Package ‘spatialsegregation’
February 20, 2015

Type Package
Title Segregation measures for multitype spatial point patterns
Version 2.40
Date 2014-03-09
Author Tuomas Rajala
Maintainer Tuomas Rajala <tuomas.rajala@iki.fi>

Suggests
Depends spatstat (>= 1.15-2)
Description Summaries for measuring segregation/mining 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.
License GPL (>= 2.0)
NeedsCompilation yes
Repository CRAN
Date/Publication 2014-03-09 19:27:16

R topics documented:

spatialsegregation-package ........................................... 2
biomassF ............................................................... 3
Dixon’s 2-type contingency table tests .............................. 4
exposurepps ............................................................ 5
helper functions ....................................................... 6
isarF ................................................................. 7
mciF ................................................................. 8
minglingF ............................................................ 9
shannonF ........................................................... 10
simpsonF .......................................................... 11
spatialsegregation-segregationFun .................................. 12

Index 14
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

________________________
segregationFun - General calculation function, please use one of the following wrappers:
minglingF - Mingling index
shannonF - Spatial Shannon index
simpsonF - Spatial Simpson index
isarF - ISAR function

mingling.index - Shortcut for a single value
shannon.index - 'sh'
 simpson.index - 'sp'
isar.index - 'isar'
dixon - Dixon's 2-type contingency table tests

Author(s)

Tuomas Rajala University of Jyvaskyla, Finland tuomas.rajala@iki.fi

References


Examples

```
data(exposureepps)
help(exposureepps)
```

---

**biomassF**

*Individual Species Area Relationship*

### Description

Compute the biomass around individuals points. WARNING: Still under development. Please contact me if you want to use this.

### Usage

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

### Arguments

- **x**: Multitype point pattern of class ppp (see package 'spatstat'). The biomass (e.g. size) is to be in an element Smass.
- **r**: Vector of sizes for neighbourhoods, e.g. geometric graph with different ranges.
- **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.
- **...**: Further parameters for the function segregationFun.

### Details

Computes the neighbourhood for each point and then sums up the biomass in that neighbourhood.

### Value

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

### Author(s)

Tuomas Rajala University of Jyvaskyla, Finland tuomas.rajala@iki.fi
Dixon’s 2-type contingency table tests

Description


The test is defined only for two-type spatial pattern.

Usage

dixon(X, prepR=∅)

Arguments

X Bivariate i.e. 2-type point pattern (see package 'spatstat')
p prepR Computes first the geometric neighbours with this distance, and then finds the nearest neighbours.

Details

See the paper by Dixon for details.

Author(s)

Tuomas Rajala University of Jyvaskyla, Finland tuomas.rajala@iki.fi

References


Example datasets for package spatialsegregation

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 University of Jyvaskyla, Finland 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

```r
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+Unif(-a,a), y+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 University of Jyvaskyla, Finland tuomas.rajala@iki.fi
Description

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

Usage

\[
\text{isarF}(X, r=NULL, target=NULL, v2=FALSE, v3=FALSE, v4=FALSE, \ldots)
\]
\[
\text{isar.index}(X, r=4, ntype="knn", \ldots)
\]

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.
- **v2**: Logical. Estimate species-to-neighbours-ratio instead of just total number of species.
- **v3**: Logical. Instead of summing number 1 for each species present, sum the average $X$\$mass of each species present.
- **v4**: Logical. Estimate ISAR using empty space probabilities instead of direct counts (equals the normal version in all my tests)
- **ntype**: Sets the n'hood type to `knn` by default in isar.index.
- **...**: Further parameters for the function segregationFun.

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.

Value

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

Author(s)

Tuomas Rajala University of Jyvaskyla, Finland tuomas.rajala@iki.fi
References


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

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

Details

The function mciF is the main calculation function. Uses function segregationFun.

Value

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

Author(s)

Tuomas Rajala University of Jyvaskyla, Finland tuomas.rajala@iki.fi

References

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 $(1-M_t)/\lambda_t$ which equals 1 for Poisson CSR.

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.

Value

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

Author(s)

Tuomas Rajala University of Jyvaskyla, Finland tuomas.rajala@iki.fi
References


shannonF

Spatial Shannon Index

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

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

Further parameters for the function segregationFun.

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 $E(o)= - \sum pi_{\tau} log(pi_{\tau})$, where the sum is over the different types present in the pattern, and $pi_{\tau}$ 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.

Author(s)

Tuomas Rajala University of Jyvaskyla, Finland tuomas.rajala@iki.fi
simpsonF

References

simpsonF  Spatial Simpson Index

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

Usage
simpsonF(X, r=NULL, ...)
simpson.index(X, spatial=FALSE, ...)

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

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. Uses 4-nn graph by default.

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

Author(s)
Tuomas Rajala University of Jyvaskyla, Finland tuomas.rajala@iki.fi

References
**spatialsegregation-segregationFun**

*Spatial Segregation Function*

**Description**

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

**Usage**

```r
segregationFun(x, fun="isar", r=NULL, ntype="geometric", funpars=NULL, toroidal=FALSE, minusRange=TRUE, included=NULL, dbg=FALSE, doDists=FALSE, prepRange=0.0, 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" and "simpson", 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 vR 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 at least minusRange from the border.
- **included**: boolean-vector of length |pp|. included[i]==TRUE => pp[i] 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 requirements, n*(n-1)!
- **prepRange**: Default 0. If >0, shrink the search space for neighbourhoods by searching only points within distance R i.e. precalculates a geometric graph.
- **prepGraph**: Precalculated graph for the point pattern. If given, The prepRange, dodists and toroidal are ignored and calculations are carried using the prepGraph as a starting point. Useful for huge datasets.
If TRUE, precalculated graph `prepGraph` is used to calculate a single function value directly, all other neighbourhood parameters are ignored.

See `isarF` for this.

Use translation correction (see e.g. documentation of `spatstat::Kest` for details). Used only in mingling index.

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

Possible neighbourhood relations for the spatial version include geometric, k-nearest neighbours, Delauney, and Gabriel. Delauney 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 for each point to consider in the calculation of the spatial exposure measures. The basic type of spatial summary uses range, or 'geometric' graph connections with varying neighbourhood parameter.

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. `prepGraph`, if given, works as the preprocessed geometric graph. But make sure `prepRange` is large enough (e.g. in geometric, `prepRange>max(r)`).

The `dodists` option speeds up calculations by precomputing the pairwise distances but takes $n*(n-1)$ 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 (1) value are used in calculation. However, the neighbourhoods are calculated with all points.

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

Tuomas Rajala University of Jyvaskyla, Finland tuomas.rajala@iki.fi
Index

*Topic package
  biomassF, 3
  isarf, 7
  mcif, 8
  minglingF, 9
  shannonF, 10
  simpsonF, 11
  spatialsegregation-package, 2
*Topic spatial
  Dixon’s 2-type contingency table tests, 4
  exposurerepps, 5
  helper functions, 6
  spatialsegregation-segregationFun, 12
  biomassF, 3
  clean.up.data (helper functions), 6
  dixon (Dixon’s 2-type contingency table tests), 4
  Dixon’s 2-type contingency table tests, 4
  exposurerepps, 5
  freqs (helper functions), 6
  helper functions, 6
  isar.index (isarf), 7
  isarf, 7
  kGraphs
    (spatialsegregation-segregationFun), 12
  mcif, 8
  mingling.index (minglingF), 9
  minglingF, 9
  minusID (helper functions), 6
  minusID.gf
    (spatialsegregation-segregationFun), 12
  print.segtest (Dixon’s 2-type contingency table tests), 4
  segregationFun, 7–11
  segregationFun
    (spatialsegregation-segregationFun), 12
  sg.modify.pp
    (spatialsegregation-segregationFun), 12
  shake (helper functions), 6
  shannon.index (shannonF), 10
  shannonF, 10
  simpson.index (simpsonF), 11
  simpsonF, 11
  spatialsegregation
    (spatialsegregation-package), 2
  spatialsegregation-package, 2
  spatialsegregation-segregationFun, 12
  spatstat, 2