Approximation Algorithm for the L_1-Fitting Circle Problem

Sariel Har-Peled*

Abstract

In this paper, we study the problem of L_1-fitting a circle to a set of points in the plane, where the target function is the sum of distances of the points to the circle. We show an $(1+\varepsilon)$-approximation algorithm, with running time $O(n + \text{poly}(\log n, 1/\varepsilon))$, where $\text{poly}(\log n, 1/\varepsilon)$ is a constant degree polynomial in $\log n$ and $1/\varepsilon$. This is the first subquadratic algorithm for this problem.

1 Introduction

Motivated by a variety of applications, considerable work has been done on measuring various descriptors of the extent of a set P of n points in $\Re^d$. We refer to such measures as *extent measures* of P. Roughly speaking, an extent measure of P either computes certain statistics of P itself or it computes certain statistics of a (possibly nonconvex) geometric shape (e.g. sphere, box, cylinder, etc.) enclosing P. Examples of the former include computing the kth largest distance between pairs of points in P, and the examples of the latter include computing the smallest radius of a sphere (or cylinder), the minimum volume (or surface area) of a box, and the smallest width of a slab (or a spherical or cylindrical shell) that contain P.

Shape fitting, a fundamental problem in computational geometry, computer vision, machine learning, data mining, and many other areas, is closely related to computing extent measures. A widely used shape-fitting problem asks for finding a shape that best fits P under some "fitting" criterion. A typical criterion for measuring how well a shape γ fits P, denoted as $\mu(P,\gamma)$, is the maximum distance between a point of P and its nearest point on γ, i.e., $\mu(P,\gamma) = \max_{p\in P}\min_{q\in\gamma} d(p,q)$. This is the *$L_\infty$-fitting problem*[1]. Here, one can define the extent measure of P to be $\mu(P) = \min_\gamma \mu(P,\gamma)$, where the minimum is taken over a family of shapes (such as points, lines, hyperplanes, spheres, etc.). For example, the problem of finding the minimum radius sphere (resp. cylinder) enclosing P is the same as finding the point (resp. line) that fits P best, and the problem of finding the smallest width slab (resp. spherical shell, cylindrical shell) is the same as finding the hyperplane (resp. sphere, cylinder) that fits P best.

The exact algorithm for computing extent measures are generally expensive, e.g., the best known algorithms for computing the smallest volume bounding box containing P in $\Re^3$ require $O(n^3)$ time. Consequently, attention has shifted to developing approximation algorithms [BH01, ZS02]. A general approximation technique was recently developed for such problems by Agarwal *et al.* [AHV04]. This implies among other things that one can $(1+\varepsilon)$-approximate the circle best L_∞-fit a set of points in the plane in $O(n + 1/\varepsilon^{O(1)})$ time (see [AAHS00] and [Cha02] and references therein for more information about this problem).

The main problem of the L_∞ measure in shape fitting is that it is very sensitive to outliers. Namely, a single outlying point can change the price of the optimal solution dramatically. There are two natural solutions. The first approach, is to change the target function to be less sensitive to outliers. For example, instead of considering the maximum distance, one can consider the sum of distances (i.e., L_1-fitting), or the sum of squared distance (i.e., L_2-fitting). The L_2-fitting in the case of a single linear subspace is well understood, and is no more than SVD (singular value decomposition). Fast approximation algorithms are known for this problem, see [FKV98, RVW04] and references therein. As for the L_1-fitting, this problem can be solved using linear programming techniques, in polynomial time in high dimensions, and linear time in constant dimension [YKII88]. Recently, Clarkson gave a faster algorithm for this problem [Cla05] which works via sampling.

The problem seems to be harder once the shape considered is not a linear subspace. There is considerable work on nonlinear regressions (i.e., extension of the least squares technique) for various shapes, but there not seems to be a guaranteed approximation algorithms for the circle case [SW89]. The hardness in the L_1-fitting of a circle seems to rise from the target function is a sum of terms, each term being an absolute value of a difference of a square root of a polynomial and a radius. It seems doubtful that analytical solution would exist for such a target function, as it is related to the Fermat-Weber problem [Wes93].

*Department of Computer Science; University of Illinois; 201 N. Goodwin Avenue; Urbana, IL, 61801, USA; sariel@uiuc.edu; http://www.uiuc.edu/~sariel/. Work on this paper was partially supported by a NSF CAREER award CCR-0132901.

[1] The L_∞-fitting comes from considering the distance of every point to the shape being a coordinate in a vector, and what metric we apply to this vector.

The second approach, is to specify the number k of outliers in advance, and find the best shape L_∞-fitting all but k of the input points. Har-Peled and Wang showed that there is a coreset for this problem [HW04], and as such it can be $(1+\varepsilon)$-approximated in $O(n + \text{poly}(k, \log n, \varepsilon))$ time, for a large family of shapes. The work of Har-Peled and Wang was mainly motivated by trying to solve the problem of L_1-fitting a circle to a set of points.

In this paper, we describe an $(1+\varepsilon)$-approximation algorithm for the L_1-fitting of a circle to a set of points in the plane. The solution has running time of $O(n + \text{poly}(\log n, 1/\varepsilon))$, and the result is correct with high probability. The only previous algorithm for this problem we are aware of, is due to Har-Peled and Koltun [HK04a], and it works in $O(n^2\varepsilon^{-2}\log^2 n)$ time.

Due to extreme space limitations, we only provide a sketch of the paper. A full version of this paper is available from the author's webpage `http://www.uiuc.edu/~sariel/papers/05/l1_fitting/`.

2 Approximate L_1 Fitting of a Circle to a Set of Points

2.1 Solution Outline.

The problem of best L_1-fitting a circle to a set of points in the plane, is equivalent to finding the points in 3D the minimizes the sum of distances to a set of cones.

As such, our solution is based on two steps. In the first step, we compute a small set of levels, and assign weights to them, such that solving the problem on those (weighted) levels would be equivalent to solving the problem on the original arrangement of cones. Unfortunately, this is by itself insufficient, as those levels by themselves might be of high complexity, and computing them would be prohibitly expensive. To this end, we replace the exact level by approximate level, using random sampling. This ensures that each such level is a shallow level (in the appropriate random sample).

This still fall short of solving the problem, because even a shallow level might have high complexity. As such, we simplify those levels in such a way that preserves the sum of vertical distances. The last step is done by applying a recent result of Har-Peled and Wang [HW04] that shows that one can do such a simplification, and it is of small size.

In the end of this process, we have a weighted arrangement of surface patches of small size (say, of complexity $O(\text{poly}(1/\varepsilon, \log n))$), such that we need to solve the problem in this arrangement. We can now solve the problem by using an reasonably efficient brute force approach. Indeed, we decompose the arrangement into vertical slabs where a vertical line intersect the surfaces in the same order. We need to solve the problem inside this prism. In the original settings, this corresponds to a (small) weighted set of points, where we want to best fit them to a circle. We use a slow (cubic time) approximation algorithm to do that.

2.2 Warmup Exercise – Slow Approximation

Lemma 1 *Let $\mathcal{G}$ be a set of n weighted cones in $\Re^3$ with total weight W, and $\varepsilon > 0$ a parameter. One can find a point $p \in \Re^3$ such that, $\nu_{\mathcal{G}}(p) \le (1+\varepsilon)\nu_{\text{opt}}(\mathcal{G})$, where $\nu_{\text{opt}}(\mathcal{G})$ is the price of the global minimum. The running time is $O(n^3 \text{poly}(\log W, \varepsilon^{-1}))$.*

The same running time holds, if the feasible region is restricted to a region of constant complexity of space.

2.3 Second Warmup Exercise – the One Dimensional Case

In this section, we consider the one dimensional problem of approximating the distance function of a point x to a set of points $Z = \langle z_1, z_2, \ldots, z_n \rangle$, where $z_1 \le z_2 \le \ldots \le z_n$. Formally, we want to approximate the function $\nu_Z(z) = \sum_{z_i \in Z} |z_i - z|$. This is no more than the one median function for those points on the line. This corresponds to a vertical line in three dimensions, where each z_i represents the intersection of the vertical line with the surface γ_i. The one dimensional problem is well understood, since it is no more than the 1-median problem, and there exists a coreset for it, see [HM04, HK04b]. Unfortunately, it is unclear how to perform the operations that corresponds to those coreset construction in a global fashion, so that the construction would hold for all vertical lines.

Our first step, is to do *chunking*. Formally, we partition Z symmetrically into subsets, such that the size of the subsets increase in size as one comes toward the middle of the set. Formally, the set $L_i = \{z_i\}$ contains the ith point on the line, for $i = 1, \ldots, M$, where $M \ge 10/\varepsilon$ is a parameter to be determined shortly. Similarly, $R_i = \{z_{n-1+1}\}$, for $i = 1, \ldots, M$. Next, let $\alpha_M = M$, and let $\alpha_{i+1} = \min(\lceil (1+\varepsilon/10)\alpha_i \rceil, n/2)$, for $i = M, \ldots, N$, where α_N is the first number in this sequence equal to $n/2$. Now, let $L_i = \{z_{\alpha_{i-1}+1}, \ldots, z_{\alpha_i}\}$ and $R_i = \{z_{n-\alpha_{i-1}}, \ldots, z_{n-\alpha_i+1}\}$, for $i = M+1, \ldots, N$. Consider the partition of Z formed by $L_1, L_2, \ldots, L_N, R_N, \ldots, R_2, R_1$. Clearly, this is a partition of Z into "exponential sets". The margin M sets on the boundary are singletons, and all the other sets grow exponentially in cardinality, till the cover the whole set Z.

We next pick a point an arbitrary point $l_i \in L_i$ and $r_i \in R_i$, and we also assign weight $|R_i| = |L_i|$ to each such point. for $i = 1, \ldots, N$. Let $\mathcal{S}$ be the resulting weighted set of points. We claim that this is a coreset for the 1-median function.

Lemma 2 *Let A be a set of n real numbers, and let a be any number of A. We have that $|\nu_A(z) - |A|\|az\|| \leq \nu_A(a)$.*

Lemma 3 $\nu_Z(z) \approx_{\varepsilon/5} \nu_{\mathcal{S}}(z)$*, for any* $z \in \Re$.

Next, we "slightly" perturb the points of the coreset $\mathcal{S}$. Formally, assume that we have points $l'_1, \ldots, l'_N, r'_1, \ldots, r'_N$ such that $\|l'_i l_i\|, \|r'_i r_i\| \leq (\varepsilon/20)\|l_i r_i\|$, for $i = 1, \ldots, N$. Let $\mathcal{S}' = \{l'_1, \ldots, l'_N, r'_N, \ldots, r'_1\}$ be the resulting weighted set. We claim that $\mathcal{S}'$ is still a good coreset.

Lemma 4 $\nu_Z(z) \approx_{\varepsilon/3} \nu_{\mathcal{S}'}(z)$*, for any* $z \in \Re$.

2.4 Additional Tools

2.4.1 Approximating a Level by a Shallow Level in a Random Sample

Lemma 5 *Let $\mathcal{G}$ be a set of n surfaces in $\Re^d$, $\varepsilon > 0$, and let k be a number between 0 and $n/2$. Let $\rho = \min\bigl(ck^{-1}\varepsilon^{-2}\log n, 1\bigr)$, and pick each surface of $\mathcal{G}$ into a random sample $\mathcal{R}$ with probability ρ. Then, with high probability, the $L = k\rho = O(\varepsilon^{-2}\log n)$ level of $\mathcal{A}(\mathcal{R})$ lies between the $(1-\varepsilon)k$-level to the $(1+\varepsilon)k$-level of $\mathcal{A}(\mathcal{G})$. This holds with high probability.*

2.4.2 Approximating the Extent of Shallow Levels

We need the result of Har-Peled and Wang [HW04]. It states that for well behaved set of functions, one can find a small subset of the functions such that the vertical extent of the subset approximates the extents of the whole set. This holds only for "shallow" levels $\leq k$. In our application k is going to be about $O(\varepsilon^{-2}\log n)$.

3 The approximation algorithm

We are now ready to put everything together. Let the input be a set P of n points in the plane. We define, as in Section 2.1, for each point of P a surface in $\Re^3$. Each such surface is a cone. Let $\mathcal{G}$ be the resulting set of cones.

We decompose the arrangement $\mathcal{A}(\mathcal{G})$ into levels. Next, we chunk the levels into sets, as done in Section 2.3, where $M = O(\varepsilon^{-2}\log n)$ and $N = M + O(\varepsilon^{-}1\log n) = O(\varepsilon^{-2}\log n)$. Let $L_1, \ldots, L_N, R_N, \ldots, R_1$ denote the resulting chunks of levels. Next, let $l_i = \mathbf{L}_{\mathcal{G},i-1}$ and $r_i = \mathbf{U}_{\mathcal{G},i-1}$ for $i = 1, \ldots, M$. For $i = N+1, \ldots, M$, we generate a random sample $\mathcal{R}_i$ of $\mathcal{G}$, according to Lemma 5, and levels $l_i = \mathbf{L}_{\mathcal{R}_i,k_i}$ and $r_i = \mathbf{U}_{\mathcal{R}_i,k_i}$, where $k_i = O(\varepsilon^{-2}\log n)$. We are guaranteed, with high probability, that l_i lies between the lowest and highest levels defined by L_i, and similarly that r_i lies between the highest and lowest levels of R_i, for $i = M+1, \ldots, N$.

Of course, computing the surfaces l_i and r_i explicitly is going to be prohibitively expensive. However, l_i and r_i are the bottom/top k_i-level in the arrangement $\mathcal{A}(\mathcal{R}_i)$. Since, k_i is relatively small, this means that those levels are shallow. As such, we can compute a subset $V_i \subseteq \mathcal{R}_i$, such that $V_i|_{k_i}^{k_i}(x) \geq (1-\varepsilon/10)\mathcal{R}_i|_{k_i}^{k_i}(x)$, for all $x \in \Re^2$, using [HW04]. Here, $s = 2$ and $|V_i| = O(k_i/\varepsilon^4) = O(\mathrm{poly}(1/\varepsilon, \log n))$.

Next, let l'_i and r'_i the k_i-bottom/top levels in $\mathcal{A}(V_i)$, respectively, for $i = M+1, \ldots, N$. Note that for $i = 1, \ldots, N$ we can use the same sample. As such, $V_1 = \cdots = V_M$. We assign the surfaces l'_i and r'_i the weight $|L_i| = |R_i|$, for $i = 1, \ldots, N$. Next, consider the set of the weighted surfaces $\mathcal{G}' = \{l'_1, \ldots, l'_N, r'_1, \ldots, r'_N\}$. It follows by Lemma 4 that

$$\nu_{\mathcal{G}'}(p) \approx_{\varepsilon} \nu_{\mathcal{G}}(p),$$

where p is any point in $\Re^3$. This holds with high probability. We still remain with the question of finding the global minimum of $\nu_{\mathcal{G}'}(p)$. To this end, we decompose $\Re^3$ into vertical prisms, such that inside such prism a vertical line intersect exactly the same surface patches in the same order. It is now straightforward to show that the number of such prisms is $O(\mathrm{poly}(\log n, 1/\varepsilon))$. Inside each such prism, the arrangement $\mathcal{A}(\mathcal{G}')$ has the same surfaces intersecting it in the same ordering. Namely, we have a portion of the parametric space we have to find the minimum for (i.e., the prism), while having a small number of weighted surfaces we have to consider.

Using the slow algorithm inside every prism, gives us an $(1+\varepsilon)$-approximation inside each such prism. Since the running time inside each such prism is $O(\mathrm{poly}(\log n, 1/\varepsilon))$, there are $O(\mathrm{poly}(\log n, 1/\varepsilon))$ prisms. Thus the overall running time is $O(n + \mathrm{poly}(\log n, 1/\varepsilon))$. We summarize:

Theorem 6 *Given a set P of n points in the plane, and parameter ε, one can compute in $O(n + \mathrm{poly}(\log n, 1/\varepsilon))$ time the circle minimizing the L_1 fitting price to P. The running time is $O(n + \mathrm{poly}(\log n, 1/\varepsilon))$. The result is correct with high probability.*

4 Conclusions

We had described in this paper an $(1+\varepsilon)$-approximation algorithm for the problem of L_1-fitting of a circle to a set of points in the plane. The running time of the new algorithm is $O(n + \mathrm{poly}(\log n, 1/\varepsilon))$, which is a linear running time for fixed ε. The constant powers hiding in the polylogarithmic term are too embarrassing to be explicitly stated, but are probably somewhere between 20 to 60. As such, this algorithm is only of theoretical interest. As such, the first open problem raised by this work is to improve this constants. A considerably more interesting problem is to develop a practical algorithm for this problem.

It is the author's belief that the techniques described in this paper, can be also applied to the problem of L_2-fitting of a circle to a set of points (i.e., best circle fitting a set of points minimizing the sum of square distances of the points to the circle). More importantly, it seems that the technique should be applicable to any of the fitting problems handled by the algorithm of Agarwal *et al.* [AHV04]. This includes the L_1-fitting of a sphere or a cylinder to a set of points.

A natural question is whether one can use the techniques of Har-Peled and Wang directly, to compute a coreset for this problem, and solve the problem on the coreset directly (our solution did a similar thing, by breaking the parametric space into a small number regions, and constructing a coreset inside each such region). There is unfortunately a nasty technicality that requires that a coreset for the L_1-fitting of linear function, is also coreset if we take the square root of the functions. It seems doubtful that this claim holds in general, but maybe a more careful construction of a coreset for the linear functions case would still work. The author leaves this as an open problem for further research.

Acknowledgments

The author would like to thank Pankaj Agarwal, Arash Farzan, Vladlen Koltun, Bardia Sadri, Kasturi Varadarajan, and Yusu Wang for useful discussions related to the problems studied in this paper.

References

[AAHS00] P. K. Agarwal, B. Aronov, S. Har-Peled, and M. Sharir. Approximation and exact algorithms for minimum-width annuli and shells. *Discrete Comput. Geom.*, 24(4):687–705, 2000.

[AHV04] P. K. Agarwal, S. Har-Peled, and K. R. Varadarajan. Approximating extent measures of points. *J. Assoc. Comput. Mach.*, 51(4):606–635, 2004.

[BH01] G. Barequet and S. Har-Peled. Efficiently approximating the minimum-volume bounding box of a point set in three dimensions. *J. Algorithms*, 38:91–109, 2001.

[Cha02] T. M. Chan. Approximating the diameter, width, smallest enclosing cylinder and minimum-width annulus. *Internat. J. Comput. Geom. Appl.*, 12(2):67–85, 2002.

[Cla05] K. L. Clarkson. Subgradient and sampling algorithms for l1 regression. In *Proc. 16th ACM-SIAM Sympos. Discrete Algorithms*, 2005. to appear.

[FKV98] A. Frieze, R. Kannan, and S. Vempala. Fast monte-carlo algorithms for finding low-rank approximations. In *FOCS '98: Proceedings of the 39th Annual Symposium on Foundations of Computer Science*, page 370. IEEE Computer Society, 1998.

[HK04a] S. Har-Peled and V. Koltun. Approximate l_1 and l_2 circle fitting in (easy) polynomial time. manuscript, 2004.

[HK04b] S. Har-Peled and A. Kushal. Smaller coresets for k-median and k-means clustering. http://www.uiuc.edu/~sariel/papers/04/small_coreset/, 2004.

[HM04] S. Har-Peled and S. Mazumdar. Coresets for k-means and k-median clustering and their applications. In *Proc. 36th Annu. ACM Sympos. Theory Comput.*, pages 291–300, 2004.

[HW04] S. Har-Peled and Y. Wang. Shape fitting with outliers. *SIAM J. Comput.*, 33(2):269–285, 2004.

[RVW04] L. Rademacher, S. Vempala, and G. Wang. Matrix approximation and projective clustering via adaptive sampling. manuscript, 2004.

[SW89] G.A.F. Seber and C.J. Wild. *Nonlinear regression*. Jonh Wiley & Sons, 1989.

[Wes93] G. Wesolowsky. The Weber problem: History and perspective. *Location Science*, 1:5–23, 1993.

[YKII88] P. Yamamoto, K. Kato, K. Imai, and H. Imai. Algorithms for vertical and orthogonal l1 linear approximation of points. In *Proc. 4th Annu. ACM Sympos. Comput. Geom.*, pages 352–361. ACM Press, 1988.

[ZS02] Y. Zhou and S. Suri. Algorithms for a minimum volume enclosing simplex in three dimensions. *SIAM J. Comput.*, 31(5):1339–1357, 2002.