

Package ‘spatialsegregation’

April 18, 2017

Type Package

Title Segregation Measures for Multitype Spatial Point Patterns

Version 2.44

Date 2017-04-18

Author Tuomas Rajala

Maintainer Tuomas Rajala <tuomas.rajala@iki.fi>

Description Summaries for measuring segregation/mingling in multitype spatial point patterns with graph based neighbourhood description. Included indices: Mingling, Shannon, Simpson (also the non-spatial) Included functionals: Mingling, Shannon, Simpson, ISAR, MCI. Included neighbourhoods: Geometric, k-nearest neighbours, Gabriel, Delaunay. Dixon's test.

License GPL

Depends spatstat

RoxygenNote 6.0.1

NeedsCompilation yes

Repository CRAN

Date/Publication 2017-04-18 16:06:17 UTC

R topics documented:

spatialsegregation-package	2
biomassF	3
dixon	4
exposurepps	4
helper functions	5
isarF	6
mciF	7
minglingF	8
print.segtest	9
segregationFun	9
sg.modify.pp	11
shannonF	11
simpsonF	12

spatialsegregation-package

Spatial Segregation Measures

Description

Collection of measures or summaries of spatial multitype exposure: segregation vs. mingling of different types of points in a plane.

Details

This is a collection of summaries for multitype spatial point patterns (see package [spatstat](#) for more).

The package is developed for an article Rajala&Illian 2010, and provides summaries for detecting simple inter-type effects in the pattern.

See the help of the functions for further information.

Package provides an example dataset object called `exposurepps`, documented separately.

Also, the Dixon bivariate test based on contingency tables is available.

Functions

<code>segregationFun</code>	-	General calculation function, please use one of the following wrappers:
<code>minglingF</code>	-	Mingling index
<code>shannonF</code>	-	Spatial Shannon index
<code>simpsonF</code>	-	Spatial Simpson index
<code>isarF</code>	-	ISAR function
<code>mingling.index</code>	-	Shortcut for a single value
<code>shannon.index</code>	-	'''
<code>simpson.index</code>	-	'''
<code>isar.index</code>	-	'''
<code>dixon</code>	-	Dixon's 2-type contingency table tests

Author(s)

Tuomas Rajala tuomas.rajala@iki.fi

References

Graz: The behaviour of the species mingling index m_{sp} in relation to species dominance and dispersion. Eur. J. forest research. 123:87-92, 2004.

Lewandowski, Pommerening: Zur Beschreibung der Waldstruktur - Erwartete und beobachtete Arten-Durchmischung. Forstwiss Centralbl, 116:129-139, 1997.

Rajala, Illian: A family of spatial biodiversity measures based on graphs, Env. Ecol. Stat. 2012

Reardon, O'sullivan: Measures of spatial segregation. *Sociological methodology*, 34:121-162, 2004.

Shimatani, Kubota: Quantitative assesment of multispecies spatial pattern with high species diversity. *Ecological Research*, 19, 2004.

Wiegand, Gunatilleke, Gunatilleke, Huth: How individual species structure diversity in tropical forests. *PNAS*, nov 16, 2007.

Examples

```
data(exposurepps)
help(exposurepps)
```

biomassF	<i>Local biomass summary</i>
----------	------------------------------

Description

Compute the biomass around points as a function of spatial scale.

Usage

```
biomassF(X, r = NULL, target = NULL, v2 = FALSE, ...)
```

Arguments

X	Multitype point pattern of class <code>ppp</code> (see package <code>'spatstat'</code>). The biomass (e.g. size) is to be in an element <code>\$mass</code> .
r	Vector of sizes for neighbourhoods, e.g. ranges in the geometric graph neighbourhoods.
target	Default NULL. Calculate only for target type. If NULL compute mean over all types.
v2	Logical. Return the average biomass instead of just sum (development nomenclature)
...	Further parameters for the function segregationFun .

Value

Returns an `fv`-object, see `spatstat` for more information.

dixon

Dixon's 2-type contingency table tests

Description

Dixon's 2-type contingency table tests

Usage

```
dixon(X, prepR = 0)
```

Arguments

X	bivariate ppp
prepR	for large patterns, makes it a bit easier to find nearest neighbours.

Details

Computes the tests of segregation using nearest neighbour contingency tables introduced by Philip Dixon in his paper "Testing spatial segregation using a nearest-neighbor contingency table", *Ecology*, 75, p.1940-1948 (1994). The tests are an improvement on the Pielou's test of segregation. The test is defined only for two-type spatial pattern.

References

Philip Dixon: "Testing spatial segregation using a nearest-neighbor contingency table", *Ecology*, 75, p.1940-1948 (1994).

exposurepps*Example datasets for package spatalsegregation*

Description

Example datasets with 9 different scattering+exposure combinations.

Usage

```
data(exposurepps)
```

Format

A list with nine elements of class ppp.

Details

A list of 9 point patterns with different degrees of intra-species clustering and inter-species mingling.

The patterns are synthetically produced using a combination of the functions of this package as an energy function of a Gibbs model.

Author(s)

Tuomas Rajala tuomas.rajala@iki.fi

Examples

```
data(exposurepps)
par(mfrow=c(3,3), mar=c(2,2,2,2))
for(i in 1:9)plot(exposurepps[[i]])

# upper row has strong inter-type mingling effect
# lower row has strong inter-type repulsion or segregation
# left column has strong intra-type clustering
# right column has strong intra-type repulsion.

par(mfrow=c(3,3), mar=c(3,3,4,3))
for(i in 1:9)plot( isarF(exposurepps[[i]]), cbind(ISARmean,theo)~par)
```

helper functions

Functions for the aid of segregation measures

Description

Small functions included in package `spatialsegregation`, used for manipulation of forest datasets which have dbh-values (pp with an element `$dbh`).

Usage

```
clean.up.data(pp, dbh = 10, atleast = 10)
freqs(pp)
minusID(pp, minusR, dbh, atleast=0)
shake(pp, a = 0.001)
```

Arguments

pp	Multitype point pattern (see package 'spatstat')
atleast	Include specii with abundance atleast atleast.
dbh	Include only those points with dbh atleast dbh.
minusR	Range from the border withing which to exluce points (used for correction of estimates).
a	Size of displacement: $x+\text{Unif}(-a,a)$, $y+\text{Unif}(-a,a)$.

Details

Small functions to manipulate multitype point patterns.

`clean.up.data`: Returns a subsample fullfilling the given constrains.

`freqs`: Returns the abundance vector.

`minusID`: Returns a 0-1-vector indicating inclusion in a simple minus-correction.

`shake`: Shakes the pattern, i.e. adds a random displacement shift to each point.

Author(s)

Tuomas Rajala tuomas.rajala@iki.fi

 isarF

Individual Species Area Relationship

Description

Compute the Individual Species Area Relationship (ISAR) or Local Species Richness, for a given multitype point pattern.

Usage

```
isarF(X, r = NULL, target = NULL, v2 = FALSE, v3 = FALSE, v4 = FALSE,
      ...)
```

```
isar.index(X, r = 4, ntype = "knn", ...)
```

Arguments

<code>X</code>	Multitype point pattern of class <code>ppp</code> (see package 'spatstat')
<code>r</code>	Vector of sizes for neighbourhoods, e.g. geometric graph with different ranges.
<code>target</code>	Default NULL. Calculate only for target type. If NULL computes for each type + mean over all types.
<code>v2</code>	Logical. Estimate species-to-neighbours-ratio instead of just total number of species.
<code>v3</code>	Logical. Instead of summing number 1 for each species present, sum the average <code>X\$mass</code> of each species present.
<code>v4</code>	Logical. Estimate ISAR using empty space probabilities instead of direct counts (equals the normal version in all my tests)
<code>...</code>	Further parameters for the function <code>segregationFun</code> .
<code>ntype</code>	Sets the n'hood type to <code>knn</code> by default in <code>isar.index</code> .

Details

Extension of ISAR-function introduced in WGGH07. In effect calculates the expected amount of different types present in the neighbourhood of a point in the pattern.

The function `isarF` is the calculation function for different neighbourhoods. Uses function `segregationFun`.

The function `isar.index` is a shortcut to get a single value for the pattern. Uses 4-nn graph by default.

Functions

- `isar.index`: Shortcut for 4-nearest neighbour value.

References

Rajala, Illian: A family of spatial biodiversity measures based on graphs, *Env. Ecol. Stat.* 2012

Wiegand, Gunatilleke, Gunatilleke, Huth: How individual species structure diversity in tropical forests. *PNAS*, nov 16, 2007.

 mciF

Mean Composite Information

Description

Compute the Mean Composite Information for a given multitype point pattern. See Podani&Czaran 1997.

Usage

```
mciF(X, r = NULL, target = NULL, ...)
```

Arguments

<code>X</code>	Multitype point pattern of class <code>ppp</code> (see package 'spatstat')
<code>r</code>	Vector of sizes for neighbourhoods, e.g. <code>geometric</code> graph with different ranges.
<code>target</code>	If given, look at the surroundings of this type only.
<code>...</code>	Further parameters for the function <code>segregationFun</code> .

Value

Returns an `fv`-object, see `spatstat` for more information.

References

Podani, Czaran: Individual-centered analysis of mapped point patterns representing multi-species assemblages. *J. Veg. Sci.* 8: 259-270, 1997.

minglingF *Spatial Mingling index*

Description

Compute the Mingling index for a given multitype point pattern.

Usage

```
minglingF(X, r = NULL, target = NULL, ratio = FALSE, ...)
```

```
mingling.index(X, r = 4, ntype = "knn", ...)
```

Arguments

X	Multitype point pattern of class ppp (see package 'spatstat')
r	Vector of sizes for neighbourhoods, e.g. geometric graph with different ranges.
target	Default NULL. Calculate only for target type. If NULL computes for each type + mean over all types.
ratio	Default FALSE. If TRUE, scale the typewise values M_{t} using formula $(1-M_{\tau})/\lambda_{\tau}$ which equals 1 for Poisson CSR.
...	Further parameters for the function segregationFun.
ntype	The original mingling index uses knn neighbourhood type.

Details

Extension of Mingling index introduced by Lewandowski & Pommerening 1997. Measures the proportion of alien points in the neighbourhood of a specific type typical point of the pattern.

If no specific type is given, the function takes mean over all types. A typewise value is more useful, so they are also included.

The function minglingF is the main calculation function. Uses function [segregationFun](#).

The function mingling.index is a shortcut to get a single value for the pattern. Uses 4-nn graph by default, which is the original Mingling index used by Lewandowski & Pommerening 1997 and Graz 2004.

Functions

- mingling.index: Shortcut to 4-nearest neighbours index.

References

Graz: The behaviour of the species mingling index $m_{\{sp\}}$ in relation to species dominance and dispersion. Eur. J. forest research. 123:87-92, 2004.

Lewandowski, Pommerening: Zur Beschreibung der Waldstruktur - Erwartete und beobachtete Arten-Durchmischung. Forstwiss Centralbl, 116:129-139, 1997.

Rajala, Illian: A family of spatial biodiversity measures based on graphs, Env. Ecol. Stat. 2012

print.segtest	<i>Print Segtest Objects</i>
---------------	------------------------------

Description

Print Segtest Objects

Usage

```
## S3 method for class 'segtest'
print(x, ...)
```

Arguments

x	segtest object
...	ignored

segregationFun	<i>Main computer function for spatialsegregation</i>
----------------	--

Description

Compute the spatial exposure (segregation vs. mingling) features from a given multitype point pattern. Usage of shortcuts minglingF, isarF, shannonF, simpsonF etc. highly recommended.

Usage

```
segregationFun(X, fun = "isar", r = NULL, ntype = "geometric",
  funpars = NULL, toroidal = FALSE, minusRange = TRUE, included = NULL,
  dbg = FALSE, doDists = FALSE, prepRange = 0, prepGraph = NULL,
  prepGraphIsTarget = FALSE, weightMatrix = NULL, translate = FALSE, ...)
```

Arguments

X	Multitype point pattern of class ppp (see package 'spatstat')
fun	Default "isar". Takes "isar", "mingling", "shannon", "simpson", "mci" and "biomass", see below.
r	Vector for the neighbourhood defining graph, e.g. "geometric" graph with different ranges. See below.
ntype	Default "geometric". Type of the neighbourhood graph. Accepts: "knn", "geometric", "delauney", "gabriel".
funpars	Default NULL. Parameter(s) for the measure. Mingling: c(i,j), where i= only for type i (0 for all), j=1 -> ratio version. ISAR: i, i=type (integer). Shannon: 0 or 1, see v2 in shannonF. Simpson: none.

toroidal	Default FALSE. If TRUE, use a toroidal correction in distance calculation. Works at the moment only for rectangular windows and "geometric" or "knn" graph.
minusRange	If TRUE, adaptive minus-sampling is employed. Overrides included-vector. If given as a positive number, included-vector is created with points with distance atleast minusRange from the border.
included	boolean-vector of length <code>lpp</code> . <code>included[i]==TRUE => pp[i]</code> included in calculations. Used for minus-sampling border correction.
dbg	Default FALSE. Print additional runtime texts.
doDists	Default TRUE. Precalculate distances for speed. Be aware of memory requirement $n*(n-1)/2$
prepRange	Default 0. If >0 , shrink the search space for neighbourhoods by searching only points within distance <code>R</code> i.e. precalculates a geometric graph.
prepGraph	Precalculated graph for the point pattern. If given, The <code>prepRange</code> , <code>doDists</code> and <code>toroidal</code> are ignored and calculations are carried using the <code>prepGraph</code> as a starting point. Useful for huge datasets.
prepGraphIsTarget	If TRUE, precalculated graph <code>prepGraph</code> is used to calculate a single function value directly, all other neighbourhood parameters are ignored.
weightMatrix	See <code>isarF</code> for this.
translate	Use translation correction (see e.g. documentation of <code>spatstat::Kest</code> for details). Used only in mingling index.
...	ignored.

Details

This is the general function for computing the spatial exposure (segregation/mingling) features. Used by `minglingF`, `shannonF`, `simpsonF`, `isarF`, `mciF` and `biomassF`, which should be preferred for better (and nicer) outcome.

Possible neighbourhood relations include geometric, k-nearest neighbours, Delaunay, and Gabriel. Delaunay and Gabriel are parameter free, so given `r` has no meaning. In geometric graph, `r` is a vector of distances (sizes of the surrounding 'disc') and for k-nn `r` is the vector of neighbourhood abundances (so `r` is `k` in k-nn) for each point to consider in the calculation of the spatial exposure measures. In general, the basic type of spatial summary in literature uses 'geometric' graph with several ranges.

For geometric and knn, the calculations are done by shrinking the graph given by the largest value of `r`. If dealing with large datasets, it is advisable to give preprocessing range, `prepRange`. The algorithm first calculates a geometric graph with parameter `prepRange`, and uses this as basis for finding the needed neighbourhoods. Speeds up calculations, but make sure `prepRange` is large enough e.g. in geometric, `prepRange > max(r)`. `prepGraph`, if given, works as the preprocessed geometric graph (overrides `prepRange`), and can be computed using the package `spatgraphs`; useful for huge datasets, where the dominating graph needs to be computed only one. The `doDists` option speeds up calculations by precomputing the pairwise distances but requires approx. $n*(n-1)/2 * 32$ bytes of memory.

For border correction, use `minusRange` for reduced border correction (for rectangular windows only). If using geometric or knn neighbourhoods, the option `toroidal` for toroidal correction is

also available. The vector `included` can be given for more specific minus-correction, only those points with TRUE (or 1) value are used in calculation. However, the neighbourhoods will include all points.

Value

Returns an object of class `fv`, see `spatstat` for more details. Basically a list with the computed values and parameter values.

<code>sg.modify.pp</code>	<i>Prepare the input data pattern object</i>
---------------------------	--

Description

Prepares given point pattern object for computations.

Usage

```
sg.modify.pp(pp)
```

Arguments

<code>pp</code>	Point pattern object
-----------------	----------------------

<code>shannonF</code>	<i>Spatial Shannon index</i>
-----------------------	------------------------------

Description

Compute the spatial and aspatial Shannon index for a given multitype point pattern.

Usage

```
shannonF(X, r = NULL, v2 = FALSE, ...)
```

```
shannon.index(X, spatial = FALSE, ...)
```

Arguments

<code>X</code>	Multitype point pattern of class <code>ppp</code> (see package 'spatstat')
<code>r</code>	Vector of sizes for neighbourhoods, e.g. geometric graph with different ranges.
<code>v2</code>	If TRUE, use the real number of types in neighbourhoods as the log-base instead of total population type count.
<code>...</code>	Further parameters for the function segregationFun .
<code>spatial</code>	If FALSE, return the classical aspatial index value.

Details

The form of Shannon index is $H = 1 - E(o)/E(N)$, where $E(N)$ is the global entropy and $E(o)$ is the local entropy calculated as $E(o) = - \sum p_{i_tau} \log(p_{i_tau})$, where the sum is over the different types present in the pattern, and p_{i_tau} is the expected frequency of type tau points in a neighbourhood of a typical point of the pattern.

The function `shannonF` is the calculation function. Uses function `segregationFun`.

The function `shannon.index` is a shortcut to get the non-spatial Shannon index.

Value

Returns an fv-object, see `spatstat` for more information. The index returns a scalar.

Functions

- `shannon.index`: Traditional index.

References

Rajala, Illian: A family of spatial biodiversity measures based on graphs, *Env. Ecol. Stat.* 2012

Reardon, O'sullivan: Measures of spatial segregation. *Sociological methodology*, 34:121-162, 2004.

simpsonF

Spatial Simpson index

Description

Compute the spatial and non-spatial Simpson index for a given multitype point pattern.

Usage

```
simpsonF(X, r = NULL, ...)
```

```
simpson.index(X, spatial = FALSE, ...)
```

Arguments

<code>X</code>	Multitype point pattern of class <code>ppp</code> (see package 'spatstat')
<code>r</code>	Vector of sizes for neighbourhoods, e.g. geometric graph with different ranges.
<code>...</code>	Further parameters for the function <code>segregationFun</code> .
<code>spatial</code>	If <code>FALSE</code> , return the classical aspatial index value.

Details

The form of Simpson index is $S = 1 - \sum pi_tau$, where the sum is over the types of the pattern, and pi_tau is like in Shimatani & Kubota 2004. The function `simpsonF` is the main calculation function. Uses function `segregationFun`.

The function `simpson.index` is a shortcut to get a single value for the pattern using 4-nearest neighbours graph by default.

Value

If spatial, returns an fv-object, see `spatstat` for more information. Otherwise a numeric value.

Functions

- `simpson.index`: The Spatial Simpson Index

References

- Rajala, Ilian: A family of spatial biodiversity measures based on graphs, *Env. Ecol. Stat.* 2012
- Shimatani, Kubota: Quantitative assesment of multispecies spatial pattern with high species diversity. *Ecological Research*, 19, 2004.

Index

- *Topic **package**
 - spatialsegregation-package, 2
- *Topic **spatial**
 - exposurepps, 4
 - helper functions, 5
- biomassF, 3, 10
- clean.up.data(helper functions), 5
- dixon, 4
- exposurepps, 4
- freqs(helper functions), 5
- helper functions, 5
- isar.index(isarF), 6
- isarF, 6, 10
- mciF, 7, 10
- mingling.index(minglingF), 8
- minglingF, 8, 10
- minusID(helper functions), 5
- ppp, 3
- print.segtest, 9
- segregationFun, 3, 7, 8, 9, 11–13
- sg.modify.pp, 11
- shake(helper functions), 5
- shannon.index(shannonF), 11
- shannonF, 10, 11
- simpson.index(simpsonF), 12
- simpsonF, 10, 12
- spatialsegregation
 - (spatialsegregation-package), 2
- spatialsegregation-package, 2
- spatstat, 2