Robust Algorithms for Online k-means Clustering

Aditya Bhaskara BHASKARAADITYA@GMAIL.COM
School of Computing, University of Utah

Aravinda Kanchana Ruwanpathirana KANCHANA@CS.UTAH.EDU
School of Computing, University of Utah

Editors: Aryeh Kontorovich and Gergely Neu

Abstract

In the online version of the classic k-means clustering problem, the points of a dataset $u_1, u_2, \ldots$ arrive one after another in an arbitrary order. When the algorithm sees a point, it should either add it to the set of centers, or let go of the point. Once added, a center cannot be removed. The goal is to end up with set of roughly k centers, while competing in k-means objective value with the best set of k centers in hindsight.

Online versions of k-means and other clustering problem have received significant attention in the literature. The key idea in many algorithms is that of adaptive sampling: when a new point arrives, it is added to the set of centers with a probability that depends on the distance to the centers chosen so far. Our contributions are as follows:

1. We give a modified adaptive sampling procedure that obtains a better approximation ratio (improving it from logarithmic to constant).

2. Our main result is to show how to perform adaptive sampling when data has outliers ($\gg k$ points that are potentially arbitrarily far from the actual data, thus rendering distance-based sampling prone to picking the outliers).

3. We also discuss lower bounds for k-means clustering in an online setting.

Keywords: Online algorithms, k-means clustering, Robust algorithms

1. Introduction

Clustering data is one of the fundamental subroutines in the analysis of large data. The general goal is to partition the data into clusters so that points within a cluster are "more similar" to one another, compared to points in different clusters. This intuitive requirement can be formalized in many ways. Some popular notions include k-means, k-center, facility location, hierarchical clustering, correlation clustering, etc. Each formulation has its merits, and the choice between them largely depends on the application and the data itself. (E.g., see Dasgupta, 2016; Hastie et al., 2009).

The subject of this paper is k-means clustering. Here, we are given an integer k, and the goal is to partition the data into k clusters, so as to minimize the sum of squared distances from the points to the corresponding *cluster centers*. In the case of points in a Euclidean space, the cluster center is simply the (empirical) mean of the points in a cluster. In general metric spaces, we need to designate one point in a cluster to be the center. k-means clustering is well-studied from an algorithmic and a complexity perspective. It is known to be APX-hard (i.e., hard to approximate to a factor > 1 assuming $\mathsf{P} \neq \mathsf{NP}$) (Bahmani et al., 2012; Krishnaswamy et al., 2018). The first constant factor

approximation is due to (Kanungo et al., 2004), who gave an algorithm based on local search. The best known approximation algorithm is due to (Ahmadian et al., 2017).

On the practical side, there are many heuristics developed for the k-means problem. Lloyd's algorithm (Lloyd, 1982) (popularly known simply as the "k-means algorithm") is an iterative EM-style procedure that works quite well in practice. A better initialization for Lloyd's algorithm was proposed by (Arthur and Vassilvitskii, 2007), based on *adaptive sampling* (an idea which will feature prominently in our results). In order to deal with much larger datasets, distributed algorithms for k-means have been extensively studied (see Ene et al., 2011; Balcan et al., 2013; Bateni et al., 2014, and references therein).

Another well-studied setting for clustering problems is the *data streaming* model. Here, the points of a dataset arrive one after another in arbitrary order. The algorithm must, in the end, arrive at an approximately optimal solution to the k-means problem on the full dataset. The trivial solution is to store all the points, and use an approximation algorithm in the end. The goal in streaming algorithms is to do much better than this, in terms of both memory and time complexity. Specifically, the goal is to use only roughly $O(k)$ memory, use as little computation as possible at each step (when a point arrives), and in the end, obtain a constant factor approximation. Surprisingly, this turns out to be possible, using a little more than k memory (see Guha et al., 2001; Ene et al., 2011).

Online model. We consider the *online* model of computation, which is more restrictive than streaming. Here, points arrive one after another in an arbitrary order. Once a point arrives, the algorithm must decide to keep it, or let it go (forever). Once a point is kept, it cannot be removed (thus the decision making is done in a one-shot manner). The goal is similar: we wish to maintain roughly k centers, whilst competing with the clustering objective for the best k centers in hindsight. The online model is well studied for problems such as hiring (the so-called secretary problem, see Babaioff et al., 2009), as well as submodular optimization (see Buchbinder et al., 2019). In the context of clustering and facility location, online algorithms were first studied in the classic work of (Meyerson, 2001). The motivation here was to open facilities so as to serve *demands* that arrive online. In this context, it is also natural that once facility/clusters are opened, they cannot be closed/moved without a cost. Later works of (Charikar et al., 2004) and (Liberty et al., 2016) studied online clustering from an approximation perspective.

The work of (Liberty et al., 2016), which is inspired by the *adaptive sampling* ideas of (Meyerson, 2001) (see also Ostrovsky et al., 2013; Arthur and Vassilvitskii, 2007), shows that assuming one has an estimate of the optimum k-means error (up to a constant factor), going over the points and sampling them with probability proportional to the squared-distance to the points chosen so far (scaled by the guess for the optimum) results in (a) choosing only $O(k \log n)$ points (assuming that the guess for the optimum error was close), and (b) achieving an objective value that is at most $O(\log n) \cdot \mathrm{OPT}$.

Our contributions improve the understanding of the online k-means problem along many axes. First, we improve the approximation factor of the k-means objective cost and study intrinsic limitations in the online model. Next, we consider the problem in the presence of *outliers* (formally defined below). Suppose we have $z \gg k \log n$ points that are outliers. Now, adaptive sampling can choose these points with high probability, resulting in $\Theta(z)$ points being chosen. We show, via a carefully designed algorithm, how we can mitigate the effect of outliers.

1.1. Our results

Our first result is a new online algorithm for k-means clustering. It is based on adaptive sampling, but done in phases. The algorithm achieves a $\log n$ factor improvement in the approximation ratio compared to the prior work of (Liberty et al., 2016).

Theorem 1 *Suppose the points $V \subset \mathbb{R}^d$ of a dataset arrive in an online manner. Let $k \geq 1$, and $\xi > 0$ and $\epsilon \in (0,1)$ be given parameters. k is the desired number of clusters and ξ is the desired clustering cost. Then there exists an algorithm, that after seeing u_i, decides to either ignore it or add it to a selection set C (which starts off empty), with the following properties:*

1. *The processing time per point is $\widetilde{O}(kd)$.*
2. *In the end $|C| \leq O\left(\frac{k}{\epsilon}\log n \cdot \max\left(1, \log\frac{\text{OPT}}{\xi}\right)\right)$, where OPT is the optimum k-means objective value for the full instance.*
3. *The k-means objective cost, defined as $\sum_{u\in V} d^2(u,C)$, is $\leq O(1)\cdot \text{OPT} + \epsilon\xi$.*

Thus we obtain a bi-criteria approximation (use $k' > k$ centers), while obtaining a constant approximation to the objective. As stated earlier, (Liberty et al., 2016) obtain a similar result, albeit with a $O(\log n)(\text{OPT}+\xi)$ bound on the objective. We obtain this improvement via an idea introduced in the recent work (Bhaskara et al., 2019) on online PCA. Adapted to our setting, it allows us to analyze clustering cost in a novel way: when bounding the distance from a point u to the centers C, we take into account not just the centers that have been chosen *so far*, but also the ones that are to appear later.

We also note that the value of ξ plays an important role in the guarantees. As such, it is the "desired" objective value (one might think of it as approaching zero). However, if ξ is too small, we pay an additional $\log\frac{\text{OPT}}{\xi}$ in the bound for $|C|$. As long as we pick it to be $\text{OPT}/\text{poly}(n)$, this will only add a logarithmic term, and we thus do not expect it to be significant in practice. More formally, ξ can be viewed as the permissible "additive" error in the clustering cost.

Our next (and main) result deals with k-means clustering, when the input also contains a certain number of outliers. In this case, the goal is to discard some points as outliers, and minimize the k-means objective on the remaining points. The problem, which we now define formally, has received a lot of attention in the *offline* case.

Formulating clustering with outliers. Let $\text{KM-OPT}(X)$ denote the optimum value of the k-means objective on a set of points X. Given a set of points V and a bound on the number of outliers z, the goal in clustering with outliers is to partition $V = V_{\text{in}} \cup V_{\text{out}}$, such that $|V_{\text{out}}| \leq z$, and $\text{KM-OPT}(V_{\text{in}})$ is minimized.

The early work of (Charikar et al., 2001) studied the problem of k-median and facility location in a similar setup. The recent work of (Gupta et al., 2017) gave a *local search* based algorithm for the problem. Both these algorithms give *bi-criteria* approximations (where $> z$ points are discarded as outliers.). In practice, this corresponds to declaring a small number of the *inliers* as outliers. In applications where the true clusters are sufficiently large, these guarantees are still valuable. The recent result of (Krishnaswamy et al., 2018) (and the earlier result of Chen, 2008, for k-median) made considerable progress, giving constant factor (~ 50) approximation algorithms without violating the constraint on z. These algorithms are intricate and are based on LP relaxations.

Our main result shows that one can obtain approximation guarantees for k-means with outliers even in an *online* model, via a slight variant of adaptive sampling. Intuitively, this is challenging because if encounter a point that is far from all the previously stored points, we are not sure if it represents a start of a new cluster, or if it is simply an outlier. Our result is the following:

Theorem 2 *Suppose the points $V \subset \mathbb{R}^d$ of a dataset arrive in an online manner. Let $k \geq 1$, and $\xi > 0$, and $L > 1$ and $\epsilon \in (0,1)$ be given parameters. Let V_{in} be the set of inlier points in V, k the desired number of clusters and ξ the desired clustering cost over the inlier points in V. Then there exists an algorithm, that after seeing u_i, decides to either ignore it, or add it to a selection set C (which starts off empty), or add it to the outlier set M with the following properties:*

1. *The processing time per point is $\widetilde{O}(kd)$.*
2. *In the end $|C| \leq O\left(\frac{k}{\epsilon}(\log n + L) \cdot \max\left(1, \log \frac{\text{OPT}}{\xi}\right)\right)$, where OPT is the optimum k-means objective value over the inlier points.*
3. *The k-means objective cost over the inlier points not marked as outliers, defined as $\sum_{u \in V_{in} \setminus M} d^2(u, C)$, is $\leq O(1) \cdot \text{OPT} + \epsilon\xi$.*
4. *In the end $|M| \leq O\left(z\left(\frac{\log n}{L} + 1\right) \cdot \max\left(1, \log \frac{\text{OPT}}{\xi}\right)\right)$*

The algorithm has a parameter L, which features in the bounds. If we have an estimate for $\log n$ (up to constants would suffice), the terms $(\log n)/L$ would reduce to $O(1)$. Note also that the algorithm discards more than z points as outliers. Assuming that L, ξ are good guesses for $\log n$, OPT respectively, the number of points discarded is $O(z)$.

Lower bounds. Finally, it is natural to ask if our algorithms' dependence on the various parameters, k, z, etc. can be improved. Lower bounds have been extensively studied for online clustering and facility location problems (Meyerson, 2001; Fotakis, 2007; Liberty et al., 2016). For k-means clustering, the first question is if we can obtain an online algorithm that achieves a constant factor approximation to the objective, while choosing only $O(k)$ points in the end. It is easy to see that this is impossible if we do not know either the number of points or an estimate of the optimum objective value in advance: consider points in one dimension, where the ith point is c^i, for $i = 0, 1, \ldots, c > 2$ and $k = 1$. The optimum solution is to choose the last point, while an online algorithm, to remain competitive, must end up keeping every point. (See Liberty et al., 2016, for a formalization of this.) The recent work of (Moshkovitz, 2019) showed also established similar guarantees.

Our main result is that *even when* the optimum objective value is known up to a constant factor (as is the case in our algorithm), we cannot avoid an overhead of $\log n$ in the number of centers. This shows that the dependence in our algorithm is essentially tight. Formally, we show the following:

Theorem 3 *For any constant C and parameter $r \in \mathbf{N}^+$, there is no randomized online algorithm which, with no prior knowledge of the number of points, can obtain an expected approximation ratio of $\frac{C^2}{16}$ for the objective, using fewer than r centers. Moreover, this holds even when the instance size n is in the range $[C^{8r}, C^{16r}]$ (in which case, $\frac{\log n}{8 \log C} \geq r \geq \frac{\log n}{16 \log C}$).*

In the instances we use for the lower bound, the optimum value is always $\Theta(r)$, thus we can assume that we know it up to a constant; however, the instance size (total number of points) is not known exactly.

2. Notation and outline

We will use the following basic notation throughout the paper: given a set S and a point v, we denote $d^2(v, S) = \min_{x \in S} \|v - x\|^2$. If $S = \emptyset$, $d^2(v, S)$ is defined to be ∞.

For consistency, we use C to denote the subset of points chosen by the algorithm (these are the candidate cluster centers), and V to denote the original set of points. In many places, we use the notation C^* to denote the optimal centers for the corresponding problem. The quantity OPT will always be used to denote the optimum objective value.

2.1. Outline of the paper

The paper is structured as follows. In Section 3, we present the algorithm for online k-means (without outliers). This will illustrate our key idea of adaptive sampling in phases. We first present a $\log n$ approximation to the objective (which is similar to the bound from prior work), and then in § 3.2, show how to improve this to a constant factor. A similar approach will be followed for the algorithm when we have outliers (Section 4). Here we have a more involved notion of phases, and the analysis is deferred to the Appendix.

Assumptions on ξ. On a technical note, throughout Sections 3 and 4, we assume that the target objective value (ξ in the statements of Theorem 1 and Theorem 2) satisfies $\xi \geq$ OPT. This assumption will be removed in Section 5, which will then complete the proofs of Theorems 1 and 2.

3. Online k-means clustering

In this section, we present an algorithm for the online k-means clustering problem. The algorithm assumes the knowledge of a guess ξ for the optimum error, and the parameter k. (It does not assume any knowledge of n, the total number of points.)

Algorithm 1: Online k-means clustering

Input: A set of points V that arrive one by one, guess ξ for the optimum error, parameter k.
Output: A subset C of the points (to be cluster centers).
Initialize $C_{\text{pre}} = \emptyset$, $C_{\text{cur}} = \emptyset$ and running sum $\alpha = 0$.;
while *points u arrive* **do**
  Let $p_u := \min(\frac{k \cdot d^2(u, C_{\text{pre}})}{\xi}, 1)$. ;
  With probability p_u, add u to C_{cur} ;
  Increment $\alpha \leftarrow \alpha + p_u$;
  If $\alpha \geq 1$, set $C_{\text{pre}} := C_{\text{pre}} \cup C_{\text{cur}}$ and reset $\alpha = 0, C_{\text{cur}} = \emptyset$ (start new phase) ;
end
Output $C = C_{\text{pre}} \cup C_{\text{cur}}$. ;

Description of Algorithm 1. The algorithm processes the points in phases. In every phase, the algorithm builds a set C_{cur} (chosen centers in the current phase), by adding each point u to C_{cur} with probability p_u that depends on the distance from u to the set C_{pre} (which is the set of centers we have accumulated until the beginning of the phase). Once the sum of the probabilities p_u exceeds 1, the phase ends, and C_{cur} is added to C_{pre}.

3.1. An $O(\log n)$ approximation to the objective

We start by showing the following theorem about the algorithm.

Theorem 4 *Suppose the points of V arrive in an online manner, and suppose that the guess ξ satisfies $\xi \geq \sum_{v \in V} d^2(v, C^*)$, where C^* is the optimal set of cluster centers. Then the algorithm satisfies: w.p. at least $\frac{4}{5}$,*

1. *The number of phases and consequently the number of chosen centers is $\leq O(k \log n)$.*
2. *The k-means objective cost for the output centers C is $\leq \xi \cdot O(\log n)$.*

Definition 5 *Let $\{u_i\}_{i=1}^r$ be the points in a phase and let C_{cur} be the set of selected points. A phase is said to be successful if:*

1. *At least one point is selected, i.e., $|C_{cur}| \geq 1$.*
2. *For the optimal set of clusters C^*, we have*

$$\sum_{u_i \in C_{cur}} \frac{1}{p_{u_i}} d^2(u_i, C^*) < 4 \sum_{i \in [r]} d^2(u_i, C^*)$$

3. *For any point $u_i \in C_{cur}$, we have $p_{u_i} \geq 1/n^2$.*

Lemma 6 *A phase is successful with probability $\geq \frac{1}{4}$ (over the choice of C_{cur}).*

Proof The proof is by simply bounding the probabilities of (1), (2) and (3) in the definition of a successful phase. The details are deferred to § D.1. ■

The next lemma is the key step in the entire argument.

Lemma 7 *The number of successful phases is $O(k \log n)$.*

Proof We will prove the lemma via contradiction. Suppose we have t successful phases, and suppose we choose one point from each successful phase. Let these points be $v_1, v_2, \ldots, v_t$.

By definition, the probability values $p_{v_i} \in (1/n^2, 1)$. Thus, by dividing $(1/n^2, 1)$ into $2 \log n$ intervals of the form $(1/2^{i+1}, 1/2^i]$, we obtain that there exists some q such that $t/(2 \log n)$ of the v_i lie in the interval $(q, 2q]$. Let I denote the indices of these points.

Now, for any $i, j \in I$, we claim that $d(v_i, v_j)^2 > \frac{\xi q}{k}$. This claim holds because of the following: suppose $i < j$. Thus in the jth phase, v_i was already present in C_{pre}. Thus, since $p_{v_j} \leq \frac{k d^2(v_j, C_{\text{pre}})}{\xi}$, we have that $p_{v_j} \leq \frac{k d(v_i, v_j)^2}{\xi}$. Since $p_{v_j} > q$, the claim follows.

Now, consider balls of squared-radius $\frac{\xi q}{4k}$ around each of the points v_i (for $i \in I$), and denote these balls by B_i (respectively). The claim above implies that for any $i, j \in I$ with $i \neq j$, $B_i \cap B_j = \emptyset$.

Finally, we claim that $|I| \leq 33k$. Suppose, for the sake of contradiction, that $|I| > 33k$. Then, since the balls are all disjoint, we must have that for at least $32k$ of the balls B_i, $C^* \cap B_i = \emptyset$ (where C^* is the optimal set of centers). Let J denote the set of i for which this holds. This means that for any $i \in J$, $d^2(v_i, C^*) > \frac{\xi q}{4k}$.

Now, let us denote by S_i the set of original points u that are in phase i. Since i is a successful phase, we have (by condition (2) of the definition), that

$$4 \sum_{u \in S_i} d^2(u, C^*) \geq \frac{1}{p_i} d^2(v_i, C^*) > \frac{1}{2q} \frac{\xi q}{4k} = \frac{\xi}{8k}.$$

Thus, if we have $|J| \geq 32k$, this would let us conclude that the sum of the distances to the optimal centers from the points in those phases is $> \xi$, a contradiction to the assumption that the optimum objective value is $\leq \xi$.

This implies that $|J| \leq 32k$, and subsequently that $|I| \leq 33k$, implying that the number of phases t satisfies $t \leq 66k \log n$. ■

The two lemmas above now let us bound the number of phases in the algorithm.

Lemma 8 *For any $\delta > 0$, the total number of phases is $O(k \log n + \log(1/\delta))$, with probability at least $1 - \delta$.*

The proof is intuitively easy, but needs some care since the points chosen in a phase depend on the previously chosen points. This is deferred to Appendix E.

We can now complete the proof of Theorem 4.

Proof [of Theorem 4] First, we bound the size of C, the selected points, using the bound on the number of phases. Suppose the number of phases is r. In each phase, the selection probabilities add up to a quantity between 1 and 2. Thus, we have that $\Pr[|C| > 20r] \leq 1/10$. Setting $\delta = 1/10$ in Lemma 8, we get that $r \leq O(k \log n)$ with probability $\geq 9/10$. Combining these, we have that $|C| \leq O(k \log n)$ with probability $\geq 4/5$.

Next, we need to bound the total cost $\sum_u d^2(u, C)$, where the sum ranges over all the points. To bound this, we will bound the sum in each phase. To this end, let $u_1, \ldots, u_r$ be the points in a phase, and let C_{pre} the set of chosen points at the start of the phase. We consider two cases.

Case 1: there is no i such that $d^2(u_i, C_{\text{pre}}) > \frac{\xi}{k}$. In this case,

$$\sum_{i \in [r]} d^2(u, C) \leq \sum_{i \in [r]} d^2(u, C_{\text{pre}}) \leq \sum_{i \in [r]} p_{u_i} \cdot \frac{\xi}{k} \leq 2\frac{\xi}{k}.$$

The last inequality follows because the sum of the selection probabilities for points in any phase is ≤ 2

Case 2: there is an i such that $d^2(u_i, C_{\text{pre}}) > \frac{\xi}{k}$. In this case, i must be r (as the phase ends at that i). Furthermore, the point i is certainly included in C_{cur}, and hence is also in C. Thus, we have

$$\sum_{i \in [r]} d^2(u, C) = \sum_{i \in [r-1]} d^2(u, C),$$

which can then be bounded exactly as before. Thus in both cases, we have that $\sum_{i \in [r]} d^2(u, C) \leq 2\xi/k$. Combined with the bound on the number of phases, this completes the proof. ■

Algorithm 2: Online k-means clustering: An $O(1)$ Approximation

Input: A set of points V that arrive one by one, guess ξ, parameters k and $\epsilon > 0$.
Output: A set C of the cluster centers.
Initialize $C_{\text{pre}} = \emptyset$, $C_{\text{cur}} = \emptyset$ and $T = \emptyset$;
while *points u arrive* **do**
  Execute Algorithm 1 with input u; this updates C_{pre} and C_{cur}. ;
  With probability $p_u := \min(\frac{40k.d^2(u,C_{\text{pre}})}{\epsilon\xi}, 1)$, add u to T ;
end
Output $C = C_{\text{pre}} \cup T$. ;

3.2. $O(1)$ Approximation to the Objective

Description of the algorithm. When a point u arrives, we first run it through Algorithm 1 and update the C_{pre} and C_{cur} as before. But additionally, we maintain another set T. With probability $p_u = \min(\frac{40k\cdot d^2(u,C_{\text{pre}})}{\epsilon\xi}, 1)$, we add u to T. However, note that T is *not* used in any way in the updates of C_{pre} and C_{cur}. The final output set is the union of C_{pre} and T. (See Algorithm 2.)

3.3. Analysis

We show the following theorem about the algorithm.

Theorem 9 *Suppose the points of a V arrive in an online manner and suppose that the guess ξ satisfies $\xi \geq \sum_{v\in V} d^2(v, C^*)$. Then for any $\epsilon > 0$, the algorithm satisfies:*

1. *w.p. at least $\frac{7}{10}$, the number of chosen centers is $\leq O(\frac{k}{\epsilon}\log(n))$.*
2. *The k-means objective cost for the output centers C is $\leq O(1)\cdot \text{OPT} + \epsilon\xi$ where OPT is the optimum k-means objective value for the full instance.*

The main trick in the analysis is to move to a slightly different algorithm, one that can be viewed as a "two-pass" procedure, and analyze that instead. Observe that in line 2 of the algorithm, we do not use T in any way, and thus it proceeds precisely as in Algorithm 1. Let C_{first} denote the final value (after we have seen all the points) of $C_{\text{pre}} \cup C_{\text{cur}}$. Now, consider replacing the probability of adding v to T (line 2) to $\min\left(1, \frac{40k\cdot d^2(v,C_{\text{first}})}{\epsilon\xi}\right)$. This is no longer an online algorithm, but it is a two-pass algorithm. The key observation is that the probability of adding u to T in our algorithm is at least as large as the probability of adding u to T in the two pass algorithm. Thus, it suffices to bound the error of the two-pass algorithm.

This is done by using the following lemma.

Lemma 10 *Let S be any set of points in our metric space, and define $Z := \sum_{v\in V} d^2(v, S)$. Suppose T is formed by adding every element $v \in V$ independently with probability p_v that satisfies $p_v \geq ck\frac{d^2(v,S)}{Z}$. Then we have*

$$\mathbb{E}\left[\sum_{v\in V} d^2(v, S\cup T)\right] \leq 16\sum_{v\in V} d^2(v, C^*) + \frac{4Z}{c}. \tag{1}$$

Remark. The lemma is reminiscent of the classic lemma of (Frieze et al., 2004) on "norm sampling" (for matrix low rank approximation). While the statement for clustering is similar in spirit, we are not aware of its proof in the literature (nor does it follow from the result of Frieze et al., 2004, directly).

The proof of the lemma is somewhat involved, and is deferred to Section B. In analyzing the two-pass algorithm, we apply Lemma 10 with $S = C_{\text{first}}$. Then we show that the condition $p_v \geq ck\frac{d^2(v,S)}{Z}$ holds with $c = \frac{40Z}{\epsilon\xi}$. To see this, note that

$$p_v = \frac{40kd^2(v, C_{\text{first}})}{\epsilon\xi} = \frac{40Z}{\epsilon\xi}\frac{kd^2(v, C_{\text{first}})}{Z}.$$

Thus with probability at least $\frac{4}{5}$ (over only the second pass), denoting by T the set of points chosen in the second pass (using Markov's inequality following (1)),

$$\sum_{v\in V} d^2(v, C_{\text{first}} \cup T) \leq 160 \sum_{v\in V} d^2(v, C^*) + \frac{40Z}{c}.$$

This results in

$$\sum_{v\in V} d^2(v, C_{\text{first}} \cup T) \leq 160 \sum_{v\in V} d^2(v, C^*) + \epsilon\xi.$$

Next, consider the expected size of the final set. This time, we need to analyze Algorithm 2 and not the two-pass algorithm above. We have that

$$\mathbb{E}[|T|] = \sum \frac{40kd^2(v, C_{\text{pre}})}{\epsilon\xi}$$

By the bound on the total error from Theorem 4, we have that with probability $\geq \frac{4}{5}$,

$$\sum d^2(v, C_{\text{pre}}) \leq O(\xi \log(n))$$

Thus the above together with Markov's inequality, we have that with probability $\geq \frac{7}{10}$, $|T| \leq O(\frac{k}{\epsilon}\log(n))$. This completes the proof of Theorem 9.

4. Online clustering in the presence of outliers

In this section, we present our main result: an online algorithm for the k-means problem when the data has outliers. The algorithm assumes the knowledge of a guess ξ for the optimum error over the inliers, an upper bound z on the number of outliers, and the parameter k (see Section 1.1 for a discussion on the assumption on ξ). In the algorithm, $L \geq 1$ is an arbitrary constant (for best results, L must be a guess for $\log n$).

Description of Algorithm 3. The algorithm is broadly similar to Algorithm 1, and it processes the points in phases. When a point u arrives, we assign a sampling probability p_u, that is proportional to the distance of u to the selected points in all previous phases and not the current phase. However, the main difference now is that if the probability is above a threshold (kL/z), we handle them separately. In this case, its probability is brought down to kL/z, and we add the points to a different set. In the analysis, we introduce the notions of type A and type B points as well as phases. There are *two* running sums, α and β. If either of them exceeds 1, the phase ends.

Algorithm 3: Online k-means clustering

Input: A set of points V that arrive one by one, guess ξ for the optimum error over the inlier points, an upper bound z on the number of outliers, and parameters k and L.

Output: A set C of the cluster centers. Initialize $C_{\text{pre}} = \emptyset$, $C_{\text{cur}} = \emptyset$, $T_{\text{cur}} = \emptyset$ and running sums $\alpha = 0$, and $\beta = 0$

while *points* u *arrive* **do**
  Let $p_u := \min(\frac{k.d^2(u,C_{\text{pre}})}{\xi}, 1)$;
  if $p_u < \frac{k.L}{z}$ **then**
    With probability p_u, add u to C_{cur} ;
    Increment $\alpha \leftarrow \alpha + p_u$;
    If $\alpha \geq 1$, set $C_{\text{pre}} := C_{\text{pre}} \cup C_{\text{cur}} \cup T_{\text{cur}}$ and reset $\alpha = 0, C_{\text{cur}} = \emptyset$ and $\beta = 0, T_{\text{cur}} = \emptyset$ (start new phase) ;
  else
    Let $p_u := \frac{k.L}{z}$;
    With probability p_u, add u to T_{cur} (i.e., decide to pick u) ;
    Mark u as an outlier ;
    Increment $\beta \leftarrow \beta + p_u$;
    If $\beta \geq 1$, set $C_{\text{pre}} := C_{\text{pre}} \cup C_{\text{cur}} \cup T_{\text{cur}}$ and reset $\alpha = 0, C_{\text{cur}} = \emptyset$ and $\beta = 0, T_{\text{cur}} = \emptyset$ (start new phase) ;
  end
end
Output $C = C_{\text{pre}} \cup C_{\text{cur}} \cup T_{\text{cur}}$;

4.1. An $O(\log n)$ approximation

We start by showing the following theorem about Algorithm 3.

Theorem 11 *Suppose the points of V arrive in an online manner and let V_{in} be the set of inliers in V. Suppose that the guess ξ satisfies $\xi \geq \sum_{v \in V_{in}} d^2(v, C^*)$, where C^* is the optimal set of cluster centers. Then the algorithm, on seeing a point v_i either adds it to C, or ignores it, or marks it as an outlier such that in the end we have, w.p. at least $4/5$ and for any parameter $L \geq 1$:*

1. *the number of phases to be $\leq O(k \log n + kL)$.*
2. *the number of selected centers (or points) is $\leq O(k \log n + kL)$.*
3. *the number of points marked as outliers is $\leq z \cdot O(\frac{\log n}{L} + 1)$*
4. *the objective cost for the inlier points (ones not marked as outliers) is $\leq \xi \cdot O(\log n + L)$.*

Definition 12 (Type A, B) *First, we say that a* point *u is type A if $\frac{kd^2(u,C_{pre})}{\xi} < \frac{kL}{z}$, and we say that it is type B otherwise.*

Next, a phase *is said to be type A if it terminates because $\alpha \geq 1$. Else, if it terminates because of $\beta \geq 1$, we say that it is type B. (The very last phase can be classified as type A.)*

Outline of the argument. At an intuitive level, the algorithm labels points as type B (and handles them differently) as long as they are sufficiently far from the ones chosen so far. This is certainly a reasonable way to handle outliers (in fact, the threshold is chosen precisely to ensure that the number of type B phases, assuming only outliers are type B points, is roughly kL). The problem is now that inliers (e.g., when a cluster is starting to be formed) can be type B points. We need to ensure that there is still a sufficiently large probability that they will be chosen. This is done via a "win-win" argument. If a cluster has more than z/k points (and assume for the sake of intuition that they appear consecutively), then once z/kL of them are seen, there is a sufficient probability of picking one of the points, and thus the cluster is *covered*. For the clusters with $< z/k$ points, the total number of points in these clusters is $< z$. Thus classifying the points as outliers only results in a factor 2 increase in the number of outliers. (This is the intuitive reason we obtain a bi-criteria bound.)

To carry this plan forth formally, we require carefully arguing that the number of phases of each type is small. We also need to define the notion of *successful phases* more carefully (depending on whether they are type A or B). The details are deferred to Appendix A.

4.2. An O(1) approximation in the presence of outliers

We will follow the main strategy from Section 4.1, where we kept a second sample T in the algorithm (which allowed the interpretation as a *two-pass* algorithm), which led to a constant factor approximation for the objective value. The details are deferred to Appendix A.1.

5. Setting the guess for optimum

Our theorems so far have assumed that the target error ξ is greater than the optimal error ($\text{OPT} := \sum_{u \in V_{\text{in}}} d^2(u, C^*)$). If this is not true, we show that a doubling procedure can be applied. The description below will use Algorithm 1, but it can be replaced with any of the algorithms we have seen.

Outline of the procedure. Suppose that we are given a target error ξ_0. If $\xi_0 < \text{OPT}$, the bounds we have shown on the number of phases (in any of the algorithms above) does not hold. On the other hand, if $\xi_0 \geq opt$, the bounds on the number of phases hold with high probability. Thus the algorithm exceeding a certain number of phases can be used as a test to see if ξ_0 is too small! This suggests the following procedure: use the given value of ξ_0. If the number of phases exceeds $ck \log n$ (c is chosen using the guarantees of Algorithm 1), then we double the value of ξ, clear all the parameters and repeat. The total number of phases in the new algorithm (which determines the number of points in C in the end) will thus be $O(k \log n) \cdot \log\left(\frac{\text{OPT}}{\xi_0}\right)$. The total error behaves better – it will turn out that the total error accumulated will increase geometrically as ξ doubles (as we use the same number of iterations for each guess), and thus only the *final* value of ξ matters. But this is precisely the first value where $\xi \geq \text{OPT}$. In this case we are back to our earlier setting.

The formal execution of this procedure (using Algorithm 1) is done in Appendix C. This then completes the proof of our main results (Theorem 1 and Theorem 2).

6. Lower bounds on approximation

In this section we show that there are fundamental lower bounds that limit the performance of any online algorithm. This will prove Theorem 3.

Proof [of Theorem 3] To prove this we rely on Yao's principle and show that for any $r \in \mathbf{N}^+$ and a constant C, there exists a probability distribution over instances such that the expected cost to optimal cost ratio of any deterministic algorithm with the number of centers limited to r is $> \frac{C^2}{16}$.

Let us define a complete hierarchically separated tree (HST) with the following properties,

1. The fan-out of each vertex is $2r$.
2. The root is labeled level 0, and for any $i \geq 0$, the distance from a level i node to a descendant at level $i+1$ is $1/C^i$.

We define a probability distribution over instances obtained as follows. First, the depth d of the tree is chosen u.a.r. in the interval $[4r, 8r]$. The points are then revealed as follows. First, one point is placed at the root (v_0). Next, a descendant of the root is chosen u.a.r. (denote it by v_1), and C^2 points are placed at that location.[1] Next, a random descendant of v_1 is chosen and C^4 points are placed there. This process continues until depth d (where C^{2d} points are placed). The target value of k (the number of centers in the optimum solution) is 1. For any such instance, in hindsight, the best solution is to place a center at the last location. This results in a cost of d, which lies between $4r$ and $8r$.

Now, consider any deterministic algorithm A. We can view A as a function mapping the requests so far to a set of centers to open, i.e., A takes as input a sequence $(v_0, v_1, \ldots, v_t)$ and outputs a subset of vertices of the HST. Because A is an online algorithm, we also have that $A(v_0, \ldots, v_{t-1}) \subseteq A(v_0, \ldots, v_t)$. For any such deterministic mapping, we now argue that the expected cost of the solution (under our distribution over instances) is $\geq \frac{1}{2} r C^2$. As a first observation, note that we can view $A(v_0, \ldots, v_t)$ as a function of v_t alone, as $v_{t-1}, \ldots, v_0$ are always the ancestors of v_t in our input distribution. Thus, let us write this as $A\langle v_t \rangle$.

Now, we say that a vertex v_t (at level t) is *bad* if $A\langle v_t \rangle$ does not contain any descendant of v_t (we will follow the convention that the set of descendants includes v_t itself). We define p_t to be the fraction of bad vertices at level t in the HST. The first claim is that for all $t \leq 7r$, $p_t < 1/10$. To see this, fix some $t \leq 7r$. Now, there is a probability of $1/8$ of having $d \in [7.5r, 8r]$. If the instance is a descendant of a bad vertex v_t at level t, then the 1-means objective cost for the instance will be at least C^r. Thus, the overall expected cost is at least $p_t \cdot \frac{1}{8} C^r$. If this needs to be $\leq C^2$ (and for C, r large enough), then we must have $p_t < 1/10$.

Next, we claim that for any $t \leq 7r$, $\mathbb{E}[|A\langle v_t \rangle|] \geq \frac{4}{10} + \mathbb{E}[|A\langle v_{t-1} \rangle|]$. I.e., the size of $A\langle v_t \rangle$ for a random vertex at level t is at least $4/10$ larger than the corresponding size for a random vertex at level $t-1$. This leads to a contradiction if we run $t = 4r, \ldots, 7r$, as it leads to an average size $> r$.

To show the claim, consider all the descendants of some v_{t-1}. At most r of the $2r$ descendants can be (a) good, and (b) have $A\langle v_t \rangle = A\langle v_{t-1} \rangle$ (since $|A\langle v_{t-1} \rangle| \leq r$). Thus, since level t contains $2r$ times the number of vertices in level $t-1$, and since at least $9/10$ of them are good, on average, at least $4/10$ of the descendants of every v_{t-1} must be good without having $A\langle v_t \rangle = A\langle v_{t-1} \rangle$. Since the former set is a super-set of the latter, the desired claim follows. This leads to a contradiction as we mentioned earlier, thus completing the proof of the theorem. ■

1. We produce an instance where points have multiplicities. This can easily be removed by small perturbation.

References

Sara Ahmadian, Ashkan Norouzi-Fard, Ola Svensson, and Justin Ward. Better guarantees for k-means and euclidean k-median by primal-dual algorithms. *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 61–72, 2017.

David Arthur and Sergei Vassilvitskii. K-means++: The advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '07, pages 1027–1035, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.

Moshe Babaioff, Michael Dinitz, Anupam Gupta, Nicole Immorlica, and Kunal Talwar. Secretary problems: Weights and discounts. In *Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '09, pages 1245–1254, Philadelphia, PA, USA, 2009. Society for Industrial and Applied Mathematics.

Bahman Bahmani, Benjamin Moseley, Andrea Vattani, Ravi Kumar, and Sergei Vassilvitskii. Scalable k-means++. *PVLDB*, 5(7):622–633, 2012.

Maria-Florina Balcan, Steven Ehrlich, and Yingyu Liang. Distributed clustering on graphs. In *Neural Information Processing Systems (NIPS)*, 2013.

MohammadHossein Bateni, Aditya Bhaskara, Silvio Lattanzi, and Vahab S. Mirrokni. Distributed balanced clustering via mapping coresets. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2591–2599, 2014.

Aditya Bhaskara, Silvio Lattanzi, Sergei Vassilvitskii, and Morteza Zadimoghaddam. Residual based sampling for online low rank approximation. In *Proceedings of the 60th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2019.

Niv Buchbinder, Moran Feldman, Yuval Filmus, and Mohit Garg. Online submodular maximization: Beating 1/2 made simple. In Andrea Lodi and Viswanath Nagarajan, editors, *Integer Programming and Combinatorial Optimization*, pages 101–114, Cham, 2019. Springer International Publishing.

M. Charikar, C. Chekuri, T. Feder, and R. Motwani. Incremental clustering and dynamic information retrieval. *SIAM Journal on Computing*, 33(6):1417–1440, 2004. doi: 10.1137/S0097539702418498. URL https://doi.org/10.1137/S0097539702418498.

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*, SODA '01, pages 642–651, Philadelphia, PA, USA, 2001. Society for Industrial and Applied Mathematics. ISBN 0-89871-490-7. URL http://dl.acm.org/citation.cfm?id=365411.365555.

Ke Chen. A constant factor approximation algorithm for k-median clustering with outliers. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '08, pages 826–835, Philadelphia, PA, USA, 2008. Society for Industrial and Applied Mathematics. URL http://dl.acm.org/citation.cfm?id=1347082.1347173.

Sanjoy Dasgupta. A cost function for similarity-based hierarchical clustering. In *Proceedings of the Forty-eighth Annual ACM Symposium on Theory of Computing*, STOC '16, pages 118–127, New York, NY, USA, 2016. ACM.

Alina Ene, Sungjin Im, and Benjamin Moseley. Fast clustering using mapreduce. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 681–689, New York, NY, USA, 2011. ACM.

Dimitris Fotakis. On the competitive ratio for online facility location. *Algorithmica*, 50(1):1–57, 2007. doi: 10.1007/s00453-007-9049-y.

Alan Frieze, Ravi Kannan, and Santosh Vempala. Fast monte-carlo algorithms for finding low-rank approximations. *Journal of the ACM*, 51(6):1025–1041, November 2004.

Michel Goemans. Chernoff bounds, and some applications, 2015. URL http://math.mit.edu/~goemans/18310S15/chernoff-notes.pdf.

S. Guha, N. Mishra, R. Motwani, and L. O'Callaghan. Clustering data streams. *STOC*, 2001.

Shalmoli Gupta, Ravi Kumar, Kefu Lu, Benjamin Moseley, and Sergei Vassilvitskii. Local search methods for k-means with outliers. *Proc. VLDB Endow.*, 10(7):757–768, March 2017. ISSN 2150-8097. doi: 10.14778/3067421.3067425. URL https://doi.org/10.14778/3067421.3067425.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference and prediction*. Springer, 2 edition, 2009.

Tapas Kanungo, David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Ruth Silverman, and Angela Y. Wu. A local search approximation algorithm for k-means clustering. *Computational Geometry*, 28(2):89 – 112, 2004. Special Issue on the 18th Annual Symposium on Computational Geometry - SoCG2002.

Ravishankar Krishnaswamy, Shi Li, and Sai Sandeep. 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*, STOC 2018, pages 646–659, New York, NY, USA, 2018. ACM.

Edo Liberty, Ram Sriharsha, and Maxim Sviridenko. *An Algorithm for Online K-Means Clustering*, pages 81–89. SIAM, 2016.

Stuart P. Lloyd. Least squares quantization in pcm. *IEEE Trans. Information Theory*, 28:129–136, 1982.

A. Meyerson. Online facility location. In *Proceedings of the 42Nd IEEE Symposium on Foundations of Computer Science*, FOCS '01, pages 426–, Washington, DC, USA, 2001. IEEE Computer Society.

Michal Moshkovitz. Unexpected effects of online k-means clustering, 2019.

Rafail Ostrovsky, Yuval Rabani, Leonard J. Schulman, and Chaitanya Swamy. The effectiveness of lloyd-type methods for the k-means problem. *J. ACM*, 59(6):28:1–28:22, January 2013.

Appendix A. Analysis of k-means with outliers

We now present the full analysis of the algorithm in the presence of outliers. Our first goal will be to prove Theorem 21.

Definition 13 (Majority outlier/inlier) *A phase is said to be a* majority outlier *phase if one of the following conditions hold:*

1. *The phase is Type A and the following inequality holds (where $\{u_i\}_{i=1}^r$ are the Type A points in the phase):*
$$\sum_{i\in[r]\wedge u_i\in V\setminus V_{in}} \frac{kd^2(u_i, C_{pre})}{\xi} \geq \frac{1}{2}.$$
2. *The phase is type B and the following holds (where $\{u_i\}_{i=1}^r$ are the Type B points in the phase): $\sum_{i\in[r]\wedge u_i\in V\setminus V_{in}} \frac{kL}{z} \geq \frac{1}{2}$.*

If a phase is not majority outlier, then it is said to be a majority inlier *phase.*

Next, as in Section 3.1, we define the notion of a successful phase.

Definition 14 (Successful phases) *A phase is successful if one of the following holds:*

1. *(Majority outliers) If it is a majority outlier phase.*
2. *(Type A phases with majority inliers) Let the phase be a majority inlier phase of Type A. Let $\{u_i\}_{i=1}^r$ be the Type A points in the phase and let C_{cur} be the set of selected Type A points. The phase is said to be successful if:*
   (a) *At least one inlier point of Type A is selected, i.e., $|C_{cur} \cap V_{in}| \geq 1$*
   (b) *For the optimal set of clusters C^*, we have*
   $$\sum_{u_i\in C_{cur}\cap V_{in}} \frac{1}{p_i} d^2(u_i, C^*) < 4 \sum_{i\in[r]\wedge u_i\in V_{in}} d^2(u_i, C^*)$$
   (c) *For any point point $u_i \in C_{cur}$, we have $p_{u_i} \geq 1/n^3$*
3. *(Type B phases with majority inliers) Let the phase be a majority inlier phase of Type B. Let $\{u_i\}_{i=1}^r$ be the Type B points in the phase and let T_{cur} be the set of selected Type B points. The phase is said to be successful if:*
   (a) *At least one inlier point of Type B is selected, i.e. $|T_{cur} \cap V_{in}| \geq 1$*
   (b) *For the optimal set of clusters C^*, we have*
   $$\sum_{u_i\in T_{cur}\cap V_{in}} \frac{1}{p_i} d^2(u_i, C^*) < 4 \sum_{i\in[r]\wedge u_i\in V_{in}} d^2(u_i, C^*).$$

Lemma 15 *A Type A phase with majority inliers, is successful with probability $\geq \frac{1}{8}$ (over the choice of C_{cur}).*

Proof The proof is again by bounding the probabilities of (a), (b) and (c) in the definition of a successful Type A phase with majority inliers.

First off, the probability that $C_{\text{cur}} \cap V_{\text{in}} = \emptyset$ is at most

$$\prod_{i \in [r] \wedge u_i \in V_{\text{in}}} (1 - p_{u_i}) \leq \exp(- \sum_{i \in [r] \wedge u_i \in V_{\text{in}}} p_{u_i}) \leq 1/e^{\frac{1}{2}},$$

since by the definition of a majority inlier Type A phase, the probabilities over the inliers, add up to a quantity $\geq \frac{1}{2}$.

Next, let Y_i be an indicator random variable that is 1 if $u_i \in C_{\text{cur}}$ and 0 otherwise. Thus $\Pr[Y_i = 1] = p_{u_i}$. Define

$$\mathbf{X} = \sum_{i \in [r] \wedge u_i \in V_{\text{in}}} Y_i \frac{1}{p_{u_i}} d^2(u_i, C^*). \tag{2}$$

By linearity of expectation, we have that

$$\mathbb{E}[\mathbf{X}] = \sum_{i \in [r] \wedge u_i \in V_{\text{in}}} \mathbb{E}[Y_i] \frac{1}{p_{u_i}} d^2(u_i, C^*) = \sum_{i \in [r] \wedge u_i \in V_{\text{in}}} d^2(u_i, C^*).$$

Thus part (b) of the definition of a successful phase holds with probability $\geq 3/4$, by Markov's inequality. For part (c), note that the probability of picking a point u with with probability $< 1/n^3$ is at most $1/n^2$ (as there are at most n points in total).

Thus all the conditions hold with probability $\geq 1 - \frac{1}{e^{\frac{1}{2}}} - \frac{1}{4} - \frac{1}{n^2} > 1/8$, for $n > 8$. ■

Lemma 16 *A Type B phase with majority inliers, is successful with probability $\geq \frac{1}{8}$ (over the choice of T_{cur}).*

Proof The proof is by simply bounding the probabilities of (a), and (b) in the definition of a successful Type B phase with majority inliers.

First off, the probability that $T_{\text{cur}} \cap V_{\text{in}} = \emptyset$ is at most

$$\prod_{i \in [r] \wedge u_i \in V_{\text{in}}} (1 - p_{u_i}) \leq \exp(- \sum_{i \in [r] \wedge u_i \in V_{\text{in}}} p_{u_i}) \leq 1/e^{\frac{1}{2}},$$

since by the definition of a successful Type B phase, the probabilities over the inliers, add up to a quantity $\geq \frac{1}{2}$.

Next, let Y_i be an indicator random variable that is 1 if $u_i \in T_{\text{cur}}$ and 0 otherwise. Thus $\Pr[Y_i = 1] = p_{u_i}$. Define

$$\mathbf{X} = \sum_{i \in [r] \wedge u_i \in V_{\text{in}}} Y_i \frac{1}{p_{u_i}} d^2(u_i, C^*). \tag{3}$$

By linearity of expectation, we have that

$$\mathbb{E}[\mathbf{X}] = \sum_{i \in [r] \wedge u_i \in V_{\text{in}}} \mathbb{E}[Y_i] \frac{1}{p_{u_i}} d^2(u_i, C^*) = \sum_{i \in [r] \wedge u_i \in V_{\text{in}}} d^2(u_i, C^*).$$

Thus part (b) of the definition of a successful phase holds with probability $\geq 3/4$, by Markov's inequality.

Thus all the conditions hold with probability $\geq 1 - \frac{1}{e^{\frac{1}{2}}} - \frac{1}{4} > 1/8$. ■

There are two main steps in the analysis. First, we need to bound the number of phases in the algorithm. This determines the expected number of selected columns. Next, we need to bound the total error. The following lemmas capture these statements.

Lemma 17 *The number of majority outlier phases, is $O(kL)$.*

Proof We will prove the lemma via contradiction. Suppose we have t successful phases because of having majority outliers.

If we consider one of these phases, if the phase is of Type B and if the Type B points in that phase are $v_1, v_2, \ldots, v_r$, by definition we would have $\sum_{i\in[r]\wedge u_i\in V\setminus V_{\text{in}}} \frac{kL}{z} > \frac{1}{2}$ and if the phase is of Type A and if the Type A points in that phase are $v_1, v_2, \ldots, v_r$, we would have $\sum_{i\in[r]\wedge u_i\in V\setminus V_{\text{in}}} \frac{kd^2(u_i,C_{\text{pre}})}{\xi} > \frac{1}{2}$ and since $\frac{kL}{z} \geq \frac{kd^2(u_i,C_{\text{pre}})}{\xi}$, we can see that $\sum_{i\in[r]\wedge u_i\in V\setminus V_{\text{in}}} \frac{kL}{z} > \frac{1}{2}$. Therefore we can see that in either case, the number of outlier points in a the phase is $> \frac{z}{2kL}$

Now, we claim that $r \leq 2kL$. Suppose, for the sake of contradiction, that $r > 2kL$. Then, since for each phase in this, the number of outlier points in a the phase is $> \frac{z}{2kL}$, we would have the number of outliers $> z$, a contradiction to the assumption on the upper bound on outliers. This implies $r \leq 2kL$, implying that the number of phases successful because of having majority outliers, is $O(kL)$ ■

Lemma 18 *The number of successful majority inlier phases of Type A, is $O(k \log n)$.*

Proof We will prove the lemma via contradiction. Suppose we have t successful Type A phases, and suppose we choose one Type A inlier point from each successful phase. Let these points be $v_1, v_2, \ldots, v_t$. By definition, the probability values $p_{v_i} \in (1/n^3, {}^{kL}/_{z})$. Thus, by dividing $(1/n^3, {}^{kL}/_{z})$ into $3\log n + \log {}^{kL}/_{z}$ intervals of the form $(1/2^{i+1}, 1/2^i]$, we obtain that there exists some q such that $t/(3\log n + \log {}^{kL}/_{z})$ of the v_i lie in the interval $(q, 2q]$. Let I denote the indices of these points.

Now, for any $i, j \in I$, we claim that $d(v_i, v_j)^2 > \frac{\xi q}{k}$. This claim holds because of the following: suppose $i < j$. Thus in the jth phase, v_i was already present in C_{pre}. Thus, since $p_{v_j} \leq \frac{kd^2(v_j,C_{\text{pre}})}{\xi}$, we have that $p_{v_j} \leq \frac{kd(v_i,v_j)^2}{\xi}$. Since $p_{v_j} > q$, the claim follows.

Now, consider balls of squared-radius $\frac{\xi q}{4k}$ around each of the points v_i (for $i \in I$), and denote these balls by B_i (respectively). The claim above implies that for any $i, j \in I$ with $i \neq j$, $B_i \cap B_j = \emptyset$.

Finally, we claim that $|I| \leq 33k$. Suppose, for the sake of contradiction, that $|I| > 33k$. Then, since the balls are all disjoint, we must have that for at least $32k$ of the balls B_i, $C^* \cap B_i = \emptyset$ (where C^* is the optimal set of centers). Let J denote the set of i for which this holds. This means that for any $i \in J$, $d^2(v_i, C^*) > \frac{\xi q}{4k}$.

Now, let us denote by S_i the set of original points u that are in phase i. Since i is a successful phase, we have (by condition (b) of the definition of Type A successful phase), that

$$4 \sum_{u\in S_i\wedge u\in V_{\text{in}}} d^2(u, C^*) \geq \frac{1}{p_i} d^2(v_i, C^*) > \frac{1}{2q}\frac{\xi q}{4k} = \frac{\xi}{8k}.$$

Thus, if we have $|J| \geq 32k$, this would let us conclude that the sum of the distances to the optimal centers from the inlier points in those phases is $> \xi$, a contradiction to the assumption that the optimum objective value is $\leq \xi$.

This implies that $|J| \leq 32k$, and subsequently that $|I| \leq 33k$, implying that the number of phases t satisfies $t \leq 99k \log n$ (Since kL/z is assumed to be < 1 we can drop that from the bound). ■

Lemma 19 *The number of successful majority inlier phases of Type B, is $O(k)$.*

Proof We will prove the lemma via contradiction. Suppose we have t successful phases, and suppose we choose one Type B inlier point from each successful phase. Let these points be $v_1, v_2, \ldots, v_t$.

By definition, the probability values $p_{u_i} = \frac{kL}{z}$ and $d^2(u_i, C_{\text{pre}}) \geq \frac{\xi L}{z}$.

Now, for any $i, j \in [t]$, we claim that $d(v_i, v_j)^2 > \frac{\xi L}{2z}$. This claim holds because of the following: suppose $i < j$. Thus in the jth phase, v_i was already present in C_{pre}. Thus, since $p_{v_j} \leq \frac{kd^2(v_j, C_{\text{pre}})}{\xi}$, we have that $p_{v_j} \leq \frac{kd(v_i, v_j)^2}{\xi}$. Since $p_{v_j} > \frac{kL}{2z}$, the claim follows.

Now, consider balls of squared-radius $\frac{\xi L}{8z}$ around each of the points v_i (for $i \in [t]$), and denote these balls by B_i (respectively). The claim above implies that for any $i, j \in [t]$ with $i \neq j$, $B_i \cap B_j = \emptyset$.

Finally, we claim that $t \leq 33k$. Suppose, for the sake of contradiction, that $t > 33k$. Then, since the balls are all disjoint, we must have that for at least $32k$ of the balls B_i, $C^* \cap B_i = \emptyset$ (where C^* is the optimal set of centers). Let J denote the set of i for which this holds. This means that for any $i \in J$, $d^2(v_i, C^*) > \frac{\xi L}{8z}$.

Now, let us denote by S_i the set of original points u that are in phase i. Since i is a successful phase, we have (by condition (b) of the definition of Type B successful phase), that

$$4 \sum_{u \in S_i \wedge u \in V_{\text{in}}} d^2(u, C^*) \geq \frac{1}{p_i} d^2(v_i, C^*) > \frac{z}{kL} \frac{\xi L}{8z} = \frac{\xi}{8k}.$$

Thus, if we have $|J| \geq 32k$, this would let us conclude that the sum of the distances to the optimal centers from the inlier points in those phases is $> \xi$, a contradiction to the assumption that the optimum objective value is $\leq \xi$. This implies that $|J| \leq 32k$, and subsequently that $t \leq 33k$. ■

Lemma 20 *For any $\delta > 0$, the total number of phases is $O(k \log n + kL + \log(1/\delta))$, with probability at least $1 - \delta$.*

The proof can be divided into two parts, first showing the for Type A and B phases each we could argue the bounds for each case with probability $1 - \delta/2$, using Appendix E. We can see that the probability of both bounds is $\geq (1 - \delta/2)^2 \geq 1 - \delta$ and thus w.p. at least $1 - \delta$ the total nmber of phases is $O(k \log n + kL + \log(1/\delta))$.

We can now complete the proof of Theorem 11.

Proof [of Theorem 11] First, we bound the size of C, the selected points, using the bound on the number of phases. Suppose the number of phases is r. In each phase, the selection probabilities add up to a quantity between 1 and 2. Thus, we have that $\Pr[|C| > 20r] \leq 1/10$. Setting $\delta = 1/10$ in

Lemma 20, we get that $r \leq O(k \log n + kL)$ with probability $\geq 9/10$. Combining these, we have that $|C| \leq O(k \log n + kL)$ with probability $\geq 4/5$.

Next we bound the number of points marked as outliers. To bound this, we will bound the number of points marked as outliers in each phase. Let the points marked as outliers be M. We consider two cases.

Case 1: the phase is of Type A,

$$\sum_{i \in [r] \wedge u \in M} \frac{kL}{z} \leq 1.$$

The inequality follows because the by definition, if the phase is of Type A then $\beta < 1$

Case 2: the phase is of Type B,

$$\sum_{i \in [r] \wedge u \in M} \frac{kL}{z} \leq 2.$$

The inequality follows because the by definition, if the phase is of Type B then $1 \leq \beta \leq 2$

Therefore, we can see that in either case $|i \in [r] \wedge u \in M| \leq \frac{2z}{kL}$. Combined with the bound on the number of phases, this completes the shows that $|M| \leq \frac{2z}{kL} O(k \log n + kL) \leq O(z(\frac{\log n}{L} + 1))$.

Next, we need to bound the total cost $\sum_{u \in V_{\text{in}} \setminus M} d^2(u, C)$, where the sum ranges over the inlier points of Type A. To bound this, we will bound the sum in each phase. To this end, let $u_1, \ldots, u_r$ be the points in a phase, and let C_{pre} the set of chosen points at the start of the phase. We now consider two cases,

Case 1: the phase is of Type A

$$\sum_{i \in [r] \wedge u_i \in V_{\text{in}} \setminus M} d^2(u_i, C) \leq \sum_{i \in [r] \wedge u_i \in V_{\text{in}} \setminus M} d^2(u_i, C_{\text{pre}}) \leq \sum_{i \in [r] \wedge u_i \in V_{\text{in}} \setminus M} p_{u_i} \cdot \frac{\xi}{k} \leq 2\frac{\xi}{k}.$$

The last inequality follows because the sum of the selection probabilities of Type A points in the phase (α) for the inlier points in C_{cur} when the phase is of Type A phase is ≤ 2.

Case 1: the phase is of Type B

$$\sum_{i \in [r] \wedge u_i \in V_{\text{in}} \setminus M} d^2(u_i, C) \leq \sum_{i \in [r] \wedge u_i \in V_{\text{in}} \setminus M} d^2(u_i, C_{\text{pre}}) \leq \sum_{i \in [r] \wedge u_i \in V_{\text{in}} \setminus M} p_{u_i} \cdot \frac{\xi}{k} < \frac{\xi}{k}.$$

The last inequality follows because the sum of the selection probabilities of Type A points in the phase (α) for the inlier points in C_{cur} when the phase is of Type B is < 1.

This implies that in either case the $\sum_{i \in [r] \wedge u_i \in V_{\text{in}}} d^2(u_i, C) \leq 2\frac{\xi}{k}$. Combined with the bound on the number of phases, this shows that the total cost over the inlier points (not marked as outliers) is $\xi O(\log n + L)$. ∎

The Algorithm 3 performs well when the number of outliers are sufficiently large ($z \gg k \log n$). However even in the case where z is small, we can see that by controlling the parameter L we can still get the algorithm to work well (at the cost of marking $O(z \cdot \frac{\log n}{L})$ many points as outliers).

A.1. Improvement to a constant factor approximation

Next, we follow the ideas from Section 3.1, where we kept a second sample T (which was then interpreted as a *two-pass* algorithm), which led to a constant factor approximation for the objective value.

Algorithm 4: Online k-means clustering $O(1)$ Approximation

Input: A set of points V that arrive one by one, guess ξ for the optimum error over the inliers, parameters k, L and ϵ.
Output: A set C of the cluster centers.
Initialize $C_{\text{pre}} = \emptyset$, $C_{\text{cur}} = \emptyset$, $T_{\text{cur}} = \emptyset$ and $T = \emptyset$;
while *points* u *arrive* **do**
  Execute Algorithm 3 with input u; this updates C_{pre}, C_{cur} and T_{cur} ;
  if $\frac{k.d^2(u,C_{pre})}{\xi} < \frac{k.L}{z}$ **then**
    With probability $p_u := \min(\frac{40k.d^2(u,C_{\text{pre}})}{\epsilon\xi}, 1)$, add u to T
  end
end
Output $C = C_{\text{pre}} \cup T.$;

Description of Algorithm 4. As in Algorithm 2, we maintain a second set T to which we add points independently. The key is that we do this only for type A points. (Intuitively, this is because we declare type B points as outliers and we do not need to *refine* the cost we incur on them.) Thus when a point u arrives, we first run it through Algorithm 3 and update the C_{pre} as necessary. Then with a sampling probability $p_u = \min(\frac{40k\cdot d^2(u,C_{\text{pre}})}{\epsilon\xi}, 1)$, we assign u to set T. The output set C is the union of these C_{pre} and T.

Theorem 21 *Suppose the set of points V arrive in an online manner. Let $k \geq 1$ be an integer, and let ξ be a (given) parameter that satisfies $\sum_{v\in V_{in}} d^2(v, C^*) \leq \xi$. Then for any $\epsilon > 0$, the algorithm satisfies:*

1. *w.p. at least $\frac{7}{10}$, the number of chosen centers is $\leq O(\frac{k}{\epsilon}(\log(n) + L))$.*
2. *The k-means objective cost for the output centers C, over the points not marked as outliers is $\leq O(1) \cdot \text{OPT} + \epsilon\xi$ where OPT is the optimum k-means objective value over the inlier points.*

A.2. Analysis

Once again, the idea is to view the algorithm as a "two-pass" procedure) and analyze that instead. Once again, we observe that in line 4 of the algorithm, we do not use T in any way, and thus it proceeds precisely as in Algorithm 3. Let C_{first} denote the final value (after we have seen all the points) of $C_{\text{pre}} \cup C_{\text{cur}} \cup T_{\text{cur}}$. Now, let us replace the probability of adding v to T (line 4) to $\min(1, \frac{40k\cdot d^2(v,C_{\text{first}})}{\epsilon\xi})$. This is no longer an online algorithm, but it is a two-pass algorithm. The key observation is that the probability of adding u to T in our algorithm is at least as large as the probability of adding u to T in the two pass algorithm. Thus, it suffices to bound the error of the two-pass algorithm.

This is done by using the lemma 10. In this case we only take into account the set $V_{\text{in}} \setminus M$, which contains the points that are not marked as outliers. (The points marked as outliers is denoted by M)

In analyzing the two-pass algorithm, we apply Lemma 10 with $S = C_{\text{first}}$. Then we show that the condition $p_v \geq ck\frac{d^2(v,S)}{Z}$ holds with $c = \frac{40Z}{\epsilon\xi}$. To see this, note that

$$p_i = \frac{40kd^2(v, C_{\text{first}})}{\epsilon\xi} = \frac{40Z}{\epsilon\xi}\frac{kd^2(v, C_{\text{first}})}{Z}.$$

Thus with probability at least $\frac{4}{5}$ (over only the second pass), denoting by T the set of points chosen in the second pass,

$$\sum_{v \in V_{\text{in}} \setminus M} d^2(v, C_{\text{first}} \cup T) \leq 160 \sum_{v \in V_{\text{in}} \setminus M} d^2(v, C^*) + \frac{40Z}{c}$$

.

which results in,

$$\sum_{v \in V_{\text{in}} \setminus M} d^2(v, C_{\text{first}} \cup T) \leq 160 \sum_{v \in V_{\text{in}} \setminus M} d^2(v, C^*) + \epsilon\xi$$

Next, consider the expected size of the final set. This time, we need to analyze Algorithm 2 and not the two-pass algorithm above. We have that

$$E[|T|] = \sum_{v \in V_{\text{in}} \setminus M} \frac{40kd^2(v, C_{\text{pre}})}{\epsilon\xi}$$

By the bound on the total error from Theorem 11, we have that with probability $\geq \frac{4}{5}$,

$$\sum_{v \in V_{\text{in}} \setminus M} d^2(v, C_{\text{pre}}) \leq O(\xi(\log(n) + L))$$

Thus the above together with Markov's inequality, we have that with probability $\geq \frac{7}{10}$,

$$|T| \leq O(\frac{k}{\epsilon}(\log(n) + L))$$

This completes the proof of Theorem 21.

Appendix B. "Norm sampling" for k-means

We will now prove Lemma 10. Before doing so, let us start with a simple technical lemma we will need.

Lemma 22 *Let $\alpha, \delta_1, \delta_2, \ldots, \delta_k$ be positive reals, and let $\sum_{i=1}^k \delta_i = 1$. Then,*

$$\sum_{i=1}^{k} e^{-\alpha k \delta_i} \delta_i \leq \frac{1}{\alpha}$$

Proof Given that $\alpha, k, \delta_i \geq 0$, we can see that $e^{-\alpha k \delta_i} \leq \frac{1}{1+\alpha k \delta_i}$. Therefore,

$$\sum_{i=1}^{k} e^{-\alpha k \delta_i} \delta_i \leq \sum_{i=1}^{k} \frac{\delta_i}{1 + \alpha k \delta_i} \leq \sum_{i=1}^{k} \frac{1}{\alpha k} = \frac{1}{\alpha}$$

■

We will also use the so-called *parallel axis theorem*, which can be stated as follows.

Lemma 23 *Let $u_1, u_2, \ldots, u_r$ be a set of points and let μ be their mean. Let x be any other point. Then we have*

$$\sum_i d^2(u_i, x) = \sum_i d^2(u_i, \mu) + r d^2(x, \mu).$$

The proof follows by a direct computation. We are now ready to prove Lemma 10.

Proof [of Lemma 10.] Let $Z = \sum_{v \in V} d^2(v, S)$ as in the statement. Let $C_1, C_2, \ldots, C_k$ be the optimal clusters and let $\mu_1, \mu_2, \ldots, \mu_k$ be their centers respectively. Define σ_i^2 to be the average squared distance from the points in cluster i to its center. I.e.,

$$\sigma_i^2 = \frac{1}{|C_i|} \sum_{u \in C_i} d^2(u, \mu_i).$$

The rough idea of the proof is as follows. Our goal will be to argue that for every cluster, the sampling procedure is quite likely to pick a point that is *close to the center* of the cluster. We then argue that this leads to the value of $d^2(u, S \cup T)$ being relatively small (compared to the average radius of the cluster). To this end, for every i, define

$$D_i = \{u \in C_i \ : \ d^2(u, \mu_i) \leq 4\sigma_i^2\}.$$

By a simple averaging (which can also be viewed as an application of Markov's inequality), we have that $|D_i| \geq 3|C_i|/4$ (i.e., D_i contains at least $3/4$th of the points of C_i). Next, for any S, we will relate $\sum_{u \in C_i} d^2(u, S)$ with $\sum_{u \in D_i} d^2(u, S)$. Specifically, we show that the former is not too much larger than the latter. Let $u' = \text{argmin}_{u \in D_i} d^2(u, S)$. Then by definition, we have

$$d^2(u', S) \leq \frac{1}{|D_i|} \sum_{u \in D_i} d^2(u, S). \tag{4}$$

Next, for any $u \in C_i \setminus D_i$, we have $d(u, S) \leq d(u, u') + d(u', S)$, and thus $d^2(u, S) \leq 2(d^2(u, u') + d^2(u', S))$. Similarly, we can use $d(u, u') \leq d(u, \mu_i) + d(u', \mu_i)$ to conclude that $d^2(u, u') \leq 2(d^2(u, \mu_i) + d^2(u', \mu_i))$. Putting the two together and summing over all $u \in C_i \setminus D_i$, we have

$$\sum_{u \in C_i \setminus D_i} d^2(u, S) \leq 2|C_i \setminus D_i| \left(d^2(u', S) + 2d^2(u', \mu_i)\right) + 4 \sum_{u \in C_i \setminus D_i} d^2(u, \mu_i).$$

Now, since $u' \in D_i$, we have $d^2(u', \mu_i) \leq 4\sigma_i^2$. Also, $\sum_{u \in C_i} d^2(u, \mu_i) = |C_i|\sigma_i^2$ by definition, so we can use this as an upper bound for the last term above. Using these, together with $|C_i \setminus D_i| \leq |C_i|/4$, we get:

$$\sum_{u \in C_i \setminus D_i} d^2(u, S) \leq \frac{|C_i|}{2} d^2(u', S) + 4|C_i|\sigma_i^2 + 4|C_i|\sigma_i^2.$$

Now using (4) along with $|D_i| \geq 3|C_i|/4$, we get:

$$\sum_{u \in C_i \setminus D_i} d^2(u, S) \leq \frac{2}{3} \sum_{u \in D_i} d^2(u, S) + 8|C_i|\sigma_i^2 \tag{5}$$

$$\implies \sum_{u \in C_i} d^2(u, S) \leq \frac{5}{3} \sum_{u \in D_i} d^2(u, S) + 8|C_i|\sigma_i^2. \tag{6}$$

Next, call a cluster *good* if $\sum_{u\in C_i} d^2(u,S) \le 16|C_i|\sigma_i^2$, and bad otherwise. If a cluster is good, we trivially have $\sum_{u\in C_i} d^2(u,S\cup T) \le 16|C_i|\sigma_i^2$. Thus, let us only focus on bad clusters. The important observation is that by (6), if C_i is a bad cluster, then

$$\sum_{u\in D_i} d^2(u,S) \ge \frac{3}{10}\sum_{u\in C_i} d^2(u,S).$$

Intuitively, this means that if one were to sample points in C_i using weights proportional to $d^2(u,S)$, there is at least a $3/10$ probability of picking a point "close" to the center.

Let us now use this to reason about $\mathbb{E}[\sum_{u\in C_i} d^2(u,S\cup T)]$. We can bound it as follows. If the sample T contains some point in $x\in D_i$, then by the parallel axis theorem,

$$\sum_{u\in C_i} d^2(u,S\cup T) \le \sum_{u\in C_i} d^2(u,x) \le |C_i|\sigma_i^2 + 4|C_i|\sigma_i^2 = 5|C_i|\sigma_i^2.$$

The probability that *no* point from D_i is chosen is at most: $\prod_{u\in D_i}(1-p_u) \le \exp(-\sum_{u\in D_i} p_u)$. By the condition we have on p_u, this can be bounded by

$$\exp\left(-\frac{ck}{Z}\sum_{u\in D_i} d^2(u,S)\right) \le \exp\left(-\frac{3ck}{10Z}\sum_{u\in C_i} d^2(u,S)\right).$$

For notational convenience, define $\frac{1}{Z}\sum_{u\in C_i} d^2(u,S) = \delta_i$ (thus $\sum_i \delta_i = 1$). The above computation shows that the probability that T contains no point of D_i is at most $\exp(-3ck\delta_i/10)$. Thus we can bound

$$\mathbb{E}\left[\sum_{u\in C_i} d^2(u,S\cup T)\right] \le 5|C_i|\sigma_i^2 + Z\delta_i\cdot\exp(-3ck\delta_i/10).$$

Now, summing this over all the bad clusters, we can bound

$$\mathbb{E}\left[\sum_{u\in V} d^2(u,S\cup T)\right] \le 16\sum_{u\in V} d^2(u,C^*) + \sum_i Z\delta_i\cdot\exp(-3ck\delta_i/10).$$

Now, setting $\alpha = 3c/10$, we can bound the second term by $10Z/3c$. This gives the desired result. ■

Appendix C. Doubling argument for guessing ξ

We now formally describe the procedure from Section 5 (see Algorithm 5).

We have the following lemmas about Algorithm 5. These are tailored to the fact that we used Algorithm 1. If we were to replace it with the other algorithms we have seen, we would obtain corresponding guarantees.

Lemma 24 *The number of phases of the Algorithm 5 is bounded by*

$$O\left(k\log n\cdot\max\left(1,\log\frac{\sum_{v\in V} d^2(v,C^*)}{\xi_0}\right)\right).$$

Algorithm 5: Doubling for small ξ

Input: A set of points V that arrive one by one, parameters k and ξ_0
Output: A set C_{final} of the cluster centers.
Initialize $C = \emptyset$ and $C_{old} = \emptyset$;
while *points u arrive* **do**
  Execute Algorithm 1 with input u; this updates C ;
  if *number of phases exceeds $ck \log n$ (c comes from the bounds for Algorithm 1)* **then**
    set $C_{old} \leftarrow C_{old} \cup C$;
    set $C = \emptyset$, $\xi \leftarrow 2\xi$;
  end
end
Output $C_{\text{final}} = C_{old} \cup C$

Proof Define $\text{OPT} = \sum_{v \in V} d^2(v, C^*)$ as before. Now, if $\xi_0 \geq \text{OPT}$, then the algorithm would not double its points and thus would only result in $O(k \log n)$ phases.[2] Consider the case where $\xi_0 < \text{OPT}$. Since we stop and double the ξ value after $ck \log n$ rounds, the total number of phases is clearly bounded above by $k \log n \log \frac{\text{OPT}}{\xi_0}$. Together, these cases imply the desired lemma. ■

The next lemma is more interesting: it shows that the error bound only increases by a constant (and not a $\log \frac{\text{OPT}}{\xi_0}$ factor).

Lemma 25 *The error of the Algorithm 5 (total squared distance from points u to the points chosen in the end), is $\leq O(\text{OPT} \cdot \log n)$.*

Proof The algorithm doubles the guess of ξ after $ck \log n$ phases. The total objective value for each ξ is thus at most $2c\xi \log n$. After the ith time ξ_0 is doubled, we will have $\xi = 2^i \xi_0$. Let $t = \log \frac{\text{OPT}}{\xi_0}$ be the bound on the number of times we double ξ. Then we have a bound on the objective value, of $2c \log n(\xi_0 + 2\xi_0 + \cdots + 2^t \xi_0) = O(c \log n \text{OPT})$. This completes the proof. ■

Appendix D. Full proofs of lemmas

D.1. Proof of Lemma 6

First off, the probability that $C_{\text{cur}} = \emptyset$ is at most

$$\prod_{i \in [r]} (1 - p_{u_i}) \leq \exp(-\sum_{i \in [r]} p_{u_i}) \leq 1/e,$$

since by the definition of a phase, the probabilities add up to a quantity ≥ 1.

Next, let Y_i be an indicator random variable that is 1 if $u_i \in C_{\text{cur}}$ and 0 otherwise. Thus $\Pr[Y_i = 1] = p_{u_i}$. Define

$$\mathbf{X} = \sum_{i \in [r]} Y_i \frac{1}{p_{u_i}} d^2(u_i, C^*). \tag{7}$$

2. Technically, this is always true – there is a failure probability of $1/10$. However, we can carry forth this failure probability into our procedure's guarantee as well.

By linearity of expectation, we have that

$$\mathbb{E}\left[\mathbf{X}\right] = \sum_{i\in[r]} \mathbb{E}[Y_i]\frac{1}{p_{u_i}} d^2(u_i, C^*) = \sum_{i\in[r]} d^2(u_i, C^*).$$

Thus part (2) of the definition of a successful phase holds with probability $\geq 3/4$, by Markov's inequality. For part (3), note that the probability of picking a point u with with probability $< 1/n^2$ is at most $1/n$ (as there are at most n points in total).

Thus all the conditions hold with probability $\geq 1 - \frac{1}{e} - \frac{1}{4} - \frac{1}{n} > 1/4$, for $n > 10$.

Appendix E. Bounding number of phases and points

The following lemma is used to bound the number of phases in the algorithm. We note that the lemma was already proved in (Bhaskara et al., 2019), and we include the proof here for completeness.

Lemma 26 *We toss a coin n times. The tosses are independent of each other and in each toss, the probability of seeing a head is at least p. Let H_m and T_m denote the number of heads and tails we observe in the first $m \leq n$ coin tosses. With probability $1-\delta$, we have $H_m \geq \frac{pm}{4} - \lceil 8 \ln \frac{2}{\delta}/p \rceil$ for any $1 \leq m \leq n$. We note that although the claim is about conjunction of all these n events, the probability does not rely on n.*

Proof We denote the expected number of heads in the first m tosses with μ which is at least pm. Applying lower tail inequality of Theorem 4 in (Goemans, 2015) implies,

$$Pr[H_m < (1 - \frac{1}{2})\mu] \leq e^{-\mu/8} \leq e^{-pm/8}$$

The error probability $e^{-pm/8}$ is at most $\delta/2$ for $m \geq m' = \lceil {8ln(2/\delta)}/{p} \rceil$. Instead of summing up the error bound for all values of m, we focus on the smaller geometrically growing sequence $M = \{2^l m' | l \in \mathbf{Z}^{\geq \mathbf{0}} \; AND \; 2^l m' \leq n\}$. Having the lower bound on H_m for every $m \in M$ helps us achieve a universal lower bound on any $1 \leq m \leq n$ as follows. For any $m \leq m'$, the bound $H_m \geq pm - m'$ holds trivially. For any other $m \leq n$, there exists an $m'' \in M$ such that $m'' \leq m \leq 2m''$. By definition H_m is at least $H_{m''}$. Assuming $H_{m''} \geq {pm''}/{2}$ implies H_m is at least ${pm}/{4}$ which proves the claim of the lemma. So we focus on bounding the error probabilities only for values in set M. For m', the error probability is at most $\delta/2$. The next value in M is $2m'$, so given the exponential form of the error, it is at most $(\delta/2)^2$. Using union bound, the aggregate error probability for set M does not exceed

$$\frac{\delta}{2} + \left(\frac{\delta}{2}\right)^2 + \left(\frac{\delta}{2}\right)^3 + \cdots \leq \frac{\delta/2}{1-\delta/2} \leq \delta$$

Therefore with probability at least $1-\delta$ we have for every $m \in M$, $H_m \geq {pm}/{2}$, and consequently for every $1 \leq m \leq n$, $H_m \geq \frac{pm}{4} - m'$ which finishes the proof. ■

The next lemma allows us to go from a bound on the number of phases to a bound on the number of points chosen.

Lemma 27 *Consider any of the algorithms 1, 2, 3, 4. If the number of phases is $\leq r$, then with probability $\geq \frac{9}{10}$, the number of points selected is $\leq 20r$.*

Proof Assume we have p phases where $p \leq r$, then $Z = \sum_{i\in[p]} \sum_{v_j \in Phase_i} Y_j$ is the number of points selected. Then $E(Z) = E(\sum_{i\in[p]} \sum_{v_j \in Phase_i} Y_j) = \sum_{i\in[p]} \sum_{v_j \in Phase_i} E(Y_j) \leq \sum_{i\in[p]} 2$ (Since the expected number of points over a single phase is ≤ 2). Therefore $E(Z) \leq 2p \leq 2r$ and applying Markov's Inequality with this we can see that $Pr(Z \geq 20r) \leq Pr(Z \geq 10E(Z)) \leq 1/10$. Thus w.p. $\geq \frac{9}{10}$, $Z \leq 20r$. ■