Selecting the Aspect Ratio of a Scatter Plot Based on Its Delaunay Triangulation

Martin Fink* Jan-Henrik Haunert* Joachim Spoerhase* Alexander Wolff*

Abstract

Scatter plots are diagrams that visualize sets of points in two dimensions. They allow users to detect correlations and clusters in the data. Whether a user can accomplish these tasks highly depends on the *aspect ratio* selected for the plot, i.e., the ratio between the horizontal and the vertical extent of the diagram. We argue that an aspect ratio is good if the Delaunay triangulation of the scatter plot has some nice geometric property, e.g., a large minimum angle or a small total edge length. In order to find an optimum aspect ratio according to a given criterion we present an algorithm that efficiently maintains the Delaunay triangulation of the point set when traversing all aspect ratios.

1 Introduction

Scatter plots are diagrams that visualize sets of points in the plane to allow humans to find patterns, clusters, and trends in the data. When drawing a scatter plot, one is usually free to choose its aspect ratio, that is, the ratio between its horizontal and its vertical extent. With a bad choice, however, humans may fail in recognizing a pattern in the data. While the automatic selection of a good aspect ratio for a line chart has been intensively discussed [1–3,8,9], methods that select the aspect ratio for a scatter plot are missing.

Most methods for aspect-ratio selection rely on properties of the line segments displayed in the diagram. To apply a line-segment-based method to scatter plots, Cleveland et al. [2] suggested to consider "virtual line segments", which humans may perceive though they do not physically exist. The virtual line segments may be the segments of a (virtual) polyline connecting all data points, the segments of a regression (poly-)line, or, as suggested by Talbot et al. [8] in order to deal with pairs of variables without a functional relationship, the segments of contour lines yielded by a kernel density estimator (KDE).

Our main concern with their method is that the KDE result (and the final aspect ratio) heavily depends on the aspect ratio of the input data: it makes a difference whether the input data is given, for example, in degree Fahrenheit or in degree Celsius. To overcome this deficiency, we define—based on the output diagram visible to the user (and thus on the aspect ratio chosen)—whether two points are linked via a virtual edge. We argue that the aspect ratio is good if the virtual edges have nice properties. We define a virtual line segment for each edge of the Delaunay triangulation $D(P)$ of the set P of points displayed. This generally defines a meaningful (usually termed *the natural*) neighborhood for P: if for two points $u, v \in P$ there exists a point $w \in \mathbb{R}^2$ such that both u and v are nearest neighbors of w in P, then there exists an edge $\{u, v\}$ in $D(P)$.

We now need to choose the aspect ratio of a scatter plot such that the displayed points have a nice Delaunay triangulation, for example, one of minimum total edge length. To solve this problem, we present an algorithm that traverses the space of aspect ratios while efficiently maintaining the Delaunay triangulation. This is closely related to the problem of maintaining a Delaunay triangulation for a set of continuously moving points. In fact, our setting corresponds to the special case where each point moves along a horizontal line at an individual but constant speed.

Roos [5] described a data structure for maintaining a dynamic Delaunay triangulation which takes $O(\log n)$ time per topological change of the triangulation. His result requires that the movement of the points meets a (weak) technical assumption that holds for many natural scenarios such as movement along parametric polynomial curves. The question of *how many* topological changes a dynamic Delaunay triangulation can undergo (under some weak and natural assumptions on the movement) is an important field of research. Recently, Rubin [6] showed that there are at most $O(n^{2+\epsilon})$ topological changes for a large class of movements (including our scenario). We argue that in our case there are in fact only $O(n^2)$ topological events. Since updating the Delaunay triangulation requires $O(\log n)$ time, we can traverse all possible topological Delaunay triangulations in $O(n^2 \log n)$ time.

2 Problem Statement

Given a set $Q = \{q_1, q_2, \ldots, q_n\} \subset \mathbb{R}^2$ of points, we search for a scale factor $s \in \mathbb{R}^+$ defining the set P of displayed points, i.e., the resulting scatter plot. We denote the coordinates of each point $q_i \in Q$ by x_i and y_i and require the scatter plot P to contain the point

[*] Lehrstuhl für Informatik I, Universität Würzburg, Germany, URL: www1.informatik.uni-wuerzburg.de/en/staff

$(s \cdot x_i, y_i/s)$. This ensures that the bounding box of P and the bounding box of Q have the same area.

In order to choose a good scale factor, we define various criteria. Our main approach is to measure the quality of a scatter plot with a function $f\colon \mathcal{S} \to \mathbb{R}$, where $\mathcal{S}$ is the set of all possible scatter plots for the given point set. Then, we search for a scale factor whose corresponding scatter plot maximizes f.

All our measures are based on the Delaunay triangulation of the point set. A characterization of the Delaunay triangulation is that it maximizes the *smallest angle* among all triangulations. The natural idea is to use this measure also over different scale factors, i.e., to define $f(P)$ as the value of the smallest angle of the Delaunay triangulation of P. We will see that we can find the scale factor optimizing this criterion. Nevertheless, there are also other criteria that make sense, and which we can, at least, approximate:

- maximize the *mean inradius* of triangles of $D(P)$,
- maximize the total *compactness* of triangles of $D(P)$; the compactness of triangle Δ with perimeter $c(\Delta)$ and area $A(\Delta)$ is $\sqrt{A(\Delta)}/c(\Delta)$ [4],
- minimize the total *uncompactness* of triangles of $D(P)$; the uncompactness is $c(\Delta)/\sqrt{A(\Delta)}$ [4],
- minimize the *total length* of all edges of $D(P)$.

3 Algorithm

Finding an optimum scale factor s can be seen as a continuous process. We continuously increase s starting with $s = 1$. In doing so, the Delaunay triangulation undergoes *topological changes* at certain *event points* which we keep track of. We output the scale factor at which our objective function is optimized. Symmetrically, we traverse all scale factors $s < 1$.

Our algorithm consists of two layers. The first layer steps through the discrete set of event points in the order in which they occur in the above described process. The second layer optimizes between consecutive event points s_i and s_{i+1}, where the topological structure of the Delaunay triangulation does not change. Each of our optimization measures is a continuous function of $s \in [s_i, s_{i+1}]$. We can then compute the scale factor (or an approximation of it) at which this function is maximized within $[s_i, s_{i+1}]$. Doing this for all such intervals allows us to determine the globally optimal scale factor (or an approximation).

Recall that a triangulation $D(P)$ of a point set $P = \{p_1, \ldots, p_n\}$ is Delaunay if the circumcircle of each triangle of $D(P)$ is empty, that is, it does not contain points of P in its interior (c1). Alternatively, for any edge p_ip_j of $D(P)$, there must exist an empty circle whose boundary contains p_i and p_j (c2).

Maintaining the Delaunay triangulation through scale space Since the number of points on the convex hull is constant over the whole scale space, the same holds for the number of triangles and edges. Hence, for each triangle (or edge) that disappears from the Delaunay triangulation at some event point s_h, a new triangle (or edge) is created and vice versa. For the sake of simplicity, we assume in what follows that no five points are co-circular at any scale factor.

Consider an event point s_h at which some triangle disappears from $D(P)$. According to criterion (c1) there is at least one point p_l that *enters* the circumcircle $C(p_i, p_j, p_k)$ of the triangle at s_h. More precisely, the interior of $C(p_i, p_j, p_k)$ contains p_l at any $s > s_h$ but not at $s = s_h$ where p_l is on the boundary.

The following lemma (whose proof we skip) characterizes the situations where topological changes occur.

Lemma 1 *Assume that the Delaunay triangulation undergoes a topological change at event point s_h. Then there is a quadrilateral p_i, p_j, p_k, p_l in the Delaunay triangulation with diagonal p_ip_k such that p_l enters the circle $C(p_i, p_j, p_k)$ at s_h.*

Consider a point p_l entering the circumcircle of a triangle $\Delta p_ip_jp_k$ at event point s_h between p_i and p_k as described in Lemma 1. If we replace edge p_ip_k with edge p_jp_l at event point s_h, we obtain a new triangulation. We call this operation a *flip*. The crucial observation is that if no further flips are to be performed at s_h, the current Delaunay triangulation is valid at $s_h + \epsilon$ for sufficiently small $\epsilon > 0$. Also note that the flip of p_ip_k corresponds to the co-circularity of the unique quadrilateral $p_ip_jp_kp_l$ that contains p_ip_k.

Our algorithm determines the sequence $s_1, \ldots, s_m$ of event points one by one in increasing order starting with $s_1 := 1$. Given an event point s_i and the corresponding Delaunay triangulation $D(P(s_i))$, we face the problem of computing the next event point s_{i+1}, that is, the smallest scale factor larger than s_i at which we have to perform flips. Note that any flip that we have to perform at s_{i+1} corresponds to the co-circularity of the unique quadrilateral of $D(P(s_{i+1}))$ containing the edge flipped. In other words, for every edge, we have to compute the smallest scale factor larger than s_i at which the corresponding quadrilateral becomes co-circular (if any). For each edge, this event point can be computed in constant time by solving a system of four linear equations [5].

Our algorithm traverses the sequence of event points $s_1, \ldots, s_m$ in increasing order as follows. Initially, we compute the Delaunay triangulation at $s_1 = 1$ and set up a priority queue Q that maintains, for each edge of the current triangulation, the event point at which this edge has to be flipped. We then iteratively compute the sequence of event points. First, we get the next event point t by extracting from queue Q the next scale factor t at which some edge e has to be flipped. When e is flipped, the queue Q has to be updated accordingly. We must update the event points at which the four edges of the quadrilateral containing e have to be flipped.

Let's analyze the running time of the algorithm. The initialization step takes $O(n \log n)$ time for the Delaunay triangulation and $O(n)$ time for building the priority queue. For each flip, we spend $O(\log n)$ time for extracting the minimum of Q and updating the four edges on the corresponding quadrilateral.

It remains to determine the maximum number of flips performed by the algorithm. Consider the situation of Lemma 1 where we flip the edge $p_i p_k$ at event point s_h. For every scale factor $s > s_h$, the circle $C(p_i, p_j, p_k)$ contains p_l in its interior. By criterion (c2) we can conclude that the edge $p_i p_k$ can not be part of a Delaunay triangulation for any $s > s_h$. Since there are at most $O(n^2)$ potential edges, and every edge that is flipped cannot be re-inserted into the Delaunay triangulation, there are only $O(n^2)$ flips.

Theorem 2 *We traverse the sequence $s_1, \ldots, s_m$ of event points in increasing order and compute for $1 \leq i \leq m-1$ the Delaunay triangulation valid in the interval $[s_i, s_{i+1}]$ in a total running time of $O(n^2 \log n)$.*

The event points can be scattered quite unevenly over the scale space. It is therefore not sufficient to consider only the event points as potential solutions.

Finding an approximate solution We describe our method for minimizing the uncompactness c_D of the Delaunay triangulation D. It can be applied to the other objective functions in a similar manner.

Fix an interval $[s_i, s_{i+1}]$ of consecutive event points and fix an arbitrarily small error parameter $\epsilon > 0$. Our goal is to find an $(1+\epsilon)$-approximate solution for c_D, that is, a scale factor s_a for which $c_D(s_\mathrm{a}) \leq (1+\epsilon) c_D(s^*)$ where s^* is the globally optimal scale factor. Given an edge e of D, its length $l_e(s)$ depends on the scale factor. It is easy to see that $l_e((1+\epsilon)s) \leq (1+\epsilon) l_e(s)$. As the area is constant, for the uncompactness c_Δ of a triangle Δ it also holds that $c_\Delta((1+\epsilon)s) \leq (1+\epsilon)c_\Delta(s)$, and similarly the total uncompactness $c_D(s) = \sum_{\Delta \in D} c_\Delta(s)$ is bounded.

We restrict ourselves to scale factors between 1 and C for some large constant C, which is sufficient for practical purposes. Let $s_1, \ldots, s_m$ denote the sequence of event points (between 1 and C) and let $s_{m+1} := C$. Now consider a fixed interval $[s_i, s_{i+1}]$ with $i = 1, \ldots, m$. Our algorithm computes $c_D(\cdot)$ for all *test values* $t_j := s_{i+1}/(1+\epsilon)^j$ where $j \in \mathbb{N}$ and $t_j \in [s_i, s_{i+1}]$. Let t_{j^*} be the test value at which $c_D(\cdot)$ is minimized and let s^* be an optimum scale factor in the interval $[s_i, s_{i+1}]$. Using the above bound, it can be shown that $c_D(t_{j^*}) \leq (1+\epsilon)c_D(s^*)$, that is, we obtain a $(1+\epsilon)$-approximation for the current interval; hence, taking the optimum over all intervals, we can find a $(1+\epsilon)$-approximation. Let's summarize.

Theorem 3 *For any fixed $\epsilon > 0$, we can compute a $(1+\epsilon)$-approximate solution for minimizing the uncompactness measure in $O(n^3)$ time. Taking ϵ into account the running time is $O(n(n^2 + 1/\log(1+\epsilon)))$.*

Maximizing the minimum angle We sketch an efficient exact algorithm for determining a globally optimal scale factor for the objective of maximizing the smallest angle of a Delaunay triangulation.

Each angle β (formed by two edges) that occurs during traversing the scale space can be described as a function of the scale s. Its domain is an interval $[l_\beta, r_\beta]$, the intersection of the life times of the edges. Let $\mathcal{A}$ be the set of all angles (functions) that appear at some scale factor in the Delaunay triangulation, and let $\mathrm{env}(\mathcal{A})$ be the lower envelope of $\mathcal{A}$. Determining the scale factor that maximizes the smallest angle of the Delaunay triangulation amounts to determining the maximum of $\mathrm{env}(\mathcal{A})$.

Consider some angle $\beta \in \mathcal{A}$, and let $p_i p_j$ and $p_j p_k$ be the edges defining β. Now consider a coordinate system whose origin is located at p_j. Then $\beta > \pi/2$ for any s if p_i and p_k lie in diagonally opposite quadrants. As we are only interested in the *lower* envelope, we can safely remove all such angles from $\mathcal{A}$.

Under this assumption, it is not hard to verify that any $\beta \in \mathcal{A}$ can be expressed as

$$\beta(s) = c_1 \pi + \arctan\left(c_2 s/(c_3 s^2 + 1)\right)$$

where $c_1 \in \{0, 1\}$ and $c_2, c_3 \in \mathbb{R}$ are easily computable constants that only depend on the edges defining β but not on s. Elementary calculations reveal that two functions of the above form can have at most one intersection. Let $m = O(n^2)$ be the number of angles in $\mathcal{A}$. Because any two functions in $\mathcal{A}$ intersect at most once, it is known from the theory of Davenport–Schinzel sequences that the lower envelope $\mathrm{env}(\mathcal{A})$ has complexity (number of distinct curve segments) at most $\lambda_3(m) = O(m\alpha(m))$ [7] where α denotes the functional inverse of the Ackermann function.

Agarwal and Sharir [7] show that the lower envelope of m partially defined functions can be computed in $O(\lambda_{r+1}(m) \log m)$ time if any two functions intersect at most r times. Hence, their algorithm runs in $O(n^2 \log n)$ time in our case. For each curve segment, the maximum can be computed in constant time. As the curve complexity is $O(n^2 \alpha(n^2))$, we conclude.

Theorem 4 *The globally optimal scale factor for the objective of maximizing the smallest angle can be computed in $O(n^2 \log n)$ time.*

4 Experimental Results

We implemented our algorithms in Java. For maximizing the minimum angle, we used a simplified version of our exact algorithm that is slower but easier

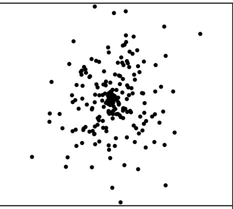
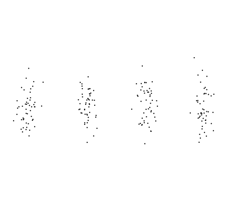
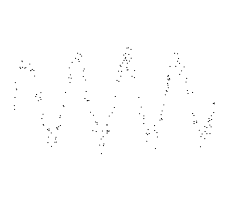
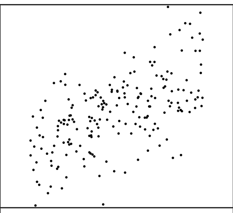
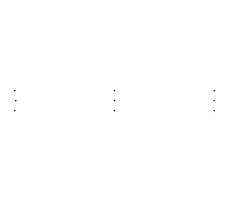
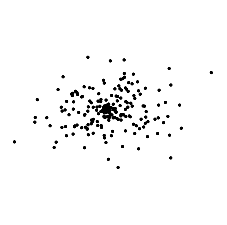
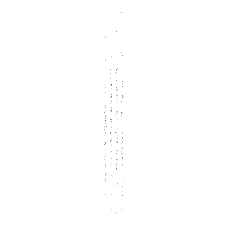
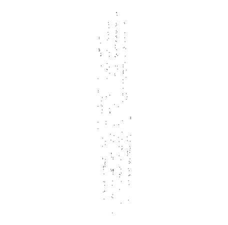
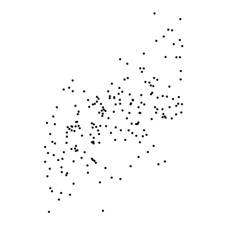
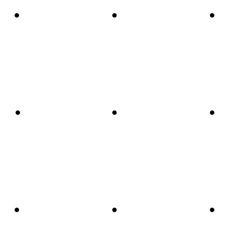
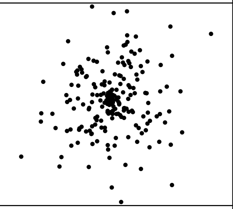
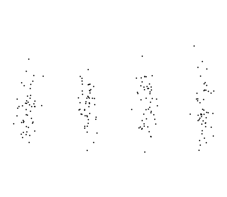
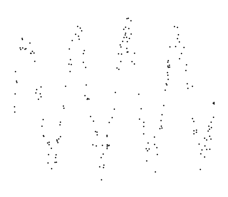
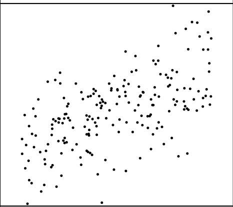
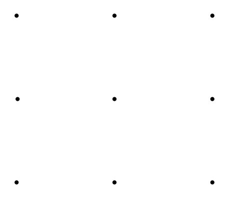
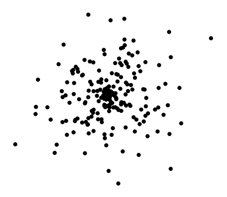
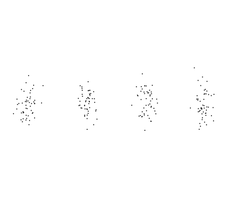
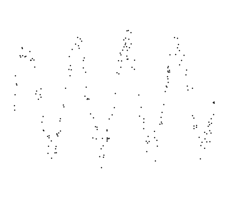
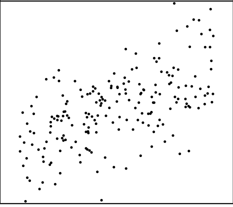
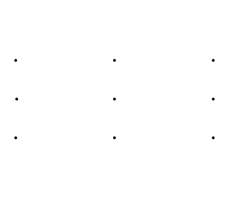
	normal distrib.	four clusters	noisy sine	rough trend	defect grid
max. min. angle					
max. mean inradius					
min. total length					
min. uncompact.					

Table 1: Test results for 4 optimization criteria on 5 generated instances (outputs scaled to fit into the boxes).

to implement. For the other criteria, we used the approximation algorithm with $\epsilon = 0.01$.

Table 1 shows results on five test instances: a cluster of points normally distributed around a center; four such clusters next to each other; points sampled along a sine function with normally distributed distance to it; the same for a linear trend; nine points lying on a grid with the exception of one of them that is moved a bit away from the grid point. We omitted the results for maximizing the compactness as they were quite similar to the ones for minimizing the total length (yet more stretched in y-direction for the sine and rough linear trend).

As visible for the four clusters and the noisy sine, the total length and especially the inradius criterion often tend to stretch the plot too much. The angle criterion, and, to a lesser degree, the uncompactness criterion are sensitive to small changes, see the defect grid. Over all tests, however, the uncompactness minimization showed the best results.

5 Conclusion and Future Work

Our tests confirm that selecting the aspect ratio of a scatter plot based on the Delaunay triangulation is a promising approach.

References

[1] W. S. Cleveland. A model for studying display methods of statistical graphics. *J. Comput. Graph. Statist.*, 2(4):323–343, 1993.

[2] W. S. Cleveland, M. E. McGill, and R. McGill. The shape parameter of a two-variable graph. *J. Am. Stat. Assoc.*, 83(289–300), 1988.

[3] J. Heer and M. Agrawala. Multi-scale banking to 45°. *IEEE T. Vis. Comput. Gr.*, pages 701–708, 2006.

[4] A. M. MacEachren. Compactness of geographic shape: Comparison and evaluation of measures. *Geografiska Annaler. Ser. B, Human Geogr.*, 67(1):53–67, 1985.

[5] T. Roos. Voronoi diagrams over dynamic scenes. *Discrete Appl. Math.*, 43(3):243–259, 1993.

[6] N. Rubin. On topological changes in the Delaunay triangulation of moving points. In *Proc. 28th ACM Symp. Comput. Geom. (SoCG'12)*, pages 1–10, 2012.

[7] M. Sharir and P. K. Agarwal. *Davenport-Schinzel sequences and their geometric applications.* Cambridge University Press, 1995.

[8] J. Talbot, J. Gerth, and P. Hanrahan. Arc length-based aspect ratio selection. *IEEE T. Vis. Comput. Gr.*, 17(12):2276–2282, 2011.

[9] J. Talbot, J. Gerth, and P. Hanrahan. An empirical model of slope ratio comparisons. *IEEE T. Vis. Comput. Gr.*, 18(12):2613–2620, 2012.