Distributed and Provably Good Seedings for k-Means in Constant Rounds

Olivier Bachem [1] **Mario Lucic** [1] **Andreas Krause** [1]

Abstract

The k−means++ algorithm is the state of the art algorithm to solve k-Means clustering problems as the computed clusterings are $\mathcal{O}(\log k)$ competitive in expectation. However, its seeding step requires k inherently sequential passes through the full data set making it hard to scale to massive data sets. The standard remedy is to use the k−means‖ algorithm which reduces the number of sequential rounds and is thus suitable for a distributed setting. In this paper, we provide a novel analysis of the k−means‖ algorithm that bounds the expected solution quality for any number of rounds and oversampling factors greater than k, the two parameters one needs to choose in practice. In particular, we show that k−means‖ provides *provably good* clusterings even for a small, constant number of iterations. This theoretical finding explains the common observation that k−means‖ performs extremely well in practice even if the number of rounds is low. We further provide a hard instance that shows that an *additive* error term as encountered in our analysis is inevitable if less than $k-1$ rounds are employed.

1. Introduction

Over the last several years, the world has witnessed the emergence of data sets of an unprecedented scale across different scientific disciplines. This development has created a need for scalable, distributed machine learning algorithms to deal with the increasing amount of data. In this paper, we consider large-scale clustering or, more specifically, the task of finding provably good seedings for k-Means in a massive data setting.

Seeding — the task of finding initial cluster centers — is critical to finding good clusterings for k-Means. In fact, the *seeding* step of the state of the art algorithm k−means++ (Arthur & Vassilvitskii, 2007) provides the theoretical guarantee on the solution quality while the subsequent refinement using *Lloyd's algorithm* (Lloyd, 1982) only guarantees that the quality does not deteriorate. While the k−means++ seeding step guarantees a solution that is $\mathcal{O}(\log k)$ competitive with the optimal solution in expectation, it also requires k inherently sequential passes through the data set. This makes it unsuitable for the massive data setting where the data set is distributed across machines and computation has to occur in parallel.

As a remedy, Bahmani et al. (2012) propose the k−means‖ algorithm which produces seedings for k-Means with a reduced number of sequential iterations. Whereas k−means++ only samples a single cluster center in each of k rounds, k−means‖ samples in expectation ℓ points in each of t iterations. Provided t is small enough, this makes k−means‖ suitable for a distributed setting as the number of synchronizations is reduced.

Our contributions. We provide a novel analysis of k−means‖ that bounds the expected solution quality for any number of rounds t and any oversampling factor $\ell \geq k$, the two parameters that need to be chosen in practice. Our bound on the expected quantization error includes both a "traditional" *multiplicative* error term based on the optimal solution as well as a scale-invariant *additive* error term based on the variance of the data. The key insight is that this additive error term vanishes at a rate of $\left(\frac{k}{e\ell}\right)^t$ if t or ℓ is increased. This shows that k−means‖ provides *provably good* clusterings even for a small, constant number of iterations and explains the commonly observed phenomenon that k−means‖ works very well even for small t.

We further provide a hard instance on which k−means‖ provably incurs an additive error based on the variance of the data and for which an exclusively multiplicative error guarantee cannot be achieved. This implies that an additive error term such as the one in our analysis is in fact necessary if less than $k-1$ rounds are employed.

2. Background & related work

k-Means clustering. Let $\mathcal{X}$ denote a set of points in $\mathbb{R}^d$. The *k-Means clustering problem* is to find a set C of k cluster centers in $\mathbb{R}^d$ that minimizes the quantization error

$$\phi_{\mathcal{X}}(C) = \sum_{x\in\mathcal{X}} \mathrm{d}(x, C)^2 = \sum_{x\in\mathcal{X}} \min_{q\in C} \|x - q\|_2^2.$$

[1] Department of Computer Science, ETH Zurich. Correspondence to: Olivier Bachem <olivier.bachem@inf.ethz.ch>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017. Copyright 2017 by the author(s).

Algorithm 1 `k-means++` seeding

```
Require: weighted data set $(\mathcal{X}, w)$, number of clusters $k$
1: $C \leftarrow$ sample single $x \in \mathcal{X}$ with probability $\frac{w_x}{\sum_{x' \in \mathcal{X}} w_{x'}}$
2: for $i = 2, \ldots, k$ do
3:     Sample $x \in \mathcal{X}$ with probability $\frac{w_x \, \mathrm{d}(x,C)^2}{\sum_{x' \in \mathcal{X}} w_{x'} \, \mathrm{d}(x',C)^2}$
4:     $C \leftarrow C \cup \{x\}$
5: Return $C$
```

We denote the *optimal quantization error* by $\phi_{\mathrm{OPT}}(\mathcal{X})$ while the *variance* of the data is defined as $\mathrm{Var}(\mathcal{X}) = \phi_{\mathcal{X}}(\{\mu(\mathcal{X})\})$ where $\mu(\mathcal{X})$ is the mean of $\mathcal{X}$.

`k-means++` seeding. Given a data set $\mathcal{X}$ and any set of cluster centers $C \subset \mathcal{X}$, the *D^2-sampling strategy* selects a new center by sampling each point $x \in \mathcal{X}$ with probability

$$p(x) = \frac{\mathrm{d}(x, C)^2}{\sum_{x' \in \mathcal{X}} \mathrm{d}(x', C')^2}.$$

The seeding step of `k-means++` (Arthur & Vassilvitskii, 2007), detailed for potentially weighted data sets in Algorithm 1, selects an initial cluster center uniformly at random and then sequentially adds $k - 1$ cluster centers using D^2 sampling whereby C is always the set of previously sampled centers. Arthur & Vassilvitskii (2007) show that the solution quality $\phi_{\texttt{k-means++}}$ of `k-means++` seeding is bounded in expectation by

$$\mathbb{E}[\phi_{\texttt{k-means++}}] \leq 8\,(\log_2 k + 2)\,\phi_{\mathrm{OPT}}(\mathcal{X}).$$

The computational complexity of `k-means++` seeding is $\mathcal{O}(nkd)$ where n is the number of data points and d the dimensionality. Unfortunately, the iterations in `k-means++` seeding are inherently sequential and, as a result, the algorithm requires k full passes through the data. This makes the algorithm unsuitable for the distributed setting.

`k-means`$\|$ seeding. As a remedy, Bahmani et al. (2012) propose the algorithm `k-means`$\|$ which aims to reduce the number of sequential iterations. The key component of `k-means`$\|$ is detailed in Algorithm 2 in what we call *`k-means`$\|$ overseeding*: First, a data point is sampled as the first cluster center uniformly at random. Then, in each of t sequential rounds, each data point $x \in \mathcal{X}$ is independently sampled with probability $\min\left(1, \frac{\ell\,\mathrm{d}(x,C)^2}{\phi_{\mathcal{X}}(C)}\right)$ and added to the set of sampled centers C at the end of the round. The parameter $\ell \geq 1$ is called the *oversampling factor* and determines the expected number of sampled points in each iteration.

Algorithm 2 `k-means`$\|$ overseeding

```
Require: data set $\mathcal{X}$, # rounds $t$, oversampling factor $\ell$
1: $C \leftarrow$ sample a point uniformly at random from $\mathcal{X}$
2: for $i = 1, 2, \ldots, t$ do
3:     $C' \leftarrow \emptyset$
4:     for $x \in \mathcal{X}$ do
5:         Add $x$ to $C'$ with probability $\min\left(1, \frac{\ell\,\mathrm{d}(x,C)^2}{\phi_{\mathcal{X}}(C)}\right)$
6:     $C \leftarrow C \cup C'$
7: Return $C$
```

Algorithm 3 `k-means`$\|$ seeding

```
Require: data set $\mathcal{X}$, # rounds $t$, oversampling factor $\ell$
1: $B \leftarrow$ Result of Algorithm 2 applied to $(\mathcal{X}, t, \ell)$
2: for $c \in B$ do
3:     $\mathcal{X}_c \leftarrow$ points $x \in \mathcal{X}$ whose closest center in $B$ is $c$
       (ties broken arbitrarily but consistently)
4:     $w_c \leftarrow |\mathcal{X}_c|$
5: $C \leftarrow$ Result of Algorithm 1 applied to $(B, w)$
6: Return $C$
```

At the end of Algorithm 2, one obtains an *oversampled solution* with $t\ell$ cluster centers in expectation. The full `k-means`$\|$ seeding algorithm as detailed in Algorithm 3 reduces such a solution to k centers as follows: First, each of the centers in the oversampled solution is weighted by the number of data points which are closer to it than the other centers. Then, `k-means++` seeding is run on the weighted oversampled solution to produce a set of k final centers. The total computational complexity of Algorithm 3 is $\mathcal{O}(nt\ell d)$ in expectation.

The key intuition behind `k-means`$\|$ is that, if we choose a large oversampling factor ℓ, the number of rounds t can be small — certainly much smaller than k, preferably even constant. The step in lines 4 and 5 in Algorithm 2 can be distributed over several machines and after each round the set C can be synchronized. Due to the low number of synchronizations (i.e., rounds), Algorithm 2 can be efficiently run in a distributed setting.[1]

Other related work. Celebi et al. (2013) provide an overview over different seeding methods for k-Means. D^2-sampling and `k-means++` style algorithms have been independently studied by both Ostrovsky et al. (2006) and Arthur & Vassilvitskii (2007). This research direction has led to polynomial time approximation schemes based on D^2-sampling (Jaiswal et al., 2014; 2015), constant factor approximations based on sampling more than k centers (Ailon et al., 2009; Aggarwal et al., 2009) and the analysis of hard instances (Arthur & Vassilvitskii, 2007; Brunsch & Röglin, 2011). Recently, algorithms to approximate `k-means++` seeding based on Markov Chain Monte Carlo have been proposed by Bachem et al. (2016b;a). Finally, `k-means++` has been used to construct coresets — small data set summaries — for k-Means clustering (Lucic et al., 2016; Bachem et al., 2015; Fichtenberger et al., 2013; Ackermann et al., 2012) and Gaussian mixture models (Lucic et al., 2017).

[1] A popular choice is the MLLib library of Apache Spark (Meng et al., 2016) which uses `k-means`$\|$ by default.

3. Intuition and key results

In this section, we provide the intuition and the main results behind our novel analysis of k−means‖ and defer the formal statements and the formal proofs to Section 4.

3.1. Solution quality of k-means‖

Solution quality of Algorithm 2. We first consider Algorithm 2 as it largely determines the final solution quality. Algorithm 3 with its use of k−means++ to obtain the final k cluster centers, only adds an additional $\mathcal{O}(\log k)$ factor as shown in Theorem 1. Our key result is Lemma 4 (see Section 4) which guarantees that, for $\ell \geq k$, the expected error of solutions computed by Algorithm 2 is at most

$$\mathbb{E}[\phi_{\mathcal{X}}(C)] \leq 2\left(\frac{k}{e\ell}\right)^t \operatorname{Var}(\mathcal{X}) + 26\phi_{\mathrm{OPT}}(\mathcal{X}). \quad (1)$$

The first term may be regarded as a scale-invariant *additive error*: It is *additive* as it does not depend on the optimal quantization error $\phi_{\mathrm{OPT}}(\mathcal{X})$. It is *scale-invariant* since both the variance and the quantization error are scaled by λ^2 if we scale the data set $\mathcal{X}$ by $\lambda > 0$. The second term is a "traditional" *multiplicative* error term based on the optimal quantization error.

Given a fixed oversampling factor ℓ, the additive error term decreases exponentially if the number of rounds t is increased. Similarly, for a fixed number of rounds t, it decreases polynomially at a rate $\mathcal{O}\left(\frac{1}{\ell^t}\right)$ if the over sampling factor ℓ is increased. This result implies that even for a constant number of rounds one may obtain good clusterings by increasing the oversampling factor ℓ. This explains the empirical observation that often even a low number of rounds t is sufficient and that increasing ℓ increases the solution quality (Bahmani et al., 2012). The practical implications of this result are non-trivial: Even for the choice of $t = 5$ and $\ell = 5k$ one retains at most 0.0004% of the variance as an additive error. Furthermore, state of the art uniform deviation bounds for k-Means include a similar additive error term (Bachem et al., 2017).

Comparison to previous result. Bahmani et al. (2012) show the following result: Let C be the set returned by Algorithm 2 with t rounds. For $\alpha = \exp\left(-(1 - e^{-\ell/(2k)})\right) \approx e^{-\frac{\ell}{2k}}$, Corollary 3 of Bahmani et al. (2012) bounds the expected quality of C by

$$\mathbb{E}[\phi_{\mathcal{X}}(C)] \leq \left(\frac{1+\alpha}{2}\right)^t \psi + \frac{16}{1-\alpha}\phi_{\mathrm{OPT}}(\mathcal{X}), \quad (2)$$

where ψ denotes the quantization error of $\mathcal{X}$ based on the first, uniformly sampled center in k−means‖. The key difference compared to our result is as follows: First, even as we increase ℓ, the factor α is always non-negative. Hence, regardless of the choice of ℓ, the additive ψ term is reduced by at most $\frac{1}{2}$ per round.[2] This means that, given the analysis in Bahmani et al. (2012), one would always obtain a constant additive error for a constant number of rounds t, even as ℓ is increased.

Guarantee for Algorithm 3. Our main result — Theorem 1 — bounds the expected quality of solutions produced by Algorithm 3. As in Bahmani et al. (2012), one loses another factor of $\mathcal{O}(\ln k)$ compared to (1) due to Algorithm 3.

Theorem 1. *Let $k \in \mathbb{N}$, $t \in \mathbb{N}$ and $\ell \geq k$. Let $\mathcal{X}$ be a data set in $\mathbb{R}^d$ and C be the set returned by Algorithm 3. Then,*

$$\mathbb{E}[\phi_{\mathcal{X}}(C)] \leq \mathcal{O}\left(\left(\frac{k}{e\ell}\right)^t \ln k\right) \operatorname{Var}(\mathcal{X}) + \mathcal{O}(\ln k)\phi_{\mathrm{OPT}}(\mathcal{X}).$$

3.2. A hard instance for k-means‖

We consider the case $t < k-1$ which captures the scenario where k−means‖ is useful in practice as for $t \geq k$ one may simply use k−means++ instead.

Theorem 2. *For any $\beta > 0$, $k \in \mathbb{N}$, $t < k - 1$ and $\ell \geq 1$, there exists a data set $\mathcal{X}$ of size $2(t+1)$ such that*

$$\mathbb{E}[\phi_{\mathcal{X}}(C)] \geq \frac{1}{4}(4\ell t)^{-t} \operatorname{Var}(\mathcal{X}),$$

where C is the output of Algorithm 2 or Algorithm 3 applied to $\mathcal{X}$ with t and ℓ. Furthermore,

$$\operatorname{Var}(\mathcal{X}) > 0, \quad \phi_{\mathrm{OPT}}(\mathcal{X}) = 0 \quad \textit{and} \quad n\Delta^2 \leq \beta$$

where Δ is the largest distance between any points in $\mathcal{X}$.

Theorem 2 shows that there exists a data set on which k−means‖ provably incurs a non-negligible error even if the optimal quantization error is zero. This implies that k−means‖ with $t < k - 1$ cannot provide a multiplicative guarantee on the expected quantization error for general data sets. We thus argue that an additive error bound such as the one in Theorem 1 is required. We note that the upper bound in (1) and the lower bound in Theorem 2 exhibit the same $\frac{1}{\ell^t}$ dependence on the oversampling factor ℓ for a given number of rounds t.

Furthermore, Theorem 2 implies that, for *general data sets*, k−means‖ cannot achieve the *multiplicative error* of $\mathcal{O}(\log k)$ in expectation as claimed by Bahmani et al. (2012).[3] In particular, if the optimal quantization error is

[2] Note that $\mathbb{E}[\psi] \leq 2\operatorname{Var}(\mathcal{X})$ (Arthur & Vassilvitskii, 2007).

[3] To see this, let $\psi = \phi_{\mathcal{X}}(c_1)$ be the quantization error of the first sampled center in Algorithm 2 and choose β small enough such that the choice of $t \in \mathcal{O}(\log \psi)$ leads to $t < k - 1$. For $\mathcal{X}$ in Theorem 2, $\phi_{\mathrm{OPT}}(\mathcal{X}) = 0$ which implies that the desired multiplicative guarantee would require $\mathbb{E}[\phi_{\mathcal{X}}(C)] = 0$. However, the non-negligible, additive error in Theorem 2 and $\operatorname{Var}(\mathcal{X}) > 0$ implies that $\mathbb{E}[\phi_{\mathcal{X}}(C)] > 0$.

zero, then `k-means`$\|$ would need to return a solution with quantization error zero. While we are guaranteed to remove a constant fraction of the error in each round, the number of required iterations may be unbounded.

4. Theoretical analysis

Proof of Theorem 1. The proof is divided into four steps: First, we relate `k-means`$\|$-style oversampling to `k-means++`-style D^2-sampling in Lemmas 1 and 2. Second, we analyze a single iteration of Algorithm 2 in Lemma 3. Third, we bound the expected solution quality of Algorithm 2 in Lemma 4. Finally, we use this to bound the expected solution quality of Algorithm 3 in Theorem 1.

Lemma 1. *Let A be a finite set and let $f : 2^A \to \mathbb{R}$ be a set function that is non-negative and monotonically decreasing, i.e., $f(V) \geq f(U) \geq 0$, for all $V \subseteq U$.*

Let P be a probability distribution where, for each $a \in A$, E_a denotes an independent event that occurs with probability $q_a \in [0, 1]$. Let C be the set of elements $a \in A$ for which the event E_a occurs.

Let Q be the probability distribution on A where a single $a \in A$ is sampled with probability $q_a / \sum_{a\in A} q_a$.

Then, with $\emptyset$ denoting the empty set, we have that

$$\mathbb{E}_P[f(C)] \leq \mathbb{E}_Q[f(\{a\})] + e^{-\sum_{a\in A} q_a} f(\emptyset).$$

Proof. To prove the claim, we first construct a series of sub-events of the events $\{E_a\}_{a\in A}$ and then use them to recursively bound $\mathbb{E}_P[f(C)]$.

Let $m \in \mathbb{N}$. For each $a \in A$, let i_a be an independent random variable drawn uniformly at random from $\{1, 2, \dots, m\}$. For each $a \in A$ and $i = 1, 2, \dots, m$, let F_{ai} be an independent event that occurs with probability

$$\mathbb{P}[F_{ai}] = \left(1 - \frac{q_a}{m}\right)^{i-1}.$$

For each $a \in A$ and $i = 1, 2, \dots, m$, denote by E_{ai} the event that occurs if $i = i_a$ and both E_a and F_{ai} occur. By design, all these events are independent and thus

$$\mathbb{P}[E_{ai}] = \mathbb{P}[E_a]\mathbb{P}[F_{ai}]\mathbb{P}[i_a = i] = \frac{q_a}{m}\left(1 - \frac{q_a}{m}\right)^{i-1} \quad (3)$$

for each $a \in A$ and $i = 1, 2, \dots, m$. Furthermore, for any $a, a' \in A$ with $a \neq a'$ and any $i, i' \in \{1, 2, \dots, m\}$, the events E_{ai} and $E_{a'i'}$ are independent.

For $i = 1, 2, \dots, m$ let G_i be the event that none of the events $\{E_{ai'}\}_{a\in A, i'\leq i}$ occur, i.e.,

$$G_i = \bigcap_{i'\leq i} \bigcap_{a\in A} \overline{E_{ai'}}$$

where $\overline{A}$ denotes the complement of A. For convenience, let G_0 be the event that occurs with probability one.

Let $(a_1, a_2, \dots, a_{|A|})$ be any enumeration of A. For $i = 1, 2, \dots, m$ and $j = 1, 2, \dots, |A|+1$, define the event

$$G_{i,j} = G_{i-1} \cap \bigcap_{0<j'<j} \overline{E_{a_{j'}i}}.$$

We note that by definition $G_{i,1} = G_{i-1}$ and $G_{i,|A|+1} = G_i$ for $i = 1, 2, \dots, m$.

For $i = 1, 2, \dots, m$ and $j = 1, 2, \dots, |A|$, we have

$$\begin{aligned}\mathbb{E}[f(C)|G_{i,j}] =& \mathbb{P}\left[E_{a_j i} \mid G_{i,j}\right]\mathbb{E}\left[f(C) \mid E_{a_j i} \cap G_{ij}\right] \\ &+ \mathbb{P}\left[\overline{E_{a_{j'} i}} \mid G_{ij}\right]\mathbb{E}[f(C) \mid G_{i,j+1}].\end{aligned} \quad (4)$$

We now bound the individual terms. The event $G_{i,j}$ implies that the events $\{E_{a_j i'}\}_{i'<i}$ did not occur. Furthermore, $E_{a_j i}$ is independent of the events $\{E_{a_{j'} i'}\}_{i'=1,2,\dots,m}$ for $j' \neq j$. Hence, we have

$$\begin{aligned}\mathbb{P}\left[E_{a_j i} \mid G_{i,j}\right] &= \mathbb{P}\left[E_{a_j i} \mid G_0 \cap \bigcap_{i'<i} \overline{E_{a_j i'}}\right] \\ &= \frac{\mathbb{P}\left[E_{a_j i}\right]}{\mathbb{P}\left[G_0 \cap \bigcap_{i'<i} \overline{E_{a_j i'}}\right]} \\ &= \frac{\mathbb{P}\left[E_{a_j i}\right]}{1 - \mathbb{P}\left[\bigcup_{i'<i} E_{a_j i'}\right]} \\ &= \frac{\mathbb{P}\left[E_{a_j i}\right]}{1 - \sum_{i'<i} \mathbb{P}\left[E_{a_j i'}\right]},\end{aligned} \quad (5)$$

where the last equality follows since the events $\{E_{a_j i'}\}_{i'<i}$ are disjoint. Using (3), we observe that $\sum_{i'<i} \mathbb{P}\left[E_{a_j i'}\right]$ is a sum of a finite geometric series and we have

$$\begin{aligned}\sum_{i'<i} \mathbb{P}\left[E_{a_j i'}\right] &= \sum_{i'<i} \frac{q_a}{m}\left(1 - \frac{q_a}{m}\right)^{i'-1} \\ &= \frac{q_a}{m} \frac{1 - \left(1 - \frac{q_a}{m}\right)^{i-1}}{1 - \left(1 - \frac{q_a}{m}\right)} \\ &= 1 - \left(1 - \frac{q_a}{m}\right)^{i-1}.\end{aligned}$$

Together with (3) and (5), this implies

$$\mathbb{P}\left[E_{a_j i} \mid G_{i,j}\right] = \frac{\frac{q_a}{m}\left(1 - \frac{q_a}{m}\right)^{i-1}}{\left(1 - \frac{q_a}{m}\right)^{i-1}} = \frac{q_a}{m}. \quad (6)$$

The event $E_{a_j i}$ implies that C contains a_j. Hence, since f is monotonically decreasing, we have

$$\mathbb{E}\left[f(C) \mid E_{a_j i} \cap G_{ij}\right] \leq f(\{a_j\}).$$

Using (4) and (6), this implies

$$\mathbb{E}[f(C)|G_{i,j}] \leq \frac{q_{a_j}}{m} f(\{a_j\}) + \left(1 - \frac{q_{a_j}}{m}\right) \mathbb{E}[f(C) \mid G_{i,j+1}].$$

Applying this result iteratively for $j = 1, 2, \ldots, |A|$ implies

$$\begin{aligned}\mathbb{E}[f(C)|G_{i,1}] = &\sum_{j=1}^{|A|} \frac{q_{a_j}}{m} \left[\prod_{j'<j} \left(1 - \frac{q_{a_{j'}}}{m}\right)\right] f(\{a_j\}) \\ &+ \left[\prod_{j=1}^{|A|} \left(1 - \frac{q_{a_j}}{m}\right)\right] \mathbb{E}\left[f(C) \mid G_{i,|A|+1}\right].\end{aligned}$$

Note that $0 \leq 1 - \frac{q_a}{m} \leq 1$ for all $a \in A$ and that f is non-negative. This implies that for $i = 1, 2, \ldots, m$

$$\mathbb{E}[f(C)|G_{i,1}] \leq \sum_{a \in A} \frac{q_a}{m} f(\{a\}) + c \cdot \mathbb{E}\left[f(C)|G_{i,|A|+1}\right]$$

where

$$c = \prod_{a \in A} \left(1 - \frac{q_a}{m}\right).$$

Since $G_{i,1} = G_{i-1}$ and $G_{i,|A|+1} = G_i$, we have for $i = 1, 2, \ldots, m$

$$\mathbb{E}[f(C)|G_{i-1}] \leq \sum_{a \in A} \frac{q_a}{m} f(\{a\}) + c \cdot \mathbb{E}[f(C)|G_i].$$

Applying this result iteratively, we obtain

$$\mathbb{E}[f(C)] \leq \left(\sum_{i=1}^{m} c^{i-1}\right) \sum_{a \in A} \frac{q_a}{m} f(\{a\}) + c^m \cdot f(\emptyset).$$

Since $0 \leq c \leq 1$, we have

$$\sum_{i=1}^{m} c^{i-1} \leq \sum_{i=1}^{\infty} c^{i-1} = \frac{1}{1-c}.$$

For $x \in [-1, 0]$ it holds that $\log(1+x) \leq x$ and hence

$$\begin{aligned}c^m = \prod_{a \in A} \left(1 - \frac{q_a}{m}\right)^m &= \exp\left(m \sum_{a \in A} \log\left(1 - \frac{q_a}{m}\right)\right) \\ &\leq \exp\left(-m \sum_{a \in A} \frac{q_a}{m}\right) = e^{-\sum_{a \in A} q_a}.\end{aligned}$$

This implies that

$$\mathbb{E}[f(C)] \leq \frac{1}{1-c} \sum_{a \in A} \frac{q_a}{m} f(\{a\}) + e^{-\sum_{a \in A} q_a} \cdot f(\emptyset). \quad (7)$$

We show the main claim by contradiction. Assume that

$$\mathbb{E}_P[f(C)] > \mathbb{E}_Q[f(\{a\})] + e^{-\sum_{a \in A} q_a} f(\emptyset).$$

If $\mathbb{E}_Q[f(\{a\})] = 0$, the contradiction follows directly from (7). Otherwise, $\mathbb{E}_Q[f(\{a\})] > 0$ implies that there exists an $\epsilon > 0$ such that

$$\mathbb{E}_P[f(C)] > (1+\epsilon)\mathbb{E}_Q[f(\{a\})] + e^{-\sum_{a \in A} q_a} f(\emptyset). \quad (8)$$

By definition, we have

$$c = \prod_{a \in A} \left(1 - \frac{q_a}{m}\right) = 1 - \sum_{a \in A} \frac{q_a}{m} + o\left(\frac{1}{m}\right).$$

Thus, there exists a $m_\epsilon \in \mathbb{N}$ sufficiently large such that

$$c = 1 - \sum_{a \in A} \frac{q_a}{m_\epsilon} + o\left(\frac{1}{m_\epsilon}\right) \leq 1 - \frac{1}{1+\epsilon} \sum_{a \in A} \frac{q_a}{m_\epsilon}.$$

Together with (7), this implies

$$\begin{aligned}\mathbb{E}[f(C)] &\leq \frac{1+\epsilon}{\sum_{a \in A} \frac{q_a}{m_\epsilon}} \sum_{a \in A} \frac{q_a}{m_\epsilon} f(\{a\}) + e^{-\sum_{a \in A} q_a} \cdot f(\emptyset) \\ &= (1+\epsilon)\mathbb{E}_Q[f(\{a\})] + e^{-\sum_{a \in A} q_a} f(\emptyset).\end{aligned}$$

which is a contradiction to (8) and thus proves the claim. □

Lemma 2 extends Lemma 1 to `k-means`‖-style sampling probabilities of the form $q_a = \min(1, \ell p_a)$.

Lemma 2. *Let $\ell \geq 1$. Let A be a finite set and let $f : 2^A \to \mathbb{R}$ be a set function that is non-negative and monotonically decreasing, i.e., $f(V) \geq f(U) \geq 0$, for all $V \subseteq U$. For each $a \in A$, let $p_a \geq 0$ and $\sum_{a \in A} p_a \leq 1$.*

Let P be the probability distribution where, for each $a \in A$, E_a denotes an independent event that occurs with probability $q_a = \min(1, \ell p_a)$. Let C be the set of elements $a \in A$ for which the event E_a occurs.

Let Q be the probability distribution on A where a single $a \in A$ is sampled with probability $p_a / \sum_{a \in A} p_a$.

Then, with $\emptyset$ denoting the empty set, we have that

$$\mathbb{E}_P[f(C)] \leq 2\mathbb{E}_Q[f(\{a\})] + e^{-\ell \sum_{a \in A} p_a} f(\emptyset).$$

Proof. Let A_1 be the set of elements $a \in A$ such that $\ell p_a \leq 1$ and A_2 the set of elements $a \in A$ such that $\ell p_a > 1$. By definition, every element in A_2 is sampled almost surely, i.e., $A_2 \subseteq C$. This implies that almost surely

$$f(C) = f\left(A_2 \cup (C \cap A_1)\right). \quad (9)$$

If $|A_1|= 0$, the result follows trivially since

$$\mathbb{E}_P[f(C)] = f(A_2) = \mathbb{E}_Q[f(\{a\})].$$

Similarly, if $|A_2|= 0$, the result follows directly from Lemma 1 with $q_a = \ell p_a$. For the remainder of the proof, we may thus assume that both A_1 and A_2 are non-empty.

For $a \in A_1$, let $q_a = \ell p_a$ and define the non-negative and monotonically decreasing function

$$g(C) = f\left(A_2 \cup C\right).$$

Let $p_1 = \sum_{a \in A_1} p_a$ and $p_2 = \sum_{a \in A_2} p_a$. Lemma 1 applied to A_1, q_a and g implies that

$$\mathbb{E}_P[f(C)] = \mathbb{E}[g(C)] \leq \sum_{a \in A_1} \frac{p_a}{p_1} g(\{a\}) + e^{-\ell p_1} g(\emptyset). \tag{10}$$

Let

$$d = \left(1 - e^{-\ell p_2}\right) e^{-\ell p_1}$$

and define

$$\alpha = \frac{p_2}{p_1 + p_2} - \frac{p_1}{p_1 + p_2} d.$$

By design, $\alpha \leq 1$. Furthermore

$$\ell p_1 \geq \log \ell p_1.$$

Since A_2 is nonempty and $p_a \geq \frac{1}{\ell}$ for all $a \in A_2$, it follows that $p_2 \geq \frac{1}{\ell}$. This implies

$$e^{\ell p_1} \geq \ell p_1 \geq \frac{p_1}{p_2}.$$

Since $0 \leq \left(1 - e^{-\ell p_2}\right) \leq 1$, we have

$$p_2 \geq p_1 e^{-\ell p_1} \geq p_1 \left(1 - e^{-\ell p_2}\right) e^{-\ell p_1} = p_1 d.$$

Hence,

$$\alpha = \frac{p_2}{p_1 + p_2} - \frac{p_1}{p_1 + p_2} \left(1 - e^{-\ell p_2}\right) e^{-\ell p_1} \geq 0.$$

Since $\alpha \in [0, 1]$ and $g(\{a\}) \leq g(\emptyset)$ for any $a \in A_1$, we may write (10), i.e.,

$$\mathbb{E}_P[f(C)] \leq (1 - \alpha) \sum_{a \in A_1} \frac{p_a}{p_1} g(\{a\}) + \left(\alpha + e^{-\ell p_1}\right) g(\emptyset). \tag{11}$$

By definition, we have

$$1 - \alpha = 1 - \frac{p_2}{p_1 + p_2} + \frac{p_1}{p_1 + p_2} d = \frac{p_1}{p_1 + p_2}(1 + d).$$

Since $g(\{a\}) \leq f(\{a\})$, we thus have

$$(1 - \alpha) \sum_{a \in A_1} \frac{p_a}{p_1} g(\{a\}) \leq (1 + d) \sum_{a \in A_1} \frac{p_a}{p_1 + p_2} f(\{a\}). \tag{12}$$

Similarly, we have

$$\begin{aligned} \alpha + e^{-\ell p_1} &= \frac{p_2}{p_1 + p_2} - \frac{p_1}{p_1 + p_2} d + e^{-\ell p_1} \\ &= \frac{p_2}{p_1 + p_2} + d \frac{p_2}{p_1 + p_2} - d + e^{-\ell p_1} \\ &= (1 + d) \frac{p_2}{p_1 + p_2} + e^{-\ell (p_1 + p_2)}. \end{aligned}$$

Since $g(\emptyset) \leq f(\emptyset)$, it follows that

$$\left(\alpha + e^{-\ell p_1}\right) g(\emptyset) \leq (1+d) \frac{p_2}{p_1 + p_2} g(\emptyset) + e^{-\ell (p_1 + p_2)} f(\emptyset). \tag{13}$$

Since $g(\emptyset) = f\left(A_2\right)$ and thus $g(\emptyset) \leq f(\{a\})$ for all $a \in A_2$, we have

$$p_2 g(\emptyset) = \sum_{a \in A_2} p_a g(\emptyset) \leq \sum_{a \in A_2} p_a f(\{a\}). \tag{14}$$

Combining (11), (12), (13), and (14) leads to

$$\mathbb{E}_P[f(C)] \leq (1 + d) \mathbb{E}_Q[f(\{a\})] + e^{-\ell \sum_{a \in A} p_a} f(\emptyset).$$

Since $p_1 \geq 0$, we have

$$1 + d = 1 + \left(1 - e^{-\ell p_2}\right) e^{-\ell p_1} \leq 2$$

which proves the main claim. □

Lemma 3 bounds the solution quality after each iteration of Algorithm 2 based on the solution before the iteration.

Lemma 3. *Let $k \in \mathbb{N}$ and $\ell \geq 1$. Let $\mathcal{X}$ be a data set in $\mathbb{R}^d$ and denote by $\phi_{\mathrm{OPT}}(\mathcal{X})$ the optimal k-Means clustering cost. Let C denote the set of cluster centers at the beginning of an iteration in Algorithm 2 and C' the random set added in the iteration. Then, it holds that*

$$\mathbb{E}[\phi_{\mathcal{X}}(C \cup C')] \leq \left(\frac{k}{e\ell}\right) \phi_{\mathcal{X}}(C) + 16 \phi_{\mathrm{OPT}}(\mathcal{X}).$$

Proof. The proof relies on applying Lemma 2 to each cluster of the optimal solution. Let OPT denote any clustering achieving the minimal cost $\phi_{\mathrm{OPT}}(\mathcal{X})$ on $\mathcal{X}$. We assign all the points $x \in \mathcal{X}$ to their closest cluster center in OPT with ties broken arbitrarily but consistently. For $c \in \mathrm{OPT}$ we denote by $\mathcal{X}_c$ the subset of $\mathcal{X}$ assigned to c. For each $c \in \mathrm{OPT}$, let

$$C'_c = C' \cap \mathcal{X}_c.$$

By definition, $a \in \mathcal{X}_c$ is included in C'_c with probability

$$q_a = \min\left(1, \frac{\ell \, \mathrm{d}(a, C)^2}{\sum_{a' \in \mathcal{X}} \mathrm{d}(a', C)^2}\right).$$

For each $c \in \mathrm{OPT}$, we define the monotonically decreasing function $f_c : 2^{\mathcal{X}_c} \to \mathbb{R}_{\geq 0}$ to be

$$f_c(C'_c) = \phi_{\mathcal{X}_c}(C \cup C'_c).$$

For each $c \in \mathrm{OPT}$, Lemma 2 applied to $\mathcal{X}_c$, C'_c and f_c implies

$$\begin{aligned}\mathbb{E}[f_c(C'_c)] \leq & 2 \sum_{a \in \mathcal{X}_c} \frac{\mathrm{d}(a, C)^2}{\sum_{a' \in \mathcal{X}_c} \mathrm{d}(a', C)^2} f_c(\{a\}) \\ & + e^{-\ell \frac{\sum_{a \in \mathcal{X}_c} \mathrm{d}(a,C)^2}{\sum_{a' \in \mathcal{X}} \mathrm{d}(a',C)^2}} f_c(\emptyset).\end{aligned} \tag{15}$$

Since $f_c(\{a\}) = \phi_{\mathcal{X}_c}(C \cup \{a\})$, the first term is equivalent to sampling a single element from $\mathcal{X}_c$ using D^2 sampling. Hence, by Lemma 3.3 of Arthur & Vassilvitskii (2007) we have for all $c \in \mathrm{OPT}$

$$\sum_{a \in \mathcal{X}_c} \frac{\mathrm{d}(a, C)^2}{\sum_{a' \in \mathcal{X}_c} \mathrm{d}(a', C)^2} f_c(\{a\}). \leq 8 \phi_{\mathrm{OPT}}(\mathcal{X}_c). \tag{16}$$

For each $c \in \mathrm{OPT}$, we further have

$$e^{-\ell \frac{\sum_{a \in \mathcal{X}_c} \mathrm{d}(a,C)^2}{\sum_{a' \in \mathcal{X}} \mathrm{d}(a',C)^2}} f_c(\emptyset) = e^{-\ell u_c} u_c \phi_{\mathcal{X}}(C).$$

where

$$u_c = \frac{\sum_{a \in \mathcal{X}_c} \mathrm{d}(a, C)^2}{\sum_{a' \in \mathcal{X}} \mathrm{d}(a', C)^2} = \frac{\phi_{\mathcal{X}_c}(C)}{\phi_{\mathcal{X}}(C)}.$$

We have that

$$\log \ell u_c \leq \ell u_c - 1 \quad \iff \quad \ell u_c \leq \frac{e^{\ell u_c}}{e}$$

which implies

$$e^{-\ell u_c} u_c \phi_{\mathcal{X}}(C) \leq \frac{1}{e\ell} \phi_{\mathcal{X}}(C). \tag{17}$$

Combining (15), (16) and (17), we obtain

$$\mathbb{E}[f_c(C'_c)] \leq 16 \phi_{\mathrm{OPT}}(\mathcal{X}_c) + \frac{1}{e\ell} \phi_{\mathcal{X}}(C). \tag{18}$$

Since

$$\mathbb{E}[\phi_{\mathcal{X}}(C \cup C')] \leq \sum_{c \in \mathrm{OPT}} \mathbb{E}[f_c(C'_c)]$$

and

$$\phi_{\mathcal{X}}(\mathrm{OPT}) = \sum_{c \in \mathrm{OPT}} \phi_{\mathcal{X}_c}(\mathrm{OPT}),$$

we thus have

$$\mathbb{E}[\phi_{\mathcal{X}}(C \cup C')] \leq \left(\frac{k}{e\ell}\right) \phi_{\mathcal{X}}(C) + 16 \phi_{\mathrm{OPT}}(\mathcal{X})$$

which concludes the proof. □

An iterated application of Lemma 3 allows us to bound the solution quality of Algorithm 2 in Lemma 4.

Lemma 4. *Let $k \in \mathbb{N}$, $t \in \mathbb{N}$ and $\ell \geq k$. Let $\mathcal{X}$ be a data set in $\mathbb{R}^d$ and C be the random set returned by Algorithm 2. Then,*

$$\mathbb{E}[\phi_{\mathcal{X}}(C)] \leq 2 \left(\frac{k}{e\ell}\right)^t \mathrm{Var}(\mathcal{X}) + 26 \phi_{\mathrm{OPT}}(\mathcal{X}).$$

Proof. The algorithm starts with a uniformly sampled initial cluster center c_1. We iteratively apply Lemma 3 for each of the t rounds to obtain

$$\mathbb{E}[\phi_{\mathcal{X}}(C)] \leq \left(\frac{k}{e\ell}\right)^t \mathbb{E}[\phi_{\mathcal{X}}(\{c_1\})] + 16 s_t \phi_{\mathrm{OPT}}(\mathcal{X}) \tag{19}$$

where

$$s_t = \sum_{i=1}^{t} \left(\frac{k}{e\ell}\right)^{i-1}.$$

For $\ell \geq k$, we have $0 \leq \frac{k}{e\ell} \leq 1/e$ and hence

$$s_t \leq \sum_{i=1}^{t} \frac{1}{e^{i-1}} \leq \sum_{i=0}^{\infty} \frac{1}{e^i} = \frac{1}{1 - 1/e}. \tag{20}$$

By Lemma 3.2 of Arthur & Vassilvitskii (2007), we have that $\mathbb{E}[\phi_{\mathcal{X}}(\{c_1\})] \leq 2 \mathrm{Var}(\mathcal{X})$. Together with (19), (20) and $16/(1 - 1/e) \approx 25.31 < 26$, this implies the required result. □

With Lemma 4, we are further able to bound the solution quality of Algorithm 3 and prove Theorem 1.

Proof of Theorem 1. Let B be the set returned by Algorithm 2. For any $x \in \mathcal{X}$, let b_x denote its closest point in B with ties broken arbitrarily. By the triangle inequality and since $(|a|+|b|)^2 \leq 2a^2 + 2b^2$, for any $x \in \mathcal{X}$

$$\mathrm{d}(x, C)^2 \leq 2 \,\mathrm{d}(x, b_x)^2 + 2 \,\mathrm{d}(b_x, C)^2$$

and hence

$$\begin{aligned}\mathbb{E}[\phi_{\mathcal{X}}(C)] &= \sum_{x \in \mathcal{X}} \mathrm{d}(x, C)^2 \\ &\leq 2 \sum_{x \in \mathcal{X}} \mathrm{d}(x, b_x)^2 + 2 \sum_{x \in \mathcal{X}} \mathrm{d}(b_x, C)^2 \\ &= 2 \phi_{\mathcal{X}}(B) + 2 \sum_{x \in B} w_x \,\mathrm{d}(x, C)^2.\end{aligned} \tag{21}$$

Let $\mathrm{OPT}_{\mathcal{X}}$ be the optimal k-Means clustering solution on $\mathcal{X}$ and $\mathrm{OPT}_{(B,w)}$ the optimal solution on the weighted set (B, w). By Theorem 1.1 of Arthur & Vassilvitskii (2007), `k-means++` produces an $\alpha = 8(\log_2 k + 2)$ approximation to the optimal solution. This implies that

$$\begin{aligned}\sum_{x\in B} w_x\, \mathrm{d}(x,C)^2 &\le \alpha \sum_{x\in B} w_x\, \mathrm{d}\big(x, \mathrm{OPT}_{(B,w)}\big)^2 \\ &\le \alpha \sum_{x\in B} w_x\, \mathrm{d}(x,\mathrm{OPT}_{\mathcal{X}})^2 \qquad (22) \\ &= \alpha \sum_{x\in\mathcal{X}} \mathrm{d}(b_x,\mathrm{OPT}_{\mathcal{X}})^2.\end{aligned}$$

By the triangle inequality and since $(|a|+|b|)^2 \le 2a^2+2b^2$, it holds for any $x\in\mathcal{X}$ that

$$\mathrm{d}(b_x,\mathrm{OPT}_{\mathcal{X}})^2 \le 2\,\mathrm{d}(x,b_x)^2 + 2\,\mathrm{d}(x,\mathrm{OPT}_{\mathcal{X}})^2$$

and hence

$$\sum_{x\in\mathcal{X}} \mathrm{d}(b_x,\mathrm{OPT}_{\mathcal{X}})^2 \le 2\phi_{\mathcal{X}}(B) + 2\phi_{\mathrm{OPT}}(\mathcal{X}). \qquad (23)$$

Combining (21), (22) and (23), we obtain

$$\mathbb{E}[\phi_{\mathcal{X}}(C)] \le 2(1+\alpha)\phi_{\mathcal{X}}(B) + 2\alpha\phi_{\mathrm{OPT}}(\mathcal{X}).$$

Finally, by Lemma 4, we have

$$\begin{aligned}\mathbb{E}[\phi_{\mathcal{X}}(C)] \le& (32\log_2 k + 68)\left(\frac{k}{e\ell}\right)^t \mathrm{Var}(\mathcal{X}) \\ &+ (432\log_2 k + 916)\phi_{\mathrm{OPT}}(\mathcal{X}). \qquad \square\end{aligned}$$

Proof of Theorem 2. For this proof, we explicitly construct a data set: Let $\beta' > 0$ and consider points in one-dimensional Euclidean space. For $i = 1, 2, \dots, t$, set

$$x_i = \sqrt{\beta'(4\ell t)^{1-i} - \beta'(4\ell t)^{-i}}$$

as well as

$$x_{t+1} = \sqrt{\beta'(4\ell t)^{-t}}.$$

Let the data set $\mathcal{X}$ consist of the $t+1$ points $\{x_i\}_{t=1,2,\dots,t+1}$ as well as $t+1$ points at the origin. Since $t < k-1$, the optimal k-Means clustering solution consists of $t+2$ points placed at each of the $\{x_i\}_{i=1,2,\dots t+1}$ and at 0. By design, this solution has a quantization error of zero and the variance is nonzero, i.e., $\phi_{\mathrm{OPT}}(\mathcal{X}) = 0$ and $\mathrm{Var}(\mathcal{X}) > 0$ as claimed.

Choose $\beta' = \frac{\beta}{2(t+1)}$. The maximal distance Δ between any two points in $\mathcal{X}$ is bounded by $\Delta = \mathrm{d}(0,x_1)^2 \le \beta'$. Since $n = 2(t+1)$, this implies $\psi \le n\Delta^2 \le \beta$ as claimed.

For $i = 1, 2, \dots, t$, let C_i consist of 0 and all x_j with $j < i$. By design, we have $\mathrm{d}(0, C_i)^2 = 0$ as well as $\mathrm{d}(x_j, C_i)^2 = 0$ for $j < i$. For $j \ge i$, we have $\mathrm{d}(x_j, C_i)^2 = \mathrm{d}(x_j, 0)^2$. For any $i = 1, 2, \dots, t+1$, we thus have

$$\sum_{j\ge i} \mathrm{d}(x_j, 0)^2 = \beta'(4\ell t)^{1-i}.$$

Consider a single iteration of Algorithm 2 where $C = C_i$. In this case, all points in $\mathcal{X}_j$ with $j < i$ are added to C' with probability zero and for $j > i$ each point x_j is added to C' with probability

$$\min\left(1, \frac{\ell\,\mathrm{d}(x_j,0)^2}{\sum_{j'\ge i}\mathrm{d}(x_{j'},0)^2}\right) = \frac{\ell\,\mathrm{d}(x_j,0)^2}{\beta'(4\ell t)^{1-i}}.$$

By the union bound, the probability that any of the points in $\bigcup_{j>i}\{x_j\}$ are sampled is bounded by

$$\sum_{j>i} \frac{\ell\,\mathrm{d}(x_j,0)^2}{\beta'(4\ell t)^{1-i}} = \frac{1}{4t}.$$

The point x_i is *not* sampled with probability at most

$$\begin{aligned}1 - \min\left(1, \frac{\ell\,\mathrm{d}(x_i,0)^2}{\sum_{j'\ge i}\mathrm{d}(x_{j'},0)^2}\right) &= 1 - \min\left(1, \ell - \frac{1}{4t}\right) \\ &\le \frac{1}{4t}.\end{aligned}$$

By the union bound, a single iteration of Algorithm 2 with $C = C_i$ hence samples exactly the set $C' = \{x_i\}$ with probability at least $\left(1 - \frac{1}{2t}\right)$.

In Algorithm 2, the first center is sampled uniformly at random from $\mathcal{X}$. Since half of the elements in $\mathcal{X}$ are placed at 0, with probability at least $\frac{1}{2}$, the first center is at 0 or equivalently $C = C_1$. With probability $\left(1 - \frac{1}{2t}\right)^t \ge \frac{1}{2}$, we then sample exactly the points $x_1, x_2, \dots, x_t$ in the t subsequent iterations. Hence, with probability at least $\frac{1}{4}$, the solution produced by Algorithm 2 consists of 0 and all x_i except x_{t+1}. Since x_{t+1} is closest to 0, this implies

$$\mathbb{E}[\phi_{\mathcal{X}}(C)] \ge \frac{1}{4}\mathrm{d}(x_{t+1},0)^2 = \frac{1}{4}\beta'(4\ell t)^{-t}. \qquad (24)$$

The variance of $\mathcal{X}$ is bounded by a single point at 0, i.e.,

$$\mathrm{Var}(\mathcal{X}) \le \phi_{\mathcal{X}}(\{0\}) = \sum_{j\ge 1}\mathrm{d}(x_j,0)^2 = \beta'.$$

Together with (24), we have that

$$\mathbb{E}[\phi_{\mathcal{X}}(C)] \ge \frac{1}{4}(4\ell t)^{-t}\,\mathrm{Var}(\mathcal{X}).$$

The same result extends to the output of Algorithm 3 as it always picks a subset of the output of Algorithm 2. $\square$

Acknowledgements

This research was partially supported by SNSF NRP 75, ERC StG 307036, a Google Ph.D. Fellowship and an IBM Ph.D. Fellowship. This work was done in part while Andreas Krause was visiting the Simons Institute for the Theory of Computing.

References

Ackermann, Marcel R, Märtens, Marcus, Raupach, Christoph, Swierkot, Kamil, Lammersen, Christiane, and Sohler, Christian. StreamKM++: A clustering algorithm for data streams. *Journal of Experimental Algorithmics (JEA)*, 17:2–4, 2012.

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

Ailon, Nir, Jaiswal, Ragesh, and Monteleoni, Claire. Streaming k-means approximation. In *Advances in Neural Information Processing Systems*, pp. 10–18, 2009.

Arthur, David and Vassilvitskii, Sergei. k-means++: The advantages of careful seeding. In *Symposium on Discrete Algorithms (SODA)*, pp. 1027–1035. SIAM, 2007.

Bachem, Olivier, Lucic, Mario, and Krause, Andreas. Coresets for nonparametric estimation - the case of DP-means. In *International Conference on Machine Learning (ICML)*, 2015.

Bachem, Olivier, Lucic, Mario, Hassani, Hamed, and Krause, Andreas. Fast and provably good seedings for k-means. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 55–63, 2016a.

Bachem, Olivier, Lucic, Mario, Hassani, S. Hamed, and Krause, Andreas. Approximate k-means++ in sublinear time. In *Conference on Artificial Intelligence (AAAI)*, February 2016b.

Bachem, Olivier, Lucic, Mario, Hassani, S. Hamed, and Krause, Andreas. Uniform deviation bounds for k-means clustering. In *To appear in International Conference on Machine Learning (ICML)*, 2017.

Bahmani, Bahman, Moseley, Benjamin, Vattani, Andrea, Kumar, Ravi, and Vassilvitskii, Sergei. Scalable K-Means++. *Very Large Data Bases (VLDB)*, 5(7):622–633, 2012.

Brunsch, Tobias and Röglin, Heiko. A bad instance for k-means++. In *International Conference on Theory and Applications of Models of Computation*, pp. 344–352. Springer, 2011.

Celebi, M Emre, Kingravi, Hassan A, and Vela, Patricio A. A comparative study of efficient initialization methods for the k-means clustering algorithm. *Expert Systems with Applications*, 40(1):200–210, 2013.

Fichtenberger, Hendrik, Gillé, Marc, Schmidt, Melanie, Schwiegelshohn, Chris, and Sohler, Christian. Bico: Birch meets coresets for k-means clustering. In *European Symposium on Algorithms*, pp. 481–492. Springer, 2013.

Jaiswal, Ragesh, Kumar, Amit, and Sen, Sandeep. A simple D^2-sampling based PTAS for k-means and other clustering problems. *Algorithmica*, 70(1):22–46, 2014.

Jaiswal, Ragesh, Kumar, Mehul, and Yadav, Pulkit. Improved analysis of D^2-sampling based PTAS for k-means and other clustering problems. *Information Processing Letters*, 115(2):100–103, 2015.

Lloyd, Stuart. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.

Lucic, Mario, Bachem, Olivier, and Krause, Andreas. Strong coresets for hard and soft Bregman clustering with applications to exponential family mixtures. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, May 2016.

Lucic, Mario, Faulkner, Matthew, Krause, Andreas, and Feldman, Dan. Training mixture models at scale via coresets. *To appear in Journal of Machine Learning Research (JMLR)*, 2017.

Meng, Xiangrui, Bradley, Joseph, Yuvaz, B, Sparks, Evan, Venkataraman, Shivaram, Liu, Davies, Freeman, Jeremy, Tsai, D, Amde, Manish, Owen, Sean, et al. Mllib: Machine learning in Apache Spark. *Journal of Machine Learning Research (JMLR)*, 17(34):1–7, 2016.

Ostrovsky, Rafail, Rabani, Yuval, Schulman, Leonard J, and Swamy, Chaitanya. The effectiveness of Lloyd-type methods for the k-means problem. In *Symposium on Foundations of Computer Science (FOCS)*, pp. 165–176. IEEE, 2006.