Abstract

The $k$-means++ algorithm of Arthur and Vassilvitskii (SODA 2007) is a state-of-the-art algorithm for solving the $k$-means clustering problem and is known to give an $O(\log k)$-approximation in expectation. Recently, Lattanzi and Sohler (ICML 2019) proposed augmenting $k$-means++ with $O(k \log \log k)$ local search steps to yield a constant approximation (in expectation) to the $k$-means clustering problem. In this paper, we improve their analysis to show that, for any arbitrarily small constant $\varepsilon > 0$, with only $\varepsilon k$ additional local search steps, one can achieve a constant approximation guarantee (with high probability in $k$), resolving an open problem in their paper.

1. Introduction

$k$-means clustering is an important unsupervised learning task often used to analyze datasets. Given a set $P$ of points in $d$-dimensional Euclidean space $\mathbb{R}^d$ and an integer $k$, the task is to partition $P$ into $k$ clusters while minimizing the total cost of the partition. Formally, the goal is to find a set $C \subseteq \mathbb{R}^d$ of $k$ centers minimizing the following objective:

$$\sum_{p \in P} \min_{c \in C} \|p - c\|^2,$$

where points $p \in P$ are assigned to the closest candidate center $c \in C$.

Finding an optimal solution to this objective was proven to be NP-hard (Aloise et al., 2009; Mahajan et al., 2009), and the problem was even shown to be hard to approximate to arbitrary precision (Awasthi et al., 2015; Lee et al., 2017). The currently best known approximation ratio is 6.357 (Ahmadian et al., 2019), while other constant factor approximation algorithms exist (Jain & Vazirani, 2001; Kanungo et al., 2004). For constant dimensions $d$, $(1 + \varepsilon)$-approximation algorithms are known (Cohen-Addad, 2018; Cohen-Addad et al., 2019; Friggstad et al., 2019; Bandyapadhyay & Varadarajan, 2015). However, these algorithms are mainly of theoretical interest and not known to be efficient in practice.

On the practical side of things, the canonical $k$-means algorithm (Lloyd, 1982) proved to be a good heuristic. Starting with $k$ initial points (e.g. chosen at random), Lloyd’s algorithm iteratively, in an alternating minimization manner, assigns points to the nearest center, and updates the centers to be the centroids of each cluster, until convergence. Although the alternating minimization provides no provable approximation guarantee, Lloyd’s algorithm never increases the cost of the initial clustering. Thus, one way to obtain theoretical guarantees is to provide Lloyd’s algorithm with a provably good initialization.

The $k$-means++ algorithm (see Algorithm 1) of Arthur and Vassilvitskii (2007) is a well-known algorithm for computing an initial set of $k$ centers with provable approximation guarantees. The initialization is performed by incrementally choosing $k$ initial seeds for Lloyd using $D^2$-sampling, i.e., sample a point with probability proportional to its squared distance to the closest existing center. They showed that the resultant clustering is an $O(\log k)$-approximation in expectation. This theoretical guarantee is substantiated by empirical results showing that $k$-means++ can heavily outperform random initialization, with only a small amount of additional computation time on top of running Lloyd’s algorithm. However, lower bound analyses (Brunsch & Röglin, 2013; Bhattacharya et al., 2016) show that there exist inputs where $k$-means++ is $\Omega(\log k)$-competitive with high probability in $k$.

Recently, Lattanzi and Sohler (2019) proposed a variant of local search after picking $k$ initial centers via $k$-means++ (see Algorithm 2): In each step, a new point is sampled with probability proportional to its current cost and used to replace an existing center such as to maximize the cost reduction. If all possible swaps increase the objective cost, the new sampled point is discarded. Following their notation, we refer to this local search procedure as
LocalSearch++. They showed that performing $O(k \log \log k)$ steps of LocalSearch++ after $k$-means++ improves the expected approximation factor from $O(\log k)$ to $O(1)$, and stated that it is an interesting open question to prove that $O(k)$ local search steps suffice to obtain a constant factor approximation.

1.1. Our contribution

In this paper, we answer the open question by Lattanzi and Sohler (2019) in the affirmative. We refine their analysis to show that with only $\varepsilon k$ additional local search steps, one can achieve an approximation guarantee of $O(1/\varepsilon^3)$ with probability $1 - \exp(-\Omega(k^{0.1}))$. Compared to (Lattanzi & Sohler, 2019), we improve the number of search steps needed to achieve a constant approximation from $O(k \log \log k)$ to just $\varepsilon k$. Furthermore, our statement holds with high probability in $k$, while their statement gives a guarantee in expectation.

**Theorem 1 (Main theorem).** Let $k \in \Omega(1/\varepsilon^{20})$ and $0 < \varepsilon \leq 1$. Suppose we run Algorithm 1 followed by $\varepsilon k$ steps of Algorithm 2. We have $\text{cost}(P, C) \leq (10^{30}/\varepsilon^3) \cdot \text{cost}(P, C^*)$ with probability at least $1 - \exp(-\Omega(k^{0.1}))$.

1.2. Related Work

Another variant of local search was analyzed by Kanungo et al. (2004): In each step, try to improve by swapping an existing center with an input point. Although they showed that this eventually yields a constant approximation, the number of required steps can be very large.

Under the bicriteria optimization setting, Aggarwal et al. (2009) and Wei (2016) proved that if one over-samples and runs $D^2$-sampling for $O(k)$ steps (instead of just $k$), one can get a constant approximation of the $k$-means objective with these $O(k)$ centers. We note that a single step of LocalSearch++ has almost the same asymptotic running time as over-sampling once using $D^2$-sampling (see Section 3.3 for details) while enforcing the constraint of exactly $k$ centers. However, their results are stronger in terms of approximation guarantees: Aggarwal et al. (2009) proved that in $O(k)$ steps one achieves a $4 + \varepsilon$ approximation to the optimal cost with constant probability, while Wei (2016) proved that after $\varepsilon k$ more sampling steps one achieves an $O(1/\varepsilon)$-approximation in expectation.

Other related work include speeding up $k$-means++ via approximate sampling (Bachem et al., 2016), approximating $k$-means++ in the streaming model (Ackermann et al., 2012), and running $k$-means++ in a distributed setting (Bahmani et al., 2012).

1.3. Our method, in a nutshell

Lattanzi and Sohler showed that given any clustering with approximation ratio of at least 300, a single step of LocalSearch++ improves the cost by a factor of $1 - 1/(100k)$, with constant probability. In general, one cannot hope to asymptotically improve their bound\(^1\). Instead, our improvement comes from structural insights on solutions provided by Algorithm 1 and Algorithm 2. We argue that if we have a bad approximation at any point in time, the next step of Algorithm 2 drastically improves the cost with a positive constant probability.

To be more specific, consider an optimal clustering $OPT$. We prove that throughout the course of Algorithm 2, most centroids of $OPT$ clusters have a candidate center in the current solution close to it. Through a chain of technical lemmas, à la (Lattanzi & Sohler, 2019), we conclude that the new sampled point is close to the centroid of one of the (very few) costly $OPT$ clusters with constant probability. Then, we argue that there is an existing candidate that can be swapped with the newly sampled point to improve the solution quality substantially. Putting everything together, we get that the solution improves by a factor of $1 - \Theta(\sqrt{\alpha}/k)$ with constant probability in a single LocalSearch++ step, where $\alpha$ is the approximation factor of the current solution. This improved multiplicative cost reduction suffices to prove our main result.

In Section 2, we introduce notation and crucial definitions, along with several helpful lemmas. In Section 3, we walk the reader through our proof while deferring some lengthy proofs to the supplementary material.

---

\(^1\)Consider a $(k - 1)$-dimensional simplex with $n/k$ points at each corner and a clustering with all $k$ centers in the same corner. Swapping one center to another corner of the simplex improves the solution by a factor of $1 - \Theta(1/k)$. However, we do not expect to get such a solution after running the $k$-means++ algorithm.
2. Preliminaries

Let $P$ be a set of points in $\mathbb{R}^d$. For two points $p, q \in \mathbb{R}^d$, let $\|p - q\|$ be their Euclidean distance. We denote $C \subseteq P$ as the set of candidate centers and $C^* = \text{OPT}$ as the centers of a (fixed) optimal solution, where $|C^*| = k$. Note that a center $c^* \in C^*$ may not be a point from $P$ while all candidates $c \in C$ are actual points from $P$. For $c^* \in C^*$, the set $Q_{c^*}$ denotes the points in $P$ that $\text{OPT}$ assigns to $c^*$. We define $\text{cost}(C, p) = \sum_{c \in C} \min_{c \in C} \|p - c\|^2$ as the cost of centers $C$, where $\text{cost}(P, C^*)$ is the cost of an optimal solution. When clear from context, we also refer to the optimal cost as $\text{OPT}$. For an arbitrary set of points $Q$, we denote their centroid by $\mu_Q = (1/|Q|) \cdot \sum_{q \in Q} q$. Note that $\mu_Q$ may not be a point from $Q$.

For the sake of readability, we drop the subscript $Q$ when there is only one set of points in discussion and we drop braces when describing singleton sets in $\text{cost}(\cdot, \cdot)$. We will also ignore rounding issues as they do not play a critical role asymptotically.

We now define $D^2$-sampling introduced in $k$-means++.

**Definition 2** ($D^2$-sampling). Given a set $C \subseteq P$ of candidate centers, we sample a point $p \in P$ with probability $P[p] = \text{cost}(p, C)/\sum_{p \in P} \text{cost}(p, C)$.

The following folklore lemma describes an important property of the cost function. This is analogous to the bias-variance decomposition in machine learning and to the parallel axis theorem in physics (Aggarwal et al., 2009). As the variable naming suggests, we will use it with $Q$ being an $\text{OPT}$ center and $c$ being a candidate center.

**Lemma 3.** Let $Q \subseteq P$ be a set of points. For any point $c \in P$ (possibly not in $Q$),

$$\text{cost}(Q, c) = |Q| \cdot \|c - \mu_Q\|^2 + \text{cost}(Q, \mu_Q)$$

To have a finer understanding of the cluster structure, we define the notions of settled and approximate clusters. Consider an arbitrary set of points $Q \subseteq P$ (e.g. some cluster of $\text{OPT}$). We define

$$R_{Q, \beta} = \{q \in Q : \|q - \mu_Q\|^2 \leq (\beta/|Q|) \cdot \text{cost}(Q, \mu_Q)\}$$

as the subset of points in $Q$ that are within a certain radius from $\mu_Q$ (i.e. “close” with respect to $\beta$). As $\beta$ decreases, the condition becomes stricter and the set $R_{Q, \beta}$ shrinks.

**Definition 4** ($\beta$-settled). An $\text{OPT}$ cluster $Q$ is $\beta$-settled if $R_\beta \cap C \neq \emptyset$. That is, there is a candidate center $c \in Q$ with distance at most $(\beta/|Q|) \cdot \text{cost}(Q, \mu_Q)$ from $\mu_Q$.

**Definition 5** ($\alpha$-approximate). An $\text{OPT}$ cluster $Q$ is $\alpha$-approximate if $\text{cost}(Q, C) \leq \alpha \cdot \text{cost}(Q, \mu)$.

Intuitively, a $\beta$-settled cluster $Q$ has small $\text{cost}(Q, C)$. As settled-ness requires a candidate $c \in C$ to belong to cluster $Q$, an unsettled cluster $Q$ could have small $\text{cost}(Q, C)$. See Fig. 1 for an illustration of $Q$, $\mu_Q$, $R_{Q, \beta}$ and $\beta$-settled. We now relate the definitions of settled and approximate.

**Lemma 6.** Suppose $Q$ is a cluster of $\text{OPT}$ that is $\beta$-settled. Then, $\text{cost}(Q, C) \leq (\beta + 1) \cdot \text{cost}(Q, \mu)$. In other words, $\beta$-settled implies $(\beta + 1)$-approximate.

**Proof.** For any $\beta$-settled cluster $Q$, there is some candidate center $c \in C$ in $R_\beta$, so

$$\text{cost}(Q, C) \leq \text{cost}(Q, c) = |Q| \cdot \|c - \mu\|^2 + \text{cost}(Q, \mu) \leq (\beta + 1) \cdot \text{cost}(Q, \mu) = (\beta + 1) \cdot \text{cost}(Q, C^*)$$

It is also useful to consider the contrapositive of Lemma 6.

**Corollary 7.** Let $Q$ be a cluster of $\text{OPT}$. If $\text{cost}(Q, C) > (\beta + 1) \cdot \text{cost}(Q, \mu)$, then $\|c - \mu\|^2 > (\beta/|Q|) \cdot \text{cost}(Q, \mu)$ for any candidate center $c \in C$. That is, $Q$ is $\beta$-unsettled.

In our analysis, we will prove statements about clusters being settled for general values of $\beta$. However, there are uncountably many possible $\beta$’s and therefore we cannot do a union bound over all possible choices of $\beta$. Using a similar idea to $\varepsilon$-net arguments, we will discretize the set of $\beta$’s into a sufficiently small finite set of legal values.

**Definition 8** (Legal $\beta$ values). A parameter $\beta$ is legal if $\beta \in \mathcal{B} = \{2^i : i \in \{3, 4, \ldots, 0.3 \cdot \log k\}\}$. In particular, this implies that all legal $\beta$ are at least 8 and at most $k^{0.3}$.

3. Analysis

In this section, we present the key ideas of our proof while deferring some details to the supplementary material.
Following the proof outline of Lattanzi and Sohler (2019), we perform a more intricate analysis of LocalSearch++. Their key lemma shows that, with constant probability, the cost of the solution decreases by a factor of $1 - 1/(100k)$ after one local search step.

**Lemma 3 in (Lattanzi & Sohler, 2019)** Let $P$ be a set of points and $C$ be a set of centers with $\text{cost}(P, C) > 500 \text{OPT}$. Denote the updated centers by $C' = \text{LocalSearch++}(P, C)$. Then, with probability $1/1000$, $\text{cost}(P, C') \leq (1 - 1/(100k)) \text{cost}(P, C)$.

The above lemma implies that we expect the cost of the current solution to drop by a constant factor after $O(k)$ LocalSearch++ steps (unless we already have a constant approximation of the optimum). Since we start with a solution that is an $O(k \log k)$-approximation in expectation, we expect that after $O(k \log k)$ iterations, the cost of our solution drops to a constant. This yields the main theorem of (Lattanzi & Sohler, 2019).

**Theorem 1 in (Lattanzi & Sohler, 2019)** Let $P$ be a set of points and $C$ be the output of $k$-means++ followed by at least $100000 k \log log k$ many local search steps. Then, we have $E(\text{cost}(P, C)) \in O(\text{cost}(P, C'))$. The running time of the algorithm is $O(dnk^2 \log \log k)$.

Our improvements rely on the following structural observation: After running $k$-means++, most of the clusters of the optimal solution are already “well approximated” with high probability in $k$.

### 3.1. Structural analysis

In this subsection, we study the event of sampling a point from a $\beta$-unsettled cluster and making it $\beta$-settled. This allows us to prove concentration results about the number of $\beta$-settled clusters, which we will use in the next subsection.

Suppose $\alpha$ is the current approximation factor. The result below states that with good probability, the new sampled point is from a cluster that is currently bad approximated. **Lemma 9.** Suppose that $\text{cost}(P, C) = \alpha \cdot \text{cost}(P, C')$ and we $D^2$-sample a point $p \in P$. Consider some fixed $\beta \geq 1$. Then, with probability at least $1 - \beta/\alpha$, the sampled point $p$ is from a cluster $Q$ with $\text{cost}(Q, C) \geq \beta \cdot \text{cost}(Q, C)$.

**Proof.** Let $\tilde{Q}$ be the union of all clusters $Q$ such that $\text{cost}(Q, C) < \beta \cdot \text{cost}(Q, C)$. By definition, $D^2$-sampling will pick a point from $Q$ with probability at most $\text{cost}(Q, C)/\text{cost}(P, C) \leq \beta/\alpha$.

Similar to most work on $k$-means++, we need a sampling lemma stating that if we sample a point within a cluster $Q$ (according to $D^2$ weights of points in $Q$), then the sampled point will be relatively close to $\mu_Q$ with good probability. **Lemma 10.** Suppose that $Q$ is an OPT cluster with $\text{cost}(Q, C) \geq \beta \cdot \text{cost}(Q, \mu_C)$ for some $\beta \geq 4$ and we $D^2$-sample a point $p \in Q$. Then, with probability at least $1 - 6/\sqrt{\beta}$, $Q$ becomes $(\beta - 1)$-settled. That is, $\|p - \mu_Q\|^2 \leq ((\beta - 1)/|Q|) \cdot \text{cost}(Q, \mu_C)$ and $\text{cost}(Q, p) \leq \beta \cdot \text{cost}(Q, \mu_C)$.

**Proof.** Let $\beta' \geq \beta$ be the exact approximation factor. i.e. $\text{cost}(Q, C) = \beta' \cdot \text{cost}(Q, \mu)$. We define sets $Q_m$ and $P'_m$:

$$Q_m = \left\{ q \in Q : \|q - \mu_Q\| \leq \frac{\beta - 2}{|Q|} \cdot \text{cost}(Q, \mu_Q) \right\}$$

$$P'_m = \left\{ p \in P : \|p - \mu_Q\| \leq \frac{\beta' - 2}{|Q|} \cdot \text{cost}(Q, \mu_Q) \right\}$$

By definition, we have $Q_m \subseteq Q$ and $Q_m \subseteq P'_m$. However, $P'_m \subseteq Q$ does not hold in general.

**Lemma 3** tells us that $P'_m \cap C = \emptyset$. Otherwise $\beta' \cdot \text{cost}(Q, \mu_Q) = \text{cost}(Q, C) \leq (\beta' - 1) \cdot \text{cost}(Q, \mu_Q)$, which is a contradiction. Furthermore, it holds that $|Q \setminus Q_m| \leq |Q|/(\beta - 2)$. Otherwise $\text{cost}(Q, \mu_Q) > ((|Q|/(\beta - 2)) \cdot ((\beta - 2)/|Q|) \cdot \text{cost}(Q, \mu_Q)$, which is a contradiction. Hence, $|Q_m| \geq 1 - (1/(\beta - 2)) \cdot |Q|$. Let $d_i = \|q_i - \mu_Q\|$ be the distance of the $i$-th point of $Q_m$ from $\mu_Q$, so $\sum_{i=1}^{\lfloor Q_m \rfloor} d_i^2 \leq \text{cost}(Q, \mu_Q)$. By the Cauchy-Schwarz inequality, we have

$$\sum_{i=1}^{\lfloor Q_m \rfloor} d_i \leq \sqrt{|Q_m|} \cdot \sqrt{\sum_{i=1}^{\lfloor Q_m \rfloor} d_i^2} \leq \sqrt{|Q|} \cdot \text{cost}(Q, \mu_Q).$$

Since $P'_m \cap C = \emptyset$, triangle inequality tells us that $\sqrt{\text{cost}(q_i, C)} \geq \sqrt{\text{cost}(\mu_Q, C)} - \sqrt{\text{cost}(q_i, \mu_Q)} \geq \sqrt{((\beta' - 2)/|Q|) \cdot \text{cost}(Q, \mu_Q) - d_i}$ for each point $q_i \in Q_m$. Thus,

$$\text{cost}(Q_m, C) = \sum_{i=1}^{\lfloor Q_m \rfloor} \text{cost}(q_i, C) \geq \sum_{i=1}^{\lfloor Q_m \rfloor} \left( \frac{\beta' - 2}{|Q|} \cdot \text{cost}(Q, \mu_Q) - d_i \right) \geq \sum_{i=1}^{\lfloor Q_m \rfloor} \frac{\beta' - 2}{|Q|} \cdot \text{cost}(Q, \mu_Q) - 2d_i \sqrt{\text{cost}(Q, \mu_Q)} \geq \frac{|Q_m|}{|Q|} \cdot (\beta' - 2) \cdot \text{cost}(Q, \mu_Q).$$

(Aggarwal et al., 2009), Lemma 6 of (Lattanzi & Sohler, 2019).
Lemma 12 allows us to conclude that, with probability at least $1 - \frac{1}{\beta - 2}$, the cost of the $i$-th $\beta$-settled,.

Recall that every legal $\beta$ is smaller than $k^{0.3}$, as for larger $\beta$ one cannot obtain strong concentration results. For reasonably small $\alpha$, Lemma 12 allows us to conclude that there are at most $O(k/\sqrt{\alpha^{2/3}}) = O(k/\sqrt{\alpha})$ clusters that are $\alpha^{2/3}$-unsettled, with high probability in $k$. Conditioned on this event, we can expect a stronger multiplicative improvement in one iteration of LocalSearch++. Compared to Lemma 3 of (Lattanzi & Sohler, 2019).

3.2. One step of LocalSearch++

Given the structural analysis of the previous section, we can now analyze the LocalSearch++ procedure. First, we will identify clusters whose removal will not significantly increase the current cost, thus making them good candidates for swapping with the newly sampled center.

We now define subsets of matched and lonely candidate centers $M \subseteq C$ and $L \subseteq C$. The notion of lonely centers came from Kanungo et al. (2004). To describe the same subsets, Lattanzi and Sohler (2019) used the notation $H$ and $L$, while we use $M$ and $L$. For an illustration of these definitions, see Fig. 2.

Definition 13 (M and L candidates). We assign OPT centers $c^* \in C^*$ to candidate centers $c \in C$, and define the notion of matched ($M$) and lonely ($L$) on candidates based on assignment outcome. For each $c^* \in C^*$, assign $c^*$ to the closest $c \in C$, breaking ties arbitrarily. We say candidate $c \in C$ is matched if there is exactly one $c^* \in C^*$ assigned to it and we call $c^*$ the mate of $c$. We say candidate $c \in C$ is lonely if there is no $c^* \in C^*$ assigned to it.

We define $M \subseteq C$ as the set of matched candidates and $L \subseteq C$ as the set of lonely candidates. We sometimes
overload notation and write \((c, c^*) \in M \) if \(c \in C \) is a matched candidate center with mate \(c^* \in C^* \).

**Observation 14.** Since \(|C| = |C^*| = k\), a counting argument tells us that \(k - |M| \leq 2|L|\).

We now define reassignment costs for candidate centers \(c \in M \cup L \subseteq C \) and the notion of \(\gamma\)-good OPT centers.

Informally, OPT center \(c^* \in C^* \) is \(\gamma\)-good if selecting a random point in \(Q_{c^*} \) and removing a suitable candidate \(c \in C \) reduces a “sufficient” fraction of the current clustering cost with a constant probability.

**Definition 15** (Reassignment costs).

If \((c, c^*) \in M\),

reassign\((P, C, c)\)

\[ = \text{cost}(P \setminus Q_{c^*}, C \setminus \{c\}) - \text{cost}(P \setminus Q_{c^*}, C) \]

If \(c \in L\),

reassign\((P, C, c)\) = \text{cost}(P \setminus \{c\}) - \text{cost}(P, C)

We will use the following lemma about reassignment costs, proven in Lemma 4 of ([Lattanzi & Sohler, 2019]).

**Lemma 16.** For \(c \in M \cup L\), with \(P_c\) as the points assigned to \(c\),

reassign\((P, C, c)\) \leq \frac{21}{100} \text{cost}(P_c, C) + 24 \text{cost}(P_c, C^*)

**Definition 17** (\(M\)-\(\gamma\)-good and \(L\)-\(\gamma\)-good).

We say that \(c^* \in C^* \cap M\) with mate \(c \in C\) is \(M\)-\(\gamma\)-good if

\[
\text{cost}(Q_{c^*}, C) - \text{reassign}(P, C, c) - 100 \cdot \text{cost}(Q_{c^*}, c^*) > \frac{\gamma}{10^4 k} \cdot \text{cost}(P, C).
\]

We say that \(c^* \in C^* \setminus M\) is \(L\)-\(\gamma\)-good if

\[
\text{cost}(Q_{c^*}, C) - \min_{c \in L} \text{reassign}(P, C, c) - 100 \cdot \text{cost}(Q_{c^*}, c^*) > \frac{\gamma}{10^4 k} \cdot \text{cost}(P, C).
\]

**Claim 18.** Let \(Q\) be a \(M\)-\(\gamma\)-good or \(L\)-\(\gamma\)-good cluster and we \(D^2\)-sample a point \(q \in Q\). Then, with probability at least \(2/5\), we have \(\text{cost}(Q, q) \leq 100 \cdot \text{cost}(Q, \mu_Q)\).

**Proof.** Let \(C\) denotes the current set of candidate centers. Suppose cluster \(Q\) is \(M\)-\(\gamma\)-good. Then,

\[
\text{cost}(Q, C) > \text{reassign}(P, C, c) + 100 \cdot \text{cost}(Q, \mu_Q) + \frac{\gamma}{10^4 k} \cdot \text{cost}(P, C) \geq 100 \cdot \text{cost}(Q, \mu_Q)
\]

By Lemma 10, we have \(\text{cost}(Q, q) \leq 100 \cdot \text{cost}(Q, \mu_Q)\) with probability at least \(1 - \frac{6}{\sqrt{10^4 k}} = 2/5\). The same argument holds when \(Q\) is an \(L\)-\(\gamma\)-good cluster by applying the definition of \(L\)-\(\gamma\)-good instead.

Conditioned on our main structural insight (Lemma 12), we sample a point from an \(M\)-\(\sqrt{\beta}\)-good or \(L\)-\(\sqrt{\beta}\)-good cluster \(Q\) with constant probability, for every legal \(\beta \in B\) such that \(4 \leq \beta \leq \alpha^{2/3}\) and \(\alpha \geq 10^9\). When this happens, the sampled point \(s\) satisfies \(\text{cost}(Q, s) \leq 100 \cdot \text{cost}(Q, \mu_Q)\) with constant probability. In that case, the definition of \(M\)-\(\sqrt{\beta}\)-good and \(L\)-\(\sqrt{\beta}\)-good implies the existence of a candidate \(t \in C\) with \(\text{cost}(P, C \setminus \{t\} \cup \{s\}) \leq (1 - \frac{\sqrt{\beta}}{10^4 k}) \cdot \text{cost}(P, C)\), so LocalSearch++ makes progress.

Similar to the analysis of Lattanzi and Sohler (2019), we partition the space of possible events into whether \(\sum_{(c, c^*) \in M} \text{cost}(Q_{c^*}, C) \geq \text{cost}(P, C) / 2\), or not. In each case, we argue that the probability of sampling a point contained in a \(M\)-\(\sqrt{\beta}\)-good or \(L\)-\(\sqrt{\beta}\)-good cluster happens with a positive constant probability for a suitable legal \(\beta\). We first refine Lemma 5 of ([Lattanzi & Sohler, 2019]).

**Lemma 19.** Suppose \(2 \cdot \sum_{(c, c^*) \in M} \text{cost}(Q_{c^*}, C) \geq \text{cost}(P, C)\) \(\alpha \cdot \text{cost}(P, C^*)\) for \(\alpha \geq 10^9\). Let \(4 \leq \beta \leq \alpha^{2/3}\) be arbitrary. If there are at most \((30k)^{1/\beta}\) clusters that are \(\beta\)-unsettled, then

\[
\sum_{(c, c^*) \in M, c^* \in M \setminus \sqrt{\beta}} \text{cost}(Q_{c^*}, C) \geq \frac{1}{500} \cdot \text{cost}(P, C).
\]

Lemma 19 tells us that if points in \(M\) have sufficiently large probability mass, then points in \(M \setminus \sqrt{\beta}\)-good clusters hold a constant fraction of the total probability mass.

**Proof.** We show that the probability mass is large by upper bounding the probability mass on its negation. To do this, we partition the summation of \(c^* \notin M \setminus \sqrt{\beta}\)-good into \(\beta\)-settled and \(\beta\)-unsettled. We denote

\[
A = \{(c, c^*) \in M, c^* \notin M \setminus \sqrt{\beta}, c^* \text{ is } \beta\text{-settled}\}
\]

\[
B = \{(c, c^*) \in M, c^* \notin M \setminus \sqrt{\beta}, c^* \text{ is } \beta\text{-unsettled}\}
\]

From Lemma 6, we know that \(C\) pays no more than \((\beta + 1) \cdot \text{cost}(P, C^*)\) for all \(\beta\)-settled clusters. So,

\[
\sum_{A} \text{cost}(Q_{c^*}, C) \leq (\beta + 1) \cdot \text{cost}(P, C^*)
\]

\[
\leq (\alpha^{2/3} + 1) \cdot \text{cost}(P, C^*) \leq \frac{2\alpha^{2/3}}{\alpha} \cdot \text{cost}(P, C)
\]

For clarity, we define using \(\gamma\). Later, we set \(\gamma = \sqrt{\beta}\).
To bound $\sum_B \text{cost}(Q_{c^*}, C)$, recall that $P$ is the set of all points and $Q_{c^*} \subseteq P$ for any $c^* \in C^*$.

\[
\sum_B \text{cost}(Q_{c^*}, C) \\
\leq \sum_B \left( \text{reassign}(P, C, c) + 100 \cdot \text{cost}(Q_{c^*}, c^*) + \frac{\sqrt{\beta}}{10^4 k} \cdot \text{cost}(P, C) \right)
\leq \left( \sum_B \text{reassign}(P, C, c) + 100 \cdot \text{cost}(P, C^*) \right) + \frac{30}{10^4} \cdot \text{cost}(P, C) \\
\leq \frac{21}{100} \cdot \text{cost}(P, C) + 24 \cdot \text{cost}(P, C^*) \\
+ 100 \cdot \text{cost}(P, C^*) + \frac{30}{10^4} \cdot \text{cost}(P, C) \\
\leq \frac{250}{1000} \cdot \text{cost}(P, C)
\]

(\text{Legend}) (\ast): \text{Definition 17}; (\dagger): because there are at most $\frac{30k}{\sqrt{\beta}}$ clusters that are $\beta$-unsettled; (\dagger\dagger): \text{Lemma 16}; (\ast\ast): $\text{cost}(P, C) \geq 10^9 \cdot \text{cost}(P, C^*)$

Thus,

\[
\sum_{(c, c^*) \in M, c^* \in M-\sqrt{\beta}-\text{good}} \text{cost}(Q_{c^*}, C) \\
\geq \left( \frac{1}{2} - \frac{1}{500} - \frac{250}{1000} \right) \cdot \text{cost}(P, C) \geq \frac{1}{500} \cdot \text{cost}(P, C)
\]

Using the same structural insight on $\beta$-unsettled clusters, we now refine Lemma 7 of (Lattanzi & Sohler, 2019). Abusing notation, we use $C^* \setminus M$ to denote the set of optimal cluster centers which don’t have a mate. That is, the point $c \in C$ which $c^*$ is assigned to is assigned to has more than one center of $C^*$ assigned to it.

\textbf{Lemma 20.} Suppose $2 \cdot \sum_{(c, c^*) \in M} \text{cost}(Q_{c^*}, C) < \text{cost}(P, C) = \alpha \cdot \text{cost}(P, C^*)$ for $\alpha \geq 10^9$. Let $4 \leq \beta \leq \alpha^{2/3}$ be arbitrary. If there are at most $(30k) / \sqrt{\beta}$ clusters that are $\beta$-unsettled, then

\[
\sum_{c^* \in C^* \setminus M, c^* \in L-\sqrt{\beta}-\text{good}} \text{cost}(Q_{c^*}, C) \geq \frac{1}{500} \cdot \text{cost}(P, C).
\]

\textbf{Lemma 20} tells us that if points in $C^* \setminus M$ have sufficiently large probability mass, then points in $L-\sqrt{\beta}$-good clusters hold a constant fraction of the total probability mass.

With Lemma 19 and Lemma 20, we can now refine Lemma 3 from (Lattanzi & Sohler, 2019).

\textbf{Lemma 21.} Suppose we have a clustering $C$ with $\text{cost}(P, C) = \alpha \cdot \text{cost}(P, C^*)$ for some $\alpha \geq 10^9$. Assume that for each legal $\beta$, where $\beta \leq \alpha^{2/3}$, there are at most $(30k) / \sqrt{\beta}$ clusters that are $\beta$-unsettled. If we update $C$ to $C'$ in one LocalSearch++ iteration, we have with probability at least $1/2000$:

\[
\text{cost}(P, C') \leq \left( 1 - \min \left\{ \sqrt{\beta}, \frac{k^{0.15}}{2 \cdot 10^4 k} \right\} \right) \cdot \text{cost}(P, C)
\]

\textbf{Proof.} Pick a legal $\beta \in B$ such that $\frac{1}{2} \min \{k^{0.3}, \alpha^{2/3}\} \leq \beta < \min \{k^{0.3}, \alpha^{2/3}\}$. We define $M$ and $L$ candidate centers as in \textbf{Definition 13} and consider the following two cases separately:

1. $\sum_{(c, c^*) \in M} \text{cost}(Q_{c^*}, C) \geq \frac{1}{2} \cdot \text{cost}(P, C)$
2. $\sum_{(c, c^*) \in M} \text{cost}(Q_{c^*}, C) < \frac{1}{2} \cdot \text{cost}(P, C)$

Let $q$ be the $D^2$-sampled point and $c \in C$ be some current candidate center, which we will define later in each case. In both cases (1) and (2), we will show that the pair of points $(q, c)$ will fulfill the condition $\text{cost}(P, C \cup \{q\} \setminus \{c\}) \leq (1 - \sqrt{\beta} / (10^4 k)) \cdot \text{cost}(P, C)$ with some constant probability. The claim follows since the algorithm takes the $c \in C$ that decreases the cost the most and swipes it with the $D^2$-sampled point $q$.

\textbf{Case (1):} $\sum_{(c, c^*) \in M} \text{cost}(Q_{c^*}, C) \geq \frac{1}{2} \cdot \text{cost}(P, C)$

\textbf{Lemma 19} tells us that we sample from a $M-\sqrt{\beta}$-good cluster with probability at least $1/500$. Denote this cluster as $Q_{c^*}$. Then, by \textbf{Claim 18}, the $D^2$-sampled point $q$ satisfies $\text{cost}(Q_{c^*}, q) \leq 100 \cdot \text{cost}(Q_{c^*}, \mu_{Q_{c^*}})$ with probability at least $2/5$. Jointly, with probability at least $2/2500$, we $D^2$-sample a “good” point $q \in Q_{c^*}$ where $(c, c^*) \in M$ and $c^* \in M-\sqrt{\beta}$-good, so

\[
\text{cost}(P, C \cup \{q\} \setminus \{c\}) \\
= \text{cost}(P, C) - (\text{cost}(P, C) - \text{cost}(P, C \cup \{q\} \setminus \{c\})) \\
\leq \text{cost}(P, C) - \left( \text{cost}(P \setminus Q_{c^*}, C) + \text{cost}(Q_{c^*}, C) - (\text{cost}(P \setminus Q_{c^*}, \{c\}) + \text{cost}(Q_{c^*}, \{c\})) \right) \\
= \text{cost}(P, C) - \left( \text{cost}(Q_{c^*}, C) - \text{cost}(P \setminus Q_{c^*}, \{c\}) \right) \\
\leq \text{cost}(P, C) - \left( \text{cost}(Q_{c^*}, C) - \text{cost}(P \setminus Q_{c^*}, \{c, c^*\}) \right) \\
- 100 \cdot \text{cost}(Q_{c^*}, \mu_{Q_{c^*}})
\]
\[\begin{align*}
&\leq \text{cost}(P, C) - \frac{\sqrt{3}}{10^4 \cdot k} \cdot \text{cost}(P, C) \\
&= \left(1 - \frac{\sqrt{3}}{10^4 \cdot k}\right) \cdot \text{cost}(P, C) \\
&\leq \left(1 - \min\{\sqrt{\alpha}, k^{0.10}\} \right) \cdot \text{cost}(P, C)
\end{align*}\]

Case (2): \[\sum_{(e, c') \in M} \text{cost}(Q_{e'}, C) < \frac{1}{2} \cdot \text{cost}(P, C)\]
This is the same as Case (1), but we use Lemma 20 instead of Lemma 19.

From this lemma, we can conclude that if the current approximation factor is very high, we drastically decrease it within just a few steps. In particular, we can show that if we start with an approximation guarantee that is no worse than \(\exp(k^{0.1})\), we can decrease it to just \(O(1)\) within \(\varepsilon k\) steps, with probability \(1 - \exp(-\Omega(k^{0.1}))\). By Markov's inequality, we know that the probability of having an approximation guarantee that is worse than \(\exp(k^{0.1})\) is at most \(\exp(-\Omega(k^{0.1}))\). Our main theorem now follows:

**Theorem 1 (Main theorem).** Let \(k \in \Omega(1/\varepsilon^{20})\) and \(0 < \varepsilon \leq 1\). Suppose we run Algorithm 1 followed by \(\ell = \varepsilon k\) steps of Algorithm 2. We have \(\text{cost}(P, C') \leq (10^{30}/\varepsilon^3) \cdot \text{cost}(P, C^*) \) with probability at least \(1 - \exp(-\Omega(k^{0.1}))\).

### 3.3. Concluding remarks

**Expectation versus high probability** An approximation guarantee in expectation only implies (via Markov inequality) that with a constant probability we get a constant approximation. So, our result is stronger as we get a constant approximation of the optimum cost with a probability of at least \(1 - \exp(-\Omega(k^{0.1}))\). To recover a guarantee in expectation, we can run the algorithm twice:

Let \(C_1\) be the solution obtained by running \(k\)-means++ plus LocalSearch++, let \(C_2\) be the output of another independent run of \(k\)-means++, and let \(\mathcal{E}\) be the event that LocalSearch++ outputs an \(O(1)\)-approximation. Then, the expected cost of \(\min\{\text{cost}(C_1), \text{cost}(C_2)\}\) is

\[
\mathbb{E}[\min\{\text{cost}(C_1), \text{cost}(C_2)\}] \\
\leq \Pr[\mathcal{E}] \cdot \mathcal{O}(1) + (1 - \Pr[\mathcal{E}]) \cdot \mathbb{E}[\text{cost}(C_2)] \\
\leq (1 - \exp(-\Omega(k^{0.1}))) \cdot \mathcal{O}(1) + \exp(-\Omega(k^{0.1})) \cdot \mathcal{O}(\log k) \\
\in \mathcal{O}(1)
\]

- The cube-root of \(\alpha\) in Lemma 21 is precisely why we obtain an approximation factor of \(O(1/\varepsilon^3)\) after \(\varepsilon k\) LocalSearch++ steps, with high probability in \(k\).
- It is not unusual to run \(k\)-means++ multiple times in practice. e.g. See documentation of sklearn.cluster.KMeans.

**Running time** On a \(d\)-dimensional data set consisting of \(n\) data points, a naive implementation of \(k\)-means++ has time complexity \(\mathcal{O}(dnk^2)\) and space complexity \(\mathcal{O}(dn)\). This running time can be improved to \(\mathcal{O}(dnk)\) if each data point tracks its distance to the current closest candidate center in \(C\). This is because \(D^2\)-sampling and subsequent updating this data structure can be done in \(\mathcal{O}(dn)\) time for each iteration of \(k\)-means++.

LocalSearch++ can be implemented in a similar manner where each data point remembers its distance to the closest two candidate centers. Lattanzi and Sohler (Lattanzi & Sohler, 2019) argue that if LocalSearch++ deletes clusters with an average size of \(\mathcal{O}(n/k)\), then an iteration of LocalSearch++ can be performed in an amortized running time of \(\mathcal{O}(dn)\).

However, in the worst case, each iteration of LocalSearch++ can still take \(\mathcal{O}(dnk)\) time. A way to provably improve the worst case complexity is to use more memory. With \(\mathcal{O}(dnk)\) space, each data point can store distances to all \(k\) centers in a binary search tree. Then, each step can be implemented in \(\mathcal{O}(dn \log k)\) time, as updating a binary search tree requires \(\mathcal{O}(\log k)\) time.

### 4. Acknowledgements
We are grateful to Zalan Borsos, Mohsen Ghaffari, and Andreas Krause for their help and discussing this problem with us. In particular, we thank Mohsen Ghaffari for giving us feedback on previous versions of this paper.

### References

Ackermann, M., Märtens, M., Raupach, C., Swierkot, K., Lammersen, C., and Sohler, C. Streamkm++: A clustering algorithm for data streams. *Journal of Experimental Algorithmics (JEA)*, 17:2–1, 2012.

Aggarwal, A., Deshpande, A., and Kannan, R. Adaptive sampling for k-means clustering. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pp. 15–28. Springer, 2009.

Ahmadian, S., Norouzi-Fard, A., Svensson, O., and Ward, J. Better guarantees for k-means and euclidean k-median by primal-dual algorithms. *SIAM Journal on Computing*, (0):FOCS17–97, 2019.

Aloise, D., Deshpande, A., Hansen, P., and Popat, P. Np-hardness of euclidean sum-of-squares clustering. *Machine learning*, 75(2):245–248, 2009.

Arthur, D. and Vassilvitskii, S. k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete
algorithm, pp. 1027–1035. Society for Industrial and Applied Mathematics, 2007.

Awasthi, P., Charikar, M., Krishnaswamy, R., and Sinop, A. K. The hardness of approximation of euclidean k-means. arXiv preprint arXiv:1502.03316, 2015.

Bachem, O., Lucic, M., Hassani, H., and Krause, A. Fast and provably good seedings for k-means. In Advances in neural information processing systems, pp. 55–63, 2016.

Bahmani, B., Moseley, B., Vattani, A., Kumar, R., and Vassilvitskii, S. Scalable k-means++. Proceedings of the VLDB Endowment, 5(7):622–633, 2012.

Bandyapadhyay, S. and Varadarajan, K. On variants of k-means clustering. arXiv preprint arXiv:1512.02985, 2015.

Bhattacharya, A., Jaiswal, R., and Ailon, N. Tight lower bound instances for k-means++ in two dimensions. Theoretical Computer Science, 634:55–66, 2016.

Brunschat, T. and Rögl, H. A bad instance for k-means++. Theoretical Computer Science, 505:19–26, 2013.

Cohen-Addad, V. A fast approximation scheme for low-dimensional k-means. In Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 430–440. SIAM, 2018.

Cohen-Addad, V., Klein, P. N., and Mathieu, C. Local search yields approximation schemes for k-means and k-median in euclidean and minor-free metrics. SIAM Journal on Computing, 48(2):644–667, 2019.

Friggstad, Z., Rezapour, M., and Salavatipour, M. R. Local search yields a ptas for k-means in doubling metrics. SIAM Journal on Computing, 48(2):452–480, 2019.

Jain, K. and Vazirani, V. V. Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and lagrangian relaxation. Journal of the ACM (JACM), 48(2):274–296, 2001.

Kanungo, T., Mount, D. M., Netanyahu, N. S., Piatko, C. D., Silverman, R., and Wu, A. Y. A local search approximation algorithm for k-means clustering. Computational Geometry, 28(2-3):89–112, 2004.

Lattanzi, S. and Sohler, C. A better k-means++ algorithm via local search. In International Conference on Machine Learning, pp. 3662–3671, 2019.

Lee, E., Schmidt, M., and Wright, J. Improved and simplified inapproximability for k-means. Information Processing Letters, 120:40–43, 2017.
Corollary 11 tells us that, with probability at least ~\(1\), which we will then use to prove Proposition 23. We will use the standard Chernoff Bound: \(\sum_{i=1}^{n} X_i\) be independent binary random variables. Let \(X = \sum_{i=1}^{n} X_i\) be their sum and \(\mu = \mathbb{E}(X)\) be the sum's expected value. For any \(\delta \in [0, 1]\), \(P(X \leq (1 - \delta)\mu) \leq \exp(-\delta^2\mu/2)\) and \(P(X \geq (1 + \delta)\mu) \leq \exp(-\delta^2\mu/3)\). Additionally, for any \(\delta \geq 1\), we have \(P(X \geq (1 + \delta)\mu) \leq \exp(-\delta\mu/3)\).

A.1. Concentration Inequalities

We will use the standard Chernoff Bound:

**Theorem 22** (Chernoff bound). Let \(X_1, X_2, \ldots, X_n\) be independent binary random variables. Let \(X = \sum_{i=1}^{n} X_i\) be their sum and \(\mu = \mathbb{E}(X)\) be the sum’s expected value. For any \(\delta \in [0, 1]\), \(P(X \leq (1 - \delta)\mu) \leq \exp(-\delta^2\mu/2)\) and \(P(X \geq (1 + \delta)\mu) \leq \exp(-\delta^2\mu/3)\). Additionally, for any \(\delta \geq 1\), we have \(P(X \geq (1 + \delta)\mu) \leq \exp(-\delta\mu/3)\).

A.2. Proof of Lemma 12

We will use Corollary 11 together with a Chernoff Bound to prove Proposition 23, which we will then use to prove Lemma 12.

**Proposition 23.** After running \(k\)-means++ (for \(k\) steps), let \(C\) denote the set of candidate centers and let \(\alpha = \frac{\text{cost}(P,C)}{\text{opt}} \geq 1\) be the approximation factor. Then, with probability at least \(1 - \exp(-\Omega(k^{0.1}))\), there are at most \(\left(\frac{10k}{\sqrt{\beta}}\right)\) \(\beta\)-unsettled clusters for any legal \(\beta \leq \alpha^{2/3}\).

**Proof.** Define \(\alpha_i = \frac{\text{cost}(P,C)}{\text{opt}}\) as the approximation factor after the \(i\)-th step of \(k\)-means++. Fix an arbitrary legal \(\beta\). Since \(\alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_k = \alpha\), \(\beta \leq \alpha^{2/3}\) implies \(\beta \leq \alpha_i^{2/3}\) for any \(i\). Note that if \(\beta > \alpha^{2/3}\), then the statement vacuously holds.

Let \(X_i\) be an indicator random variable which is 1 if \(\beta \leq \alpha_i^{2/3}\) and we do not increase the number of \(\beta\)-settled clusters by one in the \(i\)-th iteration. Then, \(E[X_iX_1, \ldots, X_{i-1}] \leq \frac{8}{\sqrt{\beta}}\) for any \(X_1, \ldots, X_{i-1}\). This is because if \(\beta \leq \alpha_i^{2/3}\), Corollary 11 tells us that in the \(i\)-th iteration, with probability at least \(1 - \frac{8}{\sqrt{\beta}}\), the new sampled point is from a \(\beta\)-unsettled cluster \(Q\) and makes it \(\beta\)-settled, so the number of \(\beta\)-settled clusters increases by one.

Define random variables \(X = X_1 + \cdots + X_k\) and \(X' = X'_1 + \cdots + X'_k\), where each \(X'_i\) is an independent Bernoulli random variable with success probability \(\frac{8}{\sqrt{\beta}}\). We see that \(E[X'] = \frac{8k}{\sqrt{\beta}}\) and \(X\) is stochastically dominated by \(X'\). By Theorem 22,

\[
P\left(X \geq \frac{10k}{\sqrt{\beta}}\right) \leq P\left(X' \geq \frac{10k}{\sqrt{\beta}}\right) = P\left(X' \geq \frac{5}{4} \cdot E[X']\right) \leq e^{-\frac{\exp \left(\frac{\mathbb{E}[X']^2}{10}\right)}{10\sqrt{\beta}}} \leq e^{-\Theta\left(\frac{k}{\sqrt{\beta}}\right)} \leq e^{-\Theta\left(k^{0.85}\right)}
\]

The last inequality holds because \(\beta \leq k^{0.3}\) for any legal \(\beta \in \mathcal{B}\). Since we start with \(k\) \(\beta\)-unsettled clusters, if \(X \leq \frac{10k}{\sqrt{\beta}}\) and \(\beta \leq \alpha^{2/3}\), then the number of \(\beta\)-unsettled clusters at the end is at most \(k - (k - \frac{10k}{\sqrt{\beta}}) = \frac{10k}{\sqrt{\beta}}\). To complete the proof, we union bound over all \(O(\log k)\) possible values for legal \(\beta \in \mathcal{B}\).

**Lemma 12.** After running \(k\)-means++ (for \(k\) steps) and \(\ell \leq k\) steps of LocalSearch++, let \(C\) denote the set of candidate centers and \(\alpha \geq 1\) be the approximation factor. Then, with probability at least \(1 - \exp(-\Omega(k^{0.1}))\), there are at most \((30k) / \sqrt{\beta}\) clusters that are \(\beta\)-unsettled, for any legal \(\beta \leq \alpha^{2/3}\).

**Proof.** Define \(\alpha_0\) as the approximation factor after \(k\)-means++ and \(\alpha_i\) as the approximation factor after running the local search for additional \(i\) steps. Fix an arbitrary legal \(\beta\). Since \(\alpha_0 \geq \alpha_1 \geq \cdots \geq \alpha_\ell = \alpha\), \(\beta \leq \alpha^{2/3}\) implies \(\beta \leq \alpha_i^{2/3}\) for any \(i\). Note that if \(\beta > \alpha^{2/3}\), then the statement vacuously holds.

Let \(X_i\) be an indicator random variable which is 1 if \(\beta \leq \alpha_i^{2/3}\) and the number of \(\beta\)-unsettled clusters increases in the \(i\)-th iteration of LocalSearch++. Then, \(E[X_i | X_1, \ldots, X_{i-1}] \leq \frac{8}{\sqrt{\beta}}\). This is because if \(\beta \leq \alpha_i^{2/3}\) in the \(i\)-th iteration, then Corollary 11 tells us that, with probability at least \(1 - \frac{8}{\sqrt{\beta}}\), the new sampled point \(p_i\) is from a \(\beta\)-unsettled cluster \(Q\) and adding \(p_i\) to \(C\) would make \(Q\) \(\beta\)-settled. By definition of \(\beta\)-settled, adding or removing a single point from \(C\) can only decrease or increase the number of \(\beta\)-unsettled clusters by at most one. Thus, if LocalSearch++ decides to swap an existing point in \(C\) for \(p_i\), the number of \(\beta\)-unsettled clusters does not increase.
Define random variables $X = X_1 + \cdots + X_l$ and $X' = X'_1 + \cdots + X'_l$, where each $X'_i$ is an independent Bernoulli random variable with success probability $\frac{8}{l^{2/3}}$. We see that $E[X'] = \frac{8k}{l^{2/3}}$ and $X$ is stochastically dominated by $X'$. By Theorem 22,

$$\Pr(X \geq \frac{20k}{\sqrt{\beta}}) \leq \Pr(X' \geq \frac{20k}{\beta}) = \Pr(X' \geq \frac{5k}{2l} \cdot E[X']) \leq e^{\frac{E[X'] \cdot \beta}{2l}} \leq e^{-\Theta(l^{1/6})} \leq e^{-\Theta(k^{0.85})}$$

The last inequality is because $\beta \leq k^{0.3}$ for any legal $\beta \in B$. Proposition 23 tells us at the start of LocalSearch++, with probability at least $1 - e^{-\Omega(k^{0.1})}$, there are at most $\frac{10k}{\sqrt{\beta}}$-settled clusters. If $X \leq \frac{20k}{\sqrt{\beta}}$ and $\beta \leq \alpha^{2/3}$, then the number of $\beta$-unsettled clusters after $l$ LocalSearch++ steps is at most $\frac{10k}{\sqrt{\beta}} + \frac{20k}{\sqrt{\beta}} \leq \frac{30k}{\sqrt{\beta}}$. To complete the proof, we union bound over all $O(\log k)$ possible values for legal $\beta \in B$.

A.3. Proof of Lemma 20

We proof Lemma 20, which is similar to the proof of Lemma 19.

**Lemma 20.** Suppose $2 \cdot \sum_{(c, c^*)} \text{cost}(Q_{c^*}, C) < \text{cost}(P, C) = \alpha \cdot \text{cost}(P, C^*)$ for $\alpha \geq 10^3$. Let $4 \leq \beta \leq \alpha^{2/3}$ be arbitrary. If there are at most $(30k)/\sqrt{\beta}$ clusters that are $\beta$-unsettled, then

$$\sum_{e^* \in C^* \setminus M, \ c^* \in L_{\sqrt{\beta}} \text{-good}} \text{cost}(Q_{e^*}, C) \geq \frac{1}{500} \cdot \text{cost}(P, C),$$

We abuse notation and denote with $C^* \setminus M$ the set of optimal cluster centers which don’t have a mate. That is, the point $c \in C$ which $e^*$ is assigned to is assigned to more than one center of $C^*$. Informally, the proposition states that if points in $C^* \setminus M$ have sufficiently large probability mass, then the probability mass on $L_{\sqrt{\beta}}$-settled clusters is a constant fraction of the total probability mass.

**Proof.** We show that the probability mass is large by upper bounding the probability mass on its negation. To do this, we partition the summation of $c^* \not\in L_{\sqrt{\beta}}$-settled and $\beta$-unsettled:

$$\sum_{e^* \in C^* \setminus M, \ c^* \in L_{\sqrt{\beta}} \text{-good, } c^* \not\in L_{\sqrt{\beta}} \text{-good, } c^* \text{ is } \beta \text{-settled}} \text{cost}(Q_{e^*}, C) = \sum_{e^* \in C^* \setminus M, \ c^* \in L_{\sqrt{\beta}} \text{-good, } c^* \text{ is } \beta \text{-settled}} \text{cost}(Q_{e^*}, C) + \sum_{e^* \in C^* \setminus M, \ c^* \in L_{\sqrt{\beta}} \text{-good, } c^* \text{ is } \beta \text{-unsettled}} \text{cost}(Q_{e^*}, C)$$

From Lemma 6, we know that $C$ pays no more than $(\beta + 1) \cdot \text{cost}(P, C^*)$ for all $\beta$-settled clusters. So,

$$\sum_{e^* \in C^* \setminus M, \ c^* \not\in L_{\sqrt{\beta}} \text{-good, } c^* \text{ is } \beta \text{-settled}} \text{cost}(Q_{e^*}, C) \leq (\beta + 1) \cdot \text{cost}(P, C^*) \leq (\alpha^{2/3} + 1) \cdot \text{cost}(P, C^*) \leq \frac{2\alpha^{2/3}}{\alpha} \cdot \text{cost}(P, C) \leq \frac{1}{500} \cdot \text{cost}(P, C)$$

It remains to bound $\sum_{e^* \in C^* \setminus M, \ c^* \not\in L_{\sqrt{\beta}} \text{-good, } c^* \text{ is } \beta \text{-unsettled}} \text{cost}(Q_{e^*}, C)$. Recall that $P$ is the set of all points and $Q_{c^*} \subseteq P$ for any $c^* \in C^*$. For $c^* \in C^* \setminus M$, if $c^* \not\in L_{\sqrt{\beta}}$-good, then for any $c \in L$, $\text{cost}(Q_{c^*}, C) \leq \text{assign}(P, C, e) + 100 \cdot \text{cost}(Q_{c^*}, c^*) + \sqrt{\beta} \cdot \text{cost}(P, C)$.

$$\sum_{e^* \in C^* \setminus M, \ c^* \not\in L_{\sqrt{\beta}} \text{-good, } c^* \text{ is } \beta \text{-unsettled}} \text{cost}(Q_{e^*}, C) \leq \sum_{c^* \in C^* \setminus M, \ c^* \not\in L_{\sqrt{\beta}} \text{-good, } c^* \text{ is } \beta \text{-unsettled}} \left(\min_{e \in L} \text{assign}(P, C, e) + 100 \cdot \text{cost}(Q_{e^*}, c^*) + \sqrt{\beta} \cdot \text{cost}(P, C)\right)$$

**Definition 17**
Observation 14

Theorem 1

Lemma 21

we show that

Proposition 26

to show that an iteration is successful with

Proposition 27

Theorem 1
.

, we will introduce the notion of a

Lemma 12
, we give a lower

Theorem 1
and

Proposition 28
holds, even if we additionally

Thus,

Using this observation, we can now state the following:

Let

Proof.

1

1

Note, as we condition on (I) and (II), we cannot directly apply Lemma 21 to show that an iteration is successful with probability at least 1/2000. However, the following is still true:

Observation 25. As

1 − \exp(-\Omega(k^{0.1})) \gg 1 − 1/4000, the probability that an iteration of local search is successful, after conditioning on (I) and (II), is still at least 1/4000.

Using this observation, we can now state the following:

Proposition 26. Assume that we run \ell \leq k local search steps and that conditions (I) and (II) hold. Then, with a probability of at least 1 − \exp(-\Omega(\ell)), we will have at least \ell/8000 successes.

Proof. Let

Xi
 denote the indicator variable for the event that the i-th local search step is a success. Note that

\{\Omega, C, c\}
 are not independent. However, it is easy to check that Observation 25 holds, even if we additionally
Algorithm 1

Algorithm 2

max \ell within 10

Assume that the approximation factor after \(N\) successes is at most \(8000\exp(-\Omega(l))\).

Proposition 27. Assume that the conditions (I) and (II) are fulfilled. Then, after \(N_0 = 2 \cdot 10^4 \cdot k^{0.95}\) successful rounds, we obtain a clustering which is no worse than \(k^{0.45}\)-approximate, assuming \(k^{0.45} > 10^9\).

Proof. As we condition on (II), the initial approximation factor is no worse than \(\exp(k^{0.1})\). For the sake of contradiction, assume that the approximation factor after \(N_0\) successes is strictly greater than \(k^{0.45}\). This implies that in each of the first \(N_0\) successful rounds, we improve the approximation by a factor of at least \((1 - \frac{2 \cdot 10^4 \cdot k^{0.05}}{45})\). Thus, the approximation factor after \(N_0\) successes is at most

\[
\left(1 - \frac{k^{0.15}}{2 \cdot 10^4 k}\right)^{2 \cdot 10^4 \cdot k^{0.95}} \cdot \exp(k^{0.1}) = \left(1 - \frac{k^{-0.85}}{2 \cdot 10^4}\right)^{2 \cdot 10^4 \cdot k^{0.95}} \cdot \exp(k^{0.1})
\]

\[
\leq \exp \left(-\frac{k^{-0.85}}{2 \cdot 10^4} \cdot 2 \cdot 10^4 \cdot k^{0.95}\right) \cdot \exp(k^{0.1})
\]

\[
\leq \exp(-k^{0.1} + k^{0.1}) \leq 1 \leq k^{0.45},
\]

a contradiction.

Proposition 28. Assume that conditions (I) and (II) are fulfilled. We define \(\gamma_i := \frac{k^{0.45}}{2} \cdot \epsilon_i\). Furthermore, for \(i \geq 1\), let \(N_i := \frac{2 \cdot 10^4 \cdot k}{\sqrt{\gamma_i}}\). Then, for each \(R \geq 0\), after \(\sum_{i=0}^{R} N_i\) successes, we have a \(\max\{\gamma_R, 10^9\}\)-approximation.

Proof. We prove the statement by induction on \(R\). For \(R = 0\), the statement directly follows from Proposition 27. Now, let \(R > 0\) be arbitrary. We assume that the statement holds for \(R\), and we show that this implies that the statement holds for \(R + 1\). For the sake of contradiction, assume that the statement does not hold for \(R + 1\), i.e., the approximation is strictly worse than \(\max\{\gamma_{R+1}, 10^9\}\) after an additional \(N_{R+1}\) successful rounds. In particular, this would mean that we never achieve a \(10^9\)-approximation. Thus, in each of the additional \(N_{R+1}\) successful iterations, we would improve the solution by a factor of at least \((1 - \frac{2 \cdot 10^4 \cdot k^{0.15}}{45})\). As we started with an approximation factor no worse than \(\gamma_R\), the approximation factor after \(N_{R+1}\) successful rounds can be upper bounded by

\[
\left(1 - \frac{\sqrt{\gamma_{R+1}}}{2 \cdot 10^4 k}\right)^{2 \cdot 10^4 k} \cdot \gamma_R \leq \exp \left(-\frac{\gamma_{R+1}}{2 \cdot 10^4 k} \cdot \frac{2 \cdot 10^4 k}{\sqrt{\gamma_{R+1}}}\right) \cdot \gamma_R
\]

\[
\leq e^{-1} \cdot \gamma_R < \gamma_{R+1},
\]

a contradiction.

Finally, we can prove Theorem 1.

Theorem 1 (Main theorem). Let \(k \in \Omega(1/\epsilon^{20})\) and \(0 < \epsilon \leq 1\). Suppose we run Algorithm 1 followed by \(\ell = \epsilon k\) steps of Algorithm 2. We have \(\text{cost}(P, C) \leq \left(\frac{10^9}{\epsilon^3}\right) \cdot \text{cost}(P, C^*)\) with probability at least \(1 - \exp(-\Omega(k^{0.1}))\).

Proof of Theorem 1. First, recall that we can assume that conditions (I) and (II) are fulfilled, which holds with probability \(1 - \exp(-\Omega(k^{0.1}))\). Let \(\alpha_0 \leq \exp(k^{0.1})\) be our approximation factor after the execution of \(k\)-\text{means++}. From Proposition 28, we know that after \(\sum_{i=0}^{R} N_i\) successful iterations we have an approximation factor of at most \(\gamma_R = \max\left\{\frac{k^{0.45}}{2}, 10^9\right\}\). Setting \(R = \log_2(k^{0.45}/\epsilon^3) - 3 \log_2(32 \cdot 10^8)\), we get that \(\gamma_R \leq \frac{k^{0.45}}{2} \leq \frac{10^9}{\epsilon^3}\). For the number of successful iterations needed, we have:

\[
\sum_{i=0}^{\log_2(k^{0.45}/\epsilon^3) - 3 \log_2(32 \cdot 10^8)} N_i \leq 2 \cdot 10^4 \cdot k^{0.95} + \sum_{i=1}^{\log_2(k^{0.45}/\epsilon^3) - 3 \log_2(32 \cdot 10^8)} 2 \cdot 10^4 \cdot k^{0.45} / 2
\]
$k$-means++: constant approximation

$$\leq 2 \cdot 10^4 \cdot k^{0.95} + 2 \cdot 10^4 \cdot k \sum_{i=1}^{\log_2(k^{0.45} \varepsilon^3) - 3 \log_2(32 \cdot 10^8)} \frac{1}{2^{i/k^{0.45}}}$$

$$\leq 2 \cdot 10^4 \cdot k^{0.95} + 2 \cdot 10^4 \cdot k \cdot \frac{1}{1 - 1/\sqrt{2}} \cdot 2^{(\log_2(k^{0.45} \varepsilon^3) - 3 \log_2(32 \cdot 10^8))/3} \cdot \frac{1}{\sqrt{k^{0.45}}}$$

$$\leq 2 \cdot 10^4 \cdot k^{0.95} + \frac{\varepsilon k}{16 \cdot 10^3} \leq \frac{\varepsilon k}{8000}.$$  

By Proposition 26, we can conclude that within $\varepsilon k$ steps of local search, at least $\varepsilon k/8000$ are successful with probability at least $1 - \exp(-\Omega(k^{0.1}))$, thus proving the theorem.  
