

Package ‘RcppAlphahull’

December 1, 2019

Type Package

Title alpha-convex hull and alpha-shape computation.

Version 1.0

Date 2019-07-31

Author Federico Airoidi

Maintainer Federico Airoidi <federico2.airoidi@mail.polimi.it>

Description This package performs alpha-convex hull and alpha-shape computation using C++ code and a library, MyGAL, to compute the Voronoi tessellation/Delaunay triangulation of a finite set of points in the plane.

License GPL (>= 2)

Depends R (>= 3.4.4)

Imports Rcpp (>= 1.0.1), alphahull

LinkingTo Rcpp

SystemRequirements C++14

RoxygenNote 6.1.1

Encoding UTF-8

R topics documented:

ahull	2
ashape	3
complement	4
delvor	5
inahull	6
plot.delvor	7
Index	9

ahull	<i>α-hull computation</i>
-------	---

Description

Computes the α -hull for the given set of points.

Usage

```
ahull(x, y = NULL, alpha)
```

Arguments

x	x coordinates of the sites. Alternatively, a matrix with two columns containing both sites coordinates, an object of class <code>delvor</code> or an object of class <code>ashape</code> .
y	y coordinates of the sites; do not insert if x is not a vector of coordinates (see x).
alpha	a strictly positive value for α (NULL if x is of class <code>ashape</code>).

Value

A list with the following components:

- `arcs`: a matrix containing the arcs that form the boundary of the α -hull;
- `length`: length of the α -hull boundary;
- `complement`: a matrix describing the complement of the α -hull, see function [complement](#) for a detailed description;
- `alpha`: the value of α for which the shape is computed;
- `ashape`: output of function [ashape](#).

See Also

[delvor](#), [ashape](#), [complement](#)

Examples

```
x = runif(10)
y = runif(10)
a.hull = ahull(x, y, alpha = 0.8)
plot(a.hull)
```

ashape	<i>α-shape computation</i>
--------	--

Description

Computes the α -shape for the given set of points.

Usage

```
ashape(x, y = NULL, alpha)
```

Arguments

x	x coordinates of the sites. Alternatively, a matrix with two columns containing both sites coordinates or an object of class <code>delvor</code> .
y	y coordinates of the sites; do not insert if x is not a vector of coordinates (see x).
alpha	a strictly positive value for α .

Value

A list with the following components:

edges a mesh describing the α -shape, this object is a submatrix of the mesh matrix contained in a "delvor" object since the alpha shape is a subset of the Delanuay triangulation.

length length of the alpha shape.

alpha the value of α for which the shape is computed.

alpha.extremes contains the indices of those sites that are α -extremes.

delvor.obj a delvor object returned by the function `delvor`, if a delvor object is provided instead of the sites coordinates, this field contains such object

See Also

[delvor](#)

Examples

```
x = runif(10)
y = runif(10)
a.shape = ashape(x, y, alpha = 0.8)
plot(a.shape)
```

complement

*Complementary α -hull computation***Description**

Computes the complement of an α -hull for the given value of α provided.

Usage

```
complement(x, y = NULL, alpha)
```

Arguments

x	coordinates of the sites. Alternatively, a matrix with two columns containing both sites coordinates or an object of class <code>delvor</code> .
y	y coordinates of the sites; do not insert if x is not a vector of coordinates (see x).
alpha	a strictly positive value for α .

Value

A matrix containing information about balls and halfplanes constituting the complement of the α hull, each row describes either an open ball or an open halfplane such that balls are saved in the following way:

- c1: x coordinate of the center of the ball;
- c2: y coordinate of the center of the ball;
- r: radius of the ball;

If the row refers to an halfplane then `complement[i, 1:3]` has the following forms:

- $x > a + bx$: (a, b, -1);
- $x < a + bx$: (a, b, -2);
- $x > a$: (a, 0, -3);
- $x < a$: (a, 0, -4).

See Also

[ahull](#)

Examples

```
x = runif(10)
y = runif(10)
alpha = 2
ahull.compl = complement(x, y, alpha)
```

delvor

*Voronoi tessellation/Delanuay triangulation***Description**

Computing Voronoi diagram and Delanuay tessellation for the specified set of points employing the C++ library **MyGAL**.

Usage

```
delvor(x, y = NULL)
```

Arguments

x	x coordinates of the sites or a matrix with two columns containing both sites coordinates.
y	y coordinates of the sites. Alternatively a single argument can be provided (see x).

Details

This function retrieves the Voronoi tessellation and the Delanuay triangulation of a given set of points in the plane; results are returned in a list of three elements (see Value).

Each row of **mesh** contains information about one of the edges of the tessellation and its dual:

- ind1 and ind2: indices of the sites to which the edge refers to;
- x1 and y1: coordinates of the site denoted by ind1;
- x2 and y2: coordinates of the site denoted by ind2;
- mx1 and my1: coordinates of the first extreme, e1, of the Voronoi tessellation edge;
- mx2 and my2: coordinates of the second extreme, e2, of the Voronoi tessellation edge;
- bp1 and bp2: denote whether one of the direction of the Voronoi edge is infinite, either from the side of e1 or e2.

tri.obj, S3 object of class "tri.mod", is a list of four elements that describe the Delanuay triangulation:

- n: number of sites of the triangulation;
- x: x coordinates of the sites;
- y: y coordinates of the sites;
- neighbours: a list of integer vectors where the i-th vectors contains the indices of neighbours sites of the i-th site in the triangulation.

Value

An invisible object of class "del.vor", a list, with the following components:

mesh a matrix describing the Voronoi tessellation and the Delanuay triangulation.

x a 2-column matrix containing the coordinates of the sites.

tri.obj an S3 object of class "tri.mod" describing the triangulation similar to the one of the package tri.mesh.

See Also

[plot.delvor](#)

Examples

```
x = runif(10)
y = runif(10)
del.vor = delvor(x, y)
plot(del.vor)
```

inahull

Determines if the provided points fall inside the α -hull or not.

Description

Determines if the provided points fall inside the α -hull or not by calling an external C++ function.

Usage

```
inahull(ahull.obj, x, y = NULL, alpha)
```

Arguments

x	x coordinates of the points to be checked. Alternatively, a matrix with two columns containing both coordinates.
y	y coordinates of the points to be checked.
ahull.obj	an object class "ahull" like the one returned by the function ahull

Value

A logical vector of the same length of the number of points and such that the i-th element denotes if the i-th point is in the α -hull or not.

See Also

[ahull](#), [complement](#)

Examples

```
x = runif(10)
y = runif(10)
inahull(ahull.obj, x, y)
```

plot.delvor

*Voronoi tessellation/Delanuay triangulation plot***Description**

Plot of an object "del.vor" with a "tri.mod" tri object.

Usage

```
## S3 method for class 'delvor'
plot(delvor.obj, add = FALSE, wlines = c("both",
    "del", "vor"), wpoints = TRUE, number = FALSE, col = NULL,
    xlim = NULL, ylim = NULL, ...)
```

Arguments

add	if TRUE the plot is added to the active graphic window.
wlines	a string specifying what has to be plotted: <ul style="list-style-type: none"> • "vor": shows the Voronoi tessellation; • "del": shows the Delanuay triangulation plot; • "both": shows both of the above structures.
wpoints	if true, the sites are added to the plot.
number	if true, the plot shows the indeces of the sites between their locations.
col	specifies in a vector the colors to be used for the different objects to be plotted and the order is: <ul style="list-style-type: none"> • col[1] -> color of the points; • col[2] -> color of Delanuay triangulation; • col[3] -> color of Voronoi tessellation • col[4] -> color for the numbers Alternatively just one color can be provided.
xlim	x axis limits.
ylim	y axis limits.
...	graphical arguments to be passes to methods (see par)
x	object of class del.vor.

See Also

[delvor](#)

Examples

```
x = runif(10)
y = runif(10)
del.vor = delvor(x, y)

plot.delvor(vor.del, wlines = "both", wpoints = FALSE, number = TRUE,
            col = c("black", "blue", "red", "black"))
```


Index

ahull, [2](#), [4](#), [6](#)

ashape, [2](#), [3](#)

complement, [2](#), [4](#), [6](#)

delvor, [2](#), [3](#), [5](#), [7](#)

inahull, [6](#)

par, [7](#)

plot.delvor, [6](#), [7](#)