Is Simple Uniform Sampling Efficient for Center-Based Clustering With Outliers: When and Why?

Hu Ding
School of Computer Science and Engineering, University of Science and Technology of China
He Fei, China
huding@ustc.edu.cn

Jiawei Huang
School of Computer Science and Engineering, University of Science and Technology of China
He Fei, China
hjw0330@mail.ustc.edu.cn

ABSTRACT

Clustering has many important applications in computer science, but real-world datasets often contain outliers. The presence of outliers can make the clustering problems to be much more challenging. In this paper, we propose a framework for solving three representative center-based clustering with outliers problems: k-center/median/means clustering with outliers. The framework actually is very simple, where we just need to take a small uniform sample from the input and run an existing approximation algorithm on the sample. However, our analysis is fundamentally different from the previous (uniform and non-uniform) sampling based ideas. To explain the effectiveness of uniform sampling in theory, we introduce a "significance" criterion and prove that the performance of our framework depends on the significance degree of the given instance. In particular, the sample size can be independent of the input data size n and the dimensionality d, if we assume the given instance is sufficiently "significant", which is in fact a fairly appropriate assumption in practice. Due to its simplicity, the uniform sampling approach also enjoys several significant advantages over the non-uniform sampling approaches. The experiments suggest that our framework can achieve comparable clustering results with existing methods, but is much easier to implement and can greatly reduce the running times. To the best of our knowledge, this is the first work that systematically studies the effectiveness of uniform sampling from both theoretical and experimental aspects.

PVLDB Reference Format:
Hu Ding and Jiawei Huang. Is Simple Uniform Sampling Efficient for Center-Based Clustering With Outliers: When and Why?. PVLDB, 14(1): XXX-XXX, 2020.
doi:XX.XX/XXX.XX

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at http://vldb.org/pvldb/format_vol14.html.

1 INTRODUCTION

Clustering has many important applications in real world [36]. An important type of clustering problems is called "center-based clustering", such as the well-known *k-center/median/means* clustering problems [4]. In general, a center-based clustering problem aims to find k cluster centers so as to minimize the induced clustering cost. For example, the k-center clustering problem is to minimize the maximum distance from the input data to the set of cluster centers [29, 33]; the k-median (means) clustering problem is to minimize the average (squared) distance [40, 41].

But in practice our datasets often contain the outliers which could seriously destroy the final clustering results [11]. A key obstacle is that the outliers can be arbitrarily located in the space. *Outlier removal* is a topic that in fact has been extensively studied before [11]. The existing outlier removal methods (like DBSCAN [26]) yet cannot simply replace the center-based clustering with outliers approaches, since we often need to use the obtained cluster centers to represent the data or perform other tasks for data mining and machine learning (*e.g.,* data compression and data selection in large-scale machine learning [3, 17, 52]).

Clustering with outliers can be viewed as a generalization of the vanilla clustering problems, however, the presence of arbitrary outliers makes the problems to be much more challenging. In particular, it may result in a potentially challenging **combinatorial optimization problem**: suppose z is the number of outliers and n is the total number of input data items ($z < n$); to optimize the clustering cost, at first glance we should consider an exponentially large number $\binom{n}{z}$ of different possible cases. Therefore, even for their approximate solutions, most existing quality-guaranteed algorithms have super-linear time complexities. So a number of sampling methods have been proposed for reducing their complexities (a detailed introduction on prior work is shown in Section 1.2). Namely, we take a small sample (uniformly or non-uniformly) from the input and run an existing approximation algorithm on the sample.

A number of **non-uniform sampling methods** have been proposed, such as the *k-means++* seeding [5, 22, 32, 35] and the successive sampling method [15]. However, these non-uniform sampling approaches may suffer several drawbacks in practice. For instance, they need to read the input dataset in multiple passes with high computational complexities, and/or may have to discard more than z outliers (if z is the pre-specified number of outliers). For example, the methods [32, 35] need to run the k-means++ seeding procedure $\Omega(k+z)$ rounds, and discard $O(kz \log n)$ and $O(kz)$ outliers respectively for achieving their theoretical quality guarantees.

Comparing with non-uniform sampling methods, **uniform sampling** enjoys several significant advantages, *e.g.,* it is very easy to implement in practice. We need to emphasize that, beyond clustering quality and complexity, **the simplicity of implementation is also a major concern in *algorithm engineering*, especially**

Proceedings of the VLDB Endowment, Vol. 14, No. 1 ISSN 2150-8097.
doi:XX.XX/XXX.XX

for dealing with big data [51]. However, the analysis on uniform sampling usually suffers the following dilemma. Let S be the uniform sample from the input. To guarantee a low clustering error, the sample size $|S|$ should be at least larger than some threshold depending on n/z and the dimensionality d, which could be very large (*e.g.,* d could be high and z could be much smaller than n). For example, as the result proposed in [34], $|S|$ should be at least $O\big((\frac{n}{\delta z})^2 kd \log k\big)$, if $\delta \in (0,1)$ and we allow to discard $(1+\delta)z$ outliers (which is slightly larger than the pre-specified number z); so if $z = \sqrt{n}$, the sample size will be even larger than n. Also, though we can apply the techniques like *JL-transform* [21] to reduce the dimensionality, the obtained new dimension $\Theta(\frac{\log n}{\epsilon^2})$ can be still large if the error parameter ϵ is required to be small; moreover, the JL-transform itself may also take a large running time if the data size is extremely large.

Though the uniform sampling method suffers the above dilemma in theory, it often achieves nice performance in practice even if the sample size is much smaller than the theoretical bounds. The main reason is that the given bounds are usually overly conservative for most practical instances, since they are obtained based on the worst-case analysis. So, from both theoretical and experimental aspects, a natural question is:

whether we can reduce the overly conservative sample size bound for a uniform sampling approach, under some realistic assumption.

Actually, for the vanilla k-median/means clustering (without outliers) problems, Meyerson *et al.* [46, Section 4 and 5] considered using uniform sampling to solve the practical case that each optimal cluster is assumed to have size $\Omega(\frac{\epsilon n}{k})$, where ϵ is a fixed parameter in $(0,1)$. But unfortunately their result cannot be easily extended to the version with outliers (as studied in the same paper [46, Section 6]). The current solution of [46, Section 6], has not only the sample size depending on $\frac{n}{z}$, but also a large error on the number of discarded outliers (it needs to discard $> 16z$ outliers).

1.1 Our Contributions

In this paper, we aim to discover that what factor indeed affects the performance of uniform sampling; in particular, we need to explain our discovery in theory, and validate it through the experiments. Our framework actually is very simple, where we just need to take a small sample uniformly at random from the input and run an existing approximation algorithm (as the black-box algorithm) on the sample. Our results are partly inspired by the work from [46], but as discussed above, we need to go deeper and develop significantly new ideas for analyzing the influence of outliers. We briefly introduce our ideas and results below.

Suppose $\{C_1^*, C_2^*, \cdots, C_k^*\}$ are the k optimal clusters. In our framework, we consider the ratio between two values: the size of the smallest optimal cluster $\min_{1\leq j\leq k} |C_j^*|$[1] and the number of outliers z. We show this ratio actually plays a key role to the effectiveness of uniform sampling. If a cluster C_j^* has size $\ll z$, then we can say that C_j^* is not a "significant" cluster. In real applications, it is often reasonable to assume that each cluster C_j^* has a size at least comparable to z, *i.e.,* $\min_{1\leq j\leq k} |C_j^*|/z = \Omega(1)$ (note this assumption allows the ratio to be a value smaller than 1, say 0.5). A cluster usually represents a certain scale of population and it is rare to have a cluster with the size much smaller than the number of outliers (except for some special scenario that we may be particularly interested in tiny clusters). Under such a realistic assumption, our framework can output $k + O(\log k)$ cluster centers that yield a 4-approximation for k-center clustering with outliers. In some scenarios, we may insist on returning exactly k cluster centers. We prove that, if $\frac{\min_{1\leq j\leq k} |C_j^*|}{z} > 1$, our framework can return exactly k cluster centers that yield a $(c+2)$-approximate solution, through running an existing c-approximation k-center with outliers algorithm (with some $c \geq 1$) on the sample. The framework also yields similar results for k-median/means clustering with outliers.

We prove that the sample size can be independent of the ratio n/z and the dimensionality d, which is fundamentally different from the previous results on uniform sampling [13, 24, 34]. Instead, the sample size only depends on k, $\min_{1\leq j\leq k} |C_j^*|$, and several parameters controlling the clustering error and success probability. If we only require to output the cluster centers, our uniform sampling approach has the sublinear time complexity that is independent of the input size[2]. Moreover, different from the previous methods which often have the errors on the number of discarded outliers, our method allows to discard exactly z outliers.

Finally, we test the performance of our framework on both synthetic and real-world datasets. Though uniform sampling was often used as a baseline method in the experiments (*e.g.,* [15, 35]), to the best of our knowledge, we are the first to conduct a systematical experiment to study its effectiveness from various aspects, such as clustering quality, running time, scalability, *etc.*

Remark 1. It is worth noting that Gupta [31] proposed a similar uniform sampling approach to solve k-means clustering with outliers. However, the analysis and results of [31] are quite different from ours. Also the assumption in [31] is stronger: it requires that each optimal cluster has size roughly $\Omega(\frac{z}{\gamma^2}\log k)$ (*i.e.,* $\frac{\min_{1\leq j\leq k} |C_j^*|}{z} = \Omega(\frac{\log k}{\gamma^2})$) where γ is a small parameter in $(0,1)$.

Remark 2. Many optimization problems (such as max-cut [6], clustering [48], and learning mixture of Gaussian distributions [20]) have shown to be challenging even for approximation, but admit efficient solutions in practice. Recent works tried to explain this phenomenon and proposed the "beyond worst-case analysis" methods under some realistic assumptions [50]. For example, Ostrovsky *et al.* [48] proposed a separation condition for k-means clustering which refers to the scenario where the clustering cost of k-means is significantly lower than that of $(k-1)$-means for a given instance, and demonstrated the effectiveness of the Lloyd heuristic [41] under the separation condition. Our study on the "significant" instances actually also falls under the umbrella of beyond worst-case analysis.

1.2 Related Work

We overview the existing center-based clustering with outliers algorithms below.

k-center clustering with outliers. Charikar *et al.* [12] proposed a 3-approximation algorithm for k-center clustering with

[1] $|C_j^*|$ indicates the number of points in C_j^*.

[2] Obviously, if we require to output the clustering assignment for each point, it needs at least linear time.

outliers in arbitrary metrics. The time complexity of their algorithm is at least quadratic in data size, since it needs to read all the pairwise distances. A following streaming $(4+\epsilon)$-approximation algorithm was proposed by McCutchen and Khuller [44]. Chakrabarty *et al.* [10] showed a 2-approximation algorithm for metric k-center clustering with outliers based on the LP relaxation techniques. Bădoiu *et al.* [7] showed a coreset based approach but having an exponential time complexity if k is not a constant. Recently, Ding *et al.* [24] provided a greedy algorithm that yields a bi-criteria approximation (returning more than k clusters) based on Gonzalez's idea [29]; independently, Bhaskara *et al.* [5] also proposed a similar greedy bi-criteria approximation algorithm. Both their results yield 2 (or $2+\epsilon$)-approximation for the radius and need to return $O(\frac{k}{\epsilon})$ cluster centers, if allowing to discard $(1+\epsilon)z$ outliers. The non-uniform and uniform sampling methods for k-center clustering with outliers were studied in [9, 13, 24, 34].

k-median/means clustering with outliers. Actually, the early work of k-means clustering with outliers dates back to 1990s, in which Cuesta-Albertos *et al.* [18] used the "trimming" idea to formulate the robust model for k-means clustering. There are a bunch of algorithms with provable guarantees that have been proposed for k-means/median clustering with outliers [12, 16, 28, 38], but they are difficult to implement due to their high complexities. Therefore, several heuristic but practical algorithms without provable guarantee have been studied, such as [14, 49].

Moreover, several sampling based methods were proposed to speed up the existing algorithms. By using the local search method, Gupta *et al.*[32] provided a 274-approximation algorithm for k-means clustering with outliers; they showed that the well known k-means++ method [2] can be used to reduce the complexity. Furthermore, Bhaskara *et al.* [5] and Deshpande *et al.* [22] respectively showed that the quality can be improved by modifying the k-means++ seeding. Partly inspired by the successive sampling method of [45], Chen *et al.* [15] proposed a novel summary construction algorithm to reduce input data size for the k-median/means clustering with outliers problems. Very recently, Im *et al.*[35] provided a sampling method for k-means clustering with outliers by combining k-means++ and uniform sampling. Charikar *et al.*[13] and Meyerson *et al.* [46] provided different uniform sampling approaches respectively; Huang *et al.* [34] and Ding and Wang [**?**] also presented the uniform sampling approaches in Euclidean space.

Furthermore, due to the rapid increase of data volumes in real world, a number of communication efficient distributed algorithms for k-center clustering with outliers [9, 30, 39, 42] and k-median /means clustering with outliers [15, 30, 39] were also proposed.

1.3 Preliminaries

In this paper, we follow the common definition for "uniform sampling" in most previous articles, that is, **we take a sample from the input independently and uniformly at random**.

We use $||\cdot||$ to denote the Euclidean norm. Let the input be a point set $P\subset\mathbb{R}^d$ with $|P|=n$. Given a set of points $H\subset\mathbb{R}^d$ and a positive integer $z<n$, we let $\texttt{dist}(p,H):=\min_{q\in H}||p-q||$ for any point $p\in\mathbb{R}^d$, and define the following notations:

$$\Delta_{\infty}^{-z}(P,H) = \min\{\max_{p\in P'}\texttt{dist}(p,H)\,|\,P'\subset P, |P'|=n-z\}; \quad (1)$$

$$\Delta_{1}^{-z}(P,H) = \min\{\frac{1}{|P'|}\sum_{p\in P'}\texttt{dist}(p,H)\,|\,P'\subset P, |P'|=n-z\}; \quad (2)$$

$$\Delta_{2}^{-z}(P,H) = \min\{\frac{1}{|P'|}\sum_{p\in P'}\left(\texttt{dist}(p,H)\right)^2\,|\,P'\subset P, |P'|=n-z\}. \quad (3)$$

The following definition follows the previous articles (mentioned in Section 1.2) on center-based clustering with outliers. As emphasized earlier, we suppose that the outliers can be arbitrarily located in the space, and thus do not impose any restriction on them.

Definition 1 (**k-Center/Median/Means Clustering with Outliers).** Given a set P of n points in $\mathbb{R}^d$ with two positive integers $k<n$ and $z<n$, the problem of k-center (resp., k-median, k-means) clustering with outliers is to find k cluster centers $C=\{c_1,\cdots,c_k\}\subset\mathbb{R}^d$, such that the objective function $\Delta_{\infty}^{-z}(P,C)$ (resp., $\Delta_{1}^{-z}(P,C)$, $\Delta_{2}^{-z}(P,C)$) is minimized.

Remark 3. Definition 1 can be simply modified for arbitrary metric space (X,μ), where X contains n vertices and $\mu(\cdot,\cdot)$ is the distance function: the Euclidean distance "$||p-q||$" is replaced by $\mu(p,q)$; the cluster centers $\{c_1,\cdots,c_k\}$ should be chosen from X.

In this paper, we always use P_{opt}, a subset of P with size $n-z$, to denote the subset yielding the optimal solution with respect to the objective functions in Definition 1. Also, let $\{C_1^*,\cdots,C_k^*\}$ be the k optimal clusters forming P_{opt}. As mentioned in Section 1.1, it is rational to assume that $\min_{1\le j\le k}|C_j^*|$ is at least comparable to z. To formally state this assumption, we introduce the following definition.

Definition 2 (**(ϵ_1,ϵ_2)-Significant Instance).** Let $\epsilon_1,\epsilon_2>0$. Given an instance of k-center (resp., k-median, k-means) clustering with outliers as described in Definition 1, if $\min_{1\le j\le k}|C_j^*|\ge\frac{\epsilon_1}{k}n$ and $z=\frac{\epsilon_2}{k}n$, we say that it is an (ϵ_1,ϵ_2)-significant instance.

In Definition 2, we do not say "$\min_{1\le j\le k}|C_j^*|=\frac{\epsilon_1}{k}n$", since we may not be able to obtain the exact value of $\min_{1\le j\le k}|C_j^*|$. Similar to the assumption from [46], we may only have an estimation of $\min_{1\le j\le k}|C_j^*|$, and therefore use "$\min_{1\le j\le k}|C_j^*|\ge\frac{\epsilon_1}{k}n$" instead. The ratio $\frac{\epsilon_1}{\epsilon_2}$, which is no larger than $\frac{\min_{1\le j\le k}|C_j^*|}{z}$, reveals the "significance" degree of the clusters to outliers. Specifically, the higher the ratio is, the more significant to outliers the clusters will be.

The rest of the paper is organized as follows. To help our analysis, we present two implications of Definition 2 in Section 2. Then we introduce our uniform sampling algorithms for k-center clustering with outliers and k-median/means clustering with outliers in Section 3 and Section 4, respectively. Finally, we discuss the implementation details and present our experimental results in Section 5.

2 IMPLICATIONS OF SIGNIFICANT INSTANCE

We introduce the following two important implications of Definition 2.

Lemma 1. Given an (ϵ_1,ϵ_2)-significant instance P as described in Definition 2, we select a set S of points from P uniformly at random. Let $\eta,\delta\in(0,1)$, and then we have

- **(i)** if $|S| \geq \frac{k}{\epsilon_1} \log \frac{k}{\eta}$, with probability at least $1-\eta$, $S \cap C^*_j \neq \emptyset$ for any $1 \leq j \leq k$;
- **(ii)** if $|S| \geq \frac{3k}{\delta^2 \epsilon_1} \log \frac{2k}{\eta}$, with probability at least $1-\eta$, $|S \cap C^*_j| \in (1 \pm \delta) \frac{|C^*_j|}{n} |S|$ for any $1 \leq j \leq k$.

Lemma 1 can be obtained by using the Chernoff bound [1], and we leave the proof to Section A. Also, we know that the expected number of outliers contained in the sample S is $\frac{\epsilon_2}{k}|S|$. So we immediately have the following result by using the Markov's inequality.

Lemma 2. Given an (ϵ_1, ϵ_2)-significant instance P as described in Definition 2, we select a set S of points from P uniformly at random. Let $\eta \in (0,1)$. Then, with probability at least $1-\eta$, $\left|S \setminus P_{\text{opt}}\right| \leq \frac{\epsilon_2}{k\eta}|S|$.

***Remark* 4.** (**The union bound of Lemma 1 and Lemma 2**) It is worth noting that the event (i) (or event (ii)) described in Lemma 1 and the event described in Lemma 2 are not completely independent. For example, if the event (i) of Lemma 1 occurs, then $\left|S \setminus P_{\text{opt}}\right|$ should be at most $\frac{\epsilon_2}{k\eta}(|S|-k)$ instead of $\frac{\epsilon_2}{k\eta}|S|$ in Lemma 2. But since $\frac{\epsilon_2}{k\eta}|S| > \frac{\epsilon_2}{k\eta}(|S|-k)$, we can still say "$\left|S \setminus P_{\text{opt}}\right| \leq \frac{\epsilon_2}{k\eta}|S|$" with probability at least $1-\eta$. So we can safely claim that with probability at least $(1-\eta)^2$, the events of Lemma 1 and Lemma 2 both occur.

3 k-CENTER CLUSTERING WITH OUTLIERS

In this section, we only focus on the problem of k-center clustering with outliers in Euclidean space, and the results also hold for the instances in arbitrary abstract metric spaces by using exactly the same idea. We present Algorithm 1 and Algorithm 2, and prove their clustering qualities in Theorem 1 and Theorem 2 respectively.

High-level idea. The two algorithms both follow the simple uniform sampling framework: take a small sample S from the input, and run an existing black-box algorithm $\mathcal{A}$ on S to obtain the solution. For Algorithm 1, the sample size $|S|$ is relatively smaller, and thus S contains only a small number of outliers, which is roughly $O(\log k)$, from $P \setminus P_{\text{opt}}$. Therefore, we can run a $\big(k + O(\log k)\big)$-center clustering algorithm (as the algorithm $\mathcal{A}$) on S, so as to achieve a constant factor approximate solution (in terms of the radius). For Algorithm 2, under the assumption $\frac{\epsilon_1}{\epsilon_2} > 1$, we can enlarge the sample size $|S|$ and safely output only k (instead of $k + O(\log k)$) cluster centers, also by running a black-box algorithm $\mathcal{A}$. The obtained approximation ratio is $c+2$ if $\mathcal{A}$ is a c-approximation algorithm with some $c \geq 1$. For example, if we apply the 3-approximation algorithm from [12], our Algorithm 2 will yield a 5-approximate solution.

3.1 The First Algorithm

For ease of presentation, we let r_{opt} be the optimal radius of the instance P, *i.e.*, each optimal cluster C^*_j is covered by a ball with radius r_{opt}. For any point $p \in \mathbb{R}^d$ and any value $r \geq 0$, we use $\texttt{Ball}(p, r)$ to denote the ball centered at p with radius r.

Theorem 1. *In Algorithm 1, the number of returned cluster centers $|H| = k + k' = k + \frac{1}{\eta}\frac{\epsilon_2}{\epsilon_1}\log\frac{k}{\eta}$. Also, with probability at least $(1-\eta)^2$, $\Delta^{-z}_{\infty}(P, H) \leq 4r_{\text{opt}}$.*

***Remark* 5.** The sample size $|S| = \frac{k}{\epsilon_1}\log\frac{k}{\eta}$ depends on k, ϵ_1, and η only. If $\frac{\epsilon_2}{\epsilon_1} = O(1)$ and η is assumed to be a fixed constant in $(0,1)$, k' will be $O(\log k)$. Also, the **running time** of the algorithm [29] used in Step 2 is $(k+k')|S|d = O(\frac{k^2}{\epsilon_1}(\log k)d)$, which is independent of the input size n.

Algorithm 1 UNI-k-CENTER OUTLIERS I

Input: An (ϵ_1, ϵ_2)-significant instance P of k-center clustering with z outliers, and $|P| = n$; a parameter $\eta \in (0,1)$.

(1) Sample a set S of $\frac{k}{\epsilon_1}\log\frac{k}{\eta}$ points uniformly at random from P.
(2) Let $k' = \frac{1}{\eta}\frac{\epsilon_2}{k}|S|$, and solve the $(k+k')$-center clustering problem on S by using the 2-approximation algorithm [29].

Output H, which is the set of $k+k'$ cluster centers returned in Step (2).

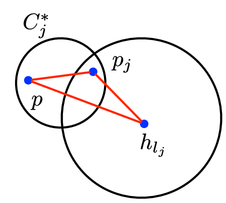

Figure 1: $||p - h_{l_j}|| \leq ||p - p_j|| + ||p_j - h_{l_j}|| \leq 4r_{\text{opt}}$.

PROOF. (**of Theorem 1**) First, it is straightforward to know that $|H| = k + k' = k + \frac{1}{\eta}\frac{\epsilon_2}{\epsilon_1}\log\frac{k}{\eta}$. Below, we assume that the sample S contains at least one point from each C^*_j, and at most $k' = \frac{\epsilon_2}{k\eta}|S|$ points from $P \setminus P_{\text{opt}}$. These events happen with probability at least $(1-\eta)^2$ according to Lemma 1 and Lemma 2 (as discussed in Remark 4 for their union bound).

Since the sample S contains at most k' points from $P \setminus P_{\text{opt}}$ and P_{opt} can be covered by k balls with radius r_{opt}, we know that S can be covered by $k+k'$ balls with radius r_{opt}. Thus, if we perform the 2-approximation $(k+k')$-center clustering algorithm [29] on S, the resulting balls should have radius no larger than $2r_{\text{opt}}$. Let $H = \{h_1, \cdots, h_{k+k'}\}$ and $\mathbb{B}_S = \{\texttt{Ball}(h_l, r) \mid 1 \leq l \leq k+k'\}$ be those balls covering S with $r \leq 2r_{\text{opt}}$. Also, for each $1 \leq j \leq k$, since $S \cap C^*_j \neq \emptyset$, there exists one ball of $\mathbb{B}_S$, say $\texttt{Ball}(h_{l_j}, r)$, covers at least one point, say p_j, from C^*_j. For any point $p \in C^*_j$, we have $||p - p_j|| \leq 2r_{\text{opt}}$ (by the triangle inequality) and $||p_j - h_{l_j}|| \leq r \leq 2r_{\text{opt}}$; therefore,

$$||p - h_{l_j}|| \leq ||p - p_j|| + ||p_j - h_{l_j}|| \leq 4r_{\text{opt}}. \tag{4}$$

See Figure 1 for an illustration. Overall, $P_{\text{opt}} = \cup_{j=1}^{k} C^*_j$ is covered by the union of the balls $\cup_{l=1}^{k+k'}\texttt{Ball}(h_l, 4r_{\text{opt}})$, *i.e.*, $\Delta^{-z}_{\infty}(P, H) \leq 4r_{\text{opt}}$. □

An "extreme" example for Theorem 1. We present an example to show that the value of k' in Step 2 cannot be reduced. That is, the clustering quality could be arbitrarily bad if we run $(k+k'')$-center clustering on S with $k'' < k'$. Let P be an (ϵ_1, ϵ_2)-significant instance in $\mathbb{R}^d$, where each optimal cluster C^*_j is a set of $|C^*_j|$ overlapping points located at its cluster center o^*_j. Let $x > 0$, and we assume (1) $||o^*_{j_1} - o^*_{j_2}|| \geq x$, $\forall j_1 \neq j_2$; (2) $||q_1 - q_2|| \geq x$,

$\forall q_1, q_2 \in P \setminus P_{\text{opt}}$; (3) $||o_j^* - q|| \geq x, \forall 1 \leq j \leq k, q \in P \setminus P_{\text{opt}}$. Obviously, the optimal radius $r_{\text{opt}} = 0$. Suppose we obtain a sample S satisfying $S \cap C_j^* \neq \emptyset$ for any $1 \leq j \leq k$ and $|S \setminus P_{\text{opt}}| = \frac{1}{\eta}\frac{\epsilon_2}{k}|S| = k'$. Given a number $k'' < k'$, we run $(k + k'')$-center clustering on S. Since the points of S take $k + k'$ distinct locations in the space, any $(k + k'')$-center clustering on S will result in a radius at least $x/2 > 0$; thus the approximation ratio is at least $\frac{x/2}{0} = +\infty$.

3.2 The Second Algorithm

We present the second algorithm (Algorithm 2) and analyze its quality in this section.

Theorem 2. If $\frac{\epsilon_1}{\epsilon_2} > \frac{1}{\eta(1-\delta)}$, with probability at least $(1-\eta)^2$, Algorithm 2 returns k cluster centers that achieve a $(c+2)$-approximation for k-center clustering with z outliers, *i.e.,* $\Delta_\infty^{-z}(P, H) \leq (c+2)r_{\text{opt}}$.

***Remark* 6.** **(i)** As an example, if we set $\eta = \delta = 1/2$, the algorithm works for any instance with $\frac{\epsilon_1}{\epsilon_2} > \frac{1}{\eta(1-\delta)} = 4$. Actually, as long as $\frac{\epsilon_1}{\epsilon_2} > 1$ (*i.e.,* $\min_{1 \leq j \leq k} |C_j^*| > z$), we can always find the appropriate values for η and δ to satisfy $\frac{\epsilon_1}{\epsilon_2} > \frac{1}{\eta(1-\delta)}$, *e.g.,* we can set $\eta = \sqrt{\frac{\epsilon_2}{\epsilon_1}}$ and $\delta < 1 - \sqrt{\frac{\epsilon_2}{\epsilon_1}}$. Obviously, if $\frac{\epsilon_1}{\epsilon_2}$ is close to 1, the success probability $(1-\eta)^2$ could be small (we will discuss that how to boost the success probability in Section 5). This also implies a simple observation: **the larger the ratio $\frac{\epsilon_1}{\epsilon_2}$ is , the better uniform sampling performs.**

(ii) Similar to Algorithm 1, the sample size $|S| = \frac{3k}{\delta^2 \epsilon_1} \log \frac{2k}{\eta}$ depends on k, ϵ_1, and the parameter δ and η only. The **running time** depends on the complexity of the subroutine c-approximation algorithm used in Step 2. For example, the algorithm of [12] takes $O\big(|S|^2 d + k|S|^2 \log |S|\big)$ time in $\mathbb{R}^d$.

Proof. (**of Theorem 2**) Similar to the proof of Theorem 1, we assume that $|S \cap C_j^*| \in (1 \pm \delta)\frac{|C_j^*|}{n}|S|$ for each C_j^*, and S has at most $z' = \frac{\epsilon_2}{k\eta}|S|$ points from $P \setminus P_{\text{opt}}$ (according to Lemma 1 and Lemma 2, and the discussion in Remark 4).

Let $\mathbb{B}_S = \{\texttt{Ball}(h_l, r) \mid 1 \leq l \leq k\}$ be the set of k balls returned in Step 2 of Algorithm 2. Since $S \cap P_{\text{opt}}$ can be covered by k balls with radius r_{opt} and $|S \setminus P_{\text{opt}}| \leq z'$, the optimal radius for the instance S with z' outliers should be at most r_{opt}. Consequently, $r \leq cr_{\text{opt}}$. Moreover, we have

$$
\begin{aligned}
|S \cap C_j^*| &\geq (1-\delta)\frac{|C_j^*|}{n}|S| \\
&\geq (1-\delta)\frac{\epsilon_1}{k}|S| \\
&> (1-\delta)\frac{\epsilon_2}{\eta(1-\delta)k}|S| \\
&= \frac{\epsilon_2}{\eta k}|S| = z'
\end{aligned} \tag{5}
$$

for any $1 \leq j \leq k$, where the last inequality comes from the assumption $\frac{\epsilon_1}{\epsilon_2} > \frac{1}{\eta(1-\delta)}$. Thus, if we perform k-center clustering with z' outliers on S, the resulting k balls must cover at least one point from each C_j^* (since $|S \cap C_j^*| > z'$ from (5)). Through a similar manner as the proof of Theorem 1, we know that $P_{\text{opt}} = \cup_{j=1}^k C_j^*$

Algorithm 2 Uni-k-Center Outliers II

Input: An (ϵ_1, ϵ_2)-significant instance P of k-center clustering with z outliers, and $|P| = n$; two parameters $\eta, \delta \in (0, 1)$.

(1) Sample a set S of $\frac{3k}{\delta^2 \epsilon_1} \log \frac{2k}{\eta}$ points uniformly at random from P.

(2) Let $z' = \frac{1}{\eta}\frac{\epsilon_2}{k}|S|$, and solve the k-center clustering with z' outliers problem on S by using any c-approximation algorithm with $c \geq 1$ (*e.g.,* the 3-approximation algorithm [12]).

Output H, which is the set of k cluster centers returned in Step (2).

---

is covered by the union of the balls $\cup_{l=1}^k \texttt{Ball}(h_l, r + 2r_{\text{opt}})$, *i.e.,* $\Delta_\infty^{-z}(P, H) \leq r + 2r_{\text{opt}} \leq (c+2)r_{\text{opt}}$. □

4 k-MEDIAN/MEANS CLUSTERING WITH OUTLIERS

For k-means clustering with outliers, we apply the similar uniform sampling ideas as Algorithm 1 and 2. However, the analyses are more complicated here. For ease of understanding, we present our high-level idea first. We then state the algorithms and the main theorems, and place the detailed proofs to Section 4.1 and 4.2. Also, we provide the extensions for k-median clustering with outliers and their counterparts in arbitrary metric space in Section B.

High-level idea. Recall $\{C_1^*, C_2^*, \cdots, C_k^*\}$ are the k optimal clusters. Denote by $O^* = \{o_1^*, \cdots, o_k^*\}$ the mean points of $\{C_1^*, \cdots, C_k^*\}$, respectively. We define the following transformation on P_{opt} to help us analyzing the clustering errors. Since the transformation forms k "stars" (see Figure 2), we call it "star shaped transformation".

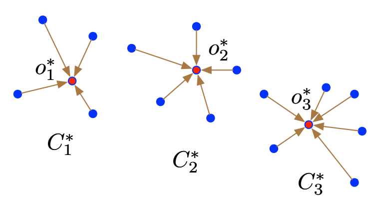

Figure 2: The transformation from P_{opt} to $\tilde{P}_{\text{opt}}$.

Definition 3 (Star Shaped Transformation). For each point in C_j^*, we translate it to o_j^*; overall, we generate a new set of $n - z$ points located at $\{o_1^*, \cdots o_k^*\}$, where each o_j^* has $|C_j^*|$ overlapping points. **(1)** For any point $p \in C_j^*$ with $1 \leq j \leq k$, denote by $\tilde{p}$ its transformed point. **(2)** For any $U \subseteq P_{\text{opt}}$, denote by $\tilde{U}$ its transformed point set.

Let S be a sufficiently large random sample from P. We first show that $S \cap C_j^*$ can well approximate C_j^* for each $1 \leq j \leq k$. Informally,

$$
\left.
\begin{aligned}
\frac{|S \cap C_j^*|}{|S|} &\approx \frac{|C_j^*|}{n}; \\
\frac{1}{|S \cap C_j^*|}\sum_{q \in S \cap C_j^*} ||q - o_j^*||^2 &\approx \frac{1}{|C_j^*|}\sum_{p \in C_j^*} ||p - o_j^*||^2.
\end{aligned}
\right\} \tag{6}
$$

Let $S_{\text{opt}} = S \cap P_{\text{opt}}$. By using (6), we can prove that the clustering costs of $\tilde{P}_{\text{opt}}$ and $\tilde{S}_{\text{opt}}$ are close (after the normalization) for any given set of cluster centers in the Euclidean space. Similar to Algorithm 1, we compute the $(k+k')$-means clustering on the sample S in Algorithm 3, where k' is roughly $O(\frac{\log k}{\xi^2})$ with a parameter $\xi \in (0,1)$. Let H be the returned set of $k+k'$ cluster centers. Then, we can use $\tilde{S}_{\text{opt}}$ and $\tilde{P}_{\text{opt}}$ as the "bridges" to connect S and P, so as to prove the following (informal) statement:

if H is a solution of the $(k+k')$-means clustering on the sample S with approximation guarantee, it also yields a solution of the $(k+k')$-means clustering with z outliers on the input data P with approximation guarantee, i.e., *$\Delta_2^{-z}(P,H)$ is bounded.*

In the second algorithm (similar to Algorithm 2), we run a k-means with z' outliers algorithm on the sample S and return exactly k (rather than $k+k'$) cluster centers. Let S_{in} be the set of obtained $|S|-z'$ inliers of S. If the ratio $\frac{\epsilon_1}{\epsilon_2} > 1$, we can prove that $|S_{\text{in}} \cap C_j^*| \approx |S \cap C_j^*|$ for each $1 \le j \le k$. Therefore, we can replace "S" by "S_{in}" in (6) and prove a similar quality guarantee for Algorithm 4.

The results. We present Algorithm 3 and Algorithm 4 to realize the above ideas, and provide the main theorems below. In Theorem 3, $\mathcal{L}$ denotes the maximum diameter of the k clusters $C_1^*, \cdots, C_k^*$, *i.e.,* $\mathcal{L} = \max_{1\le j\le k} \max_{p,q\in C_j^*} ||p-q||$. Actually, our result can be viewed as an extension of the sublinear time k-means clustering algorithms [19, 47] (who also have the additive clustering cost errors) to the case with outliers. We need to emphasize that the additive error is unavoidable even for the case without outliers, if we require the sample complexity to be independent of the input size [19, 47]. Though the k-median clustering with outliers algorithm in [46, Section 6] does not yield an additive error, as mentioned in Section 1, it needs to discard more than $16z$ outliers and the sample size depends on the ratio n/z.

Theorem 3. Let $0 < \delta, \eta, \xi < 1$. With probability at least $(1-\eta)^3$, the set of cluster centers H returned by Algorithm 3 results in a clustering cost $\Delta_2^{-z}(P,H)$ at most $\alpha\Delta_2^{-z}(P,O^*) + \beta\xi\mathcal{L}^2$, where $\alpha = \left(2 + (4+4c)\frac{1+\delta}{1-\delta}\right)$ and $\beta = (4+4c)\frac{1+\delta}{1-\delta}$.

***Remark* 7.** **(i)** In Step 2 of Algorithm 3, we can apply an $O(1)$-approximation k-means algorithm (*e.g.,* [37]). If we assume $1/\delta$ and $1/\eta$ are fixed constants, then the sample size $|S| = O(\frac{k}{\xi^2\epsilon_1}\log k)$, and both the factors α and β are $O(1)$, *i.e.,*

$$\Delta_2^{-z}(P,H) \le O(1) \cdot \Delta_2^{-z}(P,O^*) + O(\xi) \cdot \mathcal{L}^2. \tag{7}$$

Moreover, the number of returned cluster centers $|H| = k + O(\frac{\log k}{\xi^2})$ if $\frac{\epsilon_1}{\epsilon_2} = \Omega(1)$.

(ii) Furthermore, it is easy to see that the extreme example proposed at the end of Section 3.1 also works for Algorithm 3, *i.e.,* the value of k' cannot be reduced.

Theorem 4. Let $0 < \delta, \eta, \xi < 1$, and $t := \eta(1-\delta)\frac{\epsilon_1}{\epsilon_2}$. Assume $t > 1$. With probability at least $(1-\eta)^3$, the set of cluster centers H returned by Algorithm 4 results in a clustering cost $\Delta_2^{-z}(P,H)$ at most $\alpha\Delta_2^{-z}(P,O^*) + \beta\xi\mathcal{L}^2$, where $\alpha = \left(2 + (4+4c)\frac{t}{t-1}\frac{1+\delta}{1-\delta}\right)$ and $\beta = (4+4c)\frac{t}{t-1}\frac{1+\delta}{1-\delta}$.

Algorithm 3 Uni-k-Means Outliers I

Input: An (ϵ_1, ϵ_2)-significant instance P of k-means clustering with z outliers in $\mathbb{R}^d$, and $|P| = n$; three parameters $\eta, \delta, \xi \in (0,1)$.

(1) Take a uniform sample S of $\max\{\frac{3k}{\delta^2\epsilon_1}\log\frac{2k}{\eta}, \frac{k}{2\xi^2\epsilon_1(1-\delta)}\log\frac{2k}{\eta}\}$ points from P.

(2) Let $k' = \frac{1}{\eta}\frac{\epsilon_2}{k}|S|$, and solve the $(k+k')$-means clustering on S by using any c-approximation algorithm with $c \ge 1$.

Output H, which is the set of $k+k'$ cluster centers returned in Step (2).

Algorithm 4 Uni-k-Means Outliers II

Input: An (ϵ_1, ϵ_2)-significant instance P of k-means clustering with z outliers in $\mathbb{R}^d$, and $|P| = n$; three parameters $\eta, \delta, \xi \in (0,1)$.

(1) Take a uniform sample S of $\max\{\frac{3k}{\delta^2\epsilon_1}\log\frac{2k}{\eta}, \frac{k}{2\xi^2\epsilon_1(1-\delta)}\log\frac{2k}{\eta}\}$ points from P.

(2) Let $z' = \frac{1}{\eta}\frac{\epsilon_2}{k}|S|$, and solve the k-means clustering with z' outliers on S by using any c-approximation algorithm with $c \ge 1$.

Output H, which is the set of k cluster centers returned in Step (2).

***Remark* 8.** Similar to Remark 6 (i), as long as $\frac{\epsilon_1}{\epsilon_2} > 1$, we can set $\eta = \sqrt{\frac{\epsilon_2}{\epsilon_1}}$ and $\delta < 1 - \sqrt{\frac{\epsilon_2}{\epsilon_1}}$ to keep $t > 1$. Also, when $\frac{\epsilon_1}{\epsilon_2}$ is large, the success probability becomes high as well; moreover, the coefficients α and β will be lower, and then the clustering cost $\Delta_2^{-z}(P,H)$ will decrease (because $\frac{t}{t-1}$ is inversely proportional to $\frac{\epsilon_1}{\epsilon_2}$). For example, if $\frac{\epsilon_1}{\epsilon_2}$ is sufficiently large and $c = O(1)$, both α and β can be $O(1)$, and thus $\Delta_2^{-z}(P,H) \le O(1)\cdot\Delta_2^{-z}(P,O^*) + O(\xi)\cdot\mathcal{L}^2$. This also agrees with our previous observation concluded in Remark 6 (i), that is, the ratio $\frac{\epsilon_1}{\epsilon_2}$ is an important factor that affects the performance of the uniform sampling approach.

4.1 Proof of Theorem 3

Before proving Theorem 3, we need to introduce the following lemmas.

Lemma 3. We fix a cluster C_j^*. Given $\eta, \xi \in (0,1)$, if one uniformly selects a set T of $\frac{1}{2\xi^2}\log\frac{2}{\eta}$ or more points at random from C_j^*, then

$$\left|\frac{1}{|T|}\sum_{q\in T}||q-o_j^*||^2 - \frac{1}{|C_j^*|}\sum_{p\in C_j^*}||p-o_j^*||^2\right| \le \xi\mathcal{L}^2 \tag{8}$$

with probability at least $1-\eta$.

Lemma 3 can be obtained via the Hoeffding's inequality (each $||q-o_j^*||^2$ can be viewed as a random variable between 0 and $\mathcal{L}^2$) [1].

Lemma 4. If one uniformly selects a set S of $\max\{\frac{3k}{\delta^2\epsilon_1}\log\frac{2k}{\eta}, \frac{k}{2\xi^2\epsilon_1(1-\delta)}\log\frac{2k}{\eta}\}$ points at random from P,

$$\sum_{q\in S\cap C_j^*}||q-o_j^*||^2 \leq (1+\delta)\frac{|S|}{n}\Big(\sum_{p\in C_j^*}||p-o_j^*||^2+\xi|C_j^*|\mathcal{L}^2\Big) \quad (9)$$

for each $1\leq j\leq k$, with probability at least $(1-\eta)^2$.

Proof. Suppose $|S|=\max\{\frac{3k}{\delta^2\epsilon_1}\log\frac{2k}{\eta}, \frac{k}{2\xi^2\epsilon_1(1-\delta)}\log\frac{2k}{\eta}\}$. According to Lemma 1, $|S|\geq\frac{3k}{\delta^2\epsilon_1}\log\frac{2k}{\eta}$ implies

$$|S\cap C_j^*|\geq(1-\delta)\frac{|C_j^*|}{n}|S|\geq(1-\delta)\frac{\epsilon_1}{k}|S| \quad (10)$$

for each $1\leq j\leq k$, with probability at least $1-\eta$. Below, we assume (10) occurs. Further, $|S|\geq\frac{k}{2\xi^2\epsilon_1(1-\delta)}\log\frac{2k}{\eta}$ implies

$$(1-\delta)\frac{\epsilon_1}{k}|S|\geq\frac{1}{2\xi^2}\log\frac{2k}{\eta}. \quad (11)$$

Combining (10) and (11), we have $|S\cap C_j^*|\geq\frac{1}{2\xi^2}\log\frac{2k}{\eta}$. Consequently, through Lemma 3 we obtain

$$\left|\frac{1}{|S\cap C_j^*|}\sum_{q\in S\cap C_j^*}||q-o_j^*||^2-\frac{1}{|C_j^*|}\sum_{p\in C_j^*}||p-o_j^*||^2\right|\leq\xi\mathcal{L}^2 \quad (12)$$

for each $1\leq j\leq k$, with probability at least $1-\eta$ (η is replaced by η/k in Lemma 3 for taking the union bound). From (12) we have

$$\begin{aligned}\sum_{q\in S\cap C_j^*}||q-o_j^*||^2 &\leq |S\cap C_j^*|\Big(\frac{1}{|C_j^*|}\sum_{p\in C_j^*}||p-o_j^*||^2+\xi\mathcal{L}^2\Big)\\ &\leq(1+\delta)\frac{|C_j^*|}{n}|S|\Big(\frac{1}{|C_j^*|}\sum_{p\in C_j^*}||p-o_j^*||^2+\xi\mathcal{L}^2\Big)\\ &=(1+\delta)\frac{|S|}{n}\Big(\sum_{p\in C_j^*}||p-o_j^*||^2+\xi|C_j^*|\mathcal{L}^2\Big),\end{aligned} \quad (13)$$

where the second inequality comes from Lemma 1. The overall success probability $(1-\eta)^2$ comes from the success probabilities of (10) and (12). So we complete the proof. □

For ease of presentation, we define a new notation that is used in the following lemmas. Given two point sets X and $Y\subset\mathbb{R}^d$, we use $\mathbf{Cost}(X,Y)$ to denote the clustering cost of X by taking Y as the cluster centers, *i.e.*, $\mathbf{Cost}(X,Y)=\sum_{q\in X}(\mathtt{dist}(q,Y))^2$. Obviously, $\Delta_2^{-z}(P,H)=\frac{1}{n-z}\mathbf{Cost}(P_{\mathrm{opt}},H)$. Let $S_{\mathrm{opt}}=S\cap P_{\mathrm{opt}}$. Below, we prove the upper bounds of $\mathbf{Cost}(S_{\mathrm{opt}},O^*)$, $\mathbf{Cost}(\tilde{S}_{\mathrm{opt}},H)$, and $\mathbf{Cost}(\tilde{P}_{\mathrm{opt}},H)$ respectively, and use these bounds to complete the proof of Theorem 3. For convenience, we always assume that the events described in Lemma 1, Lemma 2, and Lemma 4 all occur, so that we do not need to repeatedly state the success probabilities.

Lemma 5. $\mathbf{Cost}(S_{\mathrm{opt}},O^*)\leq(1+\delta)\frac{|S|}{n}(n-z)\big(\Delta_2^{-z}(P,O^*)+\xi\mathcal{L}^2\big)$.

Proof. First, we have

$$\begin{aligned}\mathbf{Cost}(S_{\mathrm{opt}},O^*)&=\sum_{j=1}^{k}\sum_{q\in S\cap C_j^*}||q-o_j^*||^2\\ &\leq(1+\delta)\frac{|S|}{n}\sum_{j=1}^{k}\Big(\sum_{p\in C_j^*}||p-o_j^*||^2+\xi|C_j^*|\mathcal{L}^2\Big)\end{aligned} \quad (14)$$

by Lemma 4. Further, since $\sum_{j=1}^{k}\sum_{p\in C_j^*}||p-o_j^*||^2=(n-z)\Delta_2^{-z}(P,O^*)$ and $\sum_{j=1}^{k}|C_j^*|=n-z$, by plugging them into (14), we obtain Lemma 5. □

Lemma 6. $\mathbf{Cost}(\tilde{S}_{\mathrm{opt}},H)\leq(2+2c)\cdot\mathbf{Cost}(S_{\mathrm{opt}},O^*)$.

Proof. We fix a point $q\in S_{\mathrm{opt}}$, and assume that the nearest neighbors of q and $\tilde{q}$ in H are h_{j_q} and $h_{\tilde{j}_q}$, respectively. Then, we have

$$||\tilde{q}-h_{\tilde{j}_q}||^2\leq||\tilde{q}-h_{j_q}||^2\leq2||\tilde{q}-q||^2+2||q-h_{j_q}||^2. \quad (15)$$

Therefore,

$$\begin{aligned}&\sum_{q\in S_{\mathrm{opt}}}||\tilde{q}-h_{\tilde{j}_q}||^2\leq2\sum_{q\in S_{\mathrm{opt}}}||\tilde{q}-q||^2+2\sum_{q\in S_{\mathrm{opt}}}||q-h_{j_q}||^2,\\ \Longrightarrow\ &\mathbf{Cost}(\tilde{S}_{\mathrm{opt}},H)\leq2\mathbf{Cost}(S_{\mathrm{opt}},O^*)+2\mathbf{Cost}(S_{\mathrm{opt}},H).\end{aligned} \quad (16)$$

Moreover, since $S_{\mathrm{opt}}\subseteq S$ (because $S_{\mathrm{opt}}=S\cap P_{\mathrm{opt}}$) and H yields a c-approximate clustering cost of the $(k+k')$-means clustering on S, we have

$$\mathbf{Cost}(S_{\mathrm{opt}},H)\leq\mathbf{Cost}(S,H)\leq c\cdot W, \quad (17)$$

where W is the optimal clustering cost of $(k+k')$-means clustering on S. Let S' be the k' farthest points of S to O^*, then the set $O^*\cup S'$ also forms a solution for $(k+k')$-means clustering on S; namely, S is partitioned into $k+k'$ clusters where each point of S' is a cluster having a single point. Obviously, such a clustering yields a clustering cost $(|S|-k')\Delta_2^{-k'}(S,O^*)$. Consequently,

$$W\leq(|S|-k')\Delta_2^{-k'}(S,O^*). \quad (18)$$

Also, Lemma 2 shows that S contains at most k' points from $P\setminus P_{\mathrm{opt}}$, *i.e.*, $|S_{\mathrm{opt}}|\geq|S|-k'$. Thus, $\mathbf{Cost}(S_{\mathrm{opt}},O^*)\geq(|S|-k')\Delta_2^{-k'}(S,O^*)$. Together with (16), (17), and (18), we have $\mathbf{Cost}(\tilde{S}_{\mathrm{opt}},H)\leq(2+2c)\cdot\mathbf{Cost}(S_{\mathrm{opt}},O^*)$. □

Lemma 7. $\mathbf{Cost}(\tilde{P}_{\mathrm{opt}},H)\leq\frac{1}{1-\delta}\frac{n}{|S|}\mathbf{Cost}(\tilde{S}_{\mathrm{opt}},H)$.

Proof. From the constructions of $\tilde{P}_{\mathrm{opt}}$ and $\tilde{S}_{\mathrm{opt}}$, we know that they are overlapping points locating at $\{o_1^*,\cdots,o_k^*\}$. From Lemma 1, we know $|S\cap C_j^*|\geq(1-\delta)\frac{|C_j^*|}{n}|S|$, *i.e.*, $|C_j^*|\leq\frac{1}{1-\delta}\frac{n}{|S|}|S\cap C_j^*|$ for $1\leq j\leq k$. Overall, we have $\mathbf{Cost}(\tilde{P}_{\mathrm{opt}},H)=\sum_{j=1}^{k}|C_j^*|\big(\mathtt{dist}(o_j^*,H)\big)^2$ that is at most

$$\frac{1}{1-\delta}\frac{n}{|S|}\sum_{j=1}^{k}|S\cap C_j^*|\big(\mathtt{dist}(o_j^*,H)\big)^2=\frac{1}{1-\delta}\frac{n}{|S|}\mathbf{Cost}(\tilde{S}_{\mathrm{opt}},H).$$

□

Now, we are ready to prove Theorem 3.

Proof. (**of Theorem 3**) Note that $(n-z)\Delta_2^{-z}(P,H)$ actually is the $|H|$-means clustering cost of P by removing the farthest z points to H, and $|P_{\text{opt}}| = n-z$. So we have

$$(n-z)\Delta_2^{-z}(P,H) \le \textbf{Cost}(P_{\text{opt}}, H). \tag{19}$$

Further, by using a similar manner of (16), we have $\textbf{Cost}(P_{\text{opt}}, H) \le 2\textbf{Cost}(P_{\text{opt}}, O^*) + 2\textbf{Cost}(\tilde{P}_{\text{opt}}, H)$. Therefore,

$$\Delta_2^{-z}(P,H) \le \frac{1}{n-z}\textbf{Cost}(P_{\text{opt}}, H) \tag{20}$$
$$\le \frac{2}{n-z}\Big(\textbf{Cost}(P_{\text{opt}}, O^*) + \textbf{Cost}(\tilde{P}_{\text{opt}}, H)\Big)$$
$$\le \frac{2}{n-z}\Big(\textbf{Cost}(P_{\text{opt}}, O^*) + \frac{1}{1-\delta}\frac{n}{|S|}\textbf{Cost}(\tilde{S}_{\text{opt}}, H)\Big) \tag{21}$$
$$\le \frac{2}{n-z}\Big(\textbf{Cost}(P_{\text{opt}}, O^*) + \frac{1}{1-\delta}\frac{n}{|S|}(2+2c)\textbf{Cost}(S_{\text{opt}}, O^*)\Big) \tag{22}$$
$$\le \frac{2}{n-z}\Big(\textbf{Cost}(P_{\text{opt}}, O^*) + \frac{1+\delta}{1-\delta}(2+2c)(n-z)\big(\Delta_2^{-z}(P,O^*)+\xi\mathcal{L}^2\big)\Big) \tag{23}$$
$$= \big(2+(4+4c)\frac{1+\delta}{1-\delta}\big)\Delta_2^{-z}(P,O^*) + (4+4c)\frac{1+\delta}{1-\delta}\xi\mathcal{L}^2, \tag{24}$$

where (21), (22), and (23) come from Lemma 7, Lemma 6, and Lemma 5 respectively, and (24) comes from the fact $\textbf{Cost}(P_{\text{opt}}, O^*) = (n-z)\Delta_2^{-z}(P,O^*)$. The success probability $(1-\eta)^3$ comes from Lemma 4 and Lemma 2 (note that Lemma 4 already takes into account of the success probability of Lemma 1). Thus, we complete the proof of Theorem 3. □

4.2 Proof of Theorem 4

Before proving Theorem 4, we introduce the following lemmas first. Suppose the k clusters of S obtained in Step (2) of Algorithm 4 are $S_1, S_2, \cdots, S_k$, and thus the inliers $S_{\text{in}} = \cup_{j=1}^k S_j$. Similar to Section 4.1, below we always assume that the events described in Lemma 1, Lemma 2, and Lemma 4 all occur, so that we do not need to repeatedly state the success probabilities.

Lemma 8. $\frac{|C_j^*|}{|C_j^* \cap S_{\text{in}}|} \le \frac{n}{|S|}\frac{t}{(t-1)(1-\delta)}$ for each $1 \le j \le k$.

Proof. Since $z' = \frac{1}{\eta}\frac{\epsilon_2}{k}|S|$ and $|S \cap C_j^*| \ge (1-\delta)\frac{|C_j^*|}{n}|S|$ for each $1 \le j \le k$ (by Lemma 1), we have

$$|S_{\text{in}} \cap C_j^*| \ge |S \cap C_j^*| - z' \ge (1-\delta)\frac{|C_j^*|}{n}|S| - \frac{1}{\eta}\frac{\epsilon_2}{k}|S|$$
$$= \Big(1 - \frac{1}{\eta(1-\delta)}\frac{\epsilon_2}{k}\frac{n}{|C_j^*|}\Big)(1-\delta)\frac{|C_j^*|}{n}|S|$$
$$\ge \Big(1 - \frac{1}{\eta(1-\delta)}\frac{\epsilon_2}{\epsilon_1}\Big)(1-\delta)\frac{|C_j^*|}{n}|S|$$
$$= (1-\frac{1}{t})(1-\delta)\frac{|C_j^*|}{n}|S|, \tag{25}$$

where the last inequality comes from $|C_j^*| \ge \frac{\epsilon_1}{k}n$. Thus $\frac{|C_j^*|}{|C_j^* \cap S_{\text{in}}|} \le \frac{n}{|S|}\frac{t}{(t-1)(1-\delta)}$. □

Lemma 9. $\textbf{Cost}(S_{\text{in}} \cap P_{\text{opt}}, H) \le (1+\delta)\frac{|S|}{n}\cdot c \cdot \big(\textbf{Cost}(P_{\text{opt}}, O^*) + (n-z)\cdot\xi\mathcal{L}^2\big)$.

Proof. Since $z' \ge |S \setminus P_{\text{opt}}| = |S| - |S_{\text{opt}}|$, *i.e.,* $|S_{\text{opt}}| \ge |S| - z'$, we have

$$(|S|-z')\Delta_2^{-z'}(S,O^*) \le |S_{\text{opt}}|\cdot\Delta_2^{-z'}(S,O^*)$$
$$\le \textbf{Cost}(S_{\text{opt}}, O^*)$$
$$\Longrightarrow \Delta_2^{-z'}(S,O^*) \le \frac{1}{|S|-z'}\textbf{Cost}(S_{\text{opt}}, O^*), \tag{26}$$

where the second inequality is due to the same reason of (19). Because H is a c-approximation on S,

$$\Delta_2^{-z'}(S,H) \le c\cdot\Delta_2^{-z'}(S,O^*)$$
$$\le \frac{c}{|S|-z'}\textbf{Cost}(S_{\text{opt}}, O^*), \tag{27}$$

where the second inequality comes from (26). Therefore,

$$\begin{aligned}
& \textbf{Cost}(S_{\text{in}} \cap P_{\text{opt}}, H) \\
\le\ & \textbf{Cost}(S_{\text{in}}, H) \\
=\ & (|S|-z')\Delta_2^{-z'}(S,H) \\
\le\ & c\cdot\textbf{Cost}(S_{\text{opt}}, O^*) \\
\le\ & (1+\delta)\frac{|S|}{n}\cdot c\cdot(n-z)\big(\Delta_2^{-z}(P,O^*)+\xi\mathcal{L}^2\big) \\
=\ & (1+\delta)\frac{|S|}{n}\cdot c\cdot\big(\textbf{Cost}(P_{\text{opt}}, O^*) + (n-z)\cdot\xi\mathcal{L}^2\big),
\end{aligned} \tag{28}$$

where the second and third inequalities comes from (27) and Lemma 5, respectively. So we complete the proof. □

Since $S_{\text{in}} \cap P_{\text{opt}} \subseteq S_{\text{opt}}$, we immediately have the following lemma via Lemma 5.

Lemma 10. $\textbf{Cost}(S_{\text{in}} \cap P_{\text{opt}}, O^*) \le (1+\delta)\frac{|S|}{n}(n-z)\big(\Delta_2^{-z}(P,O^*) + \xi\mathcal{L}^2\big)$.

Now, we are ready to prove Theorem 4.

Proof. (**of Theorem 4**) For convenience, let $S'_{\text{in}} = S_{\text{in}} \cap P_{\text{opt}}$. Using the same manner of (16), we have

$$\textbf{Cost}(\tilde{S}'_{\text{in}}, H) \le 2\textbf{Cost}(S'_{\text{in}}, O^*) + 2\textbf{Cost}(S'_{\text{in}}, H); \tag{29}$$
$$\textbf{Cost}(P_{\text{opt}}, H) \le 2\textbf{Cost}(P_{\text{opt}}, O^*) + 2\textbf{Cost}(\tilde{P}_{\text{opt}}, H). \tag{30}$$

Also, because $\textbf{Cost}(\tilde{P}_{\text{opt}}, H) = \sum_{j=1}^k |C_j^*|(\texttt{dist}(o_j^*, H))^2$ and $\textbf{Cost}(\tilde{S}'_{\text{in}}, H) = \sum_{j=1}^k |C_j^* \cap S_{\text{in}}|(\texttt{dist}(o_j^*, H))^2$, we have

$$\frac{\textbf{Cost}(\tilde{P}_{\text{opt}}, H)}{\textbf{Cost}(\tilde{S}'_{\text{in}}, H)} \le \max_{1\le j\le k}\frac{|C_j^*|}{|C_j^* \cap S_{\text{in}}|}. \tag{31}$$

As a consequence,

$$\textbf{Cost}(\tilde{P}_{\text{opt}}, H) \le \max_{1\le j\le k}\frac{|C_j^*|}{|C_j^* \cap S_{\text{in}}|}\cdot\textbf{Cost}(\tilde{S}'_{\text{in}}, H)$$
$$\le \frac{n}{|S|}\frac{t}{(t-1)(1-\delta)}\cdot\textbf{Cost}(\tilde{S}'_{\text{in}}, H) \tag{32}$$

where the last inequality comes from Lemma 8. From (29), (30), and (32), we have

$$\mathbf{Cost}(P_{\text{opt}}, H) \leq 2\mathbf{Cost}(P_{\text{opt}}, O^*) + \frac{4n}{|S|} \frac{t}{(t-1)(1-\delta)} \cdot \left(\mathbf{Cost}(S'_{\text{in}}, O^*) + \mathbf{Cost}(S'_{\text{in}}, H)\right). \tag{33}$$

By plugging the inequalities of Lemma 9 and Lemma 10 into (33), we can obtain Theorem 4. □

5 EXPERIMENTS

The experiments were conducted on a Ubuntu workstation with 2.40GHz Intel(R) Xeon(R) CPU E5-2680 and 256GB main memory; the algorithms are implemented in Matlab R2019b. The codes have been archived in https://github.com/anon68/lightweight. We show the experimental results for k-center/means clustering with outliers. The results for k-median are quite similar to those of k-means.

Algorithms for testing. We use several baseline algorithms mentioned in Section 1.2. For k-center clustering with outliers, we consider four algorithms: the 3-approximation **Charikar** [12], the $(4+\epsilon)$-approximation **MK** [44], the 13-approximation **Malkomes** [42], and the adaptive sampling algorithm **DYW** [24] (as the non-uniform sampling approach). The distributed algorithm Malkomes partitions the dataset into $s \geq 1$ parts and processes each part separately; we set $s = 1$ for Malkomes in our experiments (since we only consider the setting with single machine). In our Algorithm 2, we apply MK as the black-box algorithm in Step 2 (though Charikar has a lower approximation ratio, we observe that MK runs faster and often achieves comparable clustering results in practice).

For k-means/median clustering with outliers, we consider the heuristic algorithm k-**means**−− [14] and two non-uniform sampling methods: the local search algorithm **LocalSearch** with k-means++ seeding [32], and the recently proposed data summary based algorithm **DataSummary** [15]. In our Algorithm 3 and Algorithm 4, we apply the k-means++ [2] and k-means−− respectively as the black-box algorithms in their Step 2.

Datasets. We generate the synthetic datasets in $\mathbb{R}^{100}$ (the values n, k, and z will be varied). We randomly generate k points as the cluster centers inside a hypercube of side length 400; around each center, we generate a cluster of points following a Gaussian distribution with standard deviation $\sqrt{1000}$; we keep the total number of points to be $n - z$; to study the performance of our algorithms with respect to the ratio $\frac{\epsilon_1}{\epsilon_2}$, we vary the size of the smallest cluster appropriately for each synthetic dataset; finally, we uniformly generate z outliers at random outside the minimum enclosing balls of these k clusters. To keep the generality of our datasets, we randomly set the ratio of the largest cluster size to the smallest cluster size in the range $[1, 50]$.

We also choose 4 real datasets from *UCI machine learning repository* [25]. **Covertype** has 7 clusters with 5.8×10^5 points in $\mathbb{R}^{54}$; **Kddcup** has 23 clusters with 4.9×10^6 points in $\mathbb{R}^{38}$; **Poking Hand** has 10 clusters with 10^6 points in $\mathbb{R}^{10}$; **Shuttle** has 7 clusters with 5.8×10^4 points in $\mathbb{R}^9$. We add 1% outliers outside the enclosing balls of the clusters as we did for the synthetic datasets.

Some implementation details. When implementing our proposed algorithms (Algorithm 1-4), it is not quite convenient to set the values for the parameters $\{\eta, \xi, \delta\}$ in practice. In fact, we only need to determine the sample size $|S|$ and k' for implementing Algorithm 1 and Algorithm 3 (and similarly, $|S|$ and z' for implementing Algorithm 2 and Algorithm 4). As an example, for Algorithm 1, it is sufficient to input $|S|$ and k' only, since we just need to compute the $(k + k')$-center clustering on the sample S; moreover, it is more intuitive to directly set $|S|$ and k' rather than to set the parameters $\{\eta, \xi, \delta\}$. Therefore, we will conduct the experiments to observe the trends when varying the values $|S|$, k', and z'.

Another practical issue for implementation is the success probability. For Algorithm 2 and 4, when $\frac{\epsilon_1}{\epsilon_2}$ is close to 1, the success probability could be low (as discussed in Remark 6). We can run the algorithm multiple times, and at least one time the algorithm will return a qualified solution with a higher probability (by simply taking the union bound). For example, if $\eta = 0.8$ and we run Algorithm 2 50 times, the success probability will be $1 - (1 - (1 - 0.8)^2)^{50} \approx 87\%$. Suppose we run the algorithm $m > 1$ times and let $H_1, \cdots, H_m$ be the set of output candidates. The remaining issue is that how to select the one that achieves the smallest objective value among all the candidates. A simple way is to directly scan the whole dataset in one pass. When reading a point p from P, we calculate its distance to all the candidates, *i.e.,* $\text{dist}(p, H_1), \cdots, \text{dist}(p, H_m)$; after scanning the whole dataset, we have calculated the clustering costs $\Delta_\infty^{-z}(P, H_l)$ (*resp.,* $\Delta_1^{-z}(P, H_l)$ and $\Delta_2^{-z}(P, H_l)$) for $1 \leq l \leq m$ and then can return the best one. Another benefit of this operation is that we can determine the clustering assignments for the data points simultaneously. When calculating $\text{dist}(p, H_l)$ for $1 \leq l \leq m$, we record the index of its nearest cluster center in H_l; finally, we can return its corresponding clustering assignment in the selected best candidate.

Experimental design. Our experiments include three parts. **(1)** We fix the sample size $|S|$ to be $5 \times 10^{-3} n$, and study the clustering quality (such as clustering cost, precision, and purity) and the running time. **(2)** We study the scalabilities of the algorithms with enlarging the data size n, and the stabilities of our proposed algorithms with varying the sampling ratio $|S|/n$. **(3)** Finally, we focus on other factors that may affect the practical performances.

5.1 Clustering Quality and Running Time

In this part, we fix the sample size $|S|$ to be $5 \times 10^{-3} n$. For the synthetic datasets, we first set $n = 10^5$, $k = 8$, and $z = 2\% n$. For Algorithm 1 and 3, the value k' should be $\frac{1}{\eta} \frac{\epsilon_2}{k} |S|$ which could be large (though it is only $O(\log k)$ in the asymptotic analysis). Alhough we have constructed the example in Section 3.1 to show that k' cannot be reduced in the worst case, we do not strictly follow this theoretical value in our experiments. Instead, we keep the ratio $\tau = \frac{k+k'}{k}$ to be 1, $\frac{4}{3}$, $\frac{5}{3}$, and 2 (*i.e.,* run the black-box algorithms from [2, 29] τk steps). For Algorithm 2 and Algorithm 4, we set $z' = 2\frac{\epsilon_2}{k}|S|$; as discussed for the implementation details before, to boost the success probability we run Algorithm 2 (*resp.,* Algorithm 4) 10 times and select the best candidate (as one trial); we count the total time of the 10 runs. For each instance, we take 20 trials and report the average results in Figure 3.

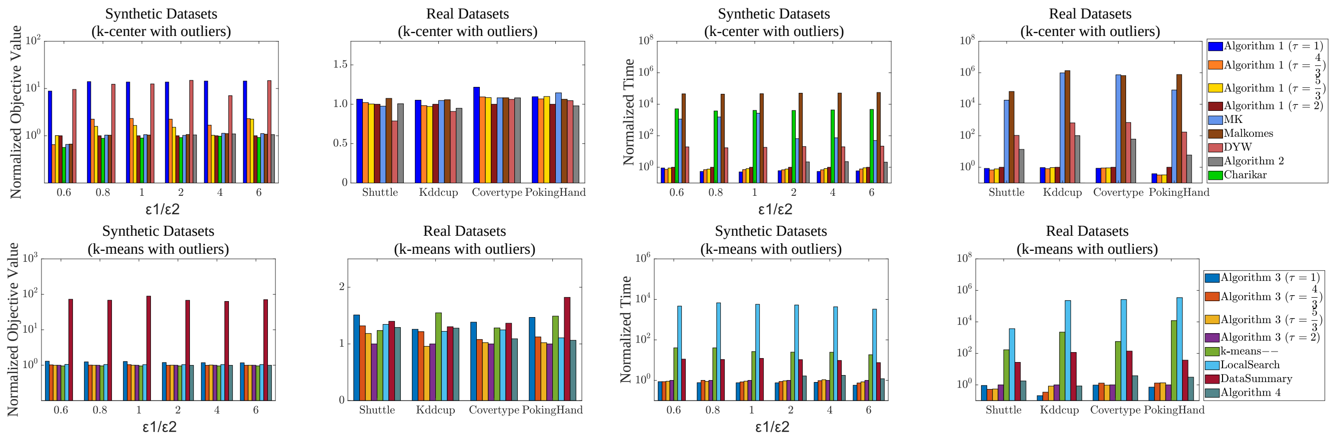

Figure 3: The normalized objective values and running times for k-center (the first line), k-means (the second line) clustering and k-median (the third line) with outliers.

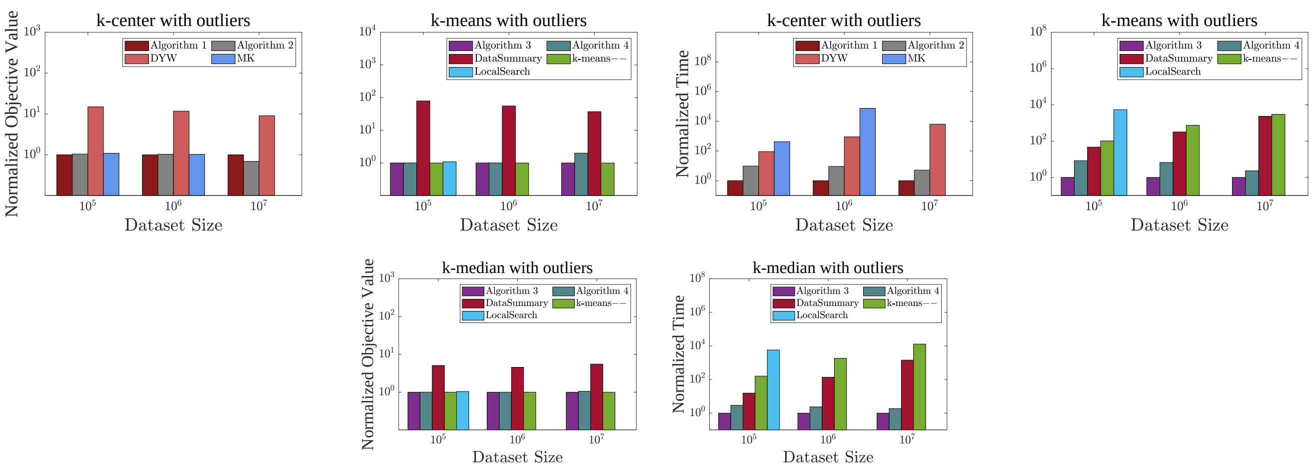

Figure 4: The results on the synthetic datasets with varying n. We did not run Charikar and Malkomes, and for $n = 10^7$, we did not run MK and LocalSearch, because their running times are too high for large n.

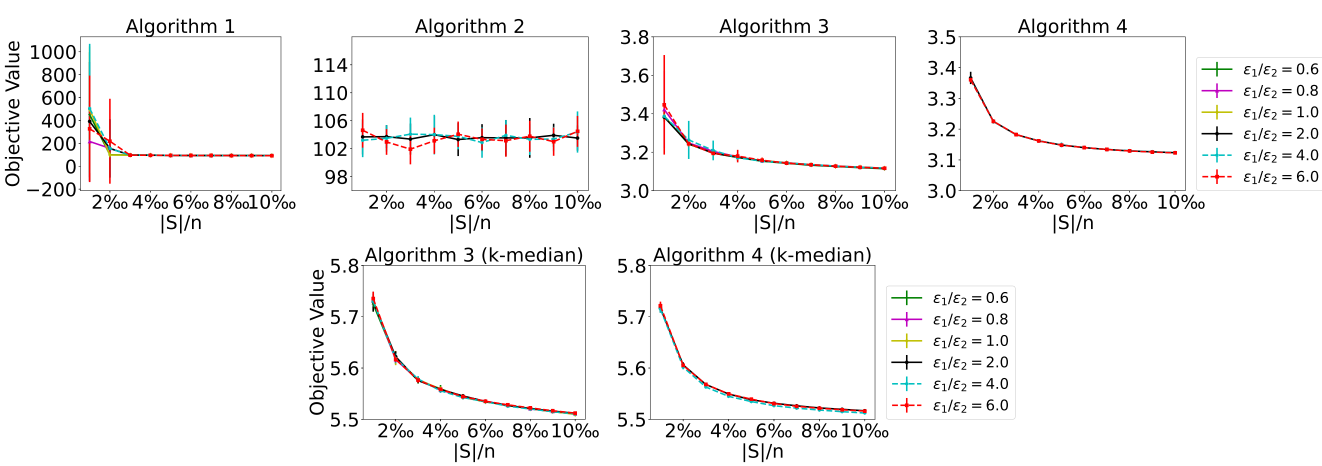

Figure 5: The performances with varying the sampling ratio $|S|/n$ (we run Algorithm 2 and 4 for $\epsilon_1/\epsilon_2 > 1$).

For k-center clustering with outliers, our algorithms (Algorithm 1 with $\tau \geq 4/3$ and Algorithm 2) and the four baseline algorithms achieve similar objective values for most of the instances (we run Algorithm 2 on the synthetic datasets with $\frac{\epsilon_1}{\epsilon_2} > 1$ only; we do not run Charikar on the real datasets due to its high complexity). For k-means clustering with outliers, Algorithm 3 with $\tau \geq 4/3$ and Algorithm 4 can achieve the results close to the best of the three baseline algorithms. Moreover, the running times of our algorithms are significantly lower comparing with the baseline algorithms. Overall, we conclude that (1) Algorithm 1 and Algorithm 3 just need to return slightly more than k cluster centers (*e.g.,* $\frac{4}{3}k$) for achieving low clustering cost; (2) Algorithm 2 and Algorithm 4 can achieve low clustering cost when $\frac{\epsilon_1}{\epsilon_2} \geq 2$.

To further evaluate their clustering qualities, we consider the measures *precision* and *purity*, which have been widely used before [43]; these two measures both aim to evaluate the difference between the obtained clusters and the ground truth. The precision is the proportion of the ground-truth outliers found by the algorithm (*i.e.,* $\frac{|Out \cap Out_{\text{truth}}|}{|Out_{\text{truth}}|}$, where Out is the set of returned outliers and Out_{truth} is the set of ground-truth outliers). For each obtained cluster, we assign it to the ground-truth cluster which is most frequent in the obtained cluster, and the purity measures the accuracy of this assignment. Specifically, let $\{C_1, C_2, \cdots, C_k\}$ be the ground-truth clusters and $\{C'_1, C'_2, \cdots, C'_k\}$ be the obtained clusters from the algorithm; the purity is equal to $\frac{1}{n-z}\sum_{j=1}^{k}\max_{1\leq l\leq k}|C'_j \cap C_l|$. In general, the precisions and the purities achieved by our uniform sampling approaches and the baseline algorithms are relatively close. The numerical results are shown in Table 1 and 2.

5.2 Scalability and Sampling Ratio

We consider the scalability first. We enlarge the data size n from 10^5 to 10^7, and illustrate the results in Figure 4. We still fix the sampling ratio $|S|/n = 5 \times 10^{-3}$ as Section 5.1. For Algorithm 1 and 3, we set $\tau = 2$. We can see that our algorithms are several orders of magnitude faster than the baselines when n is large. Actually, since our approach is just very simple uniform sampling, the advantage of our approach over the non-uniform sampling approaches will be more significant when n becomes large.

We then study the influence of the sampling ratio $|S|/n$ to the obtained clustering costs of our algorithms. We vary $|S|/n$ from 1×10^{-3} to 10×10^{-3} and run our algorithms on the synthetic datasets. We show the results (averaged across 20 trials) in Figure 5. We can see that the trends tend to be "flat" when the ratio $|S|/n > 4 \times 10^{-3}$.

5.3 Other Influence Factors

For completeness, we also consider several other influence factors. Again, we fix the sampling ratio $|S|/n = 5 \times 10^{-3}$ as Section 5.1. As discussed before, the performances of Algorithm 2 and Algorithm 4 depend on the ratio ϵ_1/ϵ_2. An interesting question is that how about their stabilities in terms of ϵ_1/ϵ_2 (in particular, when $\epsilon_1/\epsilon_2 \leq 1$). We repeat the experiments 20 times on the synthetic datasets and compute the obtained average objective values and standard deviations. In Figure 6, we can see that their performances are actually quite stable when $\epsilon_1/\epsilon_2 \geq 2$. This also agrees with our previous theoretical analysis, that is, larger ϵ_1/ϵ_2 is more friendly to uniform sampling.

In Figure 7, we study the influence of z' on the performances (averaged across 20 times). We vary $z'/\tilde{z}$ from 1.1 to 2, where $\tilde{z} = \frac{\epsilon_2}{k}|S|$ is the expected number of outliers contained in S. We

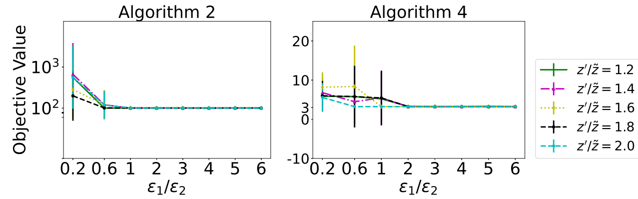

Figure 6: The average objective values and standard deviations with varying ϵ_1/ϵ_2. $\tilde{z} = \frac{\epsilon_2}{k}|S|$ is the expected number of outliers contained in S.

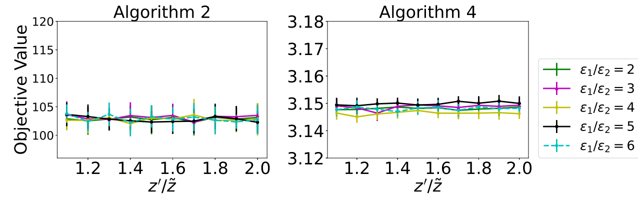

Figure 7: The average objective values and standard deviations with varying $z'/\tilde{z}$.

also illustrate the experimental results on the synthetic datasets with varying k from 8 to 24, and z from $0.2\%n$ to $10\%n$, in Figure 8 and Figure 9 respectively. Figure 10 shows the results for k-median. Overall, we observe that the influences from these parameters to the clustering qualities of our algorithms are relatively limited. And in general, our algorithms are considerably faster than the baseline algorithms.

6 FUTURE WORK

To explain the effectiveness of uniform sampling for center-based clustering with outliers problems, we introduce the significance criterion in this paper. Following this work, an interesting question is that whether the significance criterion (or some other realistic assumptions) can be applied to analyze uniform sampling for other robust optimization problems, such as *PCA with outliers* [8] and *projective clustering with outliers* [27].

A PROOF OF LEMMA 1

Lemma 1 can be directly obtained through the following claim. We just replace η by η/k in Claim 1, because we need to take the union bound over all the k clusters.

Claim 1. Let U be a set of elements and $V \subseteq U$ with $\frac{|V|}{|U|} = \tau > 0$. Given $\eta, \delta \in (0,1)$, we uniformly select a set S of elements from U at random. Then we have:

- (i) if $|S| \geq \frac{1}{\tau}\log\frac{1}{\eta}$, with probability at least $1-\eta$, S contains at least one element from V;
- (ii) if $|S| \geq \frac{3}{\delta^2\tau}\log\frac{2}{\eta}$, with probability at least $1-\eta$, we have $\big||S \cap V| - \tau|S|\big| \leq \delta\tau|S|$.

Proof. Actually, (i) is a folklore result that has been presented in several papers before (such as [23]). Since each sampled element

Table 1: Precision and Purity on real datasets (k-center with outliers)

Datasets	Shuttle		Kddcup		Covtype		Poking Hand	
Measure	Prec	Purity	Prec	Purity	Prec	Purity	Prec	Purity
Algorithm 1 $\tau = 1$	0.855	0.807	0.900	0.822	0.879	0.481	0.996	0.502
Algorithm 1 $\tau = \frac{4}{3}$	0.856	0.824	0.906	0.834	0.895	0.491	0.986	0.502
Algorithm 1 $\tau = \frac{5}{3}$	0.860	0.843	0.912	0.851	0.902	0.488	0.992	0.501
Algorithm 1 $\tau = 2$	0.864	**0.869**	0.929	0.848	**0.918**	0.495	**0.998**	0.503
Algorithm 2	0.856	0.851	**0.936**	**0.854**	0.879	0.493	0.936	0.503
MK	**0.886**	0.828	0.914	0.832	0.879	0.488	0.982	0.502
Malkomes	0.862	0.790	0.918	0.841	0.902	0.501	0.992	0.504
DYW	0.853	0.813	0.912	0.850	0.899	**0.502**	0.981	**0.510**

Table 2: Precision and Purity on real datasets (k-means with outliers)

Datasets	Shuttle		Kddcup		Covtype		Poking Hand	
Measure	Prec	Purity	Prec	Purity	Prec	Purity	Prec	Purity
Algorithm 3 $\tau = 1$	0.857	0.883	0.948	0.988	0.879	0.507	0.982	0.501
Algorithm 3 $\tau = \frac{4}{3}$	0.852	0.922	0.962	0.986	0.889	0.522	0.996	0.502
Algorithm 3 $\tau = \frac{5}{3}$	0.855	0.914	0.955	0.989	**0.913**	0.527	**0.999**	0.505
Algorithm 3 $\tau = 2$	0.861	**0.944**	**0.972**	0.991	0.906	0.515	**0.999**	0.506
Algorithm 4	0.856	0.867	0.927	0.982	0.906	0.523	**0.999**	**0.510**
k-means−−	**0.872**	0.842	0.915	0.985	0.879	0.537	**0.999**	0.502
LocalSearch	0.857	0.915	0.929	0.989	0.876	**0.541**	0.998	0.501
DataSummary	0.857	0.920	0.865	**0.994**	0.880	0.512	0.990	0.501

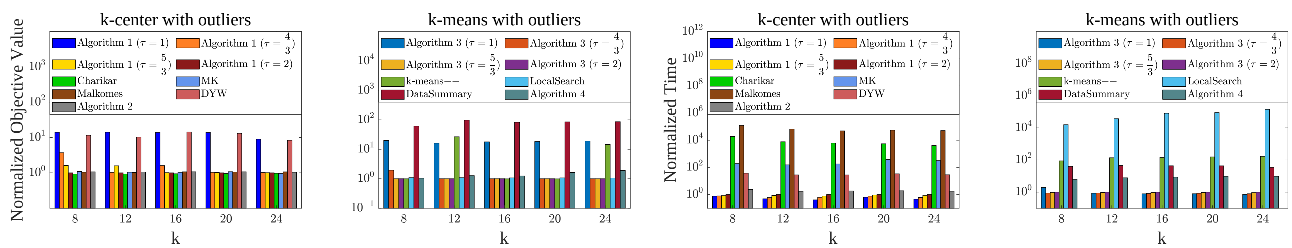

Figure 8: The normalized objective values and running times on the synthetic datasets with varying k (we set $z = 2\%n$ and $\epsilon_1/\epsilon_2 = 2$).

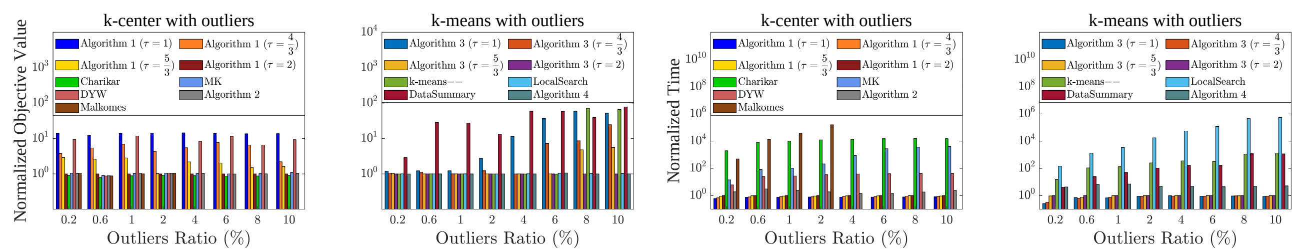

Figure 9: The normalized objective values and running times on the synthetic datasets with varying z/n (we set $k = 4$ and $\epsilon_1/\epsilon_2 = 2$). We did not run Malkomes for $z/n \geq 4\%$, since it is very slow for large z.

falls in V with probability τ, we know that the sample S contains at least one element from V with probability $1-(1-\tau)^{|S|}$. Therefore, if we want $1-(1-\tau)^{|S|} \geq 1-\eta$, $|S|$ should be at least $\frac{\log 1/\eta}{\log 1/(1-\tau)} \leq \frac{1}{\tau} \log \frac{1}{\eta}$.

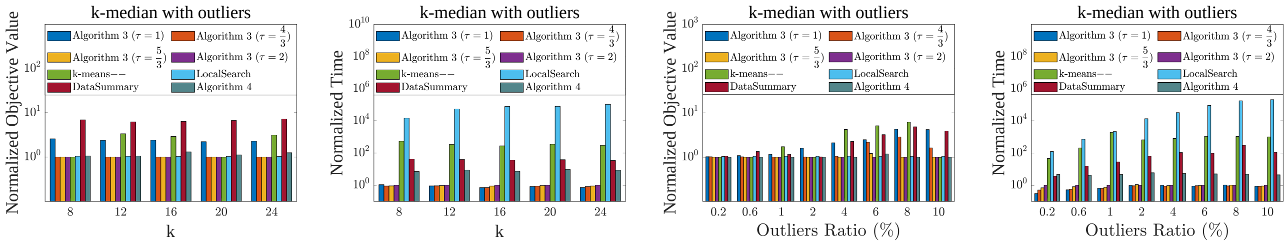

Figure 10: The normalized objective values and running times on the synthetic datasets with varying k (the left two figures) and varying z/n (the right two figures). We adopt the same parameter settings as in Figure 8 and Figure 9 respectively.

(ii) can be proved by using the Chernoff bound [1]. Define $|S|$ random variables $\{y_1, \cdots, y_{|S|}\}$: for each $1 \leq i \leq |S|$, $y_i = 1$ if the i-th sampled element falls in V, otherwise, $y_i = 0$. So $E[y_i] = \tau$ for each y_i. As a consequence, we have

$$\mathbf{Pr}(\big|\sum_{i=1}^{|S|} y_i - \tau|S|\big| \leq \delta\tau|S|) \geq 1 - 2e^{-\frac{\delta^2\tau}{3}|S|}. \tag{34}$$

If $|S| \geq \frac{3}{\delta^2\tau}\log\frac{2}{\eta}$, with probability at least $1-\eta$, $\big|\sum_{i=1}^{|S|} y_i - \tau|S|\big| \leq \delta\tau|S|$ (*i.e.*, $\big||S \cap V| - \tau|S|\big| \leq \delta\tau|S|$). □

B THE EXTENSIONS

(i). The results of Theorem 3 and Theorem 4 can be easily extended to k-median clustering with outliers in Euclidean space by using almost the same idea, where the only difference is that we can directly use triangle inequality in the proofs (*e.g.*, the inequality (15) is replaced by $||\tilde{q} - h_{\tilde{j}_q}|| \leq ||\tilde{q} - q|| + ||q - h_{j_q}||$). As a consequence, the coefficients α and β are reduced to be $\big(1 + (1+c)\frac{1+\delta}{1-\delta}\big)$ and $(1+c)\frac{1+\delta}{1-\delta}$ in Theorem 3, respectively. Similarly, α and β are reduced to be $\big(1 + (1+c)\frac{t}{t-1}\frac{1+\delta}{1-\delta}\big)$ and $(1+c)\frac{t}{t-1}\frac{1+\delta}{1-\delta}$ in Theorem 4.

(ii). To solve the metric k-median/means clustering with outliers problems for a given instance (X, μ), we should keep in mind that the cluster centers can only be selected from the vertices of X. However, the optimal cluster centers $O^* = \{o^*_1, \cdots, o^*_k\}$ may not be contained in the sample S, and thus we need to modify our analysis slightly. We observe that the sample S contains a set O' of vertices close to O^* with certain probability. Specifically, for each $1 \leq j \leq k$, there exists a vertex $o'_j \in O'$ such that $\mu(o'_j, o^*_j) \leq O(1) \times \frac{1}{|C^*_j|}\sum_{p \in C^*_j}\mu(p, o^*_j)$ (or $\big(\mu(o'_j, o^*_j)\big)^2 \leq O(1) \times \frac{1}{|C^*_j|}\sum_{p \in C^*_j}\big(\mu(p, o^*_j)\big)^2$) with constant probability (this claim can be easily proved by using the Markov's inequality). Consequently, we can use O' to replace O^* in our analysis, and achieve the similar results as Theorem 3 and Theorem 4.

REFERENCES

[1] Noga Alon and Joel H Spencer. 2004. *The probabilistic method.* John Wiley & Sons.

[2] David Arthur and Sergei Vassilvitskii. 2007. k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 1027–1035.

[3] Jordan T. Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. 2020. Deep Batch Active Learning by Diverse, Uncertain Gradient Lower Bounds. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

[4] Pranjal Awasthi and Maria-Florina Balcan. 2014. Center based clustering: A foundational perspective. (2014).

[5] Aditya Bhaskara, Sharvaree Vadgama, and Hong Xu. 2019. Greedy Sampling for Approximate Clustering in the Presence of Outliers. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (Eds.). 11146–11155.

[6] Yonatan Bilu and Nathan Linial. 2012. Are Stable Instances Easy? *Combinatorics, Probability & Computing* 21, 5 (2012), 643–660.

[7] Mihai Bădoiu, Sariel Har-Peled, and Piotr Indyk. 2002. Approximate clustering via core-sets. In *Proceedings of the ACM Symposium on Theory of Computing (STOC)*. 250–257.

[8] Emmanuel J. Candès, Xiaodong Li, Yi Ma, and John Wright. 2011. Robust principal component analysis? *J. ACM* 58, 3 (2011), 11:1–11:37.

[9] Matteo Ceccarello, Andrea Pietracaprina, and Geppino Pucci. 2019. Solving k-center Clustering (with Outliers) in MapReduce and Streaming, almost as Accurately as Sequentially. *PVLDB* 12, 7 (2019), 766–778.

[10] Deeparnab Chakrabarty, Prachi Goyal, and Ravishankar Krishnaswamy. 2016. The Non-Uniform k-Center Problem. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*. 67:1–67:15.

[11] Varun Chandola, Arindam Banerjee, and Vipin Kumar. 2009. Anomaly detection: A survey. *ACM Computing Surveys (CSUR)* 41, 3 (2009), 15.

[12] Moses Charikar, Samir Khuller, David M Mount, and Giri Narasimhan. 2001. Algorithms for facility location problems with outliers. In *Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 642–651.

[13] Moses Charikar, Liadan O'Callaghan, and Rina Panigrahy. 2003. Better streaming algorithms for clustering problems. In *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*. ACM, 30–39.

[14] Sanjay Chawla and Aristides Gionis. 2013. k-means−: A unified approach to clustering and outlier detection. In *Proceedings of the 2013 SIAM International Conference on Data Mining*. SIAM, 189–197.

[15] Jiecao Chen, Erfan Sadeqi Azer, and Qin Zhang. 2018. A Practical Algorithm for Distributed Clustering and Outlier Detection. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*. 2253–2262.

[16] Ke Chen. 2008. A constant factor approximation algorithm for k-median clustering with outliers. In *Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 826–835.

[17] Cody Coleman, Christopher Yeh, Stephen Mussmann, Baharan Mirzasoleiman, Peter Bailis, Percy Liang, Jure Leskovec, and Matei Zaharia. 2020. Selection via Proxy: Efficient Data Selection for Deep Learning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

[18] J. A. Cuesta-Albertos, A. Gordaliza, and C. Matran. 1997. Trimmed k-Means: An Attempt to Robustify Quantizers. *The Annals of Statistics* 25, 2 (1997), 553–576.

[19] Artur Czumaj and Christian Sohler. 2004. Sublinear-time approximation for clustering via random sampling. In *International Colloquium on Automata, Languages, and Programming*. Springer, 396–407.

[20] Sanjoy Dasgupta. 1999. Learning Mixtures of Gaussians. In *40th Annual Symposium on Foundations of Computer Science, FOCS '99, 17-18 October, 1999, New York, NY, USA*. IEEE Computer Society, 634–644.

[21] Sanjoy Dasgupta and Anupam Gupta. 2003. An elementary proof of a theorem of Johnson and Lindenstrauss. *Random Structures & Algorithms* 22, 1 (2003), 60–65.

[22] Amit Deshpande, Praneeth Kacham, and Rameshwar Pratap. 2020. Robust k-means++. In *Proceedings of the Thirty-Sixth Conference on Uncertainty in Artificial Intelligence, UAI 2020, virtual online, August 3-6, 2020 (Proceedings of Machine Learning Research)*, Ryan P. Adams and Vibhav Gogate (Eds.), Vol. 124. AUAI Press, 799–808.

[23] Hu Ding and Jinhui Xu. 2014. Sub-linear time hybrid approximations for least trimmed squares estimator and related problems. In *Proceedings of the International Symposium on Computational geometry (SoCG)*. 110.

[24] Hu Ding, Haikuo Yu, and Zixiu Wang. 2019. 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*. 40:1–40:16.

[25] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml

[26] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise.. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*. 226–231.

[27] Dan Feldman and Michael Langberg. 2011. A unified framework for approximating and clustering data. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*. ACM, 569–578.

[28] Zachary Friggstad, Kamyar Khodamoradi, Mohsen Rezapour, and Mohammad R Salavatipour. 2018. Approximation schemes for clustering with outliers. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 398–414.

[29] Teofilo F Gonzalez. 1985. Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science* 38 (1985), 293–306.

[30] Sudipto Guha, Yi Li, and Qin Zhang. 2017. Distributed Partial Clustering. In *Proceedings of the 29th ACM Symposium on Parallelism in Algorithms and Architectures*. ACM, 143–152.

[31] Shalmoli Gupta. 2018. *Approximation algorithms for clustering and facility location problems*. Ph.D. Dissertation. University of Illinois at Urbana-Champaign.

[32] Shalmoli Gupta, Ravi Kumar, Kefu Lu, Benjamin Moseley, and Sergei Vassilvitskii. 2017. Local search methods for k-means with outliers. *Proceedings of the VLDB Endowment* 10, 7 (2017), 757–768.

[33] Dorit S Hochbaum and David B Shmoys. 1985. A best possible heuristic for the k-center problem. *Mathematics of operations research* 10, 2 (1985), 180–184.

[34] Lingxiao Huang, Shaofeng Jiang, Jian Li, and Xuan Wu. 2018. Epsilon-Coresets for Clustering (with Outliers) in Doubling Metrics. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 814–825.

[35] Sungjin Im, Mahshid Montazer Qaem, Benjamin Moseley, Xiaorui Sun, and Rudy Zhou. 2020. Fast Noise Removal for k-Means Clustering. In *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy] (Proceedings of Machine Learning Research)*, Silvia Chiappa and Roberto Calandra (Eds.), Vol. 108. PMLR, 456–466.

[36] Anil K Jain. 2010. Data clustering: 50 years beyond K-means. *Pattern recognition letters* 31, 8 (2010), 651–666.

[37] Tapas Kanungo, David M Mount, Nathan S Netanyahu, Christine D Piatko, Ruth Silverman, and Angela Y Wu. 2004. A local search approximation algorithm for k-means clustering. *Computational Geometry* 28, 2-3 (2004), 89–112.

[38] Ravishankar Krishnaswamy, Shi Li, and Sai Sandeep. 2018. Constant approximation for k-median and k-means with outliers via iterative rounding. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*. ACM, 646–659.

[39] Shi Li and Xiangyu Guo. 2018. Distributed k-Clustering for Data with Heavy Noise. In *Advances in Neural Information Processing Systems*. 7849–7857.

[40] Shi Li and Ola Svensson. 2016. Approximating k-median via pseudo-approximation. *SIAM J. Comput.* 45, 2 (2016), 530–547.

[41] Stuart Lloyd. 1982. Least squares quantization in PCM. *IEEE transactions on information theory* 28, 2 (1982), 129–137.

[42] Gustavo Malkomes, Matt J Kusner, Wenlin Chen, Kilian Q Weinberger, and Benjamin Moseley. 2015. Fast distributed k-center clustering with outliers on massive data. In *Advances in Neural Information Processing Systems*. 1063–1071.

[43] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to information retrieval*. Cambridge University Press. https://doi.org/10.1017/CBO9780511809071

[44] Richard Matthew McCutchen and Samir Khuller. 2008. Streaming Algorithms for k-Center Clustering with Outliers and with Anonymity. In *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques*. Springer, 165–178.

[45] Ramgopal R Mettu and C Greg Plaxton. 2004. Optimal time bounds for approximate clustering. *Machine Learning* 56, 1-3 (2004), 35–60.

[46] Adam Meyerson, Liadan O'callaghan, and Serge Plotkin. 2004. A k-median algorithm with running time independent of data size. *Machine Learning* 56, 1-3 (2004), 61–87.

[47] Nina Mishra, Dan Oblinger, and Leonard Pitt. 2001. Sublinear time approximate clustering. In *Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 439–447.

[48] Rafail Ostrovsky, Yuval Rabani, Leonard J Schulman, and Chaitanya Swamy. 2012. The effectiveness of Lloyd-type methods for the k-means problem. *Journal of the ACM (JACM)* 59, 6 (2012), 28.

[49] Lionel Ott, Linsey Pang, Fabio T Ramos, and Sanjay Chawla. 2014. On integrated clustering and outlier detection. In *Advances in neural information processing systems*. 1359–1367.

[50] Tim Roughgarden. 2019. Beyond worst-case analysis. *Commun. ACM* 62, 3 (2019), 88–96.

[51] Peter Sanders. 2014. Algorithm Engineering for Big Data. In *44. Jahrestagung der Gesellschaft für Informatik, Big Data - Komplexität meistern, INFORMATIK 2014, Stuttgart, Germany, September 22-26, 2014 (LNI)*, Erhard Plödereder, Lars Grunske, Eric Schneider, and Dominik Ull (Eds.), Vol. P-232. GI, 57.

[52] Ozan Sener and Silvio Savarese. 2018. Active Learning for Convolutional Neural Networks: A Core-Set Approach. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.