Package 'fastLISA'

July 6, 2026

Type Package

Title Fast Local Indicators of Spatial Association (LISA)

Version 1.0.0

Description Computes various Local Indicators of Spatial Association (LISA) statistics, including univariate and bivariate local Moran's I, Empirical Bayes local Moran's I, univariate and multivariate local Geary's C, and Getis-Ord G and G* statistics. The methods follow Anselin (1995), Getis and Ord (1992), and Anselin (2019). Leverages a high-performance, plain-C backend with optional OpenMP multi-core support for fast permutation-based pseudo-p-value calculation. Accepts any 'spdep' listw spatial weight matrix, including custom and non-contiguity weights. Uses sample standardisation (n-1) and 'rgeoda'-style permutation p-values. Output cluster codes match 'rgeoda' conventions, including the Isolated category for observations without neighbours.

License GPL-3

Encoding UTF-8

Imports stats

Suggests spdep

NeedsCompilation yes

SystemRequirements C99, optional OpenMP

Author Lizhong Chen [aut, cre]

Maintainer Lizhong Chen <chen.l@wehi.edu.au>

Repository CRAN

Date/Publication 2026-07-06 14:30:06 UTC

Contents

local_g 2
local_geary 4
local_gstar 5

local_moran 7
local_moran_bv 9
local_moran_eb 12
local_multigeary 14

Index **16**

`local_g` *Local Getis-Ord G*

Description

`local_g` computes the Getis-Ord local G_i statistic, a Local Indicator of Spatial Association that detects local clustering of high values ("hot spots") and low values ("cold spots"). For observation i, with row-standardised spatial weights w^*_{ij} and the focal value *excluded* from both the lag and the denominator,

$$G_i = \frac{\sum_{j \neq i} w^*_{ij} x_j}{\sum_k x_k - x_i}.$$

A large G_i indicates that i is surrounded by high values; a small G_i indicates a low-value neighbourhood. G_i contains no self term; see `local_gstar` for the self-inclusive G^*_i.

Usage

```
local_g(
  x,
  listw,
  nsim = 999L,
  iseed = NULL,
  p.value = 0.05,
  n.cores = 1L,
  moments = FALSE,
  p.method = c("count", "rank")
)
```

Arguments

`x`	Numeric vector of length `n`.
`listw`	A `listw` object from **spdep**.
`nsim`	Integer; number of permutations. Default `999L`. Must be at least 1.
`iseed`	Integer seed for RNG, or `NULL`.
`p.value`	Numeric significance cutoff. Default `0.05`.
`n.cores`	Integer; number of OpenMP threads. Default `1L`; set higher to use multiple cores. Ignored on platforms without OpenMP.
`moments`	Logical; if `TRUE`, append permutation-distribution moments `E.Gi`, `Var.Gi`, `Skew.Gi`, and `Kurt.Gi`. Default `FALSE`.

`p.method` Character; permutation pseudo p-value method, `"count"` (default) or `"rank"`. `"count"` counts permutations at least as extreme as the observed value; `"rank"` uses **spdep**'s ties-averaged rank of the observed value. Both return a folded (smaller-tail) value.

Details

Inference uses a conditional permutation test: the focal value x_i is held fixed while the neighbouring values are randomly permuted `nsim` times. The pseudo p-value is folded (two-tailed),

$$p_i = \frac{\min(g,\ \mathrm{nsim} - g) + 1}{\mathrm{nsim} + 1},$$

where g is the number of permutations with $G_i^{\mathrm{perm}} \geq G_i^{\mathrm{obs}}$. The standardised score is $Z.G_i = (G_i - E_{\mathrm{perm}})/\sqrt{\mathrm{Var}_{\mathrm{perm}}}$, computed from the permutation mean and variance; `Skew.Gi` and `Kurt.Gi` (when `moments = TRUE`) follow the **e1071** type-3 convention.

Observations with a missing `x` value are labelled `Undefined` and observations with no neighbours are labelled `Isolated`; both receive `NA` for the p-value, Z-score and moments. The C backend re-seeds its random number generator per observation, so results are identical for any `n.cores`; `n.cores` is ignored when the package is built without OpenMP.

Value

A numeric matrix of class `c("localG", "matrix", "array")` with columns `Gi`, `Z.Gi`, and `Pr(folded) Sim`. When `moments = TRUE`, the permutation-moment columns are appended. It has the following attributes:

cluster A significance-filtered factor with levels `Not significant`, `High-High`, `Low-Low`, `Undefined` and `Isolated`.

gstari Logical flag set to `FALSE` indicating local G (not G*).

call The matched call.

References

Getis, A. and Ord, J. K. (1992) The Analysis of Spatial Association by Use of Distance Statistics. *Geographical Analysis* **24**(3), 189–206. doi:10.1111/j.15384632.1992.tb00261.x

Ord, J. K. and Getis, A. (1995) Local Spatial Autocorrelation Statistics: Distributional Issues and an Application. *Geographical Analysis* **27**(4), 286–306. doi:10.1111/j.15384632.1995.tb00912.x

Examples

```
lw <- spdep::nb2listw(spdep::cell2nb(7, 7))
x  <- as.numeric(seq_len(49))
res <- local_g(x, lw, nsim = 99L, n.cores = 1L)
head(res)
```

`local_geary` *Univariate Local Geary's C*

Description

`local_geary` computes the univariate local Geary's C_i, a squared-difference Local Indicator of Spatial Association that measures how much a unit differs from its neighbours. On the sample $(n-1)$ standardised variable z (when `scale = TRUE`) with row-standardised weights w^*_{ij},

$$C_i = \sum_j w^*_{ij}(z_i - z_j)^2 = z_i^2 - 2z_i \lag(z)_i + \lag(z^2)_i.$$

A small C_i means i resembles its neighbours (positive spatial association); a large C_i means it differs from them (negative association, a spatial outlier).

Usage

```
local_geary(
  x,
  listw,
  nsim = 999L,
  scale = TRUE,
  iseed = NULL,
  p.value = 0.05,
  n.cores = 1L,
  moments = FALSE,
  p.method = c("count", "rank")
)
```

Arguments

`x`	Numeric vector of length `n`.
`listw`	A `listw` object from **spdep**.
`nsim`	Integer; number of permutations. Default `999L`. Must be at least 1.
`scale`	Logical; if `TRUE` (default), standardise data in R.
`iseed`	Integer seed for RNG, or `NULL`.
`p.value`	Numeric significance cutoff. Default `0.05`.
`n.cores`	Integer; number of OpenMP threads. Default `1L`; set higher to use multiple cores. Ignored on platforms without OpenMP.
`moments`	Logical; if `TRUE`, append permutation-distribution moments `E.Ci`, `Var.Ci`, `Skew.Ci`, and `Kurt.Ci`. Default `FALSE`.
`p.method`	Character; permutation pseudo p-value method, `"count"` (default) or `"rank"`. `"count"` counts permutations at least as extreme as the observed value; `"rank"` uses **spdep**'s ties-averaged rank of the observed value. Both return a folded (smaller-tail) value.

Details

Inference uses a one-tailed conditional permutation test (`nsim` reps): the observed C_i is compared with the permutation mean to select the tail, and

$$p_i = \frac{t+1}{\text{nsim}+1},$$

where t counts permuted statistics in that tail. The standardised score is $Z.C_i = (C_i - E_{\text{perm}})/\sqrt{\text{Var}_{\text{perm}}}$; `Skew.Ci`/`Kurt.Ci` (when `moments = TRUE`) follow the **e1071** type-3 convention. The `cluster` factor splits significant positive association into `High-High`/`Low-Low`/`Other Positive` and labels significant dissimilarity `Negative`.

Observations with a missing `x` value are labelled `Undefined` and observations with no neighbours are labelled `Isolated`; both receive `NA` for the p-value, Z-score and moments. The C backend re-seeds its random number generator per observation, so results are identical for any `n.cores`; `n.cores` is ignored when the package is built without OpenMP.

Value

A numeric matrix of class `c("localC", "matrix", "array")` with columns `Ci`, `Z.Ci`, and `Pr Sim`. When `moments = TRUE`, the permutation-moment columns are appended. It has the following attributes:

cluster A significance-filtered factor with levels Not significant, High-High, Low-Low, Other Positive, Negative, Undefined, and Isolated.

call The matched call.

References

Anselin, L. (1995) Local Indicators of Spatial Association—LISA. *Geographical Analysis* **27**(2), 93–115. doi:10.1111/j.15384632.1995.tb00338.x

Examples

```
lw <- spdep::nb2listw(spdep::cell2nb(7, 7))
x  <- as.numeric(seq_len(49))
res <- local_geary(x, lw, nsim = 99L, n.cores = 1L)
head(res)
```

`local_gstar` *Local Getis-Ord G**

Description

`local_gstar` computes the Getis-Ord local G_i^* statistic, the self-inclusive companion of `local_g`: observation i is treated as its own neighbour (weight 1). With m_i valid neighbours, row-standardised neighbour weights w_{ij}, and global total $S = \sum_k x_k$,

$$G_i^* = \frac{\left(\frac{\sum_{j \in N_i} w_{ij} x_j}{\sum_j w_{ij}}\right) m_i + x_i}{(m_i + 1)\, S},$$

i.e. the average value over the focal unit and its neighbours divided by the global sum. Large G_i^* flags a hot spot and small G_i^* a cold spot, with the focal unit included.

Usage

```
local_gstar(
  x,
  listw,
  nsim = 999L,
  iseed = NULL,
  p.value = 0.05,
  n.cores = 1L,
  moments = FALSE,
  p.method = c("count", "rank")
)
```

Arguments

`x`	Numeric vector of length `n`.
`listw`	A `listw` object from **spdep**.
`nsim`	Integer; number of permutations. Default `999L`. Must be at least 1.
`iseed`	Integer seed for RNG, or `NULL`.
`p.value`	Numeric significance cutoff. Default `0.05`.
`n.cores`	Integer; number of OpenMP threads. Default `1L`; set higher to use multiple cores. Ignored on platforms without OpenMP.
`moments`	Logical; if `TRUE`, append permutation-distribution moments `E.G*i`, `Var.G*i`, `Skew.G*i`, and `Kurt.G*i`. Default `FALSE`.
`p.method`	Character; permutation pseudo p-value method, `"count"` (default) or `"rank"`. `"count"` counts permutations at least as extreme as the observed value; `"rank"` uses **spdep**'s ties-averaged rank of the observed value. Both return a folded (smaller-tail) value.

Details

Inference uses a conditional permutation test (`nsim` reps), with the focal x_i held fixed while neighbour values are permuted. The pseudo p-value is folded (two-tailed),

$$p_i = \frac{\min(g,\ \mathrm{nsim} - g) + 1}{\mathrm{nsim} + 1},$$

where g counts permutations with $G_i^{*\,\mathrm{perm}} \geq G_i^{*\,\mathrm{obs}}$. The standardised score $Z.G_i^* = (G_i^* - E_{\mathrm{perm}})/\sqrt{\mathrm{Var}_{\mathrm{perm}}}$ uses the permutation moments; `Skew.G*i`/`Kurt.G*i` (when `moments = TRUE`) follow the **e1071** type-3 convention.

Observations with a missing `x` value are labelled `Undefined` and observations with no neighbours are labelled `Isolated`; both receive `NA` for the p-value, Z-score and moments. The C backend re-seeds its random number generator per observation, so results are identical for any `n.cores`; `n.cores` is ignored when the package is built without OpenMP.

Value

A numeric matrix of class `c("localG", "matrix", "array")` with columns `G*i`, `Z.G*i`, and `Pr(folded) Sim`. When `moments = TRUE`, the permutation-moment columns are appended. It has the following attributes:

cluster A significance-filtered factor with levels `Not significant`, `High-High`, `Low-Low`, `Undefined` and `Isolated`.

gstari Logical flag set to `TRUE` indicating local G*.

call The matched call.

References

Getis, A. and Ord, J. K. (1992) The Analysis of Spatial Association by Use of Distance Statistics. *Geographical Analysis* **24**(3), 189–206. doi:10.1111/j.15384632.1992.tb00261.x

Ord, J. K. and Getis, A. (1995) Local Spatial Autocorrelation Statistics: Distributional Issues and an Application. *Geographical Analysis* **27**(4), 286–306. doi:10.1111/j.15384632.1995.tb00912.x

Examples

```
lw <- spdep::nb2listw(spdep::cell2nb(7, 7))
x  <- as.numeric(seq_len(49))
res <- local_gstar(x, lw, nsim = 99L, n.cores = 1L)
head(res)
```

`local_moran` *Univariate Local Moran's I*

Description

`local_moran` computes the univariate local Moran's I_i, the classic Local Indicator of Spatial Association measuring whether a value coincides with its neighbours' average. On the sample $(n-1)$ standardised variable z with row-standardised weights w_{ij}^*,

$$I_i = z_i \sum_j w_{ij}^* z_j = z_i \,\mathrm{lag}(z)_i.$$

A positive I_i indicates similarity to neighbours (a High-High or Low-Low cluster); a negative I_i indicates a spatial outlier (High-Low or Low-High). Standardisation uses the sample standard deviation ($n-1$ denominator).

Usage

```
local_moran(
  x,
  listw,
  nsim = 999L,
  iseed = NULL,
  p.value = 0.05,
  n.cores = 1L,
  moments = FALSE,
  p.method = c("count", "rank")
)
```

Arguments

x	Numeric vector of length n.
listw	A listw object from **spdep**.
nsim	Integer; number of permutations. Default 999L. Must be at least 1.
iseed	Integer seed for RNG, or NULL.
p.value	Numeric significance cutoff. Default 0.05.
n.cores	Integer; number of OpenMP threads. Default 1L; set higher to use multiple cores. Ignored on platforms without OpenMP.
moments	Logical; if TRUE, append the permutation-distribution moments E.Ii, Var.Ii, Skew.Ii, and Kurt.Ii to the result matrix. Default FALSE.
p.method	Character; permutation pseudo p-value method, "count" (default) or "rank". "count" counts permutations at least as extreme as the observed value; "rank" uses **spdep**'s ties-averaged rank of the observed value. Both return a folded (smaller-tail) value.

Details

This is the $x = y$ special case of bivariate local Moran's I, so the computation delegates to local_moran_bv (with scale = TRUE) and relabels the columns. Inference uses a conditional permutation test (nsim reps) with the folded two-tailed pseudo p-value $p_i = (\min(g, \mathrm{nsim} - g) + 1)/(\mathrm{nsim} + 1)$, where g is the number of permutations with $I^{\mathrm{perm}} \geq I^{\mathrm{obs}}$. The standardised score is $Z.I_i = (I_i - E_{\mathrm{perm}})/\sqrt{\mathrm{Var}_{\mathrm{perm}}}$ and Skew.Ii/Kurt.Ii (when moments = TRUE) use the **e1071** type-3 convention. NA observations are Undefined and neighbourless observations are Isolated (both NA in p-value, Z-score and moments). The C backend re-seeds its random number generator per observation, so results are identical for any n.cores.

Value

A numeric matrix of class c("localmoran", "matrix", "array") with n rows and 3 columns by default:

Ii Observed univariate Moran statistic.

Z.Ii Standardised Z-score computed from permutation moments.

Pr(folded) Sim rgeoda-style folded empirical permutation p-value.

When `moments = TRUE`, the permutation-distribution columns `E.Ii`, `Var.Ii`, `Skew.Ii`, and `Kurt.Ii` are appended. The matrix has the following attributes:

quadr Moran scatter-plot quadrant classification.

cluster A significance-filtered factor with levels `Not significant`, `High-High`, `Low-Low`, `Low-High`, `High-Low`, `Undefined` and `Isolated`.

References

Anselin, L. (1995) Local Indicators of Spatial Association—LISA. *Geographical Analysis* **27**(2), 93–115. doi:10.1111/j.15384632.1995.tb00338.x

Examples

```
lw <- spdep::nb2listw(spdep::cell2nb(7, 7))
x  <- as.numeric(seq_len(49))
res <- local_moran(x, lw, nsim = 99L, n.cores = 1L, moments = TRUE)
head(res)
```

`local_moran_bv` *Bivariate Local Moran's I*

Description

`local_moran_bv` computes the bivariate local Moran's $I_{bv,i}$, which correlates a variable x at i with the spatial lag of a second variable y over i's neighbours. On the sample $(n-1)$ standardised variables z_x, z_y (when `scale = TRUE`) with row-standardised weights w^*_{ij},

$$I_{bv,i} = z_{x,i} \sum_j w^*_{ij} z_{y,j} = z_{x,i} \operatorname{lag}(z_y)_i.$$

A positive $I_{bv,i}$ means i's x value coincides with high lagged y nearby; a negative value indicates spatial mismatch. The univariate local Moran's I is the special case $x = y$ (see `local_moran`). The backend is plain C with optional OpenMP parallelism.

Usage

```
local_moran_bv(
  x,
  y,
  listw,
  nsim = 999L,
  scale = TRUE,
  iseed = NULL,
  p.value = 0.05,
  n.cores = 1L,
```

```
  moments = FALSE,
  p.method = c("count", "rank")
)
```

Arguments

x	Numeric vector of length n; the first variable.
y	Numeric vector of length n; the second variable (lagged).
listw	A listw object from **spdep** (any style: "W", "B", "C", etc., including custom distance-decay weights). Observations with no neighbours receive cluster code 6 (Isolated).
nsim	Integer; number of permutations for the pseudo p-value. Default 999L.
scale	Logical; if TRUE (default), x and y are standardised in R (sample std dev). Set to FALSE only if you have pre-standardised the data.
iseed	Integer seed for the RNG, or NULL (default) to use the package default (123456789). Passed as the seed argument to the C backend.
p.value	Numeric; observations with $p >$ p.value are recoded to cluster 0 (Not significant). Default 0.05.
n.cores	Integer; number of OpenMP threads. Default 1L; set higher to use multiple cores. Ignored on platforms without OpenMP.
moments	Logical; if TRUE, append the permutation-distribution moments E.Ibvi, Var.Ibvi, Skew.Ibvi, and Kurt.Ibvi to the result matrix. Default FALSE.
p.method	Character; permutation pseudo p-value method, "count" (default) or "rank". "count" counts permutations at least as extreme as the observed value; "rank" uses **spdep**'s ties-averaged rank of the observed value. Both return a folded (smaller-tail) value.

Details

Inference uses a conditional permutation test (nsim reps): the neighbour y-values are permuted with the focal observation held fixed. See the *P-values* and *Cluster codes* sections below for the folded p-value and the cluster coding. The standardised score is $Z.I_{bv,i} = (I_{bv,i} - E_{\text{perm}})/\sqrt{\text{Var}_{\text{perm}}}$; Skew.Ibvi/Kurt.Ibvi (when moments = TRUE) follow the **e1071** type-3 convention. NA observations are Undefined and neighbourless observations are Isolated (both NA in p-value, Z-score and moments). The C backend re-seeds its random number generator per observation, so results are identical for any n.cores.

Value

A numeric matrix of class c("localmoran", "matrix", "array") with n rows and 3 columns by default:

Ibvi Observed bivariate Moran statistic.

Z.Ibvi Standardised Z-score computed from permutation moments.

Pr(folded) Sim rgeoda-style folded empirical permutation p-value.

When `moments = TRUE`, the permutation-distribution columns `E.Ibvi`, `Var.Ibvi`, `Skew.Ibvi`, and `Kurt.Ibvi` are appended. The matrix has the following attributes:

quadr A `data.frame` with three factor columns (`mean`, `median`, `pysal`) giving the Moran scatterplot quadrant for each observation, computed on the original (unscaled) data scale.

cluster A `factor` representing cluster classification (Not significant, High-High, Low-Low, Low-High, High-Low, Undefined, Isolated).

call The matched call.

Standardisation

Both `x` and `y` are standardised using the **sample** standard deviation ($n - 1$ denominator) in R before computing the statistic, consistent with the blisa backend.

P-values

Permutation p-values use the folded two-tailed formula matching rgeoda:

$$p = (\min(\#\{perm \geq obs\},\ \#\{perm < obs\}) + 1) \,/\, (nsim + 1)$$

No normal approximation is computed.

Cluster codes

The returned `cluster` factor attribute is based on integer codes 0–6:

0	Not significant
1	High-High
2	Low-Low
3	Low-High
4	High-Low
5	Undefined (NA input)
6	Isolated (no neighbours)

Codes 5 and 6 are preserved regardless of the significance cutoff.

References

Anselin, L. (1995) Local Indicators of Spatial Association—LISA. *Geographical Analysis* **27**(2), 93–115. doi:10.1111/j.15384632.1995.tb00338.x

Examples

```
lw  <- spdep::nb2listw(spdep::cell2nb(7, 7))
x   <- as.numeric(seq_len(49))
y   <- rev(x)
res <- local_moran_bv(x, y, lw, nsim = 99L, n.cores = 1L, moments = TRUE)
head(res)
attr(res, "quadr")
```

`local_moran_eb` *Local Moran's I with Empirical Bayes (EB) Rate*

Description

`local_moran_eb` computes local Moran's I on Empirical Bayes (EB) variance-stabilised rates, for event-count data observed over a population at risk. Raw rates $p_i = \text{event}_i/\text{base}_i$ from small populations are noisy; EB standardisation shrinks them toward the global rate $b = \sum \text{event} / \sum \text{base}$ using a variance component $\hat{a}$,

$$z_i = \frac{p_i - b}{\sqrt{\hat{a} + b/\text{base}_i}},$$

and local Moran's I is then computed on z. This follows the GeoDa/libgeoda EBLocalMoran formulation.

Usage

```
local_moran_eb(
  event,
  base,
  listw,
  nsim = 999L,
  iseed = NULL,
  p.value = 0.05,
  n.cores = 1L,
  moments = FALSE,
  p.method = c("count", "rank")
)
```

Arguments

`event`	Numeric vector of events (e.g. case counts).
`base`	Numeric vector of populations at risk.
`listw`	A `listw` object from **spdep**.
`nsim`	Integer; number of permutations. Default `999L`. Must be at least 1.
`iseed`	Integer seed for RNG, or `NULL`.
`p.value`	Numeric significance cutoff. Default `0.05`.
`n.cores`	Integer; number of OpenMP threads. Default `1L`; set higher to use multiple cores. Ignored on platforms without OpenMP.
`moments`	Logical; if `TRUE`, append the permutation-distribution moments `E.Ii`, `Var.Ii`, `Skew.Ii`, and `Kurt.Ii` to the result matrix. Default `FALSE`.
`p.method`	Character; permutation pseudo p-value method, `"count"` (default) or `"rank"`. `"count"` counts permutations at least as extreme as the observed value; `"rank"` uses **spdep**'s ties-averaged rank of the observed value. Both return a folded (smaller-tail) value.

Details

Two standardisations are applied and are not redundant: the EB rate standardisation above stabilises the rate variance, and the usual sample $(n-1)$ z-score standardisation that local Moran's I requires is then applied to the EB rates internally. Because univariate Moran's I on a standardised variable equals the bivariate statistic with $x = y$, the permutation engine is reused via `local_moran_bv`. Inference is a conditional permutation test (`nsim` reps); the folded two-tailed pseudo p-value and the score $Z.I_i = (I_i - E_{\text{perm}})/\sqrt{\text{Var}_{\text{perm}}}$ are as in `local_moran`, and `Skew.Ii`/`Kurt.Ii` (when `moments = TRUE`) use the **e1071** type-3 convention.

Observations with NA `event`/`base` or `base` ≤ 0 are labelled `Undefined`; observations with no neighbours are `Isolated`; both receive `NA` for the p-value, Z-score and moments. The C backend re-seeds its random number generator per observation, so results are identical for any `n.cores`.

Value

A numeric matrix of class `c("local_moran_eb", "matrix")` with `n` rows and 3 columns by default:

Ii Observed Local Moran statistic computed on EB-standardised rates.

Z.Ii Standardised Z-score computed from permutation moments.

Pr(folded) Sim Folded empirical permutation p-value.

When `moments = TRUE`, the permutation-distribution columns `E.Ii`, `Var.Ii`, `Skew.Ii`, and `Kurt.Ii` are appended. The matrix has the following attributes:

quadr Moran scatter-plot quadrant classification.

cluster A significance-filtered factor with levels `Not significant`, `High-High`, `Low-Low`, `Low-High`, `High-Low`, `Undefined` and `Isolated`.

call The matched call.

nsim Number of simulations used.

References

Assunção, R. M. and Reis, E. A. (1999) A new proposal to adjust Moran's I for population density. *Statistics in Medicine* **18**(16), 2147–2162. doi:10.1002/(SICI)10970258(19990830)18:16<2147::AID-SIM179>3.0.CO;2I

Anselin, L. (1995) Local Indicators of Spatial Association—LISA. *Geographical Analysis* **27**(2), 93–115. doi:10.1111/j.15384632.1995.tb00338.x

Examples

```
lw    <- spdep::nb2listw(spdep::cell2nb(7, 7))
event <- as.numeric(seq_len(49))
base  <- rep(100, 49)
res <- local_moran_eb(event, base, lw, nsim = 99L, n.cores = 1L)
head(res)
```

`local_multigeary` *Multivariate Local Geary's C*

Description

`local_multigeary` computes the multivariate local Geary's C_i (Anselin 2019), the average across K variables of the univariate squared-difference statistic. On the sample $(n-1)$ standardised variables $z^1, \ldots, z^K$ (when `scale = TRUE`) with row-standardised weights w^*_{ij},

$$C_i = \frac{1}{K} \sum_{v=1}^{K} \sum_{j} w^*_{ij} (z^v_i - z^v_j)^2.$$

A small C_i indicates that i is similar to its neighbours across all variables (positive association); a large C_i indicates multivariate dissimilarity (a spatial outlier).

Usage

```
local_multigeary(
  df,
  listw,
  nsim = 999L,
  scale = TRUE,
  iseed = NULL,
  p.value = 0.05,
  n.cores = 1L,
  moments = FALSE,
  p.method = c("count", "rank")
)
```

Arguments

`df`	A data.frame or matrix with selected variables.
`listw`	A `listw` object from **spdep**.
`nsim`	Integer; number of permutations. Default `999L`. Must be at least 1.
`scale`	Logical; if `TRUE` (default), standardise each variable in R.
`iseed`	Integer seed for RNG, or `NULL`.
`p.value`	Numeric significance cutoff. Default `0.05`.
`n.cores`	Integer; number of OpenMP threads. Default `1L`; set higher to use multiple cores. Ignored on platforms without OpenMP.
`moments`	Logical; if `TRUE`, append permutation-distribution moments `E.Ci`, `Var.Ci`, `Skew.Ci`, and `Kurt.Ci`. Default `FALSE`.
`p.method`	Character; permutation pseudo p-value method, `"count"` (default) or `"rank"`. `"count"` counts permutations at least as extreme as the observed value; `"rank"` uses **spdep**'s ties-averaged rank of the observed value. Both return a folded (smaller-tail) value.

Details

Inference uses a one-tailed conditional permutation test (`nsim` reps); each replicate applies the same permuted neighbour configuration to every variable. The observed C_i is compared with the permutation mean to choose the tail, and $p_i = (t + 1)/(\mathrm{nsim} + 1)$, where t counts permuted statistics in that tail. The standardised score is $Z.C_i = (C_i - E_{\mathrm{perm}})/\sqrt{\mathrm{Var}_{\mathrm{perm}}}$; `Skew.Ci`/`Kurt.Ci` (when `moments = TRUE`) follow the **e1071** type-3 convention. Significant units are labelled `Positive` (similar) or `Negative` (dissimilar).

Rows with any missing value are labelled `Undefined` and observations with no neighbours are labelled `Isolated`; both receive `NA` for the p-value, Z-score and moments. The C backend re-seeds its random number generator per observation, so results are identical for any `n.cores`; `n.cores` is ignored when the package is built without OpenMP.

Value

A numeric matrix of class `c("localC", "matrix", "array")` with columns `Ci`, `Z.Ci`, and `Pr Sim`. When `moments = TRUE`, the permutation-moment columns are appended. It has the following attributes:

cluster A significance-filtered factor with levels Not significant, Positive, Negative, Undefined, and Isolated.

call The matched call.

References

Anselin, L. (2019) A Local Indicator of Multivariate Spatial Association: Extending Geary's c. *Geographical Analysis* **51**(2), 133–150. doi:10.1111/gean.12164

Examples

```
lw <- spdep::nb2listw(spdep::cell2nb(7, 7))
x  <- as.numeric(seq_len(49))
df <- cbind(x, rev(x))
res <- local_multigeary(df, lw, nsim = 99L, n.cores = 1L)
head(res)
```

Index

local_g, 2, *6*
local_geary, 4
local_gstar, *2*, 5
local_moran, 7, *9*, *13*
local_moran_bv, *8*, 9, *13*
local_moran_eb, 12
local_multigeary, 14