Package ‘spatgeom’

April 26, 2023

Type Package

Title Geometric Spatial Point Analysis

Version 0.3.0

Maintainer Maikol Solís <maikol.solis@ucr.ac.cr>

Description The implementation to perform the geometric spatial point analysis developed in Hernández & Solís (2022) <doi:10.1007/s00180-022-01244-1>. It estimates the geometric goodness-of-fit index for a set of variables against a response one based on the 'sf' package. The package has methods to print and plot the results.

License MIT + file LICENSE

URL https://github.com/maikol-solis/spatgeom

BugReports https://github.com/maikol-solis/spatgeom/issues

Encoding UTF-8

Imports ggplot2, scales, sf, dplyr, lwgeom, cowplot, purrr

RoxygenNote 7.2.3

Depends R (>= 3.6.0)

Suggests rmarkdown, knitr, testthat (>= 2.1.0)

NeedsCompilation no

Author Maikol Solís [aut, cre],
Alberto Hernández [ctb],
Carlos Pasquier [ctb]

Repository CRAN

Date/Publication 2023-04-26 19:30:02 UTC

R topics documented:

`donut_data` *Donut example*

Description

Generate data points with the shape of a donut.

Usage

```
donut_data(n, a, b, theta)
```

Arguments

`n` Number of data points.

`a` Lower bound of the second variable.

`b` Upper bound of the second variable.

`theta` Angle of the donut.

Value

A data frame with three variables. Variable ’y’ is the response, variable ’x1’ makes the donut shape with ’y’, and ’x2’ is a uniform random variable between a and b. ’

Examples

```
xy <- donut_data(n = 30, a = -1, b = 1, theta = 2 * pi)
```

`linear_data` *Linear example*

Description

Generate data points with a linear relationship.

Usage

```
linear_data(n = 100, a = -3, b = 3)
```

Arguments

`n` Number of data points.

`a, b` Lower and upper bound of the uniform distribution.

Value

A data frame with three variables. Variable 'y = 0.6 * x1 + 0.3 * x2

- 0.1 * x3' is the response, and 'x1', 'x2' and 'x3' are uniform random variables between a and b.

Examples

```
xy <- linear_data(n = 30, a = -1, b = 1)
```

`plot_alpha_shape` *Plot alpha-shape for* `spatgeom` *objects*

Description

Plot alpha-shape for `spatgeom` objects.

Usage

```
plot_alpha_shape(x, alpha, font_size = 12)
```

Arguments

`x`	an object of class `spatgeom`.
`alpha`	value of `alpha` determining the maximum length between points to build the alpha-shape.
`font_size`	a integer that increases the font size in the plot.

Value

a `ggplot` object with the raw alpha-shape for the original data at resolution `alpha`

Examples

```
xy <- donut_data(n = 30, a = -1, b = 1, theta = 2 * pi)

estimation <- spatgeom(y = xy[, 1], x = xy[, -1])

plot_alpha_shape(estimation, alpha = c(0.9, 1.2))
```

plot_curve *plot* spatgeom *objects*

Description

Plot method for objects of class spatgeom.

Usage

```
plot_curve(x, type = "curve", font_size = 12)
```

Arguments

x	an object of class spatgeom
type	a string that could be curve or deriv. The option curve plots the curve of alpha against geom_corr from the function spatgeom(). The deriv option plots the numerical derivative.
font_size	a integer that increases the font size in the plot.

Value

a ggplot object with the geometric indices (or its derivative). The plot is generated with the nalphas point of alpha and geom_corr from the function spatgeom.

In each panel, the theoretical CSR process is drawn using exp(-intensity * pi * x^2). where the intensity depends on each panel.

Examples

```
xy <- donut_data(n = 30, a = -1, b = 1, theta = 2 * pi)

estimation <- spatgeom(y = xy[, 1], x = xy[, -1])

plot_curve(estimation, type = "curve")

plot_curve(estimation, type = "deriv")
```

print.spatgeom *print a* spatgeom *object*

Description

Print method for objects of class spatgeom.

Usage

```
## S3 method for class 'spatgeom'
print(x, return_table = FALSE, ...)
```

Arguments

x an object of class spatgeom

return_table if TRUE, returns a data frame with the estimated values. Otherwise, print the data frame in console. Defaults to FALSE

... further arguments passed to the plot function

Value

Print the estimate given by spatgeom.

Examples

```

xy <- donut_data(n = 30, a = -1, b = 1, theta = 2 * pi)

estimation <- spatgeom(y = xy[, 1], x = xy[, -1])

print(estimation)
```

spatgeom *Geometric Spatial Point Pattern Analysis*

Description

Function to estimate the geometric correlation between variables.

Usage

```
spatgeom(x, y, scale = FALSE, nalphas = 100, envelope = FALSE, mc_cores = 1)
```

Arguments

x	numeric matrix or data.frame of covariables.
y	numeric vector of responses in a model.
scale	boolean to make the estimations with scaled variables. Default FALSE.
nalphas	a single number for the number of alphas generated between the minimum and maximum edge distance on the Delanauy triangulation.
envelope	boolean to determine if the Monte-Carlo is estimated. Default FALSE.
mc_cores	an integer to determine how many parallel process should be run. Default mc_core=1.

Value

A list of class spatgeom with the following elements:

call The function call.

x x input.

y y output.

results A list of size ncol(x) corresponding to each column of x. Each element of the list has:

- **triangles** a data frame of class sfc (see sf::st_sf())with columns geometry, segments, max_length and alpha. The data.frame contains the whole Delanauy triangulation for the corresponding column of x and y. The segments column are the segments of each individual triangle and max_length is the maximum length of them.
- **geom_indices** a data frame with columns alpha and geom_corr. The alpha column is a numeric vector of size nalphas from the minimum to the maximum distance between points estimated in the data. The geom_corr column is the value 1 - (alpha shape Area)/(containing box Area).
- **intensity** the intensity estimated for the corresponding column of x and y.
- **mean_n** the mean number of points in the point process.
- **envelope_data** a data frame in tidy format with 40 runs of a CSR process, if envelope=TRUE, The CSR is created by generating *n* uniform points in the plane, where *n* is drawn from Poisson distribution with parameter mean_n.

References

Hernández, A.J., Solís, M. Geometric goodness of fit measure to detect patterns in data point clouds. Compute Stat (2022). https://doi.org/10.1007/s00180-022-01244-1

Examples

```
xy <- donut_data(n = 30, a = -1, b = 1, theta = 2 * pi)
estimation <- spatgeom(y = xy[, 1], x = xy[, -1])

# If you want to estimate the envelope, you can use the envelope argument to
# TRUE. This will take a while to run.
## Not run:
estimation_with_envelope <- spatgeom(
  y = xy[, 1], x = xy[, -1],
  envelope = TRUE
)

## End(Not run)
```

Index