Statistical application of curve evolution

Petra Surynková, Šárka Voráčová, Zbyněk Šír

Dep. of Mathematics Education, Fac. of Mathematics and Physics, Charles University in Prague
Sokolovská 83, 186 75 Praha 8, Czech Republic
email: petra.surynkova@mff.cuni.cz, sarka.voracova@mff.cuni.cz, zbynek.sir@mff.cuni.cz

Abstract. This contribution deals with statistical data analysis by using closed B-spline curves. Our aim is to obtain the description of a data set via these curves with respect to local density of data sample. We apply a curve evolution as a solving method. The result is the depth contour fulfilling some predefined statistical criterion.

Key words: B-spline, evolution, data depth

1 Introduction

One notion of shape estimation of a set of points in the plane can be their depth. Data depth has been used in statistics as a way to identify the centre of a bivariate distribution. This leads to a natural centre-outward ordering of the sample points. Then we can estimate important characteristics of the data. The set of depth contours of a set of points can be used in various applications.

We apply a curve evolution for finding the depth contour. The input is a finite set of points in the plane and our aim is to obtain the description of a data set by using closed B-spline curves. We have some initial closed B-spline curve. This curve moves and modifies in time such that finally we obtain depth contour fulfilling some predefined statistical criterion.

The paper is organized as follows. The notion of data depth and various definitions of statistical depth are given in section 2. The section 3 is devoted to curve evolution. In the section 4 we introduce the description of our implementation of the algorithm. Finally we put some examples of depth contours obtained by using closed B-spline curve evolution.

2 The notion of data depth

The notion of data depth generalizes the median to higher dimensions. The motivation and necessity in statistics to generalize the median is very natural. Various depth measures have been proposed to analyze a set of points in the plane. In general, the greater the depth of a point, the more densely it is surrounded by other points of the input set. We can compute over this depth measure the point in the plane with the maximum depth – the deepest point. It is analogous to the median in $\mathbb{R}$ and it is called median too. We can also find the regions of all points with depth greater than some assigned value k.

There is a wide range of data depth measures, each approaching the problem in different way. Depth measures are varied both in how well they work on different data sets and in how fast they run. We introduce several of them.

2.1 Convex hull peeling

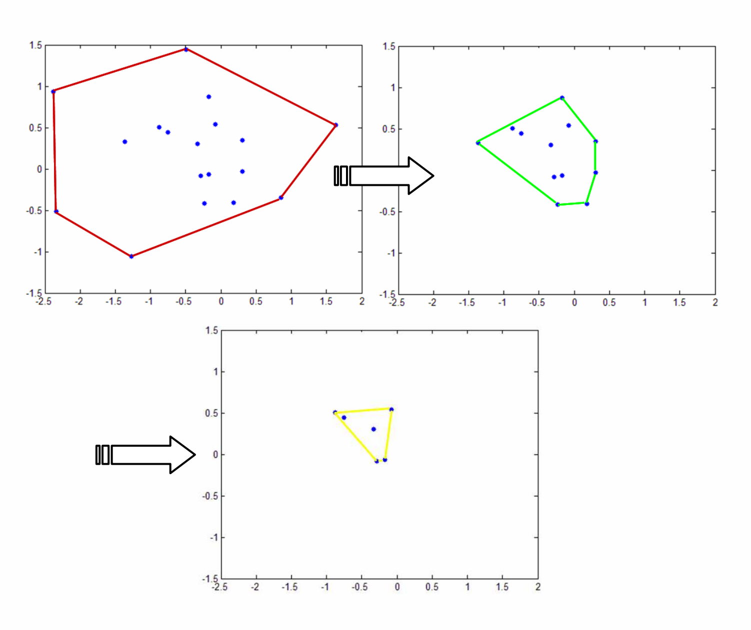

Fig. 1: Convex hull peeling process

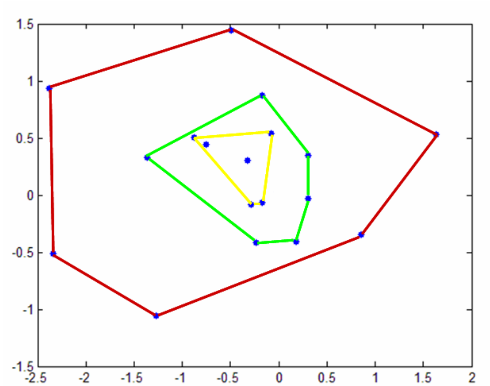

Fig. 2: Depth contours

Let $X \in \mathbb{R}^2$ be a finite planar set of points. We compute the convex hull of the set X and remove the vertices on the convex hull. Then we repeat this peeling process. This recursive peeling leads to the innermost points. The last convex hull to be peeled away is the deepest in the set of points. Vertices on some convex hull have the same depth. The depth contours are the convex regions wherein contours of some depth contain all points at that depth and deeper.

Convex hull peeling is relatively naïve approach to depth but works properly on well-behaved data. More detailed information can be found in [2]. Figure 1 shows this peeling process, figure 2 shows the depth contours.

2.2 Simplicial depth

Given a finite planar set of points $X \in \mathbb{R}^2$, n is the number of points in X. The simplicial depth, defined by Lui, of a point $p \in \mathbb{R}^2$ is the probability that a random closed simplex in $\mathbb{R}^2$ (triangle) formed by vertices of X contains p.

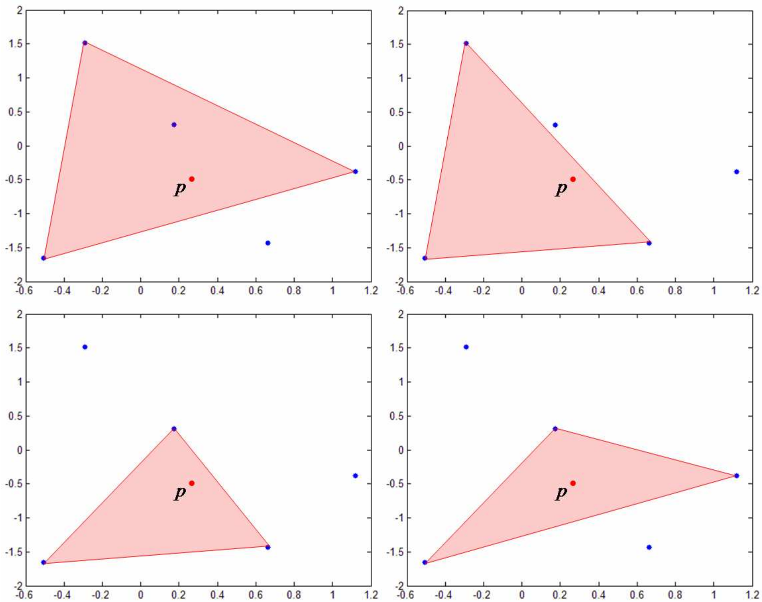

Fig. 3: Triangles containing p

Let T be the set of triangles formed by vertices of X. The set T has $\binom{n}{3}$ triangles because each triangle requires three vertices. Then the simplicial depth of a point p, denoted $SDepth(p)$, is the fraction of the number of closed simplicies containing p (T_p) and the number of all simplicies ($|T|$), thus

$$SDepth(p) = \frac{T_p}{|T|}. \qquad (1)$$

The simplicial median is the point with highest simplicial depth. The disadvantage of simplicial depth is that the computation of this depth measure can be very difficult and slow.

See figures 3, 4 which illustrated principle of simplicial depth. More detailed information can be found in [3].

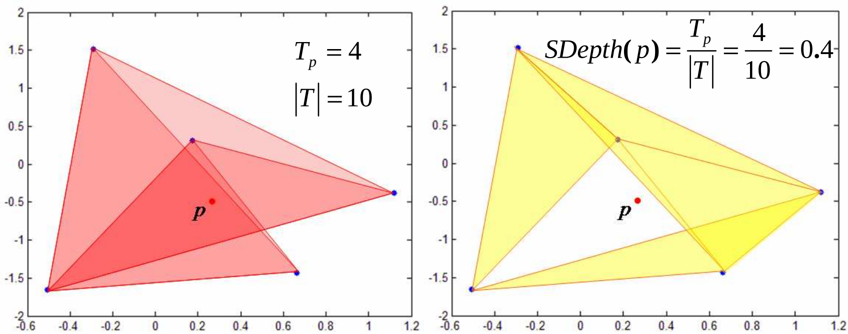

Fig. 4: On the left triangles containing p, on the right rest of triangles

2.3 Half-space depth

Let $X \in \mathbb{R}^2$ be a finite planar set of points. Half-space depth of a point $p \in \mathbb{R}^2$ is the minimum number of points contained in any half-plane passing through p; see figures 5, 6. More detailed information can be found in [4].

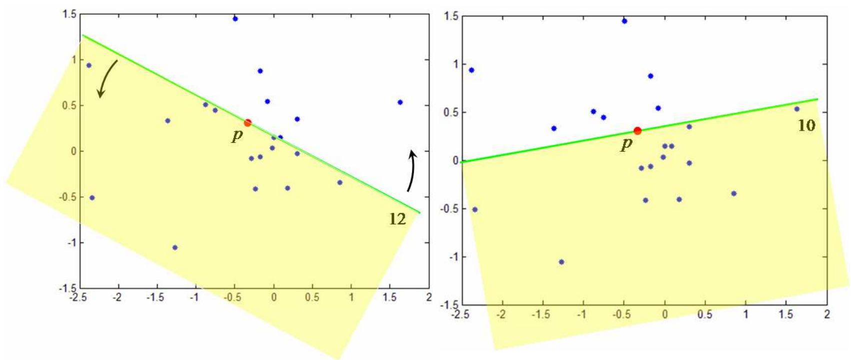

Fig. 6: The examples of half-spaces passing through p (half-space on the left contains 12 points, half-space on the right contains 10 points)

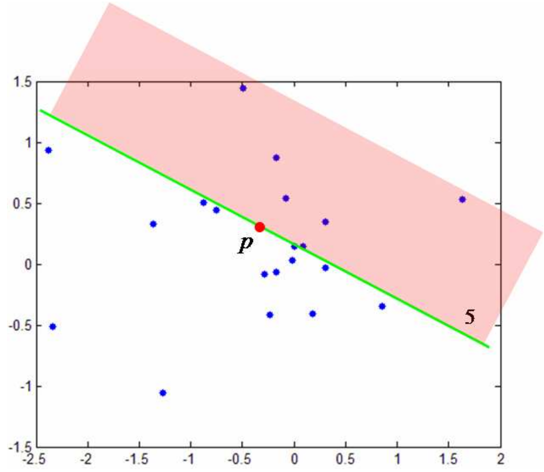

Fig. 7: Half-space containing the minimum number of points. Half-space depth of a point *p* is 5

2.4 Properties of data depth in general

Statistical data depth of a point $p \in \mathbb{R}^2$, denoted $Depth(p)$, has to satisfy the following general properties:

a) affine invariance,
b) maximality at centre,
c) monotonicity relative to deepest point,
d) vanishing at infinity.

A function which satisfies these restrictions is called the depth function. Figures 8, 9, 10 11 illustrate these properties.

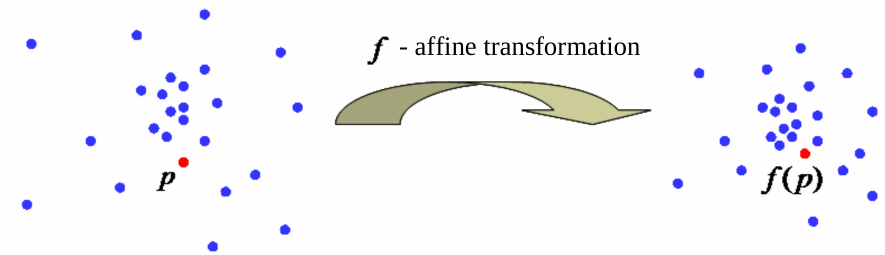

Fig. 8: Affine invariance – $\boldsymbol{Depth(p) = Depth(f(p))}$

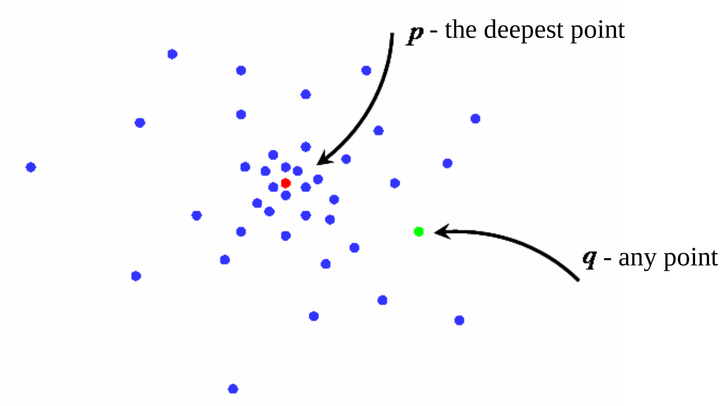

Fig. 9: Maximality at centre - $\boldsymbol{Depth}(\boldsymbol{p}) \geq \boldsymbol{Depth}(\boldsymbol{q})$

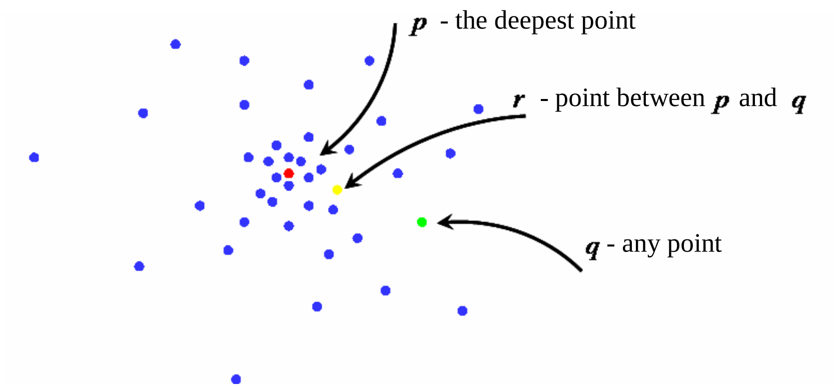

Fig. 10: Monotonicity relative to deepest point - $\boldsymbol{Depth}(\boldsymbol{q}) \leq \boldsymbol{Depth}(\boldsymbol{r})$

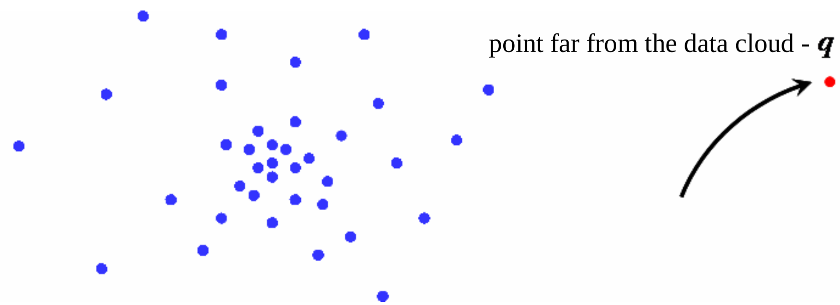

Fig.11: Vanishing at infinity - $\boldsymbol{Depth}(\boldsymbol{q}) = 0$

3 Curve evolution

We apply a curve evolution for finding the depth contour. We identify a curve that approximates a given set of data points $\{p_j\}_{j=1..N}$ in the least square sense. We consider a planar parametric curve

(2) $$c(u) = \sum_{i=0}^{m} \beta_i(u) \cdot V_i \ ,$$

where u is the curve parameter, V_i are control points and β_i are basis functions. We are looking for the curve such that

(3) $$\sum_{j=1}^{N} \min_{x_j \in c} \left\| p_j - x_j \right\|^2 \rightarrow \min \ ,$$

where x_j are points on the curve. If we denote $c_s(u) := c(s,u)$ then two different kinds of parameters appear in the representation of the curve; the curve parameter u and a vector of shape parameters $s = (s_1, s_2, ..., s_n)$. It means that we are looking for the vector of shape parameters that defines this curve. We let the shape parameters s depend smoothly on an evolution parameter t, $s(t) = \left(s_1(t), s_2(t), ..., s_n(t)\right)$. The parameter t can be identified with the time. Starting with certain initial values, these parameters are modified continuously in time such that a given initial curve moves closer to the data points. See figure 12.

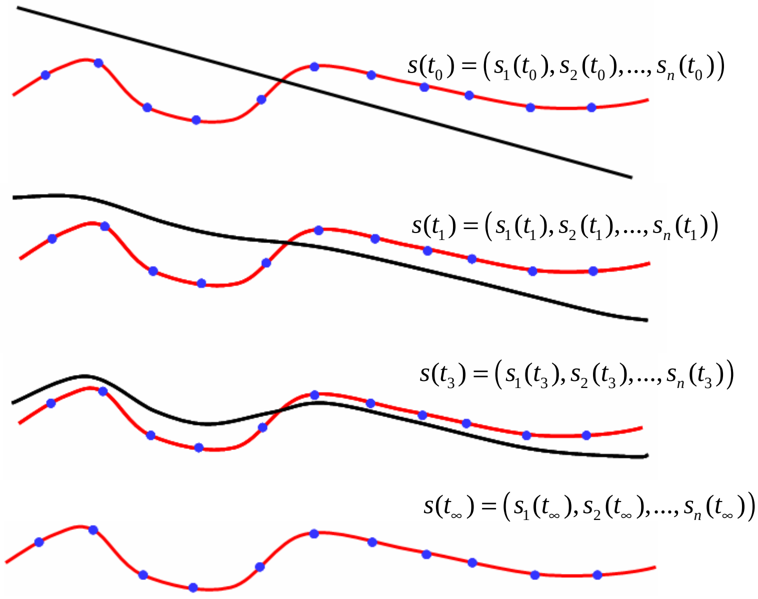

Fig. 12: Curve evolution in time

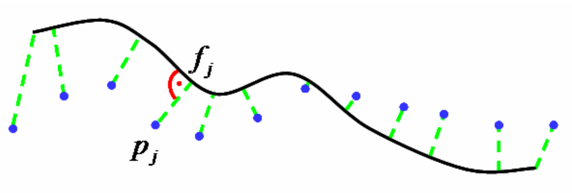

Fig. 13: Closest points

For each point $\{p_j\}_{j=1..N}$ we compute the closest points $f_j = c(u_j)$ on the initial curve, see figure 13. During the evolution of a curve $c_{s(t)}(u)$ each point f_j travels with velocity

(4) $$v_s(u_j) = \dot{c}_s(u_j) = \sum_{i=1}^{n} \frac{\partial c_s(u_j)}{\partial s_i} \dot{s}_i$$

or with normal velocity

(5) $$\left(v_s(u_j)\right)^{\mathrm{T}} n_s(u_j) = \sum_{i=1}^{n} \left(\frac{\partial c_s(u_j)}{\partial s_i} \dot{s}_i \right)^{\mathrm{T}} n_s(u_j) .$$

The dot denotes the derivative with respect to the time variable t, $n_s(u_j)$ denotes the unit normal of the curve in the point $c_s(u_j)$. We denote $d_j := p_j - f_j$. If a closest point is one of the two boundary points then we consider the velocity (4). Following (4) and (5) we can compute for each point f_j the velocity or normal velocity on the one hand and the expected velocity on the other hand. The following condition has to satisfy

(6) $$\sum_{\substack{j=1 \\ u_j \notin \{a,b\}}}^{N} \left\| \left(v_s(u_j) - d_j\right)^{\mathrm{T}} n_s(u_j) \right\|^2 + \sum_{\substack{j=1 \\ u_j \in \{a,b\}}}^{N} \left(v_s(u_j) - d_j\right)^2 \rightarrow \min_{\dot{s}} .$$

We compute $(\dot{s}_1, \dot{s}_2, ..., \dot{s}_n)$ and update the vector of shape parameters $(s_1 + \varepsilon\dot{s}_1, s_2 + \varepsilon\dot{s}_2, ..., s_n + \varepsilon\dot{s}_n)$.

More detailed information can be found in [1].

4 The algorithm

In this part we introduce our implementation of the algorithm. The input is a finite set of points in the plane. We start with computer-generated data. We consider closed B-spline as a planar parametric curve which is modified by the evolution process. We choose the initial position of closed B-spline curve. Now the shape parameters are the control points. Our curve has 8 control points and it is curve of degree 3; see figure 14.

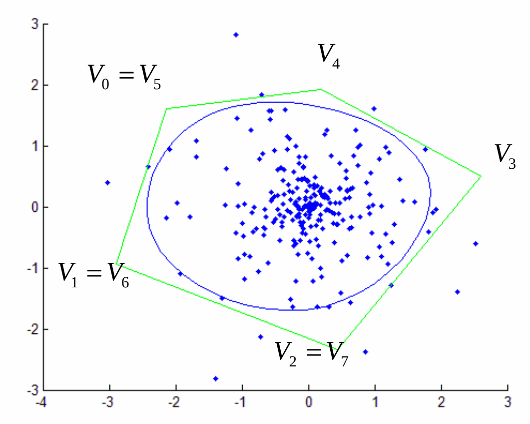

Fig. 14: The initial closed B-spline curve

Closed B-spline curve is modified in time and moves closer to points which are generated in dependence on some predefined condition. For example, we generate this points such that we compute the normals of the curve at its points, define the density of points on the curve and add multiple of this density to points on the curve. We get the points to which closed B-spline curve travels; see figure 15. We repeat this computation in each step of the algorithm.

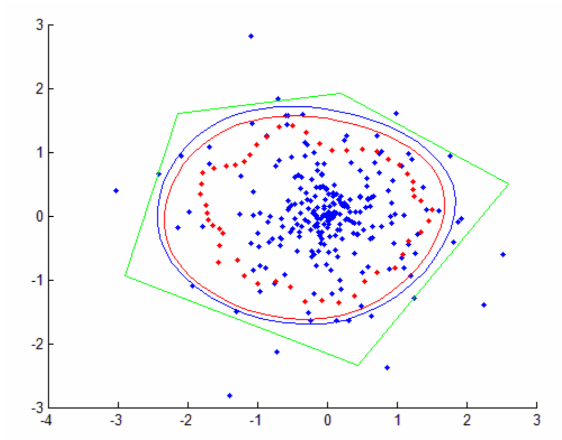

Fig. 15: Points to which closed B-spline curve travels (red), modified closed B-spline curve (red) – one step of the algorithm

4.1 Examples of closed B-spline evolution

We choose MATLAB for implementation of this algorithm. Figure 16 illustrates the outputs of our algorithm.

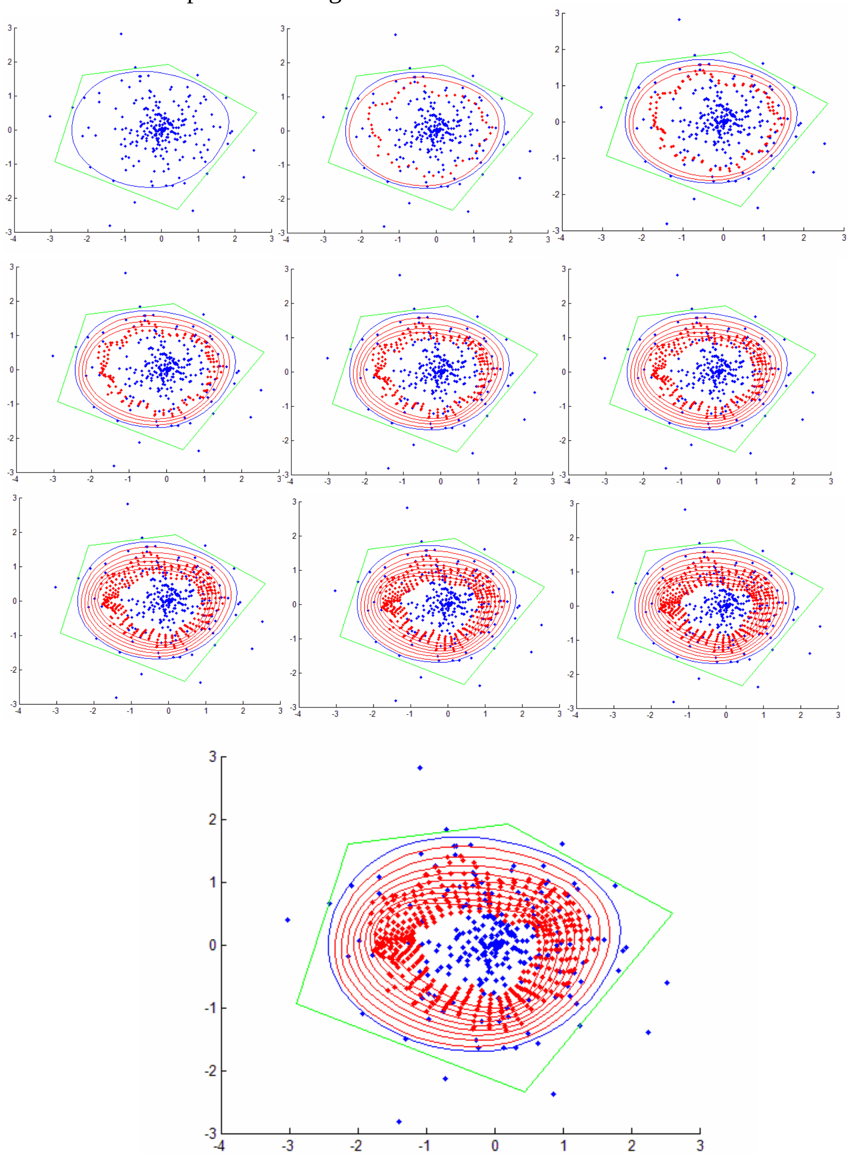

Fig. 16: Closed B-spline evolution – 10 iterations

5 Conclusion

Our contribution introduced the outline of several data depth measures. We explain curve evolution and apply it as a solving method for finding the depth contours. We suggested the algorithm of closed B-spline curve evolution and showed several examples of the outputs of our algorithm.

In our future work we will focus on the improvement of implementation of the algorithm and improve the generating of the points to which closed B-spline curve travels. Our next goal is to work with real data.

Acknowledgements

This research was supported by the project no. 201/08/0486 of the Czech Science Foundation.

References

[1] M. Aigner, Z. Šír, B. Jüttler: *Least-squares approximation by Pythagorean hodograph spline curves via an evolution process*, GMP, Springer LNCS, Vol. 4077, 2006

[2] R. Y. Liu, R. Serfling, D. L. Souvaine: *Data depth: Robust multivariate analysis, computational geometry and applications,* DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Vol. 72, 2003

[3] R. Y. Liu: *On a notion of data depth based on random simplices*, Annals of statistics, Vol. 37, 2009

[4] O. Vencálek: *Depth based classification*, ASMDA, 2009