Proximity graph depth, depth contours, and a new multimodal median*

Eynat Rafalin [†] Kathryn Seyboth[†] Diane Souvaine[†] [‡]

Abstract

We propose *proximity graph depth* as a class of depth functions, based on the minimum path length along proximity graph edges to the convex hull of a point set. We analyze the characteristics of several *proximity graph depth* functions both theoretically and experimentally, define depth contours enclosing regions of increasing depth, and present algorithms for calculating depth values in a point set and depth contours.

The contribution of this paper is in the novel approach for analyzing depth in multimodal data sets. Most existing depth function do not cope with multimodality or distributions with more then one center. We define seeds, the multimodal version of the depth median, as an estimator for the centers of the multimodal data sets and present experimental results that demonstrate that, unlike most depth functions, the proximity graph depth can indeed distinguish multimodal data sets.

1 Introduction

The proliferation of data enabled by the *information age* has driven the development of new data analysis techniques. The concept of *data depth* [21, 17] has been developed over the last decade as a method of multivariate data analysis in which no distributional assumptions are needed. Proposed *data depth* measures are inherently geometric, with a numeric value assigned to each data point that represents its *centrality* within the given data set. Most depth measures are defined with respect to a probability distribution or to a data set. We direct our attention to the latter case, and apply computational geometry techniques to its study and analysis. Many depth measures have been defined and heavily studied: e.g. halfspace or Tukey depth [27][1], convex-hull peeling depth [5, 3], and simplicial depth [17].

Two key problems emerge. First of all, proponents of *data depth* argue the effectiveness of using these methods in higher dimensions and the insights to be gained. If these depth measures were effectively computable in higher dimensions, this would indeed be the case. However, in practice, the computational complexity of most well-behaved measures is exponential in dimension. For example, the best known algorithms for computing the halfspace depth of n data points in two-dimensions run in $\Theta(n^2)$ time [18] and the generalization of these algorithms to d-dimensions will run in $\Theta(n^d)$ time, clearly impractical for dimensions larger than 5. An approximation algorithm for computing the halfspace depth in $\mathbb{R}^d$ in $O(mp^3 + mpn)$ time, where m is the number of p-subsets used was suggested in [22], but with no guarantee on the quality of the approximation. For these types of *data depth* to be practical in higher dimensions, faster algorithms, either exact or approximate, will need to be developed.

The second issue is equally important. The proponents of *data depth* advocate the fact that no distributional assumptions are needed, but indeed a key distributional assumption **is** in fact being made. The *four desirable properties* of depth function, suggested by statisticians [30] assume that each data set is centrally symmetric. If these techniques are to be applied broadly, however, in an attempt to extract information from large experimental data sets which may reflect multiple phenomena, these assumptions of centrality may obscure the fact that the data sets are multi-modal. There is need for geometric *data depth* metrics that can report a single center, if indeed there is only one, but can also detect the likelihood that the data reflects multiple distributions.

A couple of candidate *data depth* metrics that try to combat these problems have been suggested. Under the *Delaunay depth* metric, the depth of a point in a data set is the path length from that point to the convex hull of the data set along the graph that is the Delaunay triangulation of the point set. Green and Sibson [10] mentioned the idea of using the path length to the convex hull along Delaunay triangula-

*Partially supported by NSF grant CCF-0431027, Tufts Provost's Summer Scholars, and CRA DMP.

[†]Department of Computer Science, Tufts University, Medford, MA 02155, {erafalin, kseyboth, dls}@cs.tufts.edu.

[‡]2005-06 Radcliffe Institute Fellow & MIT Visiting Scientist

[1]The **half-space depth** of a point x relative to a set of points $\mathcal{S} = \{X_1, ..., X_n\}$ is the minimum number of points of $\mathcal{S}$ lying in any closed half-space determined by a line through x.

tion edges as a depth measure. Recently Abellanas *et al.* [1] have proved several combinatorial properties associated with *Delaunay depth* in the planar case. They noted that *Delaunay depth* does detect multiple "centers" of maximal depth. Note, however, that computing the Delaunay triangulation of n points in d dimensions has a lower bound of $\Omega(n^{\lceil \frac{d}{2} \rceil})$. All previous discussion is restricted to the Delaunay Triangulation and does not include experimental analysis.

In this paper, we propose *proximity graph depth* as a class of depth functions generalized from the concept of *Delaunay depth* and provide a framework from which to assess the efficacy of the various instantiations. In addition to analyzing *Delaunay depth*, we study the range of *β-skeleton graph depth* (including *Gabriel Graph depth* and *Relative Neighborhood Graph depth*). These principles can be viewed as an extension of *computational morphology* [25] where "a computational geometrical structure is intended to extract the *form* of the input." The underlying algorithms that we will use are classic, rather than new. The contribution of this paper is in the generalization of *Delaunay depth* to any type of proximity graph, in the definition of depth contours and estimators for the centers in a multimodal data set, but *most importantly* its contribution is in the experimental analysis of the efficacy of the depth functions based on these individual *proximity graphs*. We show that the *proximity graph* depth functions have the ability to detect multi-modal data sets, and that certain β-skeleton graph depth functions provide output comparable in quality to *Delaunay triangulation* depth while having only a linear computational dependency on dimension.

2 Background

2.1 Proximity Graphs

Proximity graphs are graphs in which points close to each other by some definition of closeness are connected [13]. We concentrate our analysis on the *Delaunay triangulation* [8] and *β-skeletons* [14] which are a parameterized family of neighborhood graphs, which include as a special case the *Gabriel graph* and the *relative neighborhood graph*. Other types of proximity graphs, not studied in this work, include variants like the *k-relative neighborhood graphs* [13], *rectangular influence graph* [12], *sphere of influence graph* [26] and *γ-neighborhood graphs* [28]. We denote by $\delta(p, q)$ the Euclidean distance between points p and q.

Definition 2.1 *The* ***Delaunay triangulation*** *(DT) of a d-dimensional point set S is the simplicial decomposition of the convex hull of S such that the d-sphere defined by the points of every simplex in the decomposition contains no point $r \in S$ [8].*

This decomposition is the dual of the *Voronoi diagram* and is unique for every set of points [6].

Definition 2.2 *The β* ***skeleton*** *of a point set S in $\mathbb{R}^d$ is the set of edges joining β-neighbors.*
Points p and q are ***lune-based*** ***β-neighbors*** *for $\beta \geq 1$, iff the lune defined by the* intersection *of the spheres centered at $(1 - \frac{\beta}{2})p + \frac{\beta}{2}q$ and $(1 - \frac{\beta}{2})q + \frac{\beta}{2}p$, each with radius $\frac{\beta}{2}\delta(p, q)$, contains no point $r \in S$.*
Points p and q are ***circle-based*** ***β-neighbors*** *for $\beta \geq 1$, iff the lune defined by the* union *of the two spheres of radius $\frac{\beta}{2}\delta(p, q)$ contains no point $r \in S$.*
Points p and q are ***β-neighbors*** *for $\beta < 1$, iff the lune defined by the intersection of the two sphere of radius $\frac{\beta}{2}\delta(pq)$ which contain p and q in their boundary contains no point $r \in S$ (for $\beta < 1$ the lune-based and circle-based neighbors are identical).*

For $\beta > 1$ the **lune-based** **β-skeletons** are planar and monotonic with respect to β: $G_{\beta_1}(S) \subset G_{\beta_2}(S)$, for $\beta_1 < \beta_2$. The *Gabriel graph* (GG) [9] is the lune-based 1-skeleton while the *relative neighborhood graph* (RNG) [24] is the lune-based 2-skeleton.

The **circle-based** **β-skeletons** for $\beta > 1$, are not necessarily planar and have a reverse monotonic relation with respect to β: $G_{\beta_1}(S) \subset G_{\beta_2}(S)$, for $\beta_1 > \beta_2$.

For $\beta < 1$, as β becomes smaller, the β skeleton tends towards the complete graph.

In a 2-dimensional planar graph with n points, the number of edges is linear in n: $|E| \leq 3n - 6$. Any proximity graph guaranteed to be planar (including the DT, GG, and RNG) is therefore linear in the size of the point set. A d-dimensional proximity graph approaches a complete graph as $d \to \infty$, so the number of edges in a graph of n points approaches $\frac{n(n-1)}{2}$.

2.2 Depth Contours

Depth Contours [27] are nested regions of increasing depth and serve as a topological map of the data. The jth depth contour consists of all those points in space of depth $\geq j$. Contours have applications in visualization and quantification of data sets. Section 6 defines and studies depth contours for the proximity graph depth measure.

2.3 Modes and Medians

It is often useful to find a single point that approximates a data set for simplification of calculations and analysis. In one dimension, this value is easily located as the central value, the median of the set. With the addition of dimensions, though, the goal of finding a suitable median becomes much more complicated. See [23, 2] for related surveys.

In depth-based statistics the deepest point (the median) serves as an estimator for the center of the data set in any dimension. It is possible for several points to tie for the deepest depth and then the median is the group of deepest points[2].

Definition 2.3 *The **median** of a point set S under some depth measure $D : S \to \mathbb{R}$ is the set of points M such that $\forall p \in M$, $D(p) \geq D(q)$ $\forall q \in S$.*

The use of a *single* point or group of points as the median relies on the assumption of unimodality that is common in depth measures. If the depth function reaches maximality at multiple points throughout the data set, a single median consisting of those points will be useless as an approximator of the set.

The *mode* of a point set is the most common value obtained in a set of observations [29]. We use the term mode flexibly and refer to a *bimodal* (or *multimodal*) distribution or point set as one having two (or more) local maxima. The multiple maxima can be created, for example, from two different unimodal distributions that were combined.

Often clustering algorithms are used to detect the points associated with each mode of the data set. This association, however, will not necessarily attribute a data point to the center of the distribution where it originated. For example, points located between two centers and far from each could be assigned to either of the two clusters.

3 Proximity Graph Depth

For any proximity graph we define a depth measure using each point's minimum path length along graph edges to the convex hull of the data set S. To distinguish between points included in the point set S and those not, we term members of S as *points*, while *positions* refer to points in space that are not in S.

Definition 3.1 *The **[proximity graph] depth** of a point p relative to a point set $S = \{p_1 \ldots p_n\}$ is the minimum number of edges in the [proximity graph] of S that must be traversed in order to travel from p to any point on the convex hull of S.*

Definition 3.2 *The **[proximity graph] depth** of a position $x \notin S$ relative to a set of points $S = \{p_1 \ldots p_n\}$ is the minimum number of edges in the [proximity graph] of $S \cup x$ that must be traversed to travel from x to any point on the convex hull of $S \cup x$.*

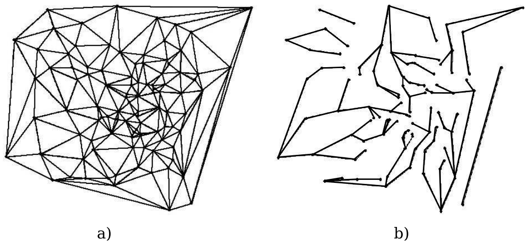


Figure 1: a) The DT of a 100-point data set and b) breadth-first search tree inward from convex hull vertices on the same point set.

3.1 Overall Complexity

The depths of all points in a proximity graph can be determined in linear time in the number of edges in the graph by using a breadth-first search (BFS) of the graph, beginning at every point on the convex hull of S (Figure 1). The depth of a point is its depth in the BFS tree. Assignment of all depths of a point set is accomplished by (1) Computation of the proximity graph; (2) Location of all convex hull points; and (3) Breadth-first search of the proximity graph.

In two dimensions there are optimal $O(n \log n)$ time algorithms to compute the DT [8], the circle-based β-skeletons for $\beta \geq 1$, and the lune-based β-skeleton for $1 \leq \beta \leq 2$ [14, 16, 11]. The lune-based β-skeletons for $\beta > 2$ and the β-skeletons for $\beta < 1$ can be computed in optimal $O(n^2)$ time [14, 11]. Points on the convex hull can be determined in $O(n \log n)$ time [20], for an overall time requirement of $O(n \log n)$ for the DT and $O(n^2)$ or $O(n \log n)$ for the β-skeleton.

In dimensions higher than 2, the DT can be calculated in $O(n^{\lceil \frac{d}{2} \rceil})$ time [7]. The β-skeletons require checking n points for interiority on n^2 lunes, which requires a distance calculation for a total of $O(dn^3)$ time. More efficient algorithms for specific graphs like the GG or RNG or for 3-dimensional space are known [13]. The set of points on the convex hull of the set can be found in $O(mn)$ time, where m is the number of extreme points [19]. Breadth-first search then requires linear time in the size of the proximity graph. Clearly, the time complexity in higher dimensions is dominated by the computation of the proximity graph itself. Assignment of all depths, then, has a total complexity of $O(n^{\lceil \frac{d}{2} \rceil})$ time for Delaunay depth and $O(dn^3)$ time for the β-skeleton depths. The exponential dependence on dimension for calculating Delaunay depth makes it impractical for use in high dimensions.

[2] In some cases the center of mass of the set of deepest points is used as a single estimator for the deepest point.

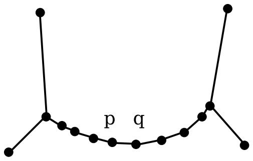


Figure 2: The 2-skeleton (RNG) for a set of 14 points. Points p, q are deepest but are also very close to the convex hull.

3.2 Performance Issues with Sparse β-skeletons

Skeletons with $\beta > 1$ are sparse, which makes them ineffective for use in the proximity graph depth measures in two dimensions: a point infinitely close to the convex hull can have an arbitrarily large depth (Figure 2). The sparsity makes the graphs overly sensitive to differences in the density of points in S: pockets of only slightly higher density than surrounding areas can create regions of high depth that are inconsistent with the overall shape of the data. Therefore, discussion will now focus on the DT and β-skeletons with $\beta \leq 1$.

Experimental analysis (see Section 4.4) shows that although for some $\beta \leq 1$, the β-skeletons do not have consistent performance and do not capture the structure of the set well for 2-dimensions, graphs with $\beta \in [.96, .97]$ exhibit performance very similar to that of the DT, at greater computational efficiency.

4 Multimodality and Seeds

Data depth is often used to estimate the location of a given point set, where the median of the set is defined as the deepest point(s) (Section 2.3). Medians are not well suited to serve as estimators in multimodal situations, estimating a set using a simple median can ignore local maxima that represent multiple modes. Instead, we define an estimator for the center of a data set that can cope with multimodal situations.

Definition 4.1 *A* ***seed*** *of a point set S under some depth measure $D : S \to \mathbb{R}$ is a connected set of points $T \subset S$ such that $\forall p, q \in T, D(p) = D(q)$ and $\forall r \in S, r \notin T$ adjacent to some $u \in T, D(r) < D(u)$.*

Most depth functions are unable to discern or cope with multimodal data sets. For example, Figure 3(a) shows the halfspace depth contours for a bimodal data set. The halfspace median is located in the region between the two clusters, clearly not a good estimator for the sets. The proximity graph seeds are powerful and unique in that they are sensitive to variance in density within a point set and can correctly locate several centers of the data set. Figure 3(b) shows the DT-depth contours for the same point set. As can clearly be seen, the DT-depth function correctly recognizes the two centers.

In the following sections we provide experimental and theoretical analysis of the concept of seeds. Section 4.1 demonstrates quantitatively how the proximity graph depth can discern multimodal data sets from unimodal sets and seeds can correctly estimate the centers of the different clusters. Section 4.2 presents algorithmic analysis of the concept of seeds. Section 4.3 suggests improvements to the concepts and Section 4.4 presents experimental results testing the concept on randomly generated data sets with one or two modes.

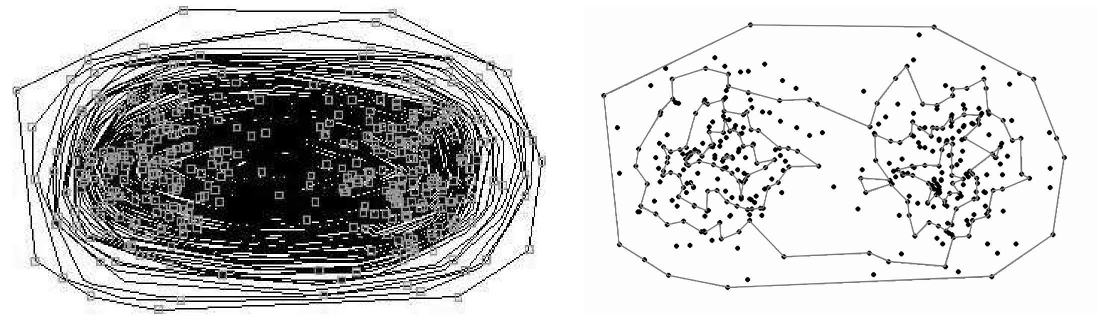
4.1 Quantitative Analysis of Proximity Depth for Unimodal and Multimodal Data Sets

To quantify the ability of the proximity graph seeds to discern unimodality and bimodality we conducted an experimental analysis on unimodal and bimodal sets in two dimensions. Bimodal sets were created by combining two 200-point normal distribution sets, each with x and y standard deviations of 10. They began centered at the same x-coordinate, but were then separated in increments of 10. The unimodal sets began with an x-coordinate standard deviation of 10, which was gradually increased to stretch the unimodal set in the x-direction. We then plotted the standard deviation of the x-values of the seeds against the x-coordinate range of the set. Bimodal sets that are close behave very similarly to unimodal sets, but as the bimodal sets pull apart their seeds separate, illustrating the bimodal behavior (Figure 4). Similar behavior was observed for each of the proximity graph depth measures. Our code was written in C++ using the LEDA library [15].

Points of different clusters are not necessarily distinguished by the proximity graph depth measures. Distributions that are too close behave as a single mode; those separated by large empty regions appear unimodal as the dearth of points between the clusters prevents paths from traveling inward quickly from the convex hull. However, two clusters that are well separated are easily distinguished using proximity-depth.

4.2 Seeds Algorithms and Complexity

Finding seeds requires a recursive search to locate all points associated with the seed and to locate all points in S that are connected to that seed. The basic process to compute the seeds is that of comparing the



(a) **(b)**

Figure 3: Depth contours for a bimodal point set consisting of two groups of 200 normally distributed points. (a) Halfspace depth contours. The deepest contour is in the region between the centers of the two sets. (b) Delaunay depth contours (for clarity only every second contour is drawn). The contours correctly recognize the two centers.

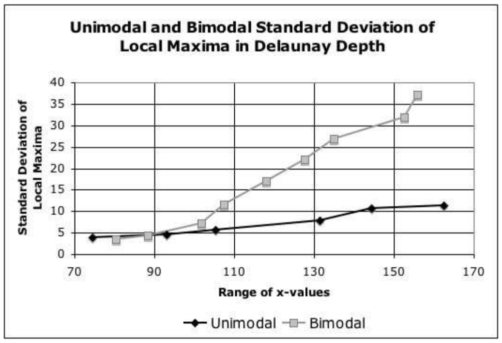


Figure 4: Computation of points of local maximal depth (seeds) using the DT depth function for unimodal and bimodal point sets. For every x-value X the unimodal point set was constructed to have comparable width to the width of the bimodal point set, whose separation is X. The standard deviation of the x-values of the seeds were computed. The results support our assumption, that in a bimodal point set we expect to find wider gaps between the seeds and therefore larger standard deviation compared to unimodal point sets.

depth of point p to the depth of its neighbors in the proximity graph and checking whether it is deeper or of the same depth as its neighbors (*compare(p)*).

One method of computation searches the list of points to locate those that are deeper or of the same depth as their neighbor. In this case, every connected grouping of points must be checked and there can be $O(n)$ such groupings. The *compare(p)* process is called recursively for each point p in a grouping exactly once and obtains a yes/no answer for that point before returning. Because *compare(p)* eliminates points of lower depth than p, it is called at most n times. Each call iterates through some portion of p's adjacency list. Each edge in the graph represents two entries in adjacency lists, which means that even if every single point in the adjacency list is accessed at every call, each edge is considered exactly twice, producing an algorithm with a running time that is linear in the size of the proximity graph. The size of the graph is linear in two dimensions and up to quadratic in higher dimensions (see Section 3). The process does not add to the overall complexity, because construction of the graphs requires at least as much time.

4.3 Improvements to the Concept of Seeds

Proximity graph measures can be overly sensitive to differences in density of points and create more seeds than desirable. If so, it may be necessary to prune the set of seeds. For values of $\beta < 1$ near 1, issues similar to those in the RNG occur: there can be points very close to the convex hull that achieve high depth because of the sparsity at the extremity of the set (Figure 2b). This means that there can be local maxima (seeds) at the very edge of the point set, which is clearly not desirable. Here, we propose several ways of combating erroneous seeds for a given β and evaluate them experimentally. Since the number of seeds is monotonic relative to β (for $\beta < 1$), an alternate method of handling erroneous seeds is to change the value of β.

4.3.1 Significant Maxima

One way to narrow the definition of a seed is to include only those seeds that are local maxima, and whose neighbors would be maxima were the points of the seed eliminated, i.e. the seed must be a local maxima by two depths rather than one:

Definition 4.2 *A* ***significant seed*** *of a point set S under some depth measure $D : S \rightarrow \mathbb{R}$ is a seed*

$T \subset S$ which for $U = \{r | r$ adjacent to $T\}$, U is a seed of $S \backslash T$.

Unfortunately, it is possible to have two seeds at the center of a mode which are separated by only one point. They form double peak in the depth, rather than a single, but if the double peaks are even and separated by two proximity graph edges, neither of them registers as a significant seed and that mode would be eliminated from notice. A simple check fixes this problem.

If there is a target or limit for the number of seeds, it is possible to modify the definition to require larger and larger peaks. For example, one could require that the removal of both the seed and the second depth leave the third deepest group as a seed. This process could be expanded to achieve the desired number of seeds, which are called the *most* significant seeds.

4.3.2 Convex-Hull Path Origins

Another way of assessing the validity of a seed is to find the number of convex hull breadth-first search roots that could have initiated paths to that seed. If many points could have created the path that labeled a point, the seed is relatively central compared to a seed for which there were only one or two path origins.

Definition 4.3 *The **CH-value** of a seed T is the number of convex hull vertices that can be reached by a path originating at T for which the depths of points visited along the path is strictly decreasing.*

Seeds can be assigned a **CH-value** by beginning at a seed and breadth-first searching outward, under the condition that every edge traversed must decrease in depth by one. The number of convex hull points reached in this manner is the **CH-value**. Once each seed is assigned its **CH-value**, it is possible to weed out weak seeds by, for example, using only those with **CH-value** great than 2 and over half of the maximum: these seeds are called **CH-point seeds**.

Limiting seeds to those having **CH-value** greater than 2 is desirable because the bad case of seeds near the convex hull (Figure 2) contains seeds with paths to only two convex-hull points.

4.4 Experimental Analysis of β-skeletons and Seeds Location Schemes

In order to study different β-skeletons and how their performance compares with the DT, we generated 300 normally distributed point sets with 400 points each and calculated the average number of seeds produced. The code was written in C++, using the LEDA library [15]. We used only seeds with **CH value** that is at least half of the maximal value, and then plotted the average number of seeds against the average deepest depth attained by the point set (Figure 5). The locations of the points indicate that the values $\beta = .962$ and $\beta = .970$ best approximate the performance of Delaunay Depth for these two characteristics. The weakness of the GG (1-skeleton) is also very visible here, as it finds many more seeds in a unimodal data set than the other graphs.

We then used the same point sets to compare different methods of seed determination. Four types of graphs were generated: the DT, the GG and the .962-skeleton and .970-skeleton. For each graph, three seed computing schemes were used (simple seeds, significant seeds and CH-point seeds). Since the sets are unimodal, a low number of seeds is desirable.

The DT demonstrated the best performance, but the .962-skeleton and .970-skeleton graphs performed well. The GG method reports more seeds than are desirable (average of 11, compared to < 6 for the other graphs). It is much more sparse than the DT, so is more sensitive to changes in density of points, which in turn creates local maxima that do not necessarily lie in separate modes of the set.

Each proximity graph depth measure reports dramatically fewer significant seeds and/or CH-point seeds.

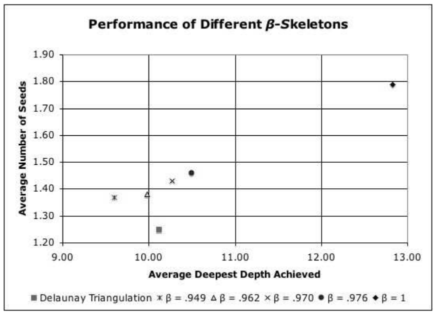


Figure 5: Performance comparison of proximity-depth schemes for 300 normally distributed unimodal point sets with 400 points. A low average number of seeds and a high average number of deepest depth values is desirable. The .962-skeleton and .970-skeleton perform most similarly to the DT The weakness of the GG (1-skeleton) is also very visible here, as it finds many more seeds than the other graphs.

Average Number of Seeds over 300 Runs

Graph	Seeds	Significant	CH-point
		Seeds	Seeds
Gabriel	11.00	2.20	1.79
Delaunay	2.80	1.00	1.25
.962-skeleton	4.70	1.25	1.38
.970-skeleton	5.12	1.27	1.43

Table 1: A comparison of seed location schemes in 400 point 2-dimensional point sets following a normal distribution. Sets are unimodal, so a low number of seeds is desirable. The DT demonstrated the best performance

5 Features of Proximity Graph Depth

5.1 Depth limits

Given a set S of n points with $n \geq 3$, the maximum *Tukey* depth of any point or position must lie between $\lfloor \frac{n}{3} \rfloor$ and $\lfloor \frac{n}{2} \rfloor$. The span of maximum depths for *proximity graph depth* is far broader. If all points of S lie on its convex hull, the highest proximity-depth of a point $p \in S$ is 0. The depth of any position inside the convex hull has depth 1. Thus, the minimum deepest depth attained is 1.

On the other hand, if the points in S can be placed arbitrarily, the maximal depth of a position in the DT is $\lceil \frac{n+1}{3} \rceil - 1$: an arrangement of equilateral triangles radiating outward, each with the same center point and same orientation will create a set in which each depth contour is a triangle and each depth is attained by only three points.

The maximal β-skeleton depth can be even higher. For example, it is possible to construct a point set such that the GG (1-skeleton) consists of a convex k-gon and a path consisting of an arbitrary number of vertices within the convex k-gon. The point at the end of the path has depth $n - k - 1$, which can be $n - 4$.

Thus, the deepest position relative to any set S with $|S| > 2$ will have depth i such that $1 \leq i \leq \lceil \frac{n+1}{3} \rceil - 1$ in the DT depth and as much as $n - 4$ in the β-skeleton depth.

5.2 Robustness

Outliers are observations that deviate from the main part of the data and can have an undesirable influence on its analysis. *Robustness*, the consistency in performance of a statistical measure in the presence of outliers, is defined by its *breakdown value* [4]:

Definition 5.1 *The* ***breakdown value*** *of an estimator R of the data set S is the smallest fraction of contamination points that must be added to S to shift the estimator arbitrarily far:*

$$\epsilon^*(R, S_n) = \min \tfrac{m}{n+m} \textit{ such that}$$
$$sup_{S_{n+m}} \|R(S_{n+m}) - R(S_n)\| = \infty$$

where S_{n+m} is a contaminated data set obtained by adding m arbitrary points.

The breakdown value indicates the imperturbability of an approximator: how stable it is in the presence of outliers. A one-dimensional median, for example, requires the addition of n points to move it outside the original set, for a breakdown value of $\frac{1}{2}$. A breakdown value of $\frac{1}{2}$ is in fact maximal for any statistical estimator [4], and has been achieved by depth measures including Tukey depth [27].

The proximity-graph-based median can be corrupted through the construction of an erroneous cluster that attains depth greater than the true median: by strategically placing points at infinity such that the maximal depth of the cluster increases with the addition of every three points, it is possible to build a cluster of points with maximal depth i with only $3i + 1$ additional points (see also Section 5.1). The breakdown value of a set with i depth values, then, requires the formation of a set of depth $i + 1$ at infinity, which requires $3(i + 1) + 1 = 3i + 4$ points.

Since the maximal depth of a set can range from 1 to $\lceil \frac{n+1}{3} \rceil - 1$ (for DT) or $n - 4$ (for β-skeleton), the breakdown value also has a wide range, dependent on the maximal depth of the set. A shallow point set (one with median of low depth) will have a low breakdown value, whereas a deeper point set will have a relatively high breakdown value. The lowest possible number of contamination points needed for breakdown occurs in the case of maximal *point* depth of 0. This case will require $3(0) + 4 = 4$, which is a breakdown value of $\frac{4}{n+4}$.

The lower bound is quite small, as analysis on a set of **any size** could be made invalid by the inclusion of only four outliers. In the best case, however, there are a maximum number of depths and the number of contamination values added is linear with the number of points. For example, for the DT the number of contamination values added is:

$$3(\lceil \tfrac{n+1}{3} \rceil - 1) + 4 = \lceil n + 1 \rceil + 1 = n + 2$$

for a breakdown value of $\frac{n+2}{n+(n-1)} = \frac{n+2}{2n+2}$. For very large sets, in the best case the breakdown value approaches $\frac{1}{2}$: $\lim_{n \to \infty} \frac{n+2}{2n+2} = \frac{1}{2}$. The upper bound is therefore maximal.

6 Depth Contours

Depth Contours, as described in Section 2.2 are regions of increasing depth that provide a topological map of the data. We provide an effective method of defining depth contours for the DT depth function

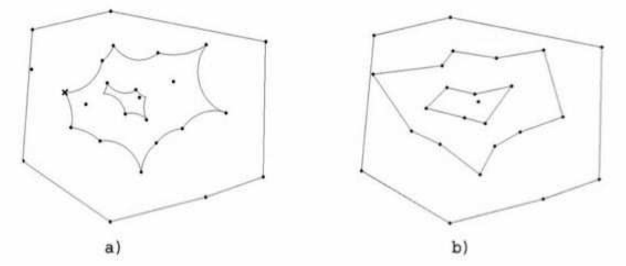


Figure 6: ((a) Delaunay contours and (b) simplified Delaunay contours for a set of 25 points. In the Delaunay contours position X is a vertex of the Delaunay depth contours but not a point of the set S. In the simplified contours all vertices are points in S.

(Section 6.1) and lune-based β-skeleton depth functions (Section 6.3).

6.1 Delaunay Contours

The depth of a position in the DT measure is defined by its depth upon its hypothetical addition to the set. Each triangle in the DT corresponds to an empty circle defined by the three points of the triangle. A point p added to a triangulated set will be connected to all defining points of the circles which contain p. In order for position x to be of depth $\geq j$, it cannot be in any Delaunay circle with a defining point of depth $< j - 1$. Therefore the borders of the contours of a point set, those places where the depth of positions change, are defined by arcs of the Delaunay circles (Figure 6a).

Definition 6.1 *The* j***th Delaunay depth contour*** *of point set S is the area inside the convex hull of S which is not inside any Delaunay circle with a defining point of depth $< j - 1$.*

The contours are nested pseudo-polygons, with edges that are arcs rather than straight segments.

A point of the set S need not lie on the boundary of a contour. Points that define boundaries may lie between the Delaunay boundaries rather than on them (see Figure 6a). In a multimodal set, some of the contours may not be continuous; each local maximum can have its own section of the jth contour.

For more properties of Delaunay depth contours in the plane, see Abellanas *et al.* [1].

6.2 Simplified Delaunay Contours

It is possible to simplify the boundaries of the Delaunay contours by changing the definition of the contours so that they are defined by straight lines rather than by arcs (Figure 6b).

Definition 6.2 *The* j***th Delaunay simplified contour*** *of point set S is the area inside the convex hull of S which is not inside any Delaunay triangle with a defining point of depth $< j - 1$.*

This definition creates a new classification system for the depth of positions. Segments on the boundaries correspond to an arc between the same two endpoints in the detailed boundaries, so the simplification maintains the general structure of the contours.

Positions affected by the slight change are those that lie between the Delaunay edge used in the simplification and the corresponding arc in the general contour boundary. The arc is more central to the data set, so positions can have a deeper depth in the simplification than in the original contours.

The simplified contours are not always convex, but are simple polygons or groups of simple polygons. Every point on a boundary has outward visibility to the next boundary; no part of the polygon obstructs its view to the contour of next lower depth. This characteristic is inherent in the definition of the depth measure, as every point must have a proximity graph edge to a point of lower depth.

The simplified contours have the useful characteristic that all vertices of the boundaries of the contours are points in S. This is not the case for the version using arcs, because the arcs can intersect at positions that are not part of the point set (Figure 6a).

6.3 β-skeleton Depth

In order to define contours for lune-based β-skeleton depth, it is necessary to refer to the angle of the arcs of the defining lune.

Definition 6.3 *The* ***arc angle****, θ_β of a β-skeleton is the angle of the arcs forming the lunes that define the graph.*

A segment p_1p_2 in the β-skeleton of set S with $p_1, p_2 \in S$ is defined by an empty lune (Section 2). A position p added to S connects to those points $q \in S$ for which the lune defined by p and q is empty.

Lemma 6.4 *Given points $p, q, r \in S$, r is interior to lune(p, q) with arc $\angle\theta_\beta$ iff angle $prq > \pi - \frac{\theta_\beta}{2}$ (Figure 7a).*

Proof 6.5 *Assume point r is on the boundary of the lune, then w.l.o.g. it is on the boundary of the circle centered at c. Let c be the center of one of the circles defining the lune(p, q), that passes through point r. Points p, q are on the boundary of the circle. Then if $\angle pcq = \theta_\beta$, $\angle cpq$ and $\angle cqp$ are each $\frac{\pi - \theta_\beta}{2}$. Now, let $\angle qpr = \alpha$ and $\angle pcr = \phi$. Then, because segments*

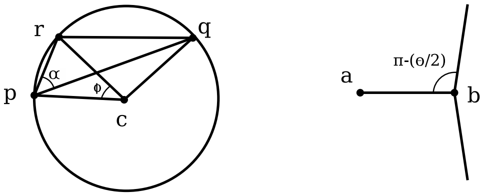


Figure 7: Proof of Lemma 6.4. a) $\angle prq = \pi - \frac{pcq}{2}$. b) The boundary dividing points that can connect to a in the β-skeleton graph from those for which b is in the way.

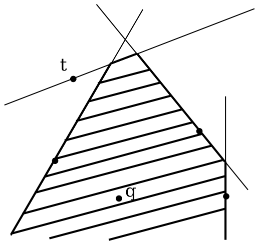


Figure 8: The region of influence in the 1-skeleton (Gabriel Graph) for point q. Point t, though not connected to q in the GG, participates in defining the region of influence of q.

pc and rc are radii and thus the same length, $\angle cpr = \angle crp = \frac{\pi-\phi}{2}$. *But* $\angle cpr = \alpha + \frac{\pi-\theta_k}{2}$ *so* $\alpha = \frac{\theta_\beta-\phi}{2}$. *By the same argument using triangle rcq, angle* $\angle pqr = \gamma = \frac{\phi}{2}$. *So:*

$$\angle prq = \pi - \alpha - \gamma = \pi - \frac{\theta_\beta - \phi}{2} - \frac{\phi}{2} = \pi - \frac{\theta_\beta}{2}$$

Since a point r on the boundary of the lune will have $\angle prq = \pi - \frac{\theta_\beta}{2}$, *any point s with* $\angle psq > \angle prq$ *will be interior to the lune.*

Lemma 6.4 indicates that in order for p to connect to $q \in S$, the angle formed by every other point $r \in S$ with p and q must be less than $\pi - \frac{\theta_\beta}{2}$.

For every $q \in S$ there is an associated convex region containing exactly those positions which, if added to S, would be connected to q.

Definition 6.6 *The* ***β-skeleton region of influence*** *of a point $q \in S$ is the set of all positions x inside the convex hull of S that are connected to q in the β-skeleton of $S \cup x$ and all points $p \in S$ connected to q in the β-skeleton of S.*

A point can affect the region of influence of q without being connected to q in the graph. In Figure 8a, for example, point t is not connected to point q in the GG, but nevertheless affects its region of influence.

Definition 6.7 *The* ***jth β-skeleton depth contour*** *of point set S is the area inside the convex hull of S which is not in the β-skeleton region of influence of any point of depth $< j - 1$.*

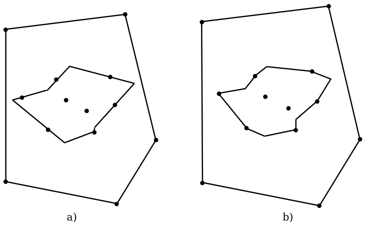


Figure 9: β-skeleton contours for a) $\beta = 1$ and b) β slightly less than 1.

These contours behave similarly to those of the DT, except that no simplification is necessary. Unlike the Delaunay simplified contours, however, the vertices on the boundaries of the contours need not be points in S (Figure 9).

7 Conclusions and Future Work

We proposed and evaluated depth measures based on path length in proximity graphs. We showed that β-skeleton depths can achieve comparable performance to DT depth in detecting the possibility of multiple underlying distributions, but can be computed far more efficiently than DT depth in high dimensions.

The work presented just begins to tap the potential of the use of proximity graphs as depth functions. Next steps include the study of additional proximity graphs and experimental analysis in higher dimensions. Additional effort will focus on possible uses of these measures for preprocessing or improving clustering algorithms.

Experimental analysis of the proximity graph abilities to discern unimodal and multimodal data sets (as described in Section 4.1) is already underway for dimensions greater then 2. In $\mathbb{R}^2$ the optimal β value to distinguish a bimodal from a unimodal distribution was empirically computed to lie between .96 and .97. We plan to determine how the optimal vales of β in other dimensions compare and the effect of using other models of distribution.

Acknowledgement: The authors would like to thank G. Toussaint for introducing us to the concept of Delaunay depth and initiating this work and to thank F. Hurtado for a fruitful discussion.

References

[1] M. Abellanas, M. Claverol, and F. Hurtado. Point set stratification and delaunay depth, 2005. ACM Computing Research Repository, cs.CG/0505017.

[2] G. Aloupis. Geometric measures of data depth. *DIMACS Series in Disc. Math. and Theoretical Comp. Sci.*, 2005. Submitted for publication.

[3] V. Barnett. The ordering of multivariate data. *J. Roy. Statist. Soc. Ser. A*, 139(3):318–355, 1976.

[4] D. L. Donoho and M. Gasko. Breakdown properties of location estimates based on halfspace depth and projected outlyingness. *The Annals of Statistics*, 20(4):1803–1827, 1992.

[5] W. Eddy. Convex hull peeling. In H. Caussinus, editor, *COMPSTAT*, pages 42–47. Physica-Verlag, Wien, 1982.

[6] H. Edelsbrunner. *Algorithms in Computational Geometry*. Springer-Verlag, 1978.

[7] H. Edelsbrunner and R. Seidel. Voronoi diagrams and arrangements. *Discrete and Computational Geometry*, 1:25–44, 1986.

[8] S. Fortune. Voronoi diagrams and Delaunay triangulations. In *Handbook of discrete and computational geometry*, CRC Press Ser. Discrete Math. Appl., pages 377–388. CRC Press, Inc., Boca Raton, FL, USA, 1997.

[9] K. Gabriel and R. Sokal. A new statistical approach to geographic variation analysis. *Systematic Zoology*, 18:259–278, 1969.

[10] P. J. Green and R. Sibson. Computing dirichlet tessellations in the plane. *The Computer Journal*, 21(2):168–173, 1978.

[11] F. Hurtado, G. Liotta, and H. Meijer. Optimal and suboptimal robust algorithms for proximity graphs. *Comput. Geom.*, 25(1-2):35–49, 2003. Special issue on the European Workshop on Computational Geometry—CG01 (Berlin).

[12] M. Ichino and J. Sklansky. The relative neighborhood graph for mixed feature variables. *Pattern Recognition*, 18(2):161–167, 1985.

[13] J. W. Jaromczyk and G. T. Toussaint. Relative neighborhood graphs and their relatives. *Proc. IEEE*, 80(9):1502–1517, sep 1992.

[14] D. G. Kirkpatrick and J. D. Radke. A framework for computational morphology. In G. Toussaint, editor, *Computational geometry*, pages 217–248. North-Holland, 1985.

[15] LEDA. Library of efficient data structures and algorithms. `www.ag2.mpi-sb.mpg.de/LEDA`.

[16] A. Lingas. A linear-time construction of the relative neighborhood graph from the Delaunay triangulation. *Comput. Geom.*, 4(4):199–208, 1994.

[17] R. Liu, J. Parelius, and K. Singh. Multivariate analysis by data depth: descriptive statistics, graphics and inference. *The Annals of Statistics*, 27:783–858, 1999.

[18] K. Miller, S. Ramaswami, P. Rousseeuw, T. Sellarés, D. Souvaine, I. Streinu, and A. Struyf. Efficient computation of location depth contours by methods of combinatorial geometry. *Statistics and Computing*, 13(2):153–162, 2003.

[19] T. Ottmann, S. Schuierer, and S. Soundaralakshmi. Enumerating extreme points in higher dimensions. In *Symposium on Theoretical Aspects of Computer Science*, pages 562–570, 1995.

[20] F. Preparata and S. Hong. Convex hulls of finite sets of points in two and three dimensions. *Commun. ACM*, 20(2):87–93, 1977.

[21] P. J. Rousseeuw. Introduction to positive-breakdown methods. In J. E. Goodman and J. O'Rourke, editors, *Handbook of discrete and computational geometry*, Discrete Mathematics and its Applications (Boca Raton), pages xviii+1539. Chapman & Hall/CRC, Boca Raton, FL, second edition, 2004.

[22] P. J. Rousseeuw and A. Struyf. Computing location depth and regression depth in higher dimensions. *Statistics and Computing*, 8:193–203, 1998.

[23] C. G. Small. A survey of multidimensional medians. *Internat. Statistical Review*, 58:263–277, 1990.

[24] G. Toussaint. The relative neighborhood graph of a finite planar set. *Pattern Recognition*, 12:261–268, 1980.

[25] G. T. Toussaint. Pattern recognition and geometrical complexity. In *Proc. Fifth International Conference on Pattern Recognition*, pages 1324–1347, 1980.

[26] G. T. Toussaint. A graph theoretical primal sketch. In G. T. Toussaint, editor, *Computational morphology*, pages 229–260. North-Holland, 1988.

[27] J. Tukey. Mathematics and the picturing of data. In *Proceedings of the International Congress of Mathematics*, pages 523–531, 1974.

[28] R. C. Veltkamp. The γ-neighborhood graph. *Comput. Geom.*, 1(4):227–246, 1992.

[29] E. W. Weisstein. Mode. From MathWorld, `http://mathworld.wolfram.com/Mode.html`.

[30] Y. Zuo and R. Serfling. General notions of statistical depth function. *The Annals of Statistics*, 28(2):461–482, 2000.