

A Sub-Linear Time Framework for Geometric Optimization with Outliers in High Dimensions

Hu Ding 

School of Computer Science and Technology, University of Science and Technology of China,
Anhui, China

<http://staff.ustc.edu.cn/~huding/index.html>

huding@ustc.edu.cn

Abstract

Many real-world problems can be formulated as geometric optimization problems in high dimensions, especially in the fields of machine learning and data mining. Moreover, we often need to take into account of outliers when optimizing the objective functions. However, the presence of outliers could make the problems to be much more challenging than their vanilla versions. In this paper, we study the fundamental minimum enclosing ball (MEB) with outliers problem first; partly inspired by the core-set method from Bădoiu and Clarkson, we propose a sub-linear time bi-criteria approximation algorithm based on two novel techniques, the Uniform-Adaptive Sampling method and Sandwich Lemma. To the best of our knowledge, our result is the first sub-linear time algorithm, which has the sample size (i.e., the number of sampled points) independent of both the number of input points n and dimensionality d , for MEB with outliers in high dimensions. Furthermore, we observe that these two techniques can be generalized to deal with a broader range of geometric optimization problems with outliers in high dimensions, including flat fitting, k -center clustering, and SVM with outliers, and therefore achieve the sub-linear time algorithms for these problems respectively.

2012 ACM Subject Classification Theory of computation → Approximation algorithms analysis

Keywords and phrases minimum enclosing ball, outliers, shape fitting, high dimensions, sub-linear time

Digital Object Identifier 10.4230/LIPIcs.ESA.2020.38

Related Version A full version of the paper is available at <https://arxiv.org/abs/2004.10090>.

Acknowledgements The author wants to thank the anonymous reviewers for their helpful comments and suggestions for improving the paper.

1 Introduction

Geometric optimization is a fundamental topic that has been extensively studied in the community of computational geometry [1]. The *minimum enclosing ball (MEB)* problem is one of the most popular geometric optimization problems who has attracted a lot of attentions in past years, where the goal is to compute the smallest ball covering a given set of points in the Euclidean space [11, 50, 32]. Though its formulation is very simple, MEB has a number of applications in real world, such as classification [68, 20, 21], preserving privacy [59, 31], and quantum cryptography [38]. A more general geometric optimization problem is called *flat fitting* that is to compute the smallest slab (centered at a low-dimensional flat) to cover the input data [42, 60, 71]. Another closely related important topic is the *k-center clustering* problem, where the goal is to find $k > 1$ balls to cover the given input data and minimize the maximum radius of the balls [35]; the problem has been widely applied to many areas, such as facility location [18] and data analysis [67]. Moreover, some geometric optimization problems are trying to maximize their size functions. As an example, the well known classification technique *support vector machine (SVM)* [17] is to maximum the margin separating two differently labeled point sets in the space.



© Hu Ding;
licensed under Creative Commons License CC-BY
28th Annual European Symposium on Algorithms (ESA 2020).

Editors: Fabrizio Grandoni, Grzegorz Herman, and Peter Sanders; Article No. 38; pp. 38:1–38:21

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Real-world datasets are often noisy and contain outliers. Moreover, outliers could seriously affect the final optimization results. For example, it is easy to see that even one outlier could make the MEB arbitrarily large. In particular, as the rapid development of machine learning, the field of *adversarial machine learning* concerning about the potential vulnerabilities of the algorithms has attracted a great amount of attentions [45, 51, 9, 36]. A small set of outliers could be added by some adversarial attacker to make the decision boundary severely deviate [8, 48]. Furthermore, the presence of outliers often results in a quite challenging combinatorial optimization problem; as an example, if m of the input n data items are outliers ($m < n$), we have to consider an exponentially large number $\binom{n}{m}$ of different possible cases when optimizing the objective function. Therefore, the design of efficient and robust optimization algorithms is urgently needed to meet these challenges.

1.1 Our Contributions

In big data era, the data size could be so large that we cannot even afford to read the whole dataset once. In this paper, we consider to develop sub-linear time algorithms for several geometric optimization problems involving outliers. We study the aforementioned MEB with outliers problem first. Informally speaking, given a set of n points in \mathbb{R}^d and a small parameter $\gamma \in (0, 1)$, the problem is to find the smallest ball covering at least $(1 - \gamma)n$ points from the input. We are aware of several existing sub-linear time bi-criteria approximation algorithms based on uniform sampling for MEB and k -center clustering with outliers [7, 46, 29], where the “bi-criteria” means that the ball (or the union of the k balls) is allowed to exclude a little more points than the pre-specified number of outliers. Their ideas are based on the theory of VC dimension [69]. But the sample size usually depends on the dimensionality d , which is roughly $O(\frac{1}{\delta^{2\gamma}}kd \cdot \text{polylog}(\frac{kd}{\delta\gamma}))$, if allowing to discard $(1 + \delta)\gamma n$ outliers with $\delta \in (0, 1)$ ($k = 1$ in the complexity for the MEB with outliers problem). A detailed overview on previous works is shown in Section 1.2.

Since many optimization problems in practice need to consider high-dimensional datasets, especially in the fields of machine learning and data mining, the above sample size from [7, 46, 29] could be very large. Partly inspired by the core-set method from Bădoiu and Clarkson [11] for computing MEB in high dimensions, **we are wondering that whether it is possible to remove the dependency on d in the sample size for MEB with outliers and other related high dimensional geometric optimization problems.** Given a parameter $\epsilon \in (0, 1)$, the method of [11] is a simple greedy algorithm that selects $\frac{2}{\epsilon}$ points (as the core-set) for constructing a $(1 + \epsilon)$ -approximate MEB, where the resulting radius is at most $1 + \epsilon$ times the optimal one. A highlight of their method is that the core-set size $\frac{2}{\epsilon}$ is independent of d . However, there are several substantial challenges when applying their method to design sub-linear time algorithm for MEB with outliers. First, we need to implement the “greedy selection” step by a random sampling manner, but it is challenging to guarantee the resulting quality especially when the data is mixed with outliers. Second, the random sampling approach often yields a set of candidates for the ball center (e.g., we may need to repeatedly run the algorithm multiple times for boosting the success probability, and each time generates a candidate solution), and thus it is necessary to design an efficient strategy to determine which candidate is the best one in sub-linear time.

To tackle these challenges, we propose two key techniques, the novel “**Uniform-Adaptive Sampling**” method and “**Sandwich Lemma**”. Roughly speaking, the Uniform-Adaptive Sampling method can help us to bound the error induced in each “randomized greedy selection” step; the Sandwich Lemma enables us to estimate the objective value of each candidate and select the best one in sub-linear time. To the best of our knowledge, our result

is the first sub-linear time approximation algorithm for the MEB with outliers problem with sample size being independent of the number of points n and the dimensionality d , which significantly improves the time complexities of existing algorithms.

Moreover, we observe that our proposed techniques can be used to solve a broader range of geometric optimization problems. We define a general optimization problem called **minimum enclosing “x” (MEX) with Outliers**, where the “x” stands for a specified kind of shape (e.g., the shape is a ball for MEB with outliers). We prove that it is able to generalize the Uniform-Adaptive Sampling method and Sandwich Lemma to adapt the shape “x”, as long as it satisfies several properties. In particular we focus on the MEX with outlier problems including flat fitting, k -center clustering, and SVM with outliers; a common characteristic of these problems is that each of them has an iterative algorithm based on greedy selection for its vanilla version (without outliers) that is similar to the MEB algorithm of [11]. Though these problems have been widely studied before, the research in terms of their sub-linear time algorithms is still quite limited.

1.2 Related Work

Sub-linear time algorithms. The research on sub-linear time algorithms design has a long history [63, 25]. For example, a number of sub-linear time clustering algorithms have been studied in [47, 56, 57, 24]. Another important application of sub-linear time algorithms is property testing on graphs or probability distributions [34].

As mentioned before, the uniform sampling idea can be used to design sub-linear time algorithms for the problems of MEB and k -center clustering with outliers [7, 46, 29], but the sample size depends on the dimensionality d that could be very large in practice. Note that Alon et al. [7] presented another sub-linear time algorithm, which has the sample size independent of d , to test whether an input point set can be covered by a ball with a given radius; however, it is difficult to apply their method to solve the MEB with outliers problem as the algorithm relies on some nice properties of minimum enclosing ball, but these properties are not easy to be utilized when inliers and outliers are mixed. In [26], we proposed a notion of stability for MEB and developed the sub-linear time MEB algorithms for stable instance. Clarkson et al. [21] developed an elegant perceptron framework for solving several optimization problems arising in machine learning, such as MEB. For a set of n points in \mathbb{R}^d , their framework can solve the MEB problem in $\tilde{O}(\frac{n}{\epsilon^2} + \frac{d}{\epsilon})^1$ time. Based on a stochastic primal-dual approach, Hazan et al. [44] provided an algorithm for solving the SVM problem in sub-linear time.

MEB and k -center clustering with outliers. *Core-set* is a popular technique to reduce the time complexities for many optimization problems [2, 61]. The core-set idea has also been used to compute approximate MEB in high dimensional space [20, 13, 50, 60, 49]. Bădoiu and Clarkson [11] showed that it is possible to find a core-set of size $\lceil 2/\epsilon \rceil$ that yields a $(1 + \epsilon)$ -approximate MEB. There are also several exact and approximation algorithms for MEB that do not rely on core-sets [32, 64, 6]. Streaming algorithms for computing MEB were also studied before [4, 16].

Bădoiu et al. [13] extended their core-set idea to the problems of MEB and k -center clustering with outliers, and achieved linear time bi-criteria approximation algorithms (if k is assumed to be a constant). Several algorithms for the low dimensional MEB with

¹ The asymptotic notation $\tilde{O}(f) = O(f \cdot \text{polylog}(\frac{nd}{\epsilon}))$.

outliers problem have been also developed [5, 30, 39, 54]. A 3-approximation algorithm for k -center clustering with outliers in arbitrary metrics was proposed by Charikar et al. [18]; Chakrabarty et al. [15] proposed a 2-approximation algorithm for k -center clustering with outliers. These algorithms often have high time complexities (e.g., $\Omega(n^2d)$). Recently, Ding et al. [29] provided a linear time greedy algorithm for k -center clustering with outliers based on the idea of the Gonzalez's algorithm [35]. Furthermore, there exist a number of works on streaming and distributed algorithms, such as [19, 55, 72, 53, 37, 14, 52].

Flat fitting with outliers. Given an integer $j \geq 0$ and a set of points in \mathbb{R}^d , the flat fitting problem is to find a j -dimensional flat having the smallest maximum distance to the input points [41]; obviously, the MEB problem is a special case with $j = 0$. In high dimensions, Har-Peled and Varadarajan [42] provided a linear time algorithm if j is assumed to be fixed; their running time was further reduced by Panigrahy [60] based on a core-set approach. There also exist several methods considering flat fitting with outliers but only for low-dimensional case [43, 3].

SVM with outliers. Given two point sets P_1 and P_2 in \mathbb{R}^d , the problem of *Support Vector Machine (SVM)* is to find the largest margin to separate P_1 and P_2 (if they are separable) [17]. SVM can be formulated as a quadratic programming problem, and a number of efficient techniques have been developed in the past, such as the soft margin SVM [22, 62], ν -SVM [65, 23], and Core-SVM [68, 20]. There also exist a number of works on designing robust algorithms of SVM with outliers [70, 40, 66, 28].

2 Definitions and Preliminaries

In this paper, we let $|A|$ denote the number of points of a given point set A in \mathbb{R}^d , and $\|x - y\|$ denote the Euclidean distance between two points x and y in \mathbb{R}^d . We use $\mathbb{B}(c, r)$ to denote the ball centered at a point c with radius $r > 0$. Below, we give several definitions used throughout this paper.

► **Definition 1** (Minimum Enclosing Ball (MEB)). *Given a set P of n points in \mathbb{R}^d , the MEB problem is to find a ball with minimum radius to cover all the points in P . The resulting ball and its radius are denoted by $MEB(P)$ and $Rad(P)$, respectively.*

A ball $\mathbb{B}(c, r)$ is called a λ -approximation of $MEB(P)$ for some $\lambda \geq 1$, if the ball covers all points in P and has radius $r \leq \lambda Rad(P)$.

► **Definition 2** (MEB with Outliers). *Given a set P of n points in \mathbb{R}^d and a small parameter $\gamma \in (0, 1)$, the MEB with outliers problem is to find the smallest ball that covers $(1 - \gamma)n$ points. Namely, the task is to find a subset of P with size $(1 - \gamma)n$ such that the resulting MEB is the smallest among all possible choices of the subset. The obtained ball is denoted by $MEB(P, \gamma)$.*

For convenience, we use P_{opt} to denote the optimal subset of P with respect to $MEB(P, \gamma)$. That is, $P_{\text{opt}} = \arg_Q \min \{Rad(Q) \mid Q \subset P, |Q| = (1 - \gamma)n\}$. From Definition 2, we can see that the main issue is to determine the subset of P . Actually, solving such combinatorial problems involving outliers are often challenging. In Section A.1, we present an example to show that it is impossible to achieve an approximation factor less than 2 for MEB with outliers, if the time complexity is required to be independent of n . Therefore, we consider finding the bi-criteria approximation. Actually, it is also a common way for solving other

optimization problems with outliers. For example, Mount et al. [58] and Meyerson et al. [56] studied the bi-criteria approximation algorithms respectively for the problems of linear regression and k -median clustering with outliers before.

► **Definition 3** (Bi-criteria Approximation). *Given an instance (P, γ) for MEB with outliers and two small parameters $0 < \epsilon, \delta < 1$, a $(1 + \epsilon, 1 + \delta)$ -approximation of (P, γ) is a ball that covers at least $(1 - (1 + \delta)\gamma)n$ points and has radius at most $(1 + \epsilon)Rad(P_{opt})$.*

When both ϵ and δ are small, the bi-criteria approximation is very close to the optimal solution with only slight changes on the number of covered points and the radius.

We also extend Definition 2 to the problem called **minimum enclosing “x” (MEX) with Outliers**, where the “x” could be any specified shape. To keep the structure of our paper more compact, we state the formal definition of MEX with outliers and the corresponding results in Section 5.

2.1 A More Careful Analysis for Core-set Construction in [11]

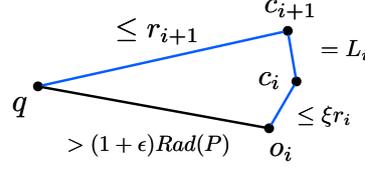
Before presenting our main results, we first revisit the core-set construction algorithm for MEB of Bădoiu and Clarkson [11], since their method will be used in our algorithms for MEB with outliers.

Let $0 < \epsilon < 1$. The algorithm in [11] yields an MEB core-set of size $2/\epsilon$ (for convenience, we always assume that $2/\epsilon$ is an integer). However, there is a small issue in their paper. The analysis assumes that the exact MEB of the core-set is computed in each iteration, but in fact one may only compute an approximate MEB. Thus, an immediate question is whether the quality is still guaranteed with such a change. Kumar et al. [50] fixed this issue, and showed that computing a $(1 + O(\epsilon^2))$ -approximate MEB for the core-set in each iteration still guarantees a core-set with size $O(1/\epsilon)$, where the hidden constant is larger than 80. Clarkson [20] systematically studied the *Frank-Wolfe* algorithm [33], and showed that the greedy core-set construction algorithm of MEB, as a special case of the Frank-Wolfe algorithm, yields a core-set with size slightly larger than $4/\epsilon$. Note that there exist several other methods yielding even lower core-set size [12, 49], but their construction algorithms are more complicated and thus not applicable to our problems. Increasing the core-set size from $2/\epsilon$ to α/ϵ (for some $\alpha > 2$) is neglectable in asymptotic analysis. But in Section 4, we will show that it could cause serious issue if outliers exist. Hence, a core-set of size $2/\epsilon$ is still desirable. **For this purpose, we provide a new analysis which is also interesting in its own right.**

For the sake of completeness, we first briefly introduce the idea of the core-set construction algorithm in [11]. Given a point set $P \subset \mathbb{R}^d$, the algorithm is a simple iterative procedure. Initially, it selects an arbitrary point from P and places it into an initially empty set T . In each of the following $2/\epsilon$ iterations, the algorithm updates the center of $MEB(T)$ and adds to T the farthest point from the current center of $MEB(T)$. Finally, the center of $MEB(T)$ induces a $(1 + \epsilon)$ -approximation for $MEB(P)$. The selected set of $2/\epsilon$ points (i.e., T) is called the core-set of MEB. To ensure the expected improvement in each iteration, [11] showed that the following two inequalities hold if the algorithm always selects the farthest point to the current center of $MEB(T)$:

$$r_{i+1} \geq (1 + \epsilon)Rad(P) - L_i; \quad r_{i+1} \geq \sqrt{r_i^2 + L_i^2}, \quad (1)$$

where r_i and r_{i+1} are the radii of $MEB(T)$ in the i -th and $(i + 1)$ -th iterations, respectively, and L_i is the shifting distance of the center of $MEB(T)$ from the i -th to $(i + 1)$ -th iteration.



■ **Figure 1** An illustration of (2).

As mentioned earlier, we often compute only an approximate $MEB(T)$ in each iteration. In the i -th iteration, we let c_i and o_i denote the centers of the exact and the approximate $MEB(T)$, respectively. Suppose that $\|c_i - o_i\| \leq \xi r_i$, where $\xi \in (0, \frac{\epsilon}{1+\epsilon})$ (we will see why this bound is needed later). Using another algorithm proposed in [11], one can obtain the point o_i in $O(\frac{1}{\xi^2}|T|d)$ time. Note that we only compute o_i rather than c_i in each iteration. Hence we can only select the farthest point (say q) to o_i . If $\|q - o_i\| \leq (1 + \epsilon)Rad(P)$, we are done and a $(1 + \epsilon)$ -approximation of MEB is already obtained. Otherwise, we have

$$(1 + \epsilon)Rad(P) < \|q - o_i\| \leq \|q - c_{i+1}\| + \|c_{i+1} - c_i\| + \|c_i - o_i\| \leq r_{i+1} + L_i + \xi r_i \quad (2)$$

by the triangle inequality (see Figure 1). In other words, we should replace the first inequality of (1) by $r_{i+1} > (1 + \epsilon)Rad(P) - L_i - \xi r_i$. Also, the second inequality of (1) still holds since it depends only on the property of the exact MEB (see Lemma 2.1 in [11]). Thus, we have

$$r_{i+1} \geq \max \left\{ \sqrt{r_i^2 + L_i^2}, (1 + \epsilon)Rad(P) - L_i - \xi r_i \right\}. \quad (3)$$

This leads to the following theorem whose proof can be found in Section A.2.

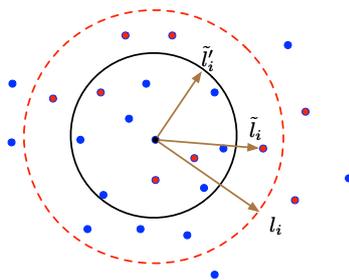
► **Theorem 4.** *In the core-set construction algorithm of [11], if one computes an approximate MEB for T in each iteration and the resulting center o_i has the distance to c_i less than $\xi r_i = s \frac{\epsilon}{1+\epsilon} r_i$ for some $s \in (0, 1)$, the final core-set size is bounded by $z = \frac{2}{(1-s)\epsilon}$. Also, the bound could be arbitrarily close to $2/\epsilon$ when s is small enough.*

► **Remark 5.** We want to emphasize a simple observation on the above core-set construction procedure, which will be used in our algorithms and analysis later on. The algorithm always selects the farthest point to o_i in each iteration. However, this is actually not necessary. As long as the selected point has distance at least $(1 + \epsilon)Rad(P)$, the inequality (2) always holds and the following analysis is still true. If no such a point exists (i.e., $P \setminus \mathbb{B}(o_i, (1 + \epsilon)Rad(P)) = \emptyset$), a $(1 + \epsilon)$ -approximate MEB (i.e., $\mathbb{B}(o_i, (1 + \epsilon)Rad(P))$) has already been obtained.

3 Two Key Lemmas for Handling Outliers

In this section, we introduce two important techniques, Lemma 6 and 7, for solving the problem of MEB with outliers in sub-linear time; the proofs are placed in Section 3.1 and 3.2, respectively. The algorithms are presented in Section 4. Moreover, these techniques can be generalized to solve a broader range of optimization problems, and we show the details in Section 5.

To shed some light on our ideas, consider using the core-set construction method in Section 2.1 to compute a bi-criteria $(1 + \epsilon, 1 + \delta)$ -approximation for an instance (P, γ) of MEB with outliers. Let o_i be the obtained ball center in the current iteration, and Q be the set of $(1 + \delta)\gamma n$ farthest points to o_i from P . A key step for updating o_i is finding a point in the set $P_{opt} \cap Q$ (the formal analysis is given in Section 4). Actually, this can be done



■ **Figure 2** The red points are the sampled n'' points in Lemma 7, and the $((1 + \delta)^2 \gamma n'' + 1)$ -th farthest point is in the ring bounded by the spheres $\mathbb{B}(o_i, \tilde{l}_i)$ and $\mathbb{B}(o_i, l_i)$.

by performing a random sampling from Q . However, it requires to compute the set Q in advance, which takes an $\Omega(nd)$ time complexity. To keep the running time to be sub-linear, we need to find a point from $P_{opt} \cap Q$ by a more sophisticated way.

Since P_{opt} is mixed with outliers in the set Q , simple uniform sampling cannot realize our goal. To remedy this issue, we propose a “two level” sampling procedure which is called “**Uniform-Adaptive Sampling**” (see Lemma 6). Roughly speaking, we take a random sample A of size n' first (i.e., the uniform sampling step), and then randomly select a point from Q' , the set of the farthest $\frac{3}{2}(1 + \delta)\gamma n'$ points from A to o_i (i.e., the adaptive sampling step). According to Lemma 6, with probability at least $(1 - \eta_1) \frac{\delta}{3(1 + \delta)}$, the selected point belongs to $P_{opt} \cap Q$; more importantly, the sample size n' is independent of n and d . The key to prove Lemma 6 is to show that the size of the intersection $Q' \cap (P_{opt} \cap Q)$ is large enough. By setting an appropriate value for n' , we can prove a lower bound of $|Q' \cap (P_{opt} \cap Q)|$.

► **Lemma 6** (Uniform-Adaptive Sampling). *Let $\eta_1 \in (0, 1)$. If we sample $n' = O(\frac{1}{\delta\gamma} \log \frac{1}{\eta_1})$ points independently and uniformly at random from P and let Q' be the set of farthest $\frac{3}{2}(1 + \delta)\gamma n'$ points to o_i from the sample, then, with probability at least $1 - \eta_1$, the following holds*

$$\frac{|Q' \cap (P_{opt} \cap Q)|}{|Q'|} \geq \frac{\delta}{3(1 + \delta)}. \quad (4)$$

The Uniform-Adaptive Sampling procedure will result in a “side-effect”. To boost the overall success probability, we have to repeatedly run the algorithm multiple times and each time the algorithm will generate a candidate solution (i.e., the ball center). Consequently we have to select the best one as our final solution. With a slight abuse of notation, we still use o_i to denote a candidate ball center; to achieve a $(1 + \epsilon, 1 + \delta)$ -approximation, its radius should be the $((1 + \delta)\gamma n + 1)$ -th largest distance from P to o_i , which is denoted as l_i . A straightforward way is to compute the value “ l_i ” in linear time for each candidate and return the one having the smallest l_i . In this section, we propose the “**Sandwich Lemma**” to estimate l_i in sub-linear time (see Lemma 7). Let B be the set of n'' sampled points from P in Lemma 7, and \tilde{l}_i be the $((1 + \delta)^2 \gamma n'' + 1)$ -th largest distance from B to o_i . The key idea is to prove that the ball $\mathbb{B}(o_i, \tilde{l}_i)$ is “sandwiched” by two balls $\mathbb{B}(o_i, \tilde{l}_i')$ and $\mathbb{B}(o_i, l_i)$, where \tilde{l}_i' is a carefully designed value satisfying (i) $\tilde{l}_i' \leq \tilde{l}_i \leq l_i$ and (ii) $|P \setminus \mathbb{B}(o_i, \tilde{l}_i')| \leq (1 + O(\delta))\gamma n$. See Figure 2 for an illustration. These two conditions of \tilde{l}_i' can imply the inequalities (5) and (6) of Lemma 7. Further, the inequalities (5) and (6) jointly imply that \tilde{l}_i is a qualified estimation of l_i : if $\mathbb{B}(o_i, l_i)$ is a $(1 + \epsilon, 1 + \delta)$ -approximation, the ball $\mathbb{B}(o_i, \tilde{l}_i)$ should be a $(1 + \epsilon, 1 + O(\delta))$ -approximation. Similar to Lemma 6, the sample size n'' is also independent of n and d .

► **Lemma 7** (Sandwich Lemma). *Let $\eta_2 \in (0, 1)$ and assume $\delta < 1/3$. If we sample $n'' = O\left(\frac{1}{\delta^2\gamma} \log \frac{1}{\eta_2}\right)$ points independently and uniformly at random from P and let \tilde{l}_i be the $((1 + \delta)^2\gamma n'' + 1)$ -th largest distance from the sample to o_i , then, with probability $1 - \eta_2$, the following holds*

$$\tilde{l}_i \leq l_i; \quad (5)$$

$$\left| P \setminus \mathbb{B}(o_i, \tilde{l}_i) \right| \leq (1 + O(\delta))\gamma n. \quad (6)$$

3.1 Proof of Lemma 6

Let A denote the set of sampled n' points from P . First, we know $|Q| = (1 + \delta)\gamma n$ and $|P_{opt} \cap Q| \geq \delta\gamma n$ (since there are at most γn outliers in Q). For ease of presentation, let $\lambda = \frac{|P_{opt} \cap Q|}{n} \geq \delta\gamma$. Let $\{x_i \mid 1 \leq i \leq n'\}$ be n' independent random variables with $x_i = 1$ if the i -th sampled point of A belongs to $P_{opt} \cap Q$, and $x_i = 0$ otherwise. Thus, $E[x_i] = \lambda$ for each i . Let σ be a small parameter in $(0, 1)$. By using the Chernoff bound, we have $\Pr\left(\sum_{i=1}^{n'} x_i \notin (1 \pm \sigma)\lambda n'\right) \leq e^{-O(\sigma^2\lambda n')}$. That is,

$$\Pr\left(|A \cap (P_{opt} \cap Q)| \in (1 \pm \sigma)\lambda n'\right) \geq 1 - e^{-O(\sigma^2\lambda n')}. \quad (7)$$

Similarly, we have

$$\Pr\left(|A \cap Q| \in (1 \pm \sigma)(1 + \delta)\gamma n'\right) \geq 1 - e^{-O(\sigma^2(1+\delta)\gamma n')}. \quad (8)$$

Note that $n' = O\left(\frac{1}{\delta\gamma} \log \frac{1}{\eta_1}\right)$. By setting $\sigma < 1/2$ in (7) and (8), we have

$$\left| A \cap (P_{opt} \cap Q) \right| > \frac{1}{2}\delta\gamma n' \quad \text{and} \quad \left| A \cap Q \right| < \frac{3}{2}(1 + \delta)\gamma n' \quad (9)$$

with probability $1 - \eta_1$. Note that Q contains all the farthest $(1 + \delta)\gamma n$ points to o_i . Denote by l_i the $((1 + \delta)\gamma n + 1)$ -th largest distance from P to o_i . Thus

$$A \cap Q = \{p \in A \mid \|p - o_i\| > l_i\}. \quad (10)$$

Also, since Q' is the set of the farthest $\frac{3}{2}(1 + \delta)\gamma n'$ points to o_i from A , there exists some $l'_i > 0$ such that

$$Q' = \{p \in A \mid \|p - o_i\| > l'_i\}. \quad (11)$$

(10) and (11) imply that either $(A \cap Q) \subseteq Q'$ or $Q' \subseteq (A \cap Q)$. Since $|A \cap Q| < \frac{3}{2}(1 + \delta)\gamma n'$ and $|Q'| = \frac{3}{2}(1 + \delta)\gamma n'$, we know $(A \cap Q) \subseteq Q'$. Therefore,

$$\left(A \cap (P_{opt} \cap Q) \right) = \left(P_{opt} \cap (A \cap Q) \right) \subseteq Q'. \quad (12)$$

Obviously,

$$\left(A \cap (P_{opt} \cap Q) \right) \subseteq (P_{opt} \cap Q). \quad (13)$$

The above (12) and (13) together imply

$$\left(A \cap (P_{opt} \cap Q) \right) \subseteq \left(Q' \cap (P_{opt} \cap Q) \right). \quad (14)$$

Moreover, since $Q' \subseteq A$, we have

$$\left(Q' \cap (P_{opt} \cap Q) \right) \subseteq \left(A \cap (P_{opt} \cap Q) \right). \quad (15)$$

Consequently, (14) and (15) together imply $Q' \cap (P_{opt} \cap Q) = A \cap (P_{opt} \cap Q)$ and hence

$$\frac{|Q' \cap (P_{opt} \cap Q)|}{|Q'|} = \frac{|A \cap (P_{opt} \cap Q)|}{|Q'|} \geq \frac{\delta}{3(1+\delta)}, \quad (16)$$

where the final inequality comes from the first inequality of (9) and the fact $|Q'| = \frac{3}{2}(1+\delta)\gamma n'$.

3.2 Proof of Lemma 7

Let B denote the set of sampled n'' points from P . For simplicity, let $t = (1+\delta)\gamma n$. Assume $\tilde{l}'_i > 0$ is the value such that $|P \setminus \mathbb{B}(o_i, \tilde{l}'_i)| = \frac{(1+\delta)^2}{1-\delta}\gamma n$. Recall that l_i is the $(t+1)$ -th largest distance from P to o_i . Since $(1+\delta)\gamma n < \frac{(1+\delta)^2}{1-\delta}\gamma n$, it is easy to know $\tilde{l}'_i \leq l_i$. Below, we aim to prove that the $((1+\delta)^2\gamma n'' + 1)$ -th farthest point from B is in the ring bounded by the spheres $\mathbb{B}(o_i, \tilde{l}'_i)$ and $\mathbb{B}(o_i, l_i)$ (see Figure 2).

Again, using the Chernoff bound (let $\sigma = \delta/2$) and the same idea for proving (9), since $|B| = n'' = O(\frac{1}{\delta^2\gamma} \log \frac{1}{\eta_2})$, we have

$$|B \setminus \mathbb{B}(o_i, \tilde{l}'_i)| \geq (1-\delta/2) \frac{(1+\delta)^2}{1-\delta} \gamma n'' > (1-\delta) \frac{(1+\delta)^2}{1-\delta} \gamma n'' = (1+\delta)^2 \gamma n''; \quad (17)$$

$$|B \cap Q| \leq (1+\delta/2) \frac{t}{n} n'' < (1+\delta) \frac{t}{n} n'' = (1+\delta)^2 \gamma n'', \quad (18)$$

with probability $1 - \eta_2$. Suppose that (17) and (18) both hold. Recall that \tilde{l}_i is the $((1+\delta)^2\gamma n'' + 1)$ -th largest distance from the sampled points B to o_i , so $|B \setminus \mathbb{B}(o_i, \tilde{l}_i)| = (1+\delta)^2\gamma n''$, and thus $\tilde{l}_i \geq \tilde{l}'_i$ by (17).

The inequality (18) implies that the $((1+\delta)^2\gamma n'' + 1)$ -th farthest point (say q_x) from B to o_i is not in Q . Then, we claim that $\mathbb{B}(o_i, \tilde{l}_i) \cap Q = \emptyset$. Otherwise, let $q_y \in \mathbb{B}(o_i, \tilde{l}_i) \cap Q$. Then we have

$$\|q_y - o_i\| \leq \tilde{l}_i = \|q_x - o_i\|. \quad (19)$$

Note that Q is the set of farthest t points to o_i of P . So $q_x \notin Q$ implies

$$\|q_x - o_i\| < \min_{q \in Q} \|q - o_i\| \leq \|q_y - o_i\| \quad (20)$$

which is in contradiction to (19). Therefore, $\mathbb{B}(o_i, \tilde{l}_i) \cap Q = \emptyset$. Further, since $\mathbb{B}(o_i, l_i)$ excludes exactly the farthest t points (i.e., Q), $\mathbb{B}(o_i, \tilde{l}_i) \cap Q = \emptyset$ implies $\tilde{l}_i \leq l_i$.

Overall, we have $\tilde{l}_i \in [\tilde{l}'_i, l_i]$, i.e., the $((1+\delta)^2\gamma n'' + 1)$ -th farthest point from B locates in the ring bounded by the spheres $\mathbb{B}(o_i, \tilde{l}'_i)$ and $\mathbb{B}(o_i, l_i)$ as shown in Figure 2. Also, $\tilde{l}_i \geq \tilde{l}'_i$ implies

$$|P \setminus \mathbb{B}(o_i, \tilde{l}_i)| \leq |P \setminus \mathbb{B}(o_i, \tilde{l}'_i)| = \frac{(1+\delta)^2}{1-\delta} \gamma n = (1+O(\delta))\gamma n, \quad (21)$$

where the last equality comes from the assumption $\delta < 1/3$. So (5) and (6) are true in Lemma 7.

4 Sub-linear Time Algorithm of MEB with Outliers

Recall the remark following Theorem 4. As long as the selected point has a distance to the center of $MEB(T)$ larger than $(1+\epsilon)$ times the optimal radius, the expected improvement will always be guaranteed. Following this observation, we investigate the following approach.

Suppose we run the core-set construction procedure described in Theorem 4 (we should replace P by P_{opt} in our following analysis). In the i -th step, we add an arbitrary point from $P_{opt} \setminus \mathbb{B}(o_i, (1 + \epsilon)Rad(P_{opt}))$ to T . We know that a $(1 + \epsilon)$ -approximation is obtained after at most $\frac{2}{(1-s)\epsilon}$ steps, that is, $P_{opt} \subset \mathbb{B}(o_i, (1 + \epsilon)Rad(P_{opt}))$ for some $i \leq \frac{2}{(1-s)\epsilon}$.

However, we need to solve two key issues in order to implement the above approach: **(i)** how to determine the value of $Rad(P_{opt})$ and **(ii)** how to correctly select a point from $P_{opt} \setminus \mathbb{B}(o_i, (1 + \epsilon)Rad(P_{opt}))$. Actually, we can implicitly avoid the first issue via replacing $(1 + \epsilon)Rad(P_{opt})$ by the t -th largest distance from the points of P to o_i , where we set $t = (1 + \delta)\gamma n$ for achieving a $(1 + \epsilon, 1 + \delta)$ -approximation in the following analysis. For the second issue, we randomly select one point from the farthest t points of P to o_i , and show that it belongs to $P_{opt} \setminus \mathbb{B}(o_i, (1 + \epsilon)Rad(P_{opt}))$ with a certain probability.

Based on the above idea, we present a sub-linear time $(1 + \epsilon, 1 + \delta)$ -approximation algorithm in this section. To better understand the algorithm, we show a linear time algorithm first (Algorithm 1 in Sections 4.1). Note that Bădoiu et al. [13] also achieved a $(1 + \epsilon, 1 + \delta)$ -approximation algorithm but with a higher complexity. **Please see more details in our analysis on the running time at the end of Sections 4.1.** More importantly, we improve the running time of Algorithm 1 to be sub-linear. For this purpose, we need to avoid computing the farthest t points to o_i , since this operation will take linear time. Also, Algorithm 1 generates a set of candidates for the solution and we need to select the best one. This process also costs linear time. By using the techniques proposed in Section 3, we can remedy these issues and develop a sub-linear time algorithm that has the sample complexity independent of n and d , in Section 4.2.

4.1 A Linear Time Algorithm

■ **Algorithm 1** $(1 + \epsilon, 1 + \delta)$ -approximation Algorithm for MEB with Outliers.

Input: A point set P with n points in \mathbb{R}^d , the fraction of outliers $\gamma \in (0, 1)$, and the parameters $0 < \epsilon, \delta < 1$, $z \in \mathbb{Z}^+$.

- 1: Let $t = (1 + \delta)\gamma n$.
- 2: Initially, randomly select a point $p \in P$ and let $T = \{p\}$.
- 3: $i = 1$; repeat the following steps until $i > z$:
 - (1) Compute the approximate MEB center o_i of T .
 - (2) Let Q be the set of farthest t points from P to o_i ; denote by l_i the $(t + 1)$ -th largest distance from P to o_i .
 - (3) Randomly select a point $q \in Q$, and add it to T .
 - (4) $i = i + 1$.
- 4: Output the ball $\mathbb{B}(o_i, l_i)$ where $\hat{i} = \arg \min\{l_i \mid 1 \leq i \leq z\}$.

In this section, we present our linear time $(1 + \epsilon, 1 + \delta)$ -approximation algorithm for MEB with outliers (see Algorithm 1). For convenience, denote by c_i and r_i the exact center and radius of $MEB(T)$ respectively in the i -th round of Step 3 of Algorithm 1. In Step 3(1), we compute the approximate center o_i with a distance to c_i of less than $\xi Rad(T) = s \frac{\epsilon}{1 + \epsilon} Rad(T)$, where $s \in (0, 1)$ as described in Theorem 4 (we will determine the value of s in our following analysis on the running time). The following theorem shows the success probability of Algorithm 1.

► **Theorem 8.** *If the input parameter $z = \frac{2}{(1-s)\epsilon}$ (we assume it is an integer for convenience), then with probability $(1 - \gamma)(\frac{\delta}{1 + \delta})^z$, Algorithm 1 outputs a $(1 + \epsilon, 1 + \delta)$ -approximation for the MEB with outliers problem.*

Before proving Theorem 8, we present the following two lemmas first.

► **Lemma 9.** *With probability $(1 - \gamma)(\frac{\delta}{1+\delta})^z$, after running z rounds in Step 3, the set $T \subset P_{opt}$ in Algorithm 1.*

Proof. Initially, because $|P_{opt}|/|P| = 1 - \gamma$, the first selected point in Step 2 belongs to P_{opt} with probability $1 - \gamma$. In each of the z rounds in Step 3, the selected point belongs to P_{opt} with probability $\frac{\delta}{1+\delta}$, since

$$\frac{|P_{opt} \cap Q|}{|Q|} = 1 - \frac{|Q \setminus P_{opt}|}{|Q|} \geq 1 - \frac{|P \setminus P_{opt}|}{|Q|} = 1 - \frac{\gamma n}{(1 + \delta)\gamma n} = \frac{\delta}{1 + \delta}. \quad (22)$$

Therefore, $T \subset P_{opt}$ with probability $(1 - \gamma)(\frac{\delta}{1+\delta})^z$. ◀

► **Lemma 10.** *In the i -th round of Step 3 for $1 \leq i \leq z$, at least one of the following two events happens: (1) o_i is the ball center of a $(1 + \epsilon, 1 + \delta)$ -approximation; (2) $r_{i+1} > (1 + \epsilon)Rad(P_{opt}) - \|c_i - c_{i+1}\| - \xi r_i$.*

Proof. If $l_i \leq (1 + \epsilon)Rad(P_{opt})$, then we are done. That is, the ball $\mathbb{B}(o_i, l_i)$ covers $(1 - (1 + \delta)\gamma)n$ points with radius $l_i \leq (1 + \epsilon)Rad(P_{opt})$ (the first event happens). Otherwise, $l_i > (1 + \epsilon)Rad(P_{opt})$ and we consider the second event. Let q be the point added to T in the i -th round. Using the triangle inequality, we have

$$\|o_i - q\| \leq \|o_i - c_i\| + \|c_i - c_{i+1}\| + \|c_{i+1} - q\| \leq \xi r_i + \|c_i - c_{i+1}\| + r_{i+1}. \quad (23)$$

Since $l_i > (1 + \epsilon)Rad(P_{opt})$ and q lies outside of $\mathbb{B}(o_i, l_i)$, i.e., $\|o_i - q\| \geq l_i > (1 + \epsilon)Rad(P_{opt})$, (23) implies that the second event happens and the proof is completed. ◀

Proof of Theorem 8. Suppose that the first event of Lemma 10 never happens. As a consequence, we obtain a series of inequalities for each pair of radii r_{i+1} and r_i , i.e., $r_{i+1} > (1 + \epsilon)Rad(P_{opt}) - \|c_i - c_{i+1}\| - \xi r_i$. Assume that $T \subset P_{opt}$ in Lemma 9, i.e., each time the algorithm correctly adds a point from P_{opt} to T . Using the almost identical idea for proving Theorem 4 in Section 2.1, we know that a $(1 + \epsilon)$ -approximate MEB of P_{opt} is obtained after at most z rounds. The success probability directly comes from Lemma 9. Overall, we obtain Theorem 8. ◀

Moreover, Theorem 8 implies the following corollary.

► **Corollary 11.** *If one repeatedly runs Algorithm 1 $O(\frac{1}{1-\gamma}(1 + \frac{1}{\delta})^z)$ times, with constant probability, the algorithm outputs a $(1 + \epsilon, 1 + \delta)$ -approximation for the problem of MEB with outliers.*

Running time. In Theorem 8, we set $z = \frac{2}{(1-s)\epsilon}$ and $s \in (0, 1)$. To keep z small, according to Theorem 4, we set $s = \frac{\epsilon}{2+\epsilon}$ so that $z = \frac{2}{\epsilon} + 1$ (only larger than the lower bound $\frac{2}{\epsilon}$ by 1). For each round of Step 3, we need to compute an approximate center o_i that has a distance to the exact one less than $\xi r_i = s \frac{\epsilon}{1+\epsilon} r_i = O(\epsilon^2)r_i$. Using the algorithm proposed in [11], this can be done in $O(\frac{1}{\epsilon^2}|T|d) = O(\frac{1}{\epsilon^5}d)$ time. Also, the set Q can be obtained in linear time by the algorithm in [10]. In total, the time complexity for obtaining a $(1 + \epsilon, 1 + \delta)$ -approximation in Corollary 11 is

$$O\left(\frac{C}{\epsilon}\left(n + \frac{1}{\epsilon^5}\right)d\right), \quad (24)$$

where $C = O(\frac{1}{1-\gamma}(1 + \frac{1}{\delta})^{\frac{2}{\epsilon}+1})$. As mentioned before, Bădoiu et al. [13] also achieved a linear time bi-criteria approximation. However, the hidden constant of their running time is exponential in $\Theta(\frac{1}{\epsilon\mu})$ (where μ is defined in [13], and should be $\delta\gamma$ to ensure a $(1 + \epsilon, 1 + \delta)$ -approximation) that is much larger than $\frac{2}{\epsilon} + 1$.

4.2 Improvement on Running Time

In this section, we show that the running time of Algorithm 1 can be further improved to be independent of the number of points n . First, we observe that it is not necessary to compute the set Q of the farthest t points in Step 3(2) of the algorithm. Actually, as long as the selected point q is part of $P_{opt} \cap Q$ in Step 3(3), a $(1 + \epsilon, 1 + \delta)$ -approximation is still guaranteed. The Uniform-Adaptive Sampling procedure proposed in Section 3 can help us to obtain a point $q \in P_{opt} \cap Q$ without computing the set Q . Moreover, in Lemma 7, we show that the radius of each candidate solution can be estimated via random sampling. Overall, we achieve a sub-linear time algorithm (Algorithm 2). Following the analysis in Section 4.1, we set $s = \frac{\epsilon}{2+\epsilon}$ so that $z = \frac{2}{(1-s)\epsilon} = \frac{2}{\epsilon} + 1$. We present the results in Theorem 12 and Corollary 13. Comparing with Theorem 8, we have an extra $(1 - \eta_1)(1 - \eta_2)$ in the success probability in Theorem 12, due to the probabilities from Lemmas 6 and 7.

■ **Algorithm 2** Sub-linear Time $(1 + \epsilon, 1 + O(\delta))$ -approximation Algorithm for MEB with Outliers.

Input: A point set P with n points in \mathbb{R}^d , the fraction of outliers $\gamma \in (0, 1)$, and the parameters $\epsilon, \eta_1, \eta_2 \in (0, 1)$, $\delta \in (0, 1/3)$, and $z \in \mathbb{Z}^+$.

- 1: Let $n' = O(\frac{1}{\delta\gamma} \log \frac{1}{\eta_1})$, $n'' = O(\frac{1}{\delta^2\gamma} \log \frac{1}{\eta_2})$, $t' = \frac{3}{2}(1 + \delta)\gamma n'$, and $t'' = (1 + \delta)^2\gamma n''$.
- 2: Initially, randomly select a point $p \in P$ and let $T = \{p\}$.
- 3: $i = 1$; repeat the following steps until $j = z$:
 - (1) Compute the approximate MEB center o_i of T .
 - (2) Sample n' points uniformly at random from P , and let Q' be the set of farthest t' points to o_i from the sample.
 - (3) Randomly select a point $q \in Q'$, and add it to T .
 - (4) Sample n'' points uniformly at random from P , and let \tilde{l}_i be the $(t'' + 1)$ -th largest distance from the sampled points to o_i .
 - (5) $i = i + 1$.
- 4: Output the ball $\mathbb{B}(o_{\hat{i}}, \tilde{l}_{\hat{i}})$ where $\hat{i} = \arg_i \min\{\tilde{l}_i \mid 1 \leq i \leq z\}$.

► **Theorem 12.** *If the input parameter $z = \frac{2}{\epsilon} + 1$, then with probability $(1 - \gamma)((1 - \eta_1)(1 - \eta_2)\frac{\delta}{3(1+\delta)})^z$, Algorithm 2 outputs a $(1 + \epsilon, 1 + O(\delta))$ -approximation for the problem of MEB with outliers.*

To boost the success probability in Theorem 12, we need to repeatedly run Algorithm 2 and output the best candidate. However, we need to be careful on setting the parameters. The success probability in Theorem 12 consists of two parts, $\mathcal{P}_1 = (1 - \gamma)((1 - \eta_1)\frac{\delta}{3(1+\delta)})^z$ and $\mathcal{P}_2 = (1 - \eta_2)^z$, where \mathcal{P}_1 indicates the probability that $\{o_1, \dots, o_z\}$ contains a qualified candidate, and \mathcal{P}_2 indicates the success probability of Lemma 7 over all the z rounds. Therefore, if we run Algorithm 2 $N = O(\frac{1}{\mathcal{P}_1})$ times, with constant probability (by taking the union bound), the set of all the generated candidates contains at least one that yields a $(1 + \epsilon, 1 + O(\delta))$ -approximation; moreover, to guarantee that we can correctly estimate the resulting radii of all the candidates via the Sandwich Lemma with constant probability, we need to set $\eta_2 = O(\frac{1}{zN})$ (because there are $O(zN)$ candidates).

► **Corollary 13.** *If one repeatedly runs Algorithm 2 $N = O\left(\frac{1}{1-\gamma}\left(\frac{1}{1-\eta_1}\left(3 + \frac{3}{\delta}\right)\right)^z\right)$ times with setting $\eta_2 = O(\frac{1}{zN})$, with constant probability, the algorithm outputs a $(1 + \epsilon, 1 + O(\delta))$ -approximation for the problem of MEB with outliers.*

The calculation of running time is similar to (24) in Section 4.1. We just replace n by $\max\{n', n''\} = O(\frac{1}{\delta^{2\gamma}} \log \frac{1}{\eta_2}) = O(\frac{1}{\delta^{2\gamma}} \log(zN)) = \tilde{O}(\frac{1}{\delta^{2\gamma\epsilon}})^2$, and change the value of C to be $O(\frac{1}{1-\gamma} (\frac{1}{1-\eta_1} (3 + \frac{3}{\delta}))^{\frac{2}{\epsilon} + 1})$. So the total running time is independent of n . Also, to convert the result from $(1 + \epsilon, 1 + O(\delta))$ -approximation to $(1 + \epsilon, 1 + \delta)$ -approximation, we just need to reduce the value of δ in the input of Algorithm 2 appropriately.

5 The Extension: MEX with Outliers

In this section, we extend Definition 2 and define a more general problem called **minimum enclosing “x” (MEX) with Outliers**. We observe that the ideas of Lemma 6 and 7 can be further extended to deal with MEX with outliers problems, as long as the shape “x” satisfies several properties. **To describe a shape “x”, we need to clarify three basic concepts: center, size, and distance function.**

Let \mathcal{X} be the set of specified shapes in \mathbb{R}^d . In this paper, we require that each shape $x \in \mathcal{X}$ is uniquely determined by the following two components: “ $c(x)$ ”, the **center** of x , and “ $s(x) \geq 0$ ”, the **size** of x . For any two shapes $x_1, x_2 \in \mathcal{X}$, $x_1 = x_2$ if and only if $c(x_1) = c(x_2)$ and $s(x_1) = s(x_2)$. Moreover, given a center o_0 and a value $l_0 \geq 0$, we use $x(o_0, l_0)$ to denote the shape x with $c(x) = o_0$ and $s(x) = l_0$. For different shapes, we have different definitions for the center and size. For example, if x is a ball, $c(x)$ and $s(x)$ should be the ball center and the radius respectively; given $o_0 \in \mathbb{R}^d$ and $l_0 \geq 0$, $x(o_0, l_0)$ should be the ball $\mathbb{B}(o_0, l_0)$. As a more complicated example, consider the k -center clustering with outliers problem, which is to find k balls to cover the input point set excluding a certain number of outliers and minimize the maximum radius (*w.l.o.g.*, we can assume that the k balls have the same radius). For this problem, the shape “x” is a union of k balls in \mathbb{R}^d ; the center $c(x)$ is the set of the k ball centers and the size $s(x)$ is the radius.

For any point $p \in \mathbb{R}^d$ and any shape $x \in \mathcal{X}$, we also need to define a **distance function** $f(c(x), p)$ between the center $c(x)$ and p . For example, if x is a ball, $f(c(x), p)$ is simply equal to $\|p - c(x)\|$; if x is a union of k balls with the center $c(x) = \{c_1, c_2, \dots, c_k\}$, the distance should be $\min_{1 \leq j \leq k} \|p - c_j\|$. Note that the distance function is only for ranking the points to $c(x)$, and not necessary to be non-negative (e.g., we define a distance function $f(c(x), p) \leq 0$ for SVM). By using this distance function, we can define the set “ Q ” and the value “ l_i ” when generalizing Lemma 6 and 7 below. To guarantee their correctnesses, we also require \mathcal{X} to satisfy the following three properties.

► **Property 1.** For any two shapes $x_1 \neq x_2 \in \mathcal{X}$, if $c(x_1) = c(x_2)$, then

$$s(x_1) \leq s(x_2) \iff x_1 \text{ is covered by } x_2, \quad (25)$$

where “ x_1 is covered by x_2 ” means “for any point $p \in \mathbb{R}^d$, $p \in x_1 \Rightarrow p \in x_2$ ”.

► **Property 2.** Given any shape $x \in \mathcal{X}$ and any point $p_0 \in x$, the set

$$\{p \mid p \in \mathbb{R}^d \text{ and } f(c(x), p) \leq f(c(x), p_0)\} \subseteq x. \quad (26)$$

► **Property 3.** Given any shape center o_0 and any point $p_0 \in \mathbb{R}^d$, they together determine a value $r_0 = \min\{r \mid r \geq 0, p_0 \in x(o_0, r)\}$; that is, $p_0 \in x(o_0, r_0)$ and $p_0 \notin x(o_0, r)$ for any $r < r_0$. (**Note:** usually the value r_0 is just the distance from p_0 to the shape center o_0 ; but for some cases, such as the SVM problem, the shape size and distance function have different meanings).

² The asymptotic notation $\tilde{O}(f) = O(f \cdot \text{polylog}(\frac{1}{\eta_1 \delta^{1-\gamma}}))$.

Intuitively, Property 1 shows that $s(x)$ defines an order of the shapes sharing the same center $c(x)$. Property 2 shows that the distance function f defines an order of the points to a given shape center $c(x)$. Property 3 shows that a center o_0 and a point p_0 can define a shape just “touching” p_0 . We can take $\mathcal{X} = \{\text{all } d\text{-dimensional balls}\}$ as an example. For any two concentric balls, the smaller one is always covered by the larger one (Property 1); if a point p_0 is inside a ball x , any point p having the distance $\|p - c(x)\| \leq \|p_0 - c(x)\|$ should be inside x too (Property 2); also, given a ball center o_0 and a point p_0 , $p_0 \in \mathbb{B}(o_0, \|p_0 - o_0\|)$ and $p_0 \notin \mathbb{B}(o_0, r)$ for any $r < \|p_0 - o_0\|$ (Property 3).

Now, we introduce the formal definitions of the MEX with outliers problem and its bi-criteria approximation.

► **Definition 14** (MEX with Outliers). *Suppose the shape set \mathcal{X} satisfies Property 1, 2, and 3. Given a set P of n points in \mathbb{R}^d and a small parameter $\gamma \in (0, 1)$, the MEX with outliers problem is to find the smallest shape $x \in \mathcal{X}$ that covers $(1 - \gamma)n$ points. Namely, the task is to find a subset of P with size $(1 - \gamma)n$ such that its minimum enclosing shape of \mathcal{X} is the smallest among all possible choices of the subset. The obtained solution is denoted by $\text{MEX}(P, \gamma)$.*

► **Definition 15** (Bi-criteria Approximation). *Given an instance (P, γ) for MEX with outliers and two small parameters $0 < \epsilon, \delta < 1$, a $(1 + \epsilon, 1 + \delta)$ -approximation of (P, γ) is a solution $x \in \mathcal{X}$ that covers at least $(1 - (1 + \delta)\gamma)n$ points and has the size at most $(1 + \epsilon)s(x_{\text{opt}})$, where x_{opt} is the optimal solution.*

It is easy to see that Definition 2 of MEB with outliers actually is a special case of Definition 14. Similar to MEB with outliers, we still use P_{opt} , where $P_{\text{opt}} \subset P$ and $|P_{\text{opt}}| = (1 - \gamma)n$, to denote the subset covered by the optimal solution of MEX with outliers.

Now, we provide the generalized versions of Lemma 6 and 7. Similar to the core-set construction method in Section 2.1, we assume that there exists an iterative algorithm Γ to compute MEX (without outliers); actually, this is an important prerequisite to design the sub-linear time algorithms under our framework (we will discuss the iterative algorithms for the MEX with outliers problems considered in our paper, including flat fitting, k -center clustering, and SVM, in the appendix). In the i -th iteration of Γ , it maintains a shape center o_i . Also, let Q be the set of $(1 + \delta)\gamma n$ farthest points from P to o_i with respect to the distance function f . First, we need to define the value “ l_i ” by Q in the following claim.

▷ **Claim 16.** There exists a value $l_i \geq 0$ satisfying $P \setminus x(o_i, l_i) = Q$.

Proof. The points of P can be ranked based on their distances to o_i . Without loss of generality, let $P = \{p_1, p_2, \dots, p_n\}$ with $f(o_i, p_1) > f(o_i, p_2) > \dots > f(o_i, p_n)$ (for convenience, we assume that any two distances are not equal; if there is a tie, we can arbitrarily decide their order to o_i). Therefore, the set $Q = \{p_j \mid 1 \leq j \leq (1 + \delta)\gamma n\}$. Moreover, from Property 3, we know that each point $p_j \in P$ corresponds to a value r_j that $p_j \in x(o_i, r_j)$ and $p_j \notin x(o_i, r)$ for any $r < r_j$. Denote by x_j the shape $x(o_i, r_j)$. We select the point p_{j_0} with $j_0 = (1 + \delta)\gamma n + 1$. From Property 2, we know that $p_j \in x_{j_0}$ for any $j \geq j_0$, i.e., **(a)** $P \setminus Q \subseteq x_{j_0}$. We also need to prove that $p_j \notin x_{j_0}$ for any $j < j_0$. Assume there exists some $p_{j_1} \in x_{j_0}$ with $j_1 < j_0$. Then we have $r_{j_1} < r_{j_0}$ and thus $p_{j_0} \notin x_{j_1}$ (by Property 3). By Property 2, $p_{j_0} \notin x_{j_1}$ implies $f(o_i, p_{j_0}) > f(o_i, p_{j_1})$, which is in contradiction to the fact $f(o_i, p_{j_0}) < f(o_i, p_{j_1})$. So we have **(b)** $Q \cap x_{j_0} = \emptyset$.

From the above **(a)** and **(b)**, we know $P \setminus x_{j_0} = Q$. Therefore, we can set the value $l_i = r_{j_0}$ and then $P \setminus x(o_i, l_i) = Q$. ◁

► **Lemma 17** (Generalized Uniform-Adaptive Sampling). *Let $\eta_1 \in (0, 1)$. If we sample $n' = O(\frac{1}{\delta\gamma} \log \frac{1}{\eta_1})$ points independently and uniformly at random from P and let Q' be the set of farthest $\frac{3}{2}(1 + \delta)\gamma n'$ points to o_i from the sample, then, with probability at least $1 - \eta_1$, the following holds*

$$\frac{|Q' \cap (P_{opt} \cap Q)|}{|Q'|} \geq \frac{\delta}{3(1 + \delta)}. \quad (27)$$

Proof. Let A denote the set of sampled n' points from P . Similar to (9), we have

$$|A \cap (P_{opt} \cap Q)| > \frac{1}{2}\delta\gamma n' \quad \text{and} \quad |A \cap Q| < \frac{3}{2}(1 + \delta)\gamma n' \quad (28)$$

with probability $1 - \eta_1$. Similar to (10), we have

$$A \cap Q = \{p \in A \mid f(o_i, p) > f(o_i, p_{j_0})\}, \quad (29)$$

where p_{j_0} is the point selected in the proof of Claim 16. By using the same manner of Claim 16, we also can select a point $p_{j'_0} \in A$ with

$$Q' = \{p \in A \mid f(o_i, p) > f(o_i, p_{j'_0})\}. \quad (30)$$

Then, we can prove

$$(A \cap (P_{opt} \cap Q)) = (Q' \cap (P_{opt} \cap Q)). \quad (31)$$

by using the same idea of (14). Hence,

$$\frac{|Q' \cap (P_{opt} \cap Q)|}{|Q'|} = \frac{|A \cap (P_{opt} \cap Q)|}{|Q'|} \geq \frac{\delta}{3(1 + \delta)}, \quad (32)$$

where the final inequality comes from the first inequality of (28) and the fact $|Q'| = \frac{3}{2}(1 + \delta)\gamma n'$. ◀

► **Lemma 18** (Generalized Sandwich Lemma). *Let $\eta_2 \in (0, 1)$ and assume $\delta < 1/3$. l_i is the value from Claim 16. We sample $n'' = O(\frac{1}{\delta^2\gamma} \log \frac{1}{\eta_2})$ points independently and uniformly at random from P and let q be the $((1 + \delta)^2\gamma n'' + 1)$ -th farthest one from the sampled points to o_i . If $\tilde{l}_i = \min\{r \mid r \geq 0, q \in x(o_i, r)\}$ (similar to the way defining “ r_0 ” in Property 3), then, with probability $1 - \eta_2$, the following holds*

$$\tilde{l}_i \leq l_i; \quad (33)$$

$$|P \setminus x(o_i, \tilde{l}_i)| \leq (1 + O(\delta))\gamma n. \quad (34)$$

Proof. Let B denote the set of sampled n'' points from P . By using the same manner of Claim 16, we know that there exists a value $\tilde{l}'_i > 0$ satisfying $|P \setminus x(o_i, \tilde{l}'_i)| = \frac{(1 + \delta)^2}{1 - \delta}\gamma n$. Similar to the proof of Lemma 7, we can prove that $\tilde{l}_i \in [\tilde{l}'_i, l_i]$. Due to Property 1, we know that $x(o_i, \tilde{l}_i)$ is “sandwiched” by the two shapes $x(o_i, \tilde{l}'_i)$ and $x(o_i, l_i)$. Further, since $x(o_i, \tilde{l}_i)$ is covered by $x(o_i, \tilde{l}'_i)$, we have

$$|P \setminus x(o_i, \tilde{l}_i)| \leq |P \setminus x(o_i, \tilde{l}'_i)| = \frac{(1 + \delta)^2}{1 - \delta}\gamma n = (1 + O(\delta))\gamma n, \quad (35)$$

where the last equality comes from the assumption $\delta < 1/3$. So (33) and (34) are true. ◀

Due to the space limit, we place other parts to the full version of our paper [27]. We propose the sub-linear time bi-criteria approximation algorithms for three different MEX with outlier problems: k -center clustering, flat fitting, and SVM with outliers. All of these problems have important applications in real world, and our results significantly reduce their time complexities comparing with existing approaches.

References

- 1 Pankaj K. Agarwal, Esther Ezra, and Kyle Fox. Geometric optimization revisited. In Bernhard Steffen and Gerhard J. Woeginger, editors, *Computing and Software Science - State of the Art and Perspectives*, volume 10000 of *Lecture Notes in Computer Science*, pages 66–84. Springer, 2019.
- 2 Pankaj K. Agarwal, Sariel Har-Peled, and Kasturi R. Varadarajan. Geometric approximation via coresets. *Combinatorial and Computational Geometry*, 52:1–30, 2005.
- 3 Pankaj K. Agarwal, Sariel Har-Peled, and Hai Yu. Robust shape fitting via peeling and grating coresets. *Discrete & Computational Geometry*, 39(1-3):38–58, 2008.
- 4 Pankaj K. Agarwal and R. Sharathkumar. Streaming algorithms for extent problems in high dimensions. *Algorithmica*, 72(1):83–98, 2015.
- 5 Alok Aggarwal, Hiroshi Imai, Naoki Katoh, and Subhash Suri. Finding k points with minimum diameter and related problems. *Journal of algorithms*, 12(1):38–56, 1991.
- 6 Zeyuan Allen Zhu, Zhenyu Liao, and Yang Yuan. Optimization algorithms for faster computational geometry. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, pages 53:1–53:6, 2016.
- 7 Noga Alon, Seannie Dar, Michal Parnas, and Dana Ron. Testing of clustering. *SIAM Journal on Discrete Mathematics*, 16(3):393–417, 2003.
- 8 Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning attacks against support vector machines. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*, 2012.
- 9 Battista Biggio and Fabio Roli. Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition*, 84:317–331, 2018.
- 10 Manuel Blum, Robert W. Floyd, Vaughan Pratt, Ronald L. Rivest, and Robert E. Tarjan. Time bounds for selection. *Journal of Computer and System Sciences*, 7(4):448–461, 1973.
- 11 Mihai Bădoiu and Kenneth L. Clarkson. Smaller core-sets for balls. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 801–802, 2003.
- 12 Mihai Bădoiu and Kenneth L. Clarkson. Optimal core-sets for balls. *Computational Geometry*, 40(1):14–22, 2008.
- 13 Mihai Bădoiu, Sariel Har-Peled, and Piotr Indyk. Approximate clustering via core-sets. In *Proceedings of the ACM Symposium on Theory of Computing (STOC)*, pages 250–257, 2002.
- 14 Matteo Ceccarello, Andrea Pietracaprina, and Geppino Pucci. Solving k -center clustering (with outliers) in mapreduce and streaming, almost as accurately as sequentially. *PVLDB*, 12(7):766–778, 2019.
- 15 Deeparnab Chakrabarty, Prachi Goyal, and Ravishankar Krishnaswamy. The non-uniform k -center problem. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, pages 67:1–67:15, 2016.
- 16 Timothy M. Chan and Vinayak Pathak. Streaming and dynamic algorithms for minimum enclosing balls in high dimensions. *Comput. Geom.*, 47(2):240–247, 2014.
- 17 Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM TIST*, 2(3), 2011.
- 18 Moses Charikar, Samir Khuller, David M Mount, and Giri Narasimhan. Algorithms for facility location problems with outliers. In *Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms*, pages 642–651. Society for Industrial and Applied Mathematics, 2001.

- 19 Moses Charikar, Liadan O’Callaghan, and Rina Panigrahy. Better streaming algorithms for clustering problems. In *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*, pages 30–39. ACM, 2003.
- 20 Kenneth L. Clarkson. Coresets, sparse greedy approximation, and the frank-wolfe algorithm. *ACM Transactions on Algorithms*, 6(4):63, 2010.
- 21 Kenneth L. Clarkson, Elad Hazan, and David P. Woodruff. Sublinear optimization for machine learning. *J. ACM*, 59(5):23:1–23:49, 2012.
- 22 Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20:273, 1995.
- 23 David J. Crisp and Christopher J. C. Burges. A geometric interpretation of v-SVM classifiers. In Sara A. Solla, Todd K. Leen, and Klaus-Robert Müller, editors, *NIPS*, pages 244–250. The MIT Press, 1999.
- 24 Artur Czumaj and Christian Sohler. Sublinear-time approximation for clustering via random sampling. In *International Colloquium on Automata, Languages, and Programming*, pages 396–407. Springer, 2004.
- 25 Artur Czumaj and Christian Sohler. Sublinear-time algorithms. In *Property Testing - Current Research and Surveys*, pages 41–64, 2010.
- 26 Hu Ding. Minimum enclosing ball revisited: Stability and sub-linear time algorithms. *CoRR*, abs/1904.03796, 2019.
- 27 Hu Ding. A sub-linear time framework for geometric optimization with outliers in high dimensions. *CoRR*, abs/2004.10090, 2020. [arXiv:2004.10090](https://arxiv.org/abs/2004.10090).
- 28 Hu Ding and Jinhui Xu. Random gradient descent tree: A combinatorial approach for svm with outliers. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 2561–2567, 2015.
- 29 Hu Ding, Haikuo Yu, and Zixiu Wang. Greedy strategy works for k-center clustering with outliers and coreset construction. In *27th Annual European Symposium on Algorithms, ESA 2019, September 9-11, 2019, Munich/Garching, Germany.*, pages 40:1–40:16, 2019.
- 30 Alon Efrat, Micha Sharir, and Alon Ziv. Computing the smallest k-enclosing circle and related problems. *Computational Geometry*, 4(3):119–136, 1994.
- 31 Dan Feldman, Chongyuan Xiang, Ruihao Zhu, and Daniela Rus. Coresets for differentially private k-means clustering and applications to privacy in mobile sensor networks. In *Proceedings of the 16th ACM/IEEE International Conference on Information Processing in Sensor Networks, IPSN 2017, Pittsburgh, PA, USA, April 18-21, 2017*, pages 3–15, 2017.
- 32 Kaspar Fischer, Bernd Gärtner, and Martin Kutz. Fast smallest-enclosing-ball computation in high dimensions. In *Algorithms - ESA 2003, 11th Annual European Symposium, Budapest, Hungary, September 16-19, 2003, Proceedings*, pages 630–641, 2003.
- 33 Marguerite Frank and Philip Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3(1-2):95–110, 1956.
- 34 Oded Goldreich, Shafi Goldwasser, and Dana Ron. Property testing and its connection to learning and approximation. *J. ACM*, 45(4):653–750, 1998.
- 35 Teofilo F Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science*, 38:293–306, 1985.
- 36 Ian J. Goodfellow, Patrick D. McDaniel, and Nicolas Papernot. Making machine learning robust against adversarial inputs. *Commun. ACM*, 61(7):56–66, 2018.
- 37 Sudipto Guha, Yi Li, and Qin Zhang. Distributed partial clustering. In *Proceedings of the 29th ACM Symposium on Parallelism in Algorithms and Architectures*, pages 143–152. ACM, 2017.
- 38 Laszlo Gyongyosi and Sandor Imre. Geometrical analysis of physically allowed quantum cloning transformations for quantum cryptography. *Information Sciences*, 285:1–23, 2014.
- 39 Sariel Har-Peled and Soham Mazumdar. Fast algorithms for computing the smallest k-enclosing circle. *Algorithmica*, 41(3):147–157, 2005.

- 40 Sarel Har-Peled, Dan Roth, and Dav Zimak. Maximum margin coresets for active and noise tolerant learning. In *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007*, pages 836–841, 2007.
- 41 Sarel Har-Peled and Kasturi R. Varadarajan. Approximate shape fitting via linearization. In *42nd Annual Symposium on Foundations of Computer Science, FOCS 2001, 14-17 October 2001, Las Vegas, Nevada, USA*, pages 66–73, 2001.
- 42 Sarel Har-Peled and Kasturi R. Varadarajan. High-dimensional shape fitting in linear time. *Discret. Comput. Geom.*, 32(2):269–288, 2004.
- 43 Sarel Har-Peled and Yusu Wang. Shape fitting with outliers. *SIAM Journal on Computing*, 33(2):269–285, 2004.
- 44 Elad Hazan, Tomer Koren, and Nati Srebro. Beating SGD: learning svms in sublinear time. In John Shawe-Taylor, Richard S. Zemel, Peter L. Bartlett, Fernando C. N. Pereira, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain*, pages 1233–1241, 2011.
- 45 Ling Huang, Anthony D Joseph, Blaine Nelson, Benjamin IP Rubinstein, and J Doug Tygar. Adversarial machine learning. In *Proceedings of the 4th ACM workshop on Security and artificial intelligence*, pages 43–58, 2011.
- 46 Lingxiao Huang, Shaofeng Jiang, Jian Li, and Xuan Wu. Epsilon-coresets for clustering (with outliers) in doubling metrics. In *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 814–825, 2018.
- 47 Piotr Indyk. Sublinear time algorithms for metric space problems. In *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing, May 1-4, 1999, Atlanta, Georgia, USA*, pages 428–434, 1999.
- 48 Matthew Jagielski, Alina Oprea, Battista Biggio, Chang Liu, Cristina Nita-Rotaru, and Bo Li. Manipulating machine learning: Poisoning attacks and countermeasures for regression learning. In *2018 IEEE Symposium on Security and Privacy, SP 2018, Proceedings, 21-23 May 2018, San Francisco, California, USA*, pages 19–35, 2018.
- 49 Michael Kerber and R. Sharathkumar. Approximate \check{c} ech complex in low and high dimensions. In *Algorithms and Computation - 24th International Symposium, ISAAC 2013, Hong Kong, China, December 16-18, 2013, Proceedings*, pages 666–676, 2013.
- 50 Piyush Kumar, Joseph S. B. Mitchell, and E. Alper Yildirim. Approximate minimum enclosing balls in high dimensions using core-sets. *ACM Journal of Experimental Algorithmics*, 8, 2003.
- 51 Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *arXiv preprint*, 2016. [arXiv:1611.01236](https://arxiv.org/abs/1611.01236).
- 52 Shi Li and Xiangyu Guo. Distributed k -clustering for data with heavy noise. In *Advances in Neural Information Processing Systems*, pages 7849–7857, 2018.
- 53 Gustavo Malkomes, Matt J Kusner, Wenlin Chen, Kilian Q Weinberger, and Benjamin Moseley. Fast distributed k -center clustering with outliers on massive data. In *Advances in Neural Information Processing Systems*, pages 1063–1071, 2015.
- 54 Jiří Matoušek. On enclosing k points by a circle. *Information Processing Letters*, 53(4):217–221, 1995.
- 55 Richard Matthew McCutchen and Samir Khuller. Streaming algorithms for k -center clustering with outliers and with anonymity. In *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques*, pages 165–178. Springer, 2008.
- 56 Adam Meyerson, Liadan O’Callaghan, and Serge A. Plotkin. A k -median algorithm with running time independent of data size. *Machine Learning*, 56(1-3):61–87, 2004.
- 57 Nina Mishra, Dan Oblinger, and Leonard Pitt. Sublinear time approximate clustering. In *Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms*, pages 439–447. Society for Industrial and Applied Mathematics, 2001.
- 58 David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Ruth Silverman, and Angela Y. Wu. On the least trimmed squares estimator. *Algorithmica*, 69(1):148–183, 2014.

- 59 Kobbi Nissim, Uri Stemmer, and Salil P. Vadhan. Locating a small cluster privately. In *Proceedings of the 35th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS 2016, San Francisco, CA, USA, June 26 - July 01, 2016*, pages 413–427, 2016.
- 60 Rina Panigrahy. Minimum enclosing polytope in high dimensions. *arXiv preprint*, 2004. [arXiv:cs/0407020](https://arxiv.org/abs/cs/0407020).
- 61 Jeff M. Phillips. Coresets and sketches. *Computing Research Repository*, 2016.
- 62 J. Platt. Fast training of support vector machines using sequential minimal optimization. In B. Schölkopf, C. J. C. Burges, and A. J. Smola, editors, *Advances in Kernel Methods — Support Vector Learning*, pages 185–208, Cambridge, MA, 1999. MIT Press.
- 63 Ronitt Rubinfeld. Sublinear time algorithms. In *Survey*. Citeseer, 2006.
- 64 Ankan Saha, S. V. N. Vishwanathan, and Xinhua Zhang. New approximation algorithms for minimum enclosing convex shapes. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011, San Francisco, California, USA, January 23-25, 2011*, pages 1146–1160, 2011.
- 65 B. Scholkopf, A. J. Smola, K. R. Muller, and P. L. Bartlett. New support vector algorithms. *Neural Computation*, 12:1207–1245, 2000.
- 66 Shinya Suzumura, Kohei Ogawa, Masashi Sugiyama, and Ichiro Takeuchi. Outlier path: A homotopy algorithm for robust svm. In Tony Jebara and Eric P. Xing, editors, *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1098–1106, 2014.
- 67 Pang-Ning Tan, Michael S. Steinbach, and Vipin Kumar. *Introduction to Data Mining*. Addison-Wesley, 2005.
- 68 Ivor W. Tsang, James T. Kwok, and Pak-Ming Cheung. Core vector machines: Fast SVM training on very large data sets. *Journal of Machine Learning Research*, 6:363–392, 2005. URL: <http://jmlr.org/papers/v6/tsang05a.html>.
- 69 Vladimir N Vapnik and A Ya Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. In *Measures of complexity*, pages 11–30. Springer, 2015.
- 70 Linli Xu, Koby Crammer, and Dale Schuurmans. Robust support vector machine training via convex outlier ablation. In *AAAI*, pages 536–542. AAAI Press, 2006.
- 71 Hai Yu, Pankaj K. Agarwal, Raghunath Poredy, and Kasturi R. Varadarajan. Practical methods for shape fitting and kinetic data structures using coresets. *Algorithmica*, 52(3):378–402, 2008.
- 72 Hamid Zarrabi-Zadeh and Asish Mukhopadhyay. Streaming 1-center with outliers in high dimensions. In *Proceedings of the Canadian Conference on Computational Geometry (CCCG)*, pages 83–86, 2009.

A Appendix

A.1 The Lower Bound of Sample Size for MEB with Outliers

Actually, it is easy to obtain a sub-linear time randomized 2-approximation algorithm for MEB with outliers (if only returning the center): one can randomly select one point $p \in P$, and it belongs P_{opt} with probability $1 - \gamma$; thus, with probability $1 - \gamma$, the point p yields a 2-approximation by using the triangle inequality. Below, we consider an example and show that it is impossible to achieve a single-criterion λ -approximation for any $\lambda < 2$, if the time complexity is required to be independent of n .

Let q_a , q_b , and q_c be three colinear points in \mathbb{R}^d , where $\|q_a - q_b\| = x$ and $\|q_b - q_c\| = y$ (see Figure 3). Let $\gamma \in (0, 1)$ and the point set $P = P_a \cup P_b \cup P_c$, where P_a contains a single point located at q_a , P_b contains $(1 - \gamma)n - 1$ points overlapping at q_b , and P_c contains γn points overlapping at q_c . Consider the instance (P, γ) for the problem of MEB with outliers. Suppose $x \ll y$. Consequently, the optimal subset P_{opt} should be $P_a \cup P_b$ and the optimal radius is $x/2$.



■ **Figure 3** $\|q_a - q_b\| = x$ and $\|q_b - q_c\| = y$.

If we take a random sample S of size $m = o(n)$ from P , with high probability, $P_a \cap S = \emptyset$ (even if we repeat our sampling a constant number of times, the single point P_a will still be missing with high probability). Therefore, S only contains the points from P_b and P_c . If we run an existing algorithm on S , the resulting ball center, say o , should always be inside the convex hull of S , i.e., the segment $\overline{q_b q_c}$. Let $\mathbb{B}(o, r)$ be the ball covering $(1 - \gamma)n$ points of P . We consider two cases: (1) $q_a \in \mathbb{B}(o, r)$ and (2) $q_a \notin \mathbb{B}(o, r)$. For case (1), since $o \in \overline{q_b q_c}$, it is easy to know that the radius $r \geq \|q_a - q_b\| = x$ and therefore the approximation ratio is at least 2. For case (2), since $|P_b| = (1 - \gamma)n - 1$, $\mathbb{B}(o, r)$ must cover some point from P_c and therefore $r = y/2$; because $x \ll y$, the approximation ratio is also larger than 2.

A.2 Proof of Theorem 4

Similar to the analysis in [11], we let $\lambda_i = \frac{r_i}{(1+\epsilon)\text{Rad}(P)}$. Because r_i is the radius of $\text{MEB}(T)$ and $T \subset P$, we know $r_i \leq \text{Rad}(P)$ and then $\lambda_i \leq 1/(1 + \epsilon)$. By simple calculation, we know that when $L_i = \frac{((1+\epsilon)\text{Rad}(P) - \xi r_i)^2 - r_i^2}{2((1+\epsilon)\text{Rad}(P) - \xi r_i)}$ the lower bound of r_{i+1} in (3) achieves the minimum value. Plugging this value of L_i into (3), we have

$$\lambda_{i+1}^2 \geq \lambda_i^2 + \frac{((1 - \xi \lambda_i)^2 - \lambda_i^2)^2}{4(1 - \xi \lambda_i)^2}. \quad (36)$$

To simplify inequality (36), we consider the function $g(x) = \frac{(1-x)^2 - \lambda_i^2}{1-x}$, where $0 < x < \xi$. Its derivative $g'(x) = -1 - \frac{\lambda_i^2}{(1-x)^2}$ is always negative, thus we have

$$g(x) \geq g(\xi) = \frac{(1 - \xi)^2 - \lambda_i^2}{1 - \xi}. \quad (37)$$

Because $\xi < \frac{\epsilon}{1+\epsilon}$ and $\lambda_i \leq 1/(1 + \epsilon)$, we know that the right-hand side of (37) is always non-negative. Using (37), the inequality (36) can be simplified to

$$\begin{aligned} \lambda_{i+1}^2 &\geq \lambda_i^2 + \frac{1}{4}(g(\xi))^2 \\ &= \lambda_i^2 + \frac{((1 - \xi)^2 - \lambda_i^2)^2}{4(1 - \xi)^2}. \end{aligned} \quad (38)$$

(38) can be further rewritten as

$$\begin{aligned} \left(\frac{\lambda_{i+1}}{1 - \xi}\right)^2 &\geq \frac{1}{4}\left(1 + \left(\frac{\lambda_i}{1 - \xi}\right)^2\right)^2 \\ \implies \frac{\lambda_{i+1}}{1 - \xi} &\geq \frac{1}{2}\left(1 + \left(\frac{\lambda_i}{1 - \xi}\right)^2\right). \end{aligned} \quad (39)$$

Now, we can apply a similar transformation of λ_i which was used in [11]. Let $\gamma_i = \frac{1}{1 - \frac{\lambda_i}{1 - \xi}}$.

We know $\gamma_i > 1$ (note $0 \leq \lambda_i \leq \frac{1}{1+\epsilon}$ and $\xi < \frac{\epsilon}{1+\epsilon}$). Then, (39) implies that

$$\begin{aligned} \gamma_{i+1} &\geq \frac{\gamma_i}{1 - \frac{1}{2\gamma_i}} \\ &= \gamma_i \left(1 + \frac{1}{2\gamma_i} + \left(\frac{1}{2\gamma_i}\right)^2 + \dots \right) \\ &> \gamma_i + \frac{1}{2}, \end{aligned} \tag{40}$$

where the equation comes from the fact that $\gamma_i > 1$ and thus $\frac{1}{2\gamma_i} \in (0, \frac{1}{2})$. Note that $\lambda_0 = 0$ and thus $\gamma_0 = 1$. As a consequence, we have $\gamma_i > 1 + \frac{i}{2}$. In addition, since $\lambda_i \leq \frac{1}{1+\epsilon}$, that is, $\gamma_i \leq \frac{1}{1 - \frac{1}{(1+\epsilon)(1-\xi)}}$, we have

$$i < \frac{2}{\epsilon - \xi - \epsilon\xi} = \frac{2}{\left(1 - \frac{1+\epsilon}{\epsilon}\xi\right)\epsilon}. \tag{41}$$

Consequently, we obtain the theorem.