

phull: p -hull in R

Marek Gaḡolewski^{1,2}
gagolews@ibspan.waw.pl

WZUR, Warsaw, October 4, 2009



¹ Systems Research Institute,
Polish Academy of Sciences



² Faculty of Mathematics
and Information Science,
Warsaw University of Technology

Outline

1 p -hull and its properties

2 Examples

3 Computation

Preliminaries

Given an arbitrary $0 < p < \infty$, $x_0, y_0 \in \mathbb{R}$, $a \geq 0$ and $b \geq 0$, let

$$E_{p,a,b}^{(x_0,y_0)} = \left\{ (x, y) \in \mathbb{R}^2 : \left| \frac{y - y_0}{b} \right|^p + \left| \frac{x - x_0}{a} \right|^p \leq 1 \right\}. \quad (1)$$

Moreover, for $p = \infty$ we have

$$E_{p,a,b}^{(x_0,y_0)} = \left\{ (x, y) \in \mathbb{R}^2 : \max \left\{ \left| \frac{y - y_0}{b} \right|, \left| \frac{x - x_0}{a} \right| \right\} \leq 1 \right\}, \quad (2)$$

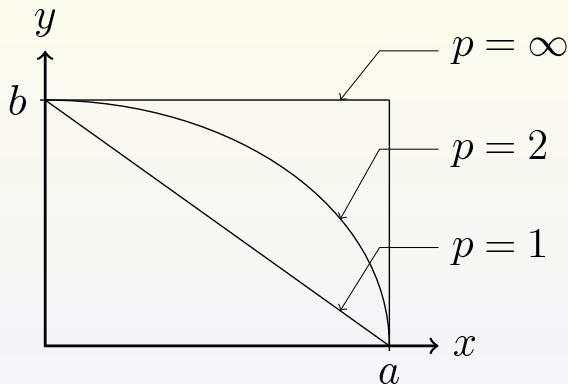
and for $p = 0$

$$E_{p,a,b}^{(x_0,y_0)} = \left\{ (x, y) \in \mathbb{R}^2 : \begin{array}{ll} x \in [x_0 - a, x_0 + a] & \wedge \quad y = y_0 \\ \vee \quad y \in [y_0 - b, y_0 + b] & \wedge \quad x = x_0 \end{array} \right\}. \quad (3)$$

We call $E_{p,a,b}^{(x_0,y_0)}$ the **p -ellipse of size (a, b) centered at (x_0, y_0)** .

Preliminaries

Illustration: $\partial E_{p,a,b}^{(0,0)} \cap \mathbb{R}_0^+ \times \mathbb{R}_0^+$.



Preliminaries

We are given a finite planar set $Q = \{q_1, q_2, \dots, q_n\}$, such that $q_i = (x_i, y_i) \in \mathbb{R}^2$, $i = 1, \dots, n$ ($n \geq 4$).

Let

$$x_l = \min_{p_i \in P} x_i,$$

$$x_r = \max_{p_i \in P} x_i,$$

$$y_b = \min_{p_i \in P} y_i,$$

$$y_t = \max_{p_i \in P} y_i.$$

Then $B(Q) = [x_l, x_r] \times [y_b, y_t]$ is the **minimal bounding rectangle** of Q .

Preliminaries

For a fixed $p \geq 0$ let

$$\begin{aligned}C_p^{\text{bl}}(Q) &= \bigcup_{a,b: Q \notin \text{int } E_{p,a,b}^{(x_l, y_b)}} E_{p,a,b}^{(x_l, y_b)}, \\C_p^{\text{br}}(Q) &= \bigcup_{a,b: Q \notin \text{int } E_{p,a,b}^{(x_r, y_b)}} E_{p,a,b}^{(x_r, y_b)}, \\C_p^{\text{tr}}(Q) &= \bigcup_{a,b: Q \notin \text{int } E_{p,a,b}^{(x_r, y_t)}} E_{p,a,b}^{(x_r, y_t)}, \\C_p^{\text{tl}}(Q) &= \bigcup_{a,b: Q \notin \text{int } E_{p,a,b}^{(x_l, y_t)}} E_{p,a,b}^{(x_l, y_t)}.\end{aligned}$$

We further on assume $\text{int } C_p^{\text{bl}}(Q)$, $\text{int } C_p^{\text{br}}(Q)$, $\text{int } C_p^{\text{tr}}(Q)$, $\text{int } C_p^{\text{tl}}(Q)$ are mutually exclusive.

Definition

Let $Q = \{q_1, q_2, \dots, q_n\} \subset \mathbb{R}^2$ and $p \geq 0$. The p -hull of Q , denoted by $H_p(Q)$, is defined by

$$H_p(Q) = \partial \left(B(Q) \setminus C_p^{\text{bl}}(Q) \setminus C_p^{\text{br}}(Q) \setminus C_p^{\text{tr}}(Q) \setminus C_p^{\text{tl}}(Q) \right). \quad (4)$$

Properties of a p -hull

Proposition

Let $Q = \{q_1, q_2, \dots, q_n\} \subset \mathbb{R}^2$ and $p \geq 0$. Then we have the following.

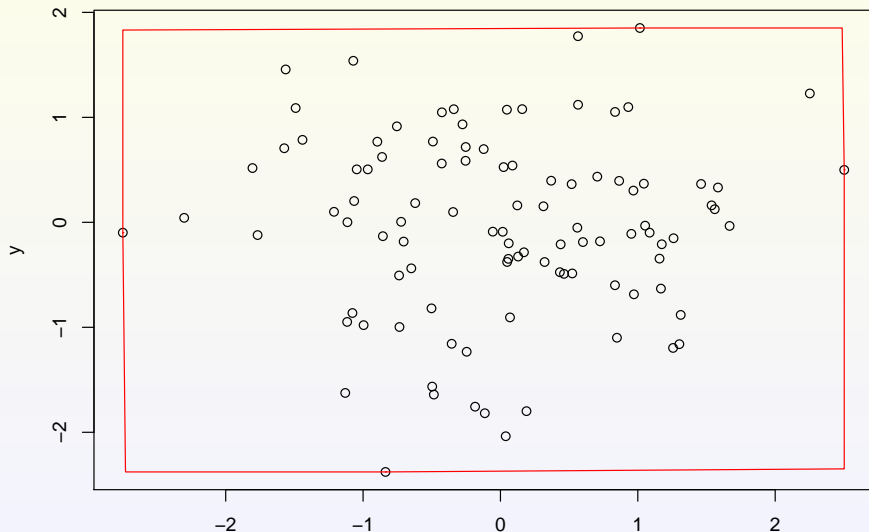
- 1 If $p = 1$ then $H_p(Q)$ is the convex hull of Q .
- 2 If $p = \infty$ then $H_p(Q)$ is the X - Y hull of Q (see Nicholl et al, 1983).
- 3 If $p = 0$ then $H_p(Q) = \partial B(Q)$.

Properties of a p -hull

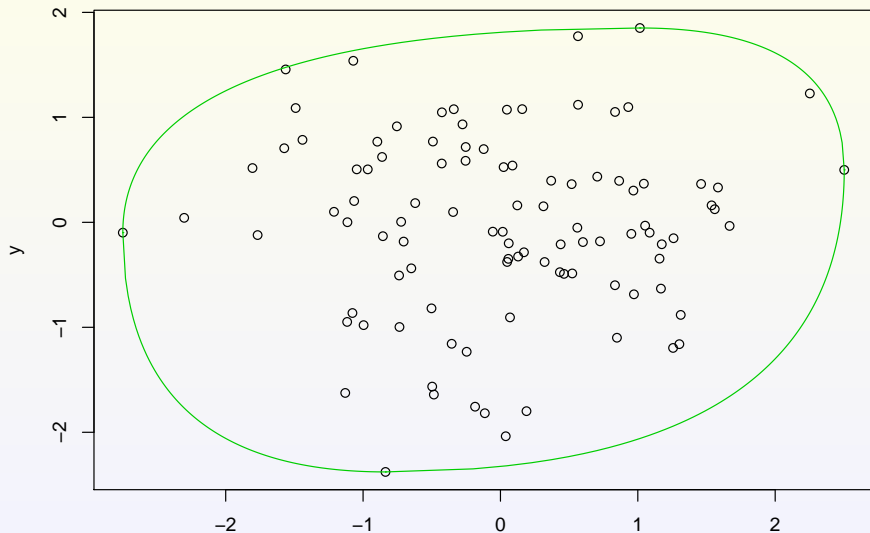
Other properties:

- ❶ $H_p(Q)$ is translation- and scale-invariant for any $p \geq 0$, i.e.
 $\otimes H_p(Q) = H_p(\otimes Q)$.
- ❷ $H_p(Q)$ is not rotation-invariant (thus it is orientation-dependent) for $p \neq 1$.
- ❸ $H_p(Q)$ is convex for $p \leq 1$.
- ❹ If $p' \geq p$, then $H_{p'}(Q) \subseteq H_p(Q)$.

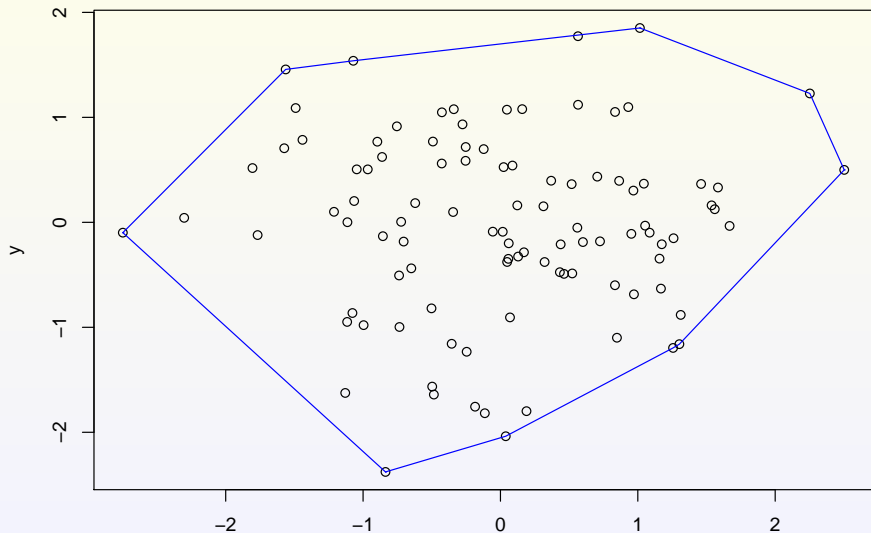
Example: $p = 0.1$



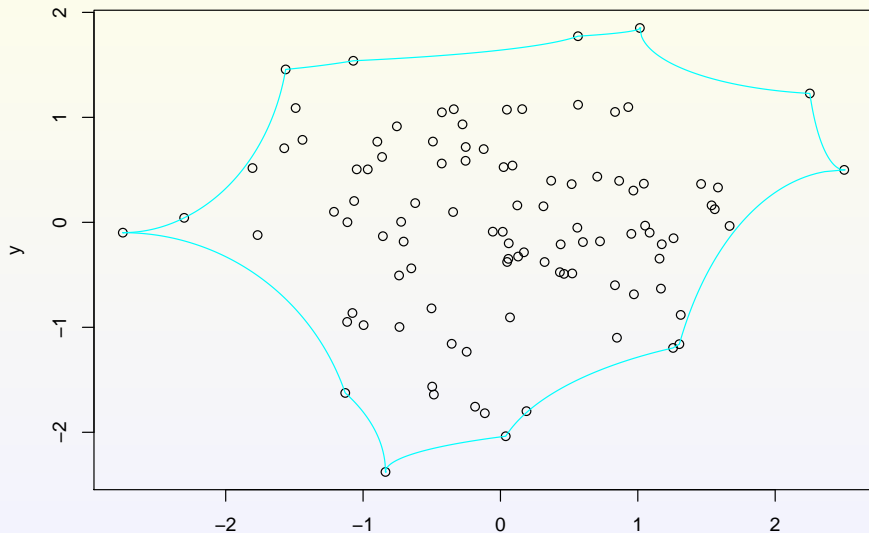
Example: $p = 0.5$



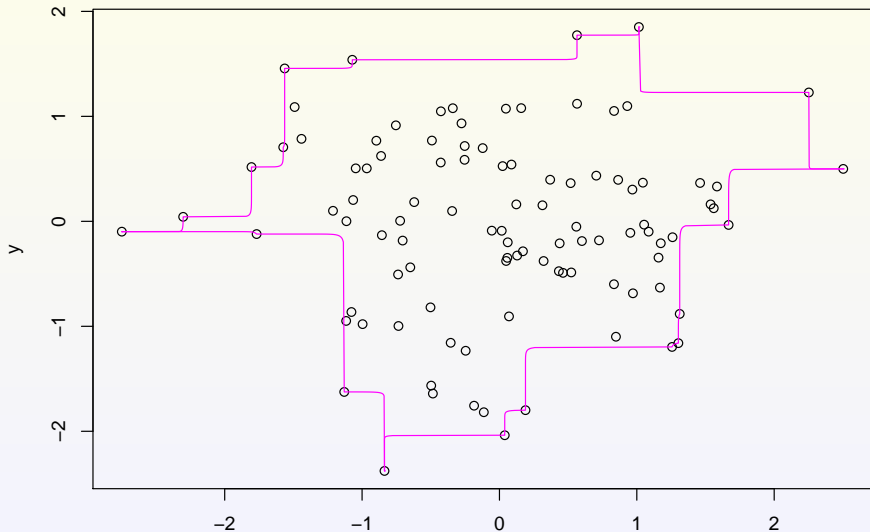
Example: $p = 1.0$



Example: $p = 2.0$



Example: $p = 50$



Applications

Among possible applications are:

- 1 Scientometrics: calculation of so-called L_p indices (Gągolewski, Grzegorzewski, 2009),
- 2 p.d.f. support estimation: given a random sample of points from distribution given by unknown p.d.f. $f(x, y)$, estimate $\text{supp } f$,
- 3 ...

Computation

Let

$$\begin{aligned}q_{\text{bl}_1} &= \arg \min_{q_i \in Q: x_i = x_l} y_i, & q_{\text{bl}_2} &= \arg \min_{q_i \in Q: y_i = y_b} x_i, \\q_{\text{br}_1} &= \arg \max_{q_i \in Q: y_i = y_b} x_i, & q_{\text{br}_2} &= \arg \min_{q_i \in Q: x_i = x_r} y_i, \\q_{\text{tr}_1} &= \arg \max_{q_i \in Q: x_i = x_r} y_i, & q_{\text{tr}_2} &= \arg \max_{q_i \in Q: y_i = y_t} x_i, \\q_{\text{tl}_1} &= \arg \min_{q_i \in Q: y_i = y_t} x_i, & q_{\text{tl}_2} &= \arg \max_{q_i \in Q: x_i = x_l} y_i.\end{aligned}$$

Note that all the points $\in \partial B(Q)$.

Computation (cont'd)

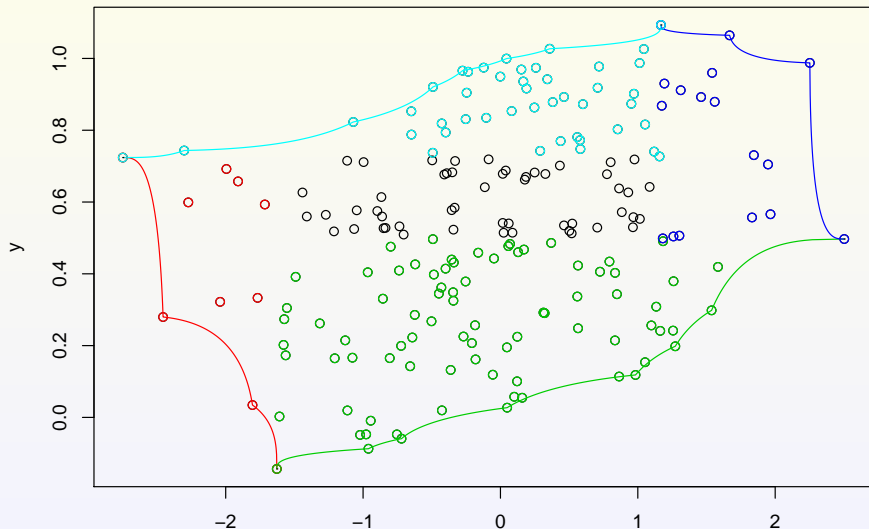
Decomposition:

$$\begin{aligned} H_p(Q) &= \partial \left(B(Q) \setminus C_p^{\text{bl}}(Q) \setminus C_p^{\text{br}}(Q) \setminus C_p^{\text{tr}}(Q) \setminus C_p^{\text{tl}}(Q) \right) \\ &= \left(\partial C_p^{\text{bl}}(Q) \cup \partial C_p^{\text{br}}(Q) \cup \partial C_p^{\text{tr}}(Q) \cup \partial C_p^{\text{tl}}(Q) \right) \cap B(Q) \\ &\cup \overline{q_{\text{bl}_2} q_{\text{br}_1}} \cup \overline{q_{\text{br}_2} q_{\text{tr}_1}} \cup \overline{q_{\text{tr}_2} q_{\text{tl}_1}} \cup \overline{q_{\text{tl}_2} q_{\text{bl}_1}}. \end{aligned} \tag{5}$$

Moreover:

$$\begin{aligned} \partial C_p^{\text{bl}}(Q) &= \partial C_p^{\text{bl}}(\{q_i \in Q : x_i \leq x_{\text{bl}_2} \wedge y_i \leq y_{\text{bl}_1}\}), \\ \partial C_p^{\text{br}}(Q) &= \partial C_p^{\text{br}}(\{q_i \in Q : x_i \geq x_{\text{br}_1} \wedge y_i \leq y_{\text{br}_2}\}), \\ \partial C_p^{\text{tr}}(Q) &= \partial C_p^{\text{tr}}(\{q_i \in Q : x_i \geq x_{\text{tr}_2} \wedge y_i \geq y_{\text{tr}_1}\}), \\ \partial C_p^{\text{tl}}(Q) &= \partial C_p^{\text{tl}}(\{q_i \in Q : x_i \leq x_{\text{tl}_1} \wedge y_i \geq y_{\text{tl}_2}\}). \end{aligned} \tag{6}$$

Computation (cont'd)



Computation (cont'd)

- 1 Naïve algorithm: $O(n^3)$ time. :-(
2 Algorithm proposed by Gągolewski, Nowakiewicz, Dębski (2009)
— generalizes Graham's scan (Graham, 1972): $O(n \log n)$ time,
 $O(n)$ memory.
3 Output: $P \cap \partial C_p^{\text{bl}}(Q)$ (without loss of generality).
4 Input: $p \geq 0$, $W = \{q_i \in Q : x_i \leq x_{\text{bl}_2} \wedge y_i \leq y_{\text{bl}_1}\}$ as an array
sorted by x coordinate $w[1], \dots, w[m]$.
5 Denotation: by $E_{p,q_i,q_j}^{(x_0,y_0)}$ we mean an p -ellipse centered at (x_0, y_0)
interpolating $q_i \neq q_j$.

Computation (cont'd)

```
1  Create an empty stack S;
2  Push  $w[1]$  into S;
3   $i := 2$ ;
4  while ( $i < n$ ) and ( $w[i]_y \geq w[1]_y$ ) do
5       $i := i + 1$ ;
6  Push  $w[i]$  into S;
7  for  $j = i + 1, i + 2, \dots, n$  do
8      if ( $S[\#S]_y < w[j]_y$ ) then {
9          while ( $\#S \geq 2$ ) and ( $S[\#S - 1] \in E_{p, S[\#S], w[j]}^{(x_{bl}, y_{bl})}$ ) do
10             Pop from S;
11             Push  $w[j]$  into S;
12      }
13 return S;
```

Computation (cont'd)

Implementation: `phull` 0.1-2 — package available on CRAN.
(<http://cran.r-project.org/web/packages/phull/index.html>)

Example: axes rotation.

```
library(phull); # load the library  
  
translateAndRotate <- function(data, x0, y0, angle)  
{ ... }  
  
rotateAndTranslate <- function(data, x0, y0, angle)  
{ ... }
```

Computation (cont'd)

```
set.seed(98765); n <- 1000; p <- 3.0;
data <- matrix(c(rnorm(n), rt(n, 10)), ncol=2); # input data
nres <- 50; # "resolution"

ptest <- phull(data, p=p);           # compute the p-hull
discr_0 <- as.matrix(ptest, nres=nres); # sample

print(ptest)

      p-hull, p=3

data: data
1000 points, bounding rectangle: (...)
```

Computation (cont'd)

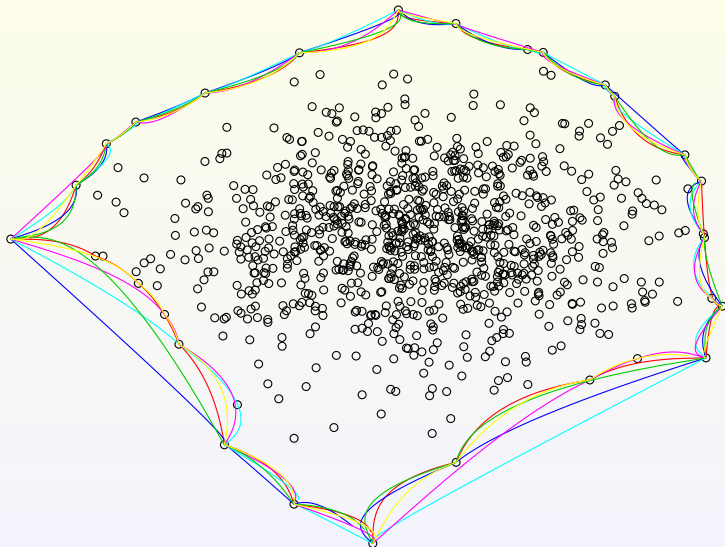
```
data2 <- translateAndRotate(data, angle=-pi/6
  -ptest$xrange[1], -ptest$yrange[1]);
ptest2 <- phull(data2, p=p);           # compute the p-hull
discr_30 <- as.matrix(ptest2, nres=nres); # sample
discr_30 <- rotateAndTranslate(discr_30, angle=pi/6,
  ptest$xrange[1], ptest$yrange[1]);

plot(data, type="p", pch=1);
lines(discr_0, col=2);
lines(discr_30, col=4);
```

...and so on...

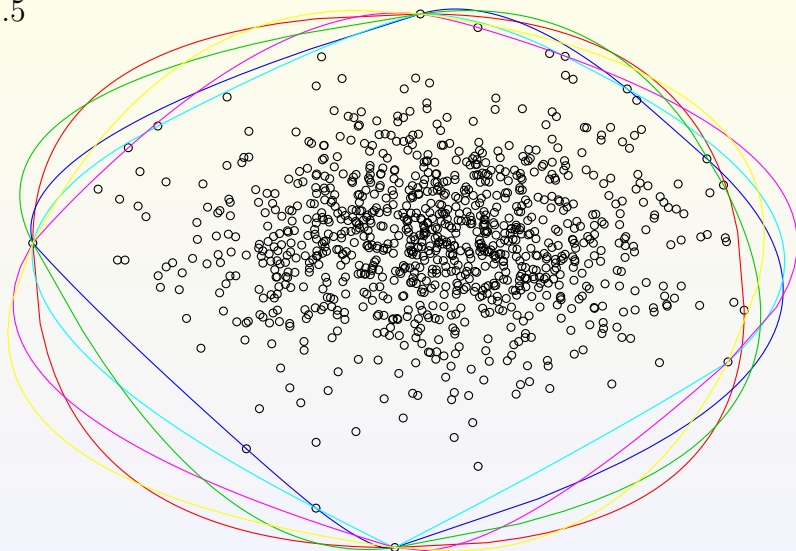
Computation (cont'd)

$$p = 3$$



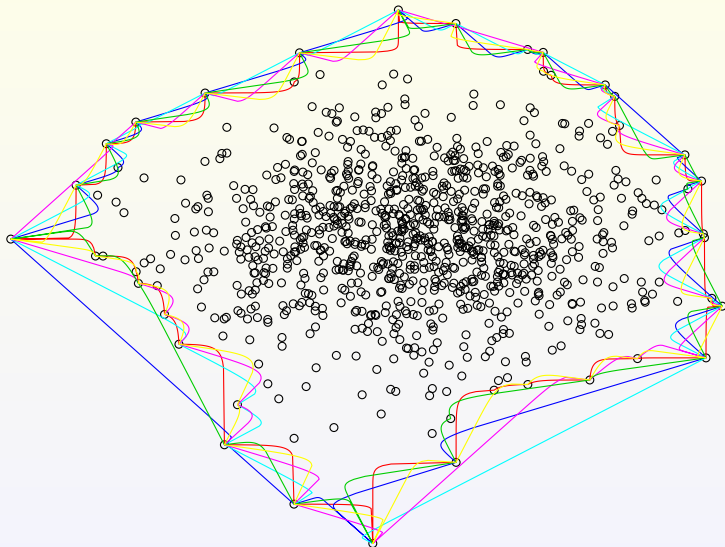
Computation (cont'd)

$p = 0.5$



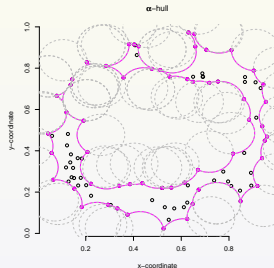
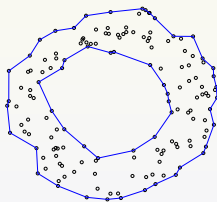
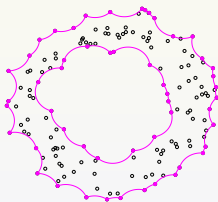
Computation (cont'd)

$p = 20$









Related packages

`alphahull` (Pateiro-Lopez, Rodriguez-Casal, 2009): α -shapes
(Edelsbrunner et al, 1983).



References

-  H. Edelsbrunner, D. G. Kirkpatrick, R. Seidel (1983). On the shape of a set of points in the plane. *IEEE Trans. Inf. Theor.* 29(4), 551–559.
-  M. Gągolewski, P. Grzegorzewski (2009). A geometric approach to the construction of scientific impact indices. *Scientometrics*. In press. DOI:10.1007/s11192-008-2253-y.
-  M. Gągolewski, M. Nowakiewicz, M. Dębski (2009). *Efficient algorithms for computing “geometric” scientific impact indices*. Submitted for publication.
-  R. L. Graham (1972). An efficient algorithm for determining the convex hull of a finite planar set. *Information Processing Letters* 1, 132–133.
-  T. M. Nicholl, D. T. Lee, Y. Z. Liao, C. K. Wong (1983). On the X-Y convex hull of a set of X-Y polygons. *BIT* 23, 456–471.
-  B. Pateiro-Lopez, A. Rodriguez-Casal (2009). *alphahull: Generalization of the convex hull of a sample of points in the plane*, see `alphahull` @ CRAN.

Thank you for your attention.