), omit_zero = TRUE)

## S3 method for class 'matrix'
trunc_pois(x, lambda_grid = seq(0.01, 100, by = 0.01), omit_zero = TRUE)
```

```
## S3 method for class 'data.frame'
trunc_pois(x, lambda_grid = seq(0.01, 100, by = 0.01), omit_zero = TRUE)

## S3 method for class 'xtabs'
trunc_pois(x, lambda_grid = seq(0.01, 100, by = 0.01), omit_zero = TRUE)

## S3 method for class 'table'
trunc_pois(x, lambda_grid = seq(0.01, 100, by = 0.01), omit_zero = TRUE)
```

Arguments

`x`	A matrix of cross-tabulated counts.
`lambda_grid`	The resolutation at which to sample for the rate parameter. Default is `seq(0.1, 100, by = 0.01)`.
`omit_zero`	Whether to omit zeros. Default is `TRUE`.

Value

A contingency table Y of the same size as X, with y_{ij} drawn according to a truncated Poisson distribution that ensures $y_{ij} \geq x_{ij}$.

Examples

```
x1 <- c(1,2,2,5,7,0,0)
x2 <- c(9,2,5,15,7,90,0)
x <- matrix(c(x1,x2), ncol = 2)

pois_rcens(x)
pois_rcens(x, omit_zero = FALSE)
```

`VB` *Bias-Corrected Cramer's V*

Description

For a matrix of cross-tabulated counts of observations, estimates Cramer's V using Bergsma's bias correction (Bergsma 2013), using the Cressie-Read power divergence statistic (see `CR`).

Usage

```
VB(x, lambda = 2/3)

## S3 method for class 'matrix'
VB(x, lambda = 2/3)

## S3 method for class 'data.frame'
VB(x, lambda = 2/3)
```

```
## S3 method for class 'xtabs'
VB(x, lambda = 2/3)

## S3 method for class 'table'
VB(x, lambda = 2/3)
```

Arguments

`x`	A matrix or data frame of cross-tabulated counts.
`lambda`	Parameter of the Cressie-Read power-divergence statistic. Default is the recommended value of 2/3.

Value

Bergsma's bias-corrected etimate of Cramer's V.

References

Bergsma W (2013). "A Bias-Correction for Cramér's V and Tschuprow's T." *Journal of the Korean Statistical Society*, **42**, 323–328. doi:10.1016/j.jkss.2012.10.002.

Examples

```
x1 <- c(2, 0, 10, 11, 5)
x2 <- c(1, 1, 17, 23, 3)
x3 <- c(0, 0, 2, 81, 11)

x <- matrix(c(x1, x2, x3), ncol = 3)

VB_pair(x)
VB(x, lambda = 1)
```

`VB_LOO_type` *Leave-One-Out Type Routine for Pairwise Effect Sizes of Archaeological Contexts*

Description

To evaluate the stability of estimates of effect sizes between archaeological contexts in light of the inclusion or exclusion of a given type, this routine computes bias-corrected Cramer's V `VB`) omitting a type on each iteration.

Usage

```
VB_LOO_type(x, lambda = 2/3)

## S3 method for class 'matrix'
VB_LOO_type(x, lambda = 2/3)

## S3 method for class 'data.frame'
VB_LOO_type(x, lambda = 2/3)

## S3 method for class 'xtabs'
VB_LOO_type(x, lambda = 2/3)

## S3 method for class 'table'
VB_LOO_type(x, lambda = 2/3)
```

Arguments

`x`	A contingency table as a matrix or data frame expressing counts, with contexts along columns and types along rows.
`lambda`	Parameter of the Cressie-Read power-divergence statistic. Default is the recommended value of 2/3.

Value

A three-dimensional array of the pairwise context-by-context effect sizes given the ommission of a type, for all types given in the input data frame.

Examples

```
x1 <- c(2, 0, 10, 11, 5)
x2 <- c(1, 1, 17, 23, 3)
x3 <- c(0, 0, 2, 81, 11)

x <- matrix(c(x1, x2, x3), ncol = 3)

rownames(x) <- LETTERS[1:nrow(x)]
colnames(x) <- c("S1", "S2", "S3")

VB_LOO_type(x)
```

`VB_pair` *Bias-Corrected Cramer's V Pairwise between Columns*

Description

For a matrix or data frame of cross-tabulated counts of observations, estimates Cramer's V using Bergsma's bias correction (Bergsma 2013) by subsetting the table by pairs of columns (see `VB`). In subsetting, zero row/columns are automatically removed from the subset matrix.

Usage

```
VB_pair(x, lambda = 2/3)

## S3 method for class 'matrix'
VB_pair(x, lambda = 2/3)

## S3 method for class 'data.frame'
VB_pair(x, lambda = 2/3)

## S3 method for class 'xtabs'
VB_pair(x, lambda = 2/3)

## S3 method for class 'table'
VB_pair(x, lambda = 2/3)
```

Arguments

`x`	A matrix or data frame of cross-tabulated counts.
`lambda`	Parameter of the Cressie-Read power-divergence statistic. Default is the recommended value of 2/3.

Value

A matrix of Bergsma's bias-corrected etimate of Cramer's V, pairwise between columns of the input matrix, of class `effect_sizes`.

References

Bergsma W (2013). "A Bias-Correction for Cramér's V and Tschuprow's T." *Journal of the Korean Statistical Society*, **42**, 323–328. doi:10.1016/j.jkss.2012.10.002.

Examples

```
x1 <- c(2, 0, 10, 11, 5)
x2 <- c(1, 1, 17, 23, 3)
x3 <- c(0, 0, 2, 81, 11)

x <- matrix(c(x1, x2, x3), ncol = 3)

VB_pair(x)
VB_pair(x, lambda = 1)
```

`VB_trunc_pois` *Bias-Corrected Cramer's V for Random Right-Censored Data*

Description

Given a matrix of cross-tabulated counts of observations which constitute a minimum threshold, returns the distribution of bias-corrected Cramer's V by resampling from a truncated Poisson distribution, with rates determined column-wise (see `VB`, `trunc_pois`).

Usage

```
VB_trunc_pois(
  x,
  lambda = 2/3,
  lambda_grid = seq(0.01, 100, by = 0.01),
  omit_zero = TRUE,
  n_iter = 10^5
)

## S3 method for class 'matrix'
VB_trunc_pois(
  x,
  lambda = 2/3,
  lambda_grid = seq(0.01, 100, by = 0.01),
  omit_zero = TRUE,
  n_iter = 10^5
)

## S3 method for class 'data.frame'
VB_trunc_pois(
  x,
  lambda = 2/3,
  lambda_grid = seq(0.01, 100, by = 0.01),
  omit_zero = TRUE,
  n_iter = 10^5
)

## S3 method for class 'xtabs'
VB_trunc_pois(
  x,
  lambda = 2/3,
  lambda_grid = seq(0.01, 100, by = 0.01),
  omit_zero = TRUE,
  n_iter = 10^5
)

## S3 method for class 'table'
```

```
VB_trunc_pois(
  x,
  lambda = 2/3,
  lambda_grid = seq(0.01, 100, by = 0.01),
  omit_zero = TRUE,
  n_iter = 10^5
)
```

Arguments

`x`	A matrix of cross-tabulated counts.
`lambda`	The value of lambda (default is 2/3) for the Cressie-Read power divergence statistic used to estimate the bias-corrected Cramer's V. Default is `2/3`.
`lambda_grid`	The resolutation at which to sample for the rate parameter. Default is `seq(0.1, 100, by = 0.01)`.
`omit_zero`	Whether to omit zeros. Default is `TRUE`.
`n_iter`	Number of samples of V to take. Default is `10^5`.

Value

A contingency table Y of the same size as X, with y_{ij} drawn according to a truncated Poisson distribution, $y_{ij} \geq x_{ij}$.

Examples

```
x1 <- c(1,2,2,5,7,0,0)
x2 <- c(9,2,5,15,7,90,0)
x <- matrix(c(x1,x2), ncol = 2)

VB_trunc_pois(x, n_iter = 10^2)
VB_trunc_pois(x, omit_zero = FALSE, n_iter = 10^2)
```

Index