Package: hotspotr (via r-universe)

May 22, 2026

Title Compare Spatial Structure of Hotspots with Neutral Models

Version 0.0.0.012

Date 2016-08-24

Description Analyses whether the statistical properties of a spatial pattern of hotspots may be reproduced with a simple neutral model.

License GPL-3

URL https://github.com/mpadge/hotspotr

Depends R (>= 3.2.4)

Imports devtools, distr, methods, msm, Rcpp (>= 0.12.4), spdep, truncnorm

Suggests testthat

Remotes mpadge/truncnorm

LinkingTo Rcpp

Encoding UTF-8

LazyData true

NeedsCompilation yes

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Repository https://mpadge.r-universe.dev

Date/Publication 2026-03-23 12:13:11 UTC

RemoteUrl https://github.com/mpadge/hotspotr

RemoteRef HEAD

RemoteSha 8859545e174c9996552bd9392d6be387054e5201

Contents

`fit_hotspot_model` *fit_hotspot_model*

Description

Fits a neutral model to an observed series of hotspot values in terms of standard deviation of environmental variables, spatial autocorrelation parameters, and number of iterations of spatial autocorrelation.

Usage

```
fit_hotspot_model(
  z,
  nbs,
  wts,
  ac_type = "moran",
  ntests = 100,
  verbose = FALSE,
  plot = FALSE
)
```

Arguments

`z`	Vector of observed values to be tested
`nbs`	An spdep nb object listing all neighbours of each point
`wts`	Weighting factors for each neighbour; must have same length as nbs. Uniform weights used if not given.
`ac_type`	type of autocorrelation statistic to use in tests (`moran`, `geary`, or `getis-org=go`)
`ntests`	Number of repeats of neutral model used to calculate mean rank–scale distribution
`verbose`	If TRUE, dump progress details to screen
`plot`	If TRUE, produces a plot of rank–scale distributions

Value

A vector of three values as estimated by the neutral model:

1. sd0 = standard deviation of normal distribution
2. alpha = temporal autocorrelation coefficient
3. niters = number of iterations of spatial autocorrelation

Note

Fitting these neutral models is **not** a standard optimisation problem because the models are very noisy. Although `optim` with `method="SANN"` may be used, it often generates extremely large values for `alpha` (for example, > 10). `DEoptim` could also be applied, yet in generally does not explore anything useful—if given starting parameters, it will generally remain exactly in that place.

The approach employed here reflects the comment of https://stat.ethz.ch/pipermail/r-help/2015-May/428751.html through simply producing regular series, fitting loess models, and taking the corresponding minima.

Examples

```
## Not run:
xy <- cbind (rep (seq (size), each=size), rep (seq (size), size))
dhi <- 1 # for rook; dhi=1.5 for queen
nbs <- spdep::dnearneigh (xy, 0, dhi)
z <- runif (length (nbs))
test <- fit_hotspot_model (z=z, nbs=dat$nbs, alpha=0.1, sd=0.1, ntests=100)

## End(Not run)
```

`generate_hotspot_model` *generate_hotspot_model*

Description

Uses the parameters returned by `fit_hotspot_model` to generate rank-scale distributions both of raw values and associated spatial autocorrelation statistics.

Usage

```
generate_hotspot_model(
  n,
  alpha = 0.1,
  sd0 = 0.1,
  ac_type = "moran",
  niters = 1,
  plot = FALSE
)
```

Arguments

n	Number of observations to generate
alpha	Strength of spatial autocorrelation
sd0	Standard deviation of truncated normal distribution used to model environmental variation (with mean of 1)
ac_type	Type of autocorrelation statistic to use in tests (moran, geary, or getis-org=go)
niters	Number of successive layers of spatial autocorrelation
plot	If TRUE, produces a plot of rank–scale distributions

Value

A matrix of two columns containing sorted and scaled versions of

1. z = raw values
2. ac = associated spatial autocorrelation statistics

---

hotspotr *hotspotr*

Description

Analyses whether the statistical properties of a spatial pattern of hotspots may be reproduced with a simple neutral model.

Functions

ives2d	Simulate model of Ives & Klopfer (Ecology 1997)
neutral2d	Neutral model in two dimensions
run_tests	Test observed data with range of possible (1-D & 2-D) models
test2d	Test observed data against a two dimensional neutral model

Author(s)

Maintainer: Mark Padgham <mark.padgham@email.com>

See Also

Useful links:

- https://github.com/mpadge/hotspotr

neutral_hotspots *neutral_hotspots*

Description

Implements neutral model of hotspot values and associated autocorrelation statustics. Current version is a simply internal R loop, while `neutral_hotspots_ntests` is a parallel version of exactly the same thing.

Usage

```
neutral_hotspots(
  nbs,
  wts,
  alpha = 0.1,
  sd0 = 0.1,
  ac_type = "moran",
  niters = 1,
  log_scale = TRUE,
  ntests = 100,
  parallel = FALSE,
  seed
)
```

Arguments

`nbs`	An spdep nb object listing all neighbours of each point
`wts`	Weighting factors for each neighbour; must have same length as nbs. Uniform weights used if not given.
`alpha`	strength of spatial autocorrelation
`sd0`	Standard deviation of truncated normal distribution used to model environmental variation (with mean of 1)
`ac_type`	Type of autocorrelation statistic to use in tests (`moran`, `geary`, or `getis-org=go`)
`niters`	Number of successive layers of spatial autocorrelation
`log_scale`	If TRUE, raw hotspot values are log-transformed
`ntests`	Number of tests over which to generate an average result
`parallel`	If true, the tests are conducted using the `R` package **parallel**, otherwise using (non-parallel) **Rcpp** loops.
`seed`	Random seed

Value

A vector of hotspot values sorted from high to low

See Also

ives

Examples

```
# First set up a grid of rectangular neighbours
size <- 10
xy <- cbind (rep (seq (size), each=size), rep (seq (size), size))
dhi <- 1 # for rook; dhi=1.5 for queen
nbs <- spdep::dnearneigh (xy, 0, dhi)
dat <- neutral_hotspots (nbs, ntests=1000)
```

---

neutral_hotspots2 *neutral_hotspots2*

Description

Implements neutral model of hotspot values and associated autocorrelation statustics. Current version is a simply internal R loop, while neutral_hotspots_ntests is a parallel version of exactly the same thing.

Usage

```
neutral_hotspots2(
  nbs,
  wts,
  alpha = 0.1,
  sd0 = 0.1,
  ac_type = "moran",
  niters = 1,
  log_scale = TRUE,
  ntests = 100,
  seed
)
```

Arguments

nbs	An spdep nb object listing all neighbours of each point
wts	Weighting factors for each neighbour; must have same length as nbs. Uniform weights used if not given.
alpha	strength of spatial autocorrelation
sd0	Standard deviation of truncated normal distribution used to model environmental variation (with mean of 1)
ac_type	Type of autocorrelation statistic to use in tests (moran, geary, or getis-org=go)
niters	Number of successive layers of spatial autocorrelation

log_scale	If TRUE, raw hotspot values are log-transformed
ntests	Number of tests over which to generate an average result
seed	Random seed

Value

A vector of hotspot values sorted from high to low

See Also

ives

Examples

```
# First set up a grid of rectangular neighbours
size <- 10
xy <- cbind (rep (seq (size), each=size), rep (seq (size), size))
dhi <- 1 # for rook; dhi=1.5 for queen
nbs <- spdep::dnearneigh (xy, 0, dhi)
dat <- neutral_hotspots (nbs, ntests=1000)
```

---

order_one *order_one*

Description

First order statistic for normal distribution.

Usage

```
order_one(sigma, n, ntrials = 1000)
```

Arguments

sigma	Standard deviation of normal distribution
n	Number of samples of normal distribution
ntrials	Number of trials over which to average order statistics

Value

First order statistic

p_values *p_values*

Description

Tests observed data (z) against a series of neutral models

Usage

```
p_values(
  z,
  nbs,
  wts,
  sd0 = 0.1,
  alpha = 0.1,
  niters = 1,
  ntests = 1000,
  ac_type = "moran",
  log_scale = FALSE,
  plot = FALSE,
  verbose = FALSE
)
```

Arguments

z	Vector of observed values to be tested
nbs	An spdep nb object listing all neighbours of each point
wts	Weighting factors for each neighbour; must have same length as nbs. Uniform weights used if not given.
sd0	Standard deviation of truncated normal distribution used to model environmental variation (with mean of 1)
alpha	Strength of spatial autocorrelation
niters	Number of successive layers of spatial autocorrelation
ntests	Number of tests to run, with statistics calculated from the mean of ntests
ac_type	type of autocorrelation statistic to use in tests (moran, geary, or getis-org=go)
log_scale	If TRUE, raw hotspot values are log-transformed
plot	If TRUE, plot mean and observed distributions of z and associated autocorrelation statistics
verbose	If TRUE, dump progress details to screen

Value

Nothing (dumps statistics to screen)

rs_dist_diff *rs_dist_diff*

Description

Averages results of a number of neutral models

Usage

```
rs_dist_diff(
  nbs,
  wts,
  alpha = 0.1,
  sd0 = 0.1,
  niters = 1,
  ntests = 1000,
  ac_type = "moran",
  log_scale = TRUE,
  mean_stats
)
```

Arguments

nbs	An spdep nb object listing all neighbours of each point
wts	Weighting factors for each neighbour; must have same length as nbs. Uniform weights used if not given.
alpha	Strength of spatial autocorrelation
sd0	Standard deviation of truncated normal distribution used to model environmental variation (with mean of 1)
niters	Number of sequential iterations of spatial autocorrelation
ntests	Number of repeats of neutral model used to calculate mean rank–scale distribution
ac_type	type of autocorrelation statistic to use in tests (moran, geary, or getis-org=go)
log_scale	If TRUE, raw hotspot values are log-transformed
mean_stats	Mean rank–scale distributions returned from neutral_hotspots: a matrix of 2 columns

Value

A vector of hotspot values sorted from high to low

See Also

ives

Index