SIAM J. COMPUT.
Vol. 39, No. 3, pp. 923–947

ON CORESETS FOR k-MEDIAN AND k-MEANS CLUSTERING IN METRIC AND EUCLIDEAN SPACES AND THEIR APPLICATIONS[*]

KE CHEN[†]

Abstract. We present new approximation algorithms for the k-median and k-means clustering problems. To this end, we obtain small coresets for k-median and k-means clustering in general metric spaces and in Euclidean spaces. In $\mathbb{R}^d$, these coresets are of size with *polynomial* dependency on the dimension d. This leads to $(1+\varepsilon)$-approximation algorithms to the optimal k-median and k-means clustering in $\mathbb{R}^d$, with running time $O(ndk + 2^{(k/\varepsilon)^{O(1)}} d^2 \log^{k+2} n)$, where n is the number of points. This improves over previous results. We use those coresets to maintain a $(1+\varepsilon)$-approximate k-median and k-means clustering of a stream of points in $\mathbb{R}^d$, using $O(d^2k^2\varepsilon^{-2}\log^8 n)$ space. These are the first streaming algorithms, for those problems, that have space complexity with *polynomial* dependency on the dimension.

Key words. k-median clustering, k-means clustering, coreset, random sampling, high dimensions, approximation algorithms

AMS subject classifications. 68W25, 68Q25, 62D05

DOI. 10.1137/070699007

1. Introduction.

Clustering is the process of classifying a set of objects into groups such that objects of each group are similar. It is an important problem in computer science with applications in many domains such as data mining, pattern recognition, and bioinformatics. Two widely studied variants are (i) *k-median clustering*, where we compute a set of k centers and the clustering cost is the sum of distances from the data points to their nearest centers, and (ii) *k-means clustering*, where the clustering cost is the sum of squared distances. The *k-median problem* (resp., *k-means problem*) requires computing a set of centers of size k such that the k-median (resp., k-means) clustering cost is minimized.

For the k-median problem in the metric space settings, the first constant factor approximation algorithm is by Charikar et al. [9]. There have been several improvements to the approximation ratio and the running time. We focus on the most relevant results here. For further information, see [10] and the references therein. Indyk [20, 21] gave a randomized constant factor approximation algorithm to produce $O(k)$ centers in $O(nk \operatorname{polylog}(nk))$ time (see also Appendix A). Based on Indyk's construction, Guha et al. [16] presented a $(300 + o(1))$-approximation algorithm with $O(nk \operatorname{polylog}(nk))$ running time. Mettu and Plaxton [29] provided a constant factor approximation algorithm that runs in $O(nk + n\log n + k^2 \log^2 n)$ time. In addition, they showed that $\Omega(nk)$ time is necessary (when using a distance oracle), even for a randomized algorithm. Thorup [33] gave a randomized constant factor approximation algorithm for k-median on a weighted undirected graph, under the shortest path metric. His algorithm runs in $O(m \operatorname{polylog} m)$ time for a graph with m edges.

[*] Received by the editors August 2, 2007; accepted for publication (in revised form) June 15, 2009; published electronically August 26, 2009. A preliminary version of this paper appeared in *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms*, Miami, 2006, pp. 1177–1185. Work on this paper was done when the author was doing his Ph.D. study at the University of Illinois at Urbana-Champaign, under the supervision of Prof. Sariel Har-Peled. The research was partially supported by NSF award CCR-0132901.

http://www.siam.org/journals/sicomp/39-3/69900.html

[†] Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL 61801 (kechen@engineering.uiuc.edu). Current address: Kosmix Corp, Mountain View, CA 94041.

For the k-median problem in the Euclidean settings, one is interested in a $(1+\varepsilon)$-approximation algorithm. Arora, Raghavan, and Rao [5] presented a $(1+\varepsilon)$-approximation algorithm for the case when points are in the plane, with running time $O(n^{O(1/\varepsilon)+1})$. Kolliopoulos and Rao [25] improved the running time to $O(\rho n \log^6 n)$ for the discrete k-median problem, where $\rho = \exp(O([\log(1/\varepsilon)/\varepsilon]^{d-1}))$. (In the discrete k-median problem, the clustering centers can be selected only from among the input points.) Further improvement can be achieved by using coresets [4]. Informally speaking, a coreset for a clustering problem is a small (weighted) subset of the input, such that for any set of centers, the cost of clustering the coreset by the set of centers is close to the true cost (that is, the cost of clustering the original input by the same set of centers). In particular, Har-Peled and Mazumdar [18] improved the running time to $O(n + \rho k^{O(1)} \log^{O(1)} n)$, using a coreset of size $O(k\varepsilon^{-d} \log n)$, for k-median clustering. Har-Peled and Kushal [17] showed that one can construct coresets for this problem with size independent of n. Kumar, Sabharwal, and Sen [26, 27] showed a $(1+\varepsilon)$-approximation algorithm for k-median, in high dimensions, with running time $O(2^{(k/\varepsilon)^{O(1)}} dn)$.

For the k-means problem, one is usually interested in the Euclidean settings. See [18] and the references therein for further information.

There was growing interest in performing clustering in the streaming model of computation [16, 10, 22, 18, 14]. Here points arrive one by one in a stream and one is interested in maintaining a clustering of the points seen so far. Typically, the input is too large to fit in memory. Therefore, it is necessary to maintain a data structure to sketch the data seen so far. In this model, the complexity measure includes the overall space used and the time required to update the data structure. Guha et al. [16] presented an algorithm that uses $O(n^{\varepsilon})$ space to compute a $2^{O(1/\varepsilon)}$-approximation for k-median clustering of points taken from a metric space. Charikar, O'Callaghan, and Panigrahy [10] improved the result significantly by proposing a constant factor approximation algorithm using $O(k \log^2 n)$ space. In the Euclidean settings, Har-Peled and Mazumdar used coresets to compute an $(1+\varepsilon)$-approximation for k-median using $O(k\varepsilon^{-d} \log^{2d+2} n)$ space. The above algorithms handle streams with insertions only. Indyk [22] showed how to handle both insertions and deletions, under the restriction that the points are from a finite resolution grid. Frahling and Sohler [14] extended the work of Indyk, by showing how to extract the coreset quickly and cluster it in the insertion-deletion streaming model.

Subsequent to this work, there have been several new developments in the study of k-median clustering and related problems. Feldman, Fiat, and Sharir [12] generalized coresets for k-median and k-means clustering to the cases where centers can be more complex geometric structures such as lines and flats. Feldman, Monemizadeh, and Sohler [13] provided a PTAS for the Euclidean k-means problem with running time $O(ndk + d(k/\varepsilon)^{O(1)} + 2^{O(k/\varepsilon)})$, by using weak coresets. Ackermann, Blömer, and Sohler [2] studied a generalization of the k-median problem. Based on a new analysis of an algorithm by Kumar, Sabharwal, and Sen [26], they presented an $O(n2^{(k/\varepsilon)^{O(1)}})$ time $(1+\varepsilon)$-approximation algorithm for the k-median problem with respect to certain distance functions D when, under this distance measure D, the 1-median problem can be approximated within a factor of $(1+\varepsilon)$ by taking a random sample of constant size and solving the 1-median problem on the sample exactly. Recently, Ackermann and Blömer [1] further improved their result.

Our results. In the following, we denote the input size by n, the number of clusters by k, the desired approximation quality by ε, and the dimension of the underlying space by d (if the input is in Euclidean space). We present fast approximation algorithms for k-clustering using coresets. (For simplicity of exposition, the phrase *k-clustering* will refer to either k-median or k-means clustering in the remainder of the paper.) We use a bicriteria approximation for k-clustering to guide random sampling from the original input. The sampling allows us to extract a (k, ε)-coreset (see section 2 for the formal definition) of size (roughly) $O(k^2\varepsilon^{-2}\log^2 n)$ in a general metric space, and a (k, ε)-coreset of size (roughly) $O(dk^2\varepsilon^{-2}\log n \log(k/\varepsilon))$ in $\mathbb{R}^d$. These two constructions of coresets are the main results of this paper.

In $\mathbb{R}^d$, the small coreset construction leads to an algorithm for finding a $(1 + \varepsilon)$-approximation to the optimal k-clustering, in $O(ndk+2^{(k/\varepsilon)^{O(1)}} d^2 \log^{k+2} n)$ time (with constant probability of success). This result improves over the algorithm of Kumar, Sabharwal, and Sen [26, 27], which has running time $O(2^{(k/\varepsilon)^{O(1)}} dn)$. In the streaming model, our main result implies an algorithm that uses $O(d^2k^2\varepsilon^{-2}\log^8 n)$ space for $(1+\varepsilon)$-approximation to the optimal k-clustering. The algorithm assumes that the points arrive one by one, and removal of points is not allowed. Upon the arrival of a new point, the amortized time to update the data structure is $O(dk \operatorname{polylog}(ndk/\varepsilon))$. In comparison, previous algorithms require space and time exponential in the dimension.

In a general metric space, the coreset construction leads to a $(10+\varepsilon)$-approximation algorithm for the k-median problem running in $O(nk+k^7\varepsilon^{-5}\log^5 n)$ time using known techniques [6]. This result provides better trade-offs between overall running time and approximation quality over previous results when k is small. In particular, all previous algorithms with $O(nk \operatorname{polylog}(nk))$ running time [20, 16, 29] provided constant approximation, where the constant is considerably larger than the one in our algorithm. The coreset can also be used to stream k-median clustering using small space, such that one can compute $(1 + \varepsilon)$-approximation to the optimal k-median clustering using this data structure. To our knowledge, this is the first algorithm, for general metric spaces, that uses small space and can provide a $(1 + \varepsilon)$-approximation to the optimal clustering cost. Of course, since it is not known how to compute the $(1 + \varepsilon)$-approximation efficiently (i.e., in polynomial time in n and k), this may be of limited interest.

The main tool we use to extract small coresets is random sampling. Mishra, Oblinger, and Pitt [31] used a similar approach to obtain a fast k-median algorithm in metric spaces. Their algorithm uses $O((M/\Delta)^2(k \log n + \log(1/\delta)))$ samples to approximately represent a set P of n points, where M is the diameter of P. They show that, with probability $\geq 1 - \delta$, the difference between the average clustering cost on the samples and the average clustering cost on P is at most Δ. Note that, to obtain a constant factor approximation, their algorithm may yield running time as high as $O(n^2)$, depending on the diameter M. Our approach can be interpreted as combining the approach of Mishra, Oblinger, and Pitt with the use of coresets and exponential grids of Har-Peled and Mazumdar [18], such that we can obtain "good" samples with size independent of M and with low dependency on the dimension.

The rest of the paper is organized as follows. In section 3, we present an algorithm to compute a small (k, ε)-coreset for metric k-median clustering. In section 4, we prove the existence of a small (k, ε)-coreset for Euclidean k-median clustering. In section 5, we extend the coreset constructions to the k-means clustering. In section 6, we present fast approximation algorithms for k-clustering using those (k, ε)-coresets. We conclude in section 7.

2. Problem definition.

Let $(X, \mathbf{d})$ be a *metric space*, where $\mathbf{d}$ is the distance function defined over the points of X. Let $\mathbf{d}(Q, v) = \min_{q\in Q} \mathbf{d}(q, v)$ denote the distance between a point $v \in X$ and a set $Q \subseteq X$. For $Q, V \subseteq X$, let $\mathbf{d}(Q,V) = \min_{v \in V} \mathbf{d}(Q,v)$ denote the distance between Q and V. Let $\text{diam}(Q) = \max_{s,t \in Q} \mathbf{d}(s,t)$ denote the *diameter* of a set $Q \subseteq X$.

Let $P \subseteq X$ be a given set of n points. Let $\text{ball}(c, r)$ denote the close ball of radius r centered at c; formally, $\text{ball}(c,r) = \{p \in P \mid \mathbf{d}(c,p) \leq r\}$. In the following, we assume that each point $p \in P$ is associated with a positive integer *weight* $\mathbf{w}(p)$. An unweighted set P can be considered as weighted, with unit weight assigned to each point. Let $\mathbf{w}(P) = \sum_{p \in P} \mathbf{w}(p)$ denote the *total weight* of P.

DEFINITION 2.1 (k-median and k-means clustering). *A* clustering *of P is a partition induced by a* center set $C = \{c_1, \ldots, c_k\} \subseteq X$*; that is, each point of P is assigned to its nearest center in C. The point $p \in P$ is* served by c_i *if the nearest neighbor to p in C is c_i.*

Let $\nu(C,p) = \mathbf{d}(C,p)\mathbf{w}(p)$ denote the cost *of the* k-median clustering *of p using C. The* cost *of the k-median clustering of P by C is $\nu(C,P) = \sum_{p \in P} \nu(C,p)$.*

Let $\mu(C,p) = (\mathbf{d}(C,p))^2\mathbf{w}(p)$ denote the cost *of the* k-means clustering *of p using C. The* cost *of the k-means clustering of P by C is $\mu(C,P) = \sum_{p \in P} \mu(C,p)$.*

The metric k-median *(resp.,* k-means*) problem is to find a set of k centers $C \subseteq P$ that minimizes the cost $\nu(C,P)$ (resp., $\mu(C,P)$). Let $\nu_{\text{opt}}(k,P)$ (resp., $\mu_{\text{opt}}(k,P)$) denote the cost of the optimal k-median (resp., k-means) clustering of P.*

DEFINITION 2.2 ((k,ε)-coreset). *Given a point set P in a metric space, a weighted subset $\mathcal{S} \subseteq P$ is a* (k,ε)-coreset *of P for the k-median clustering if*

$$|\nu(C,\mathcal{S}) - \nu(C,P)| \leq \varepsilon\nu(C,P)$$

for all sets $C \subseteq P$ satisfying $|C| \leq k$. The (k,ε)-coreset *of P for the k-means clustering is defined similarly.*

DEFINITION 2.3 ($[\alpha,\beta]$-bicriteria approximation). *A set $\mathcal{A} = \{a_1, \ldots, a_m\}$ is the center set of an* $[\alpha,\beta]$-bicriteria approximation *for the k-median (resp., k-means) clustering of P if $m \leq \alpha k$ and $\nu(\mathcal{A},P) \leq \beta\nu_{\text{opt}}(k,P)$ (resp., $\mu(\mathcal{A},P) \leq \beta\mu_{\text{opt}}(k,P)$).*

3. Coreset for metric k-median clustering.

In this section, we present an algorithm to compute a (k,ε)-coreset for metric k-median clustering. The input consists of a set P of n points and parameters k, ε, and λ. There is also an associated metric distance function $\mathbf{d}$ defined over the points of P, which we can evaluate for any pair of points of P in constant time. We shall compute a weighted sample set $\mathcal{S}$ from P such that $\mathbf{w}(\mathcal{S}) = \mathbf{w}(P)$ and $\mathcal{S}$ is a (k,ε)-coreset of P, with probability $\geq 1 - \lambda$.

3.1. The coreset construction.

The algorithm consists of two steps: (i) partitioning the input P into several disjoint subsets, and (ii) taking a random sample from each such subset. The union of these samples forms the desired coreset.

3.1.1. Step 1: Partitioning P.

For simplicity of exposition, we assume that the input P is unweighted unless explicitly stated otherwise. The results also hold when P is weighted, with slightly worse bounds.

Assume that $\mathcal{A} \subseteq P$ is the center set of an $[\alpha,\beta]$-bicriteria approximation to the optimal k-median clustering of P. That is, $\mathcal{A} = \{a_1, \ldots, a_m\}$ satisfies $\nu(\mathcal{A},P) \leq \beta\nu_{\text{opt}}(k,P)$, where $m \leq \alpha k$ and $\alpha, \beta \geq 1$ (here, α and β are some constants).

Let $P_i \subseteq P$ be the set of points served by the center a_i for $i = 1, \ldots, m$, and let $R = \nu(\mathcal{A},P)/(\beta n)$ be a lower bound on the average radius of the optimal k-median clustering. Set $\phi = \lceil \log(\beta n) \rceil$. For $i = 1, \ldots, m$ and $j = 0, \ldots, \phi$, let

$$P_{i,j} = \begin{cases} P_i \cap \text{ball}(a_i, R), & j = 0, \\ P_i \cap \left[\text{ball}(a_i, 2^j R) \setminus \text{ball}(a_i, 2^{j-1}R)\right], & j \geq 1, \end{cases}$$

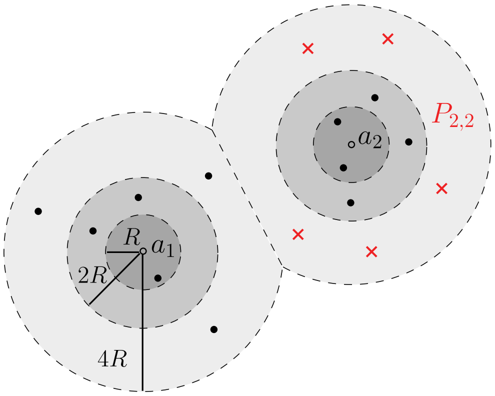

FIG. 1. *Illustrating ring sets. Here $\mathcal{A} = \{a_1, a_2\}$.*

be the jth *ring set* for the center a_i. See Figure 1. It is easy to verify that every point in P lies in exactly one ring set, since no point of P can be in distance greater than $\beta n R$ from all the centers of $\mathcal{A}$. Therefore, these ring sets partition P into disjoint sets.

To compute the center set $\mathcal{A}$, in $O(nk)$ time, we use the algorithm of Indyk [20]; see Appendix A for details.

Remark 1. The set $P_{i,j}$ is computed as follows: For each point $p \in P$, we first compute $\mathbf{d}(p, a_1), \mathbf{d}(p, a_2), \ldots, \mathbf{d}(p, a_m)$, and then we can decide to which set P_i the point p belongs. Now, based on $\mathbf{d}(p, a_i)$ and R, we can immediately decide to which set $P_{i,j}$ the point p belongs, by letting $j = 0$ if $\mathbf{d}(p, a_i) \leq R$ and letting $j = \lceil \log(\mathbf{d}(p, a_i)/R) \rceil$ if $\mathbf{d}(p, a_i) > R$. This can be done for all points of P in $O(mn) = O(\alpha k n) = O(nk)$ time, since $\alpha = O(1)$.

3.1.2. Step 2: Random sampling.

Let

$$s = \left\lceil \frac{\mathbf{c}\beta^2}{\varepsilon^2} \left(k \ln n + \ln \frac{1}{\lambda} \right) \right\rceil, \tag{1}$$

where $\mathbf{c}$ is a sufficiently large constant. For $i = 1, \ldots, m$ and $j = 0, \ldots, \phi$, if $|P_{i,j}| \leq s$, then set $\mathcal{S}_{i,j} = P_{i,j}$. Otherwise, randomly pick s points from $P_{i,j}$ independently and uniformly (with replacement), assign each point weight $|P_{i,j}|/s$, and let $\mathcal{S}_{i,j}$ be the resulting weighted sample. We assume that $|P_{i,j}|/s$ is an integer number.[1]

We claim that the set $\mathcal{S} = \cup_{i,j} \mathcal{S}_{i,j}$ is the desired (k, ε)-coreset of P.

3.2. Proof of correctness.

We start with the following observations.

OBSERVATION 3.1.

(i) *For each $p \in P_{i,0}$, it holds that $0 \leq \mathbf{d}(\mathcal{A}, p) \leq R$.*

(ii) *For each $p \in P_{i,j}$, where $j \geq 1$, it holds that $2^{j-1}R < \mathbf{d}(\mathcal{A}, p) \leq 2^j R$.*

(iii) *$\beta n R = \nu(\mathcal{A}, P) \leq \beta \nu_{\mathrm{opt}}$, where $\nu_{\mathrm{opt}} = \nu_{\mathrm{opt}}(k, P)$.*

The following lemma is an easy variant of the result of Haussler [19].

[1] This is a minor technicality that can be easily resolved. Indeed, if $|P_{i,j}|$ is not a multiple of s, we arbitrarily choose a set $Q_{i,j}$ of less than s points from $P_{i,j}$ such that $|P_{i,j} \setminus Q_{i,j}|$ is a multiple of s. Draw a set of s points from $P_{i,j} \setminus Q_{i,j}$ independently and uniformly, assign each sample point the weight $|P_{i,j} \setminus Q_{i,j}|/s$, and let $\mathcal{S}_{i,j}$ be the union of the weighted sample set (from $P_{i,j} \setminus Q_{i,j}$) and $Q_{i,j}$. It is easy to verify that $\mathbf{w}(\mathcal{S}_{i,j}) = \mathbf{w}(P_{i,j})$ and $|\mathcal{S}_{i,j}| \leq 2s$.

Lemma 3.2. *Let $M \geq 0$ and η be fixed constants, and let $h(\cdot)$ be a function defined on a set V such that $\eta \leq h(p) \leq \eta + M$ for all $p \in V$. Let $U = \{p_1, \ldots, p_s\}$ be a set of s samples drawn independently and uniformly from V, and let $\delta > 0$ be a parameter. If $s \geq (M^2/2\delta^2)\ln(2/\lambda)$, then $\mathbf{Pr}[|\frac{h(V)}{|V|} - \frac{h(U)}{|U|}| \geq \delta] \leq \lambda$, where $h(U) = \sum_{u\in U} h(u)$ and $h(V) = \sum_{v\in V} h(v)$.*

Lemma 3.3. *Let V be a set of points in a metric space $(X, \mathbf{d})$, and let $\lambda', \xi > 0$ be given parameters. Let U be a sample of $s' = \lceil \xi^{-2}\ln(2/\lambda')\rceil$ points picked from V, independently and uniformly, where each point of U is assigned weight $|V|/|U|$ such that $\mathbf{w}(U) = |V|$. For a fixed set C, where C is not necessarily a subset of V, we have that $|\nu(C, V) - \nu(C, U)| \leq \xi |V| \operatorname{diam}(V)$, with probability $\geq 1 - \lambda'$.*

Proof. Consider the function $h(v) = \mathbf{d}(C, v)$ defined over the points of V. By the triangle inequality, for every point $v \in V$, it holds that

$$\mathbf{d}(C, V) \leq h(v) = \mathbf{d}(C, v) \leq \mathbf{d}(C, V) + \operatorname{diam}(V).$$

By Lemma 3.2, setting $\eta = \mathbf{d}(C, V)$, $M = \operatorname{diam}(V)$, and $\delta = \xi M$, we have that, for a sample U of size $s' = \lceil \xi^{-2}\ln(2/\lambda')\rceil \geq (M^2/2\delta^2)\ln(2/\lambda')$ from V, it holds that

$$\mathbf{Pr}\left[\left|\frac{\sum_{v\in V}\mathbf{d}(C,v)}{|V|} - \frac{\sum_{u\in U}\mathbf{d}(C,u)}{|U|}\right| \geq \xi \operatorname{diam}(V)\right] = \mathbf{Pr}\left[\left|\frac{h(V)}{|V|} - \frac{h(U)}{|U|}\right| \geq \delta\right] \leq \lambda'.$$

This implies that

$$\begin{aligned}|\nu(C,V) - \nu(C,U)| &= |V| \cdot \left|\frac{\sum_{v\in V}\mathbf{d}(C,v)}{|V|} - \frac{\sum_{u\in U}\mathbf{d}(C,u)\mathbf{w}(u)}{|V|}\right| \\ &= |V| \cdot \left|\frac{\sum_{v\in V}\mathbf{d}(C,v)}{|V|} - \frac{\sum_{u\in U}\mathbf{d}(C,u)}{|U|}\right| \leq \xi |V| \operatorname{diam}(V),\end{aligned}$$

with probability $\geq 1 - \lambda'$, since $\mathbf{w}(u) = |V|/|U|$ for all $u \in U$. □

Claim 3.4. *Let $\nu_{\text{opt}} = \nu_{\text{opt}}(k, P)$. We have $\sum_{i,j} |P_{i,j}| 2^j R \leq 3\nu(\mathcal{A}, P) \leq 3\beta\nu_{\text{opt}}$ and $\sum_{i,j} |P_{i,j}| \operatorname{diam}(P_{i,j}) \leq 6\nu(\mathcal{A}, P) \leq 6\beta\nu_{\text{opt}}$.*

Proof. Let p be an arbitrary point in $P_{i,j}$. By Observation 3.1, we have $2^j R = R$ if $j = 0$, and $2^j R \leq 2\mathbf{d}(\mathcal{A}, p)$ if $j \geq 1$. Therefore, $2^j R \leq \max(2\mathbf{d}(\mathcal{A}, p), R) \leq 2\mathbf{d}(\mathcal{A}, p) + R$. Thus,

$$\begin{aligned}\sum_{i,j} |P_{i,j}| 2^j R &= \sum_{i,j}\sum_{p\in P_{i,j}} 2^j R \leq \sum_{i,j}\sum_{p\in P_{i,j}} (2\mathbf{d}(\mathcal{A}, p) + R) = \sum_{p\in P}(2\mathbf{d}(\mathcal{A}, p) + R) \\ &= 2\nu(\mathcal{A}, P) + |P| R = 2\nu(\mathcal{A}, P) + nR \leq 3\nu(\mathcal{A}, P) \leq 3\beta\nu_{\text{opt}},\end{aligned}$$

by Observation 3.1(iii). Now, since $\operatorname{diam}(P_{i,j}) \leq 2(2^j R)$, the above inequality also implies the second part of the claim. □

Lemma 3.5. *For all sets $C \subseteq P$ of size at most k, it holds that $|\nu(C, P) - \nu(C, \mathcal{S})| \leq \varepsilon\nu(C, P)$, with probability $\geq 1 - \lambda/2$.*

Proof. Fix an arbitrary set C of at most k centers. By Lemma 3.3, setting $\xi = \varepsilon/(6\beta)$ and $\lambda' = n^{-k}\lambda/(2m(\phi + 1))$, it holds that

$$|\nu(C, P_{i,j}) - \nu(C, \mathcal{S}_{i,j})| \leq \frac{\varepsilon}{6\beta} |P_{i,j}| \operatorname{diam}(P_{i,j}),$$

with probability $\geq 1 - \lambda'$ for $i = 1, \ldots, m$ and $j = 0, \ldots, \phi$. Here, the sample required is of size $s' = \lceil \xi^{-2}\ln(2/\lambda')\rceil = \lceil (6\beta/\varepsilon)^2 \ln(4n^k m(\phi + 1)/\lambda)\rceil$. This is smaller than s, the actual number of points drawn from $P_{i,j}$, if $\mathbf{c}$ is sufficiently large; see (1). Now, by Claim 3.4, we have

$$
\begin{aligned}
|\nu(C,P) - \nu(C,\mathcal{S})| &\le \sum_{i,j} |\nu(C,P_{i,j}) - \nu(C,\mathcal{S}_{i,j})| \\
&\le \frac{\varepsilon}{6\beta} \sum_{i,j} |P_{i,j}| \operatorname{diam}(P_{i,j}) \le \frac{\varepsilon}{6\beta} 6\beta\nu_{\mathrm{opt}} \le \varepsilon\nu(C,P),
\end{aligned}
$$

and this holds with probability $\ge 1 - m(\phi+1)\lambda' = 1 - n^{-k}\lambda/2$.

There are at most n^k different ways to select a set C of at most k centers from P. As such, the above inequality holds for every set C of size at most k, with probability $\ge 1 - n^k \cdot (n^{-k}\lambda/2) = 1 - \lambda/2$. □

THEOREM 3.6. *Given a set P of n points in a metric space and parameters $1 > \varepsilon > 0$ and $\lambda > 0$, one can compute a weighted set $\mathcal{S}$ in $O(nk\log(1/\lambda))$ time such that $|\mathcal{S}| = O(k\varepsilon^{-2}(k\log n + \log(1/\lambda))\log n)$ and $\mathcal{S}$ is a (k,ε)-coreset of P for k-median clustering, with probability $\ge 1-\lambda$.*

If P is a weighted point set, with total weight W, then the running time is $O(nk\log(1/\lambda)\log\log W)$, and the coreset size is $O(k\varepsilon^{-2}(k\log n + \log(1/\lambda))\log^2 W)$.

Proof. The algorithm is described in section 3.1. By Theorem A.4, the assumption that $\nu(\mathcal{A},P) \le \beta\nu_{\mathrm{opt}}(k,P)$ holds with probability $\ge 1-\lambda/2$. Therefore, by Lemma 3.5, it holds that $|\nu(C,P) - \nu(C,\mathcal{S})| \le \varepsilon\nu(C,P)$ for all sets $C \subseteq P$ of at most k centers, with probability $\ge 1-\lambda/2-\lambda/2 = 1-\lambda$. If P is unweighted, then the size of the coreset $\mathcal{S}$ is $|\mathcal{S}| = O(m\phi s) = O(k\varepsilon^{-2}(k\log n + \log(1/\lambda))\log n)$, and if P is weighted, then $|\mathcal{S}| = O(m\phi s) = O(k\varepsilon^{-2}(k\log n + \log(1/\lambda))\log^2 W)$.

The overall running time is dominated by the computation of the set $\mathcal{A}$. This takes $O(nk\log(1/\lambda))$ time if P is unweighted, and $O(nk\log(1/\lambda)\log\log W)$ time if P is weighted, by Theorem A.4. □

4. Coreset for Euclidean k-median clustering.

In this section, we present an algorithm for computing coresets for Euclidean k-median clustering.

DEFINITION 4.1 (Euclidean k-clustering). *Let P be a set of n points in $\mathbb{R}^d$. The* Euclidean k-median *(resp.,* Euclidean k-means*) problem is to find a set of k centers $C \subseteq \mathbb{R}^d$ that minimizes the cost $\nu(C,P)$ (resp., $\mu(C,P)$), where the distance function $\mathbf{d}$ used is the usual Euclidean distance.*

A weighted subset $\mathcal{S} \subseteq P$ is a (k,ε)-coreset *of P for Euclidean k-median clustering, if $|\nu(C,\mathcal{S}) - \nu(C,P)| \le \varepsilon\nu(C,P)$ for all sets C of at most k centers in $\mathbb{R}^d$. The* (k,ε)-coreset *of P for the Euclidean k-means clustering is defined similarly. Note that the* coreset size *is the number of points in the coreset, not the actual space used.*

Note that, unlike the metric case, the center set under consideration can be any k-tuple of points in $\mathbb{R}^d$, which is not necessarily a subset of P.

4.1. The coreset construction.

The algorithm is analogous to its metric variant. We point out the differences in the following. In the partitioning step, we use the same algorithm as described in section 3.1. Let $\mathcal{A} = \{a_1,\dots,a_m\}$ be a set of centers in $\mathbb{R}^d$ such that $\nu(\mathcal{A},P) \le \beta\nu_{\mathrm{opt}}(k,P)$, where $m \le \alpha k$ and $\alpha,\beta \ge 1$ (here, α and β are some constants). As before, $\mathcal{A}$ is computed using the algorithm of Indyk [20]; see Appendix A. Let $R = \nu(\mathcal{A},P)/(\beta n)$ and $\phi = \lceil \log(\beta n) \rceil$. As in section 3.1, we partition P into ring sets $P_{i,j}$. In the sampling step, we select s points from each ring set, where

$$
s = \left\lceil \frac{\mathbf{c}'\beta^2}{\varepsilon^2} \left(k\ln(\alpha k) + k\ln\ln n + dk\ln\frac{\beta}{\varepsilon} + \ln\frac{1}{\lambda} \right) \right\rceil \tag{2}
$$

and $\mathbf{c}'$ is a sufficiently large constant. Let $\mathcal{S}_{i,j}$ be the sample taken from $P_{i,j}$ for $i = 1, \dots, m$ and $j = 0, \dots, \phi$. We shall prove that $\mathcal{S} = \bigcup_{i,j} \mathcal{S}_{i,j}$ a (k, ε)-coreset.

Remark 2. A minor technicality here is that the algorithm of Indyk [20] approximates the optimal metric k-median clustering (i.e., the *discrete* version of the Euclidean k-median clustering, where the centers must belong to P). Fortunately, the cost of the optimal Euclidean k-median clustering is at least half of the cost of the optimal discrete solution. As such, this algorithm still provides an $[O(1), O(1)]$-bicriteria approximation to the optimal Euclidean k-median clustering.

4.2. Proof of correctness.

The main challenge in proving the correctness of the above coreset construction is that there are infinite number of ways to select a set of at most k centers in $\mathbb{R}^d$ (versus only a finite number of ways to do so in the finite metric case). Thus, the arguments used in the proof of Lemma 3.5 are no longer valid. To circumvent this problem, we define a finite set $\mathcal{G}$ (note that the set $\mathcal{G}$ is used only in the analysis), and we will show that it is sufficient to prove correctness for center sets taken from $\mathcal{G}$. A similar (but weaker) notion of a witness set was used by Matoušek [28].

Definition 4.2. *Let $\mathcal{U}$ be the union of "huge" balls centered at the points of $\mathcal{A}$. Formally, let $\Phi = \lceil \log(7\beta n/\varepsilon) \rceil$ and let $\mathcal{U} = \bigcup_{i=1}^{m} \mathrm{ball}(a_i, 2^{\Phi} R)$, where $a_i \in \mathcal{A}$. For $i = 1, \dots, m$ and $j = 0, \dots, \Phi$, let*

$$L_{i,j} = \begin{cases} \mathrm{ball}(a_i, R), & j = 0, \\ \mathrm{ball}(a_i, 2^j R) \setminus \mathrm{ball}(a_i, 2^{j-1} R), & j \geq 1. \end{cases}$$

We use an axis-parallel grid with side length $\varrho_j = 2^j \varepsilon R / (\mathsf{b}\,\beta\sqrt{d})$ to partition $L_{i,j}$ into cells, where $\mathsf{b} = 50$. Inside each grid cell of $L_{i,j}$, pick an arbitrary point (say, the center of the cell) as its representative point. *Let $\mathcal{G}_{i,j}$ denote the set of representative points for $L_{i,j}$, and let $\mathcal{G} = \bigcup_{i,j} \mathcal{G}_{i,j}$.*

Claim 4.3. *We have $\ln|\mathcal{G}| = O(\log(\alpha k) + \log\log n + d \log(\beta/\varepsilon))$.*

Proof. Fix $L_{i,j}$, and consider a cell $\mathsf{c}_{\mathsf{i,j}}$ in the grid partitioning $L_{i,j}$. The volume of $\mathsf{c}_{\mathsf{i,j}}$ is

$$\mathrm{vol}(\mathsf{c}_{\mathsf{i,j}}) = (\varrho_j)^d = \left(\frac{2^j R \varepsilon}{\mathsf{b}\,\beta\sqrt{d}} \right)^d.$$

Note that the distance from any point of $\mathsf{c}_{\mathsf{i,j}}$ to a_i is at most $2^j R + \mathrm{diam}(\mathsf{c}_{\mathsf{i,j}}) < 2^{j+1} R$, which implies $\mathsf{c}_{\mathsf{i,j}} \subseteq B_{i,j} = \mathrm{ball}(a_i, 2^{j+1} R)$. Therefore, the number of cells inside $L_{i,j}$, denoted by $\omega_{i,j}$, is at most $\mathrm{vol}(B_{i,j})/\mathrm{vol}(\mathsf{c}_{\mathsf{i,j}})$. By applying the formula of the volume of a ball in $\mathbb{R}^d$ to $B_{i,j}$, we obtain that

$$\mathrm{vol}(B_{i,j}) = \frac{\pi^{d/2} (2^{j+1} R)^d}{\Gamma(d/2 + 1)},$$

where $\Gamma(\cdot)$ is the gamma function (which is an extension of the factorial function). In particular, $\Gamma(d/2 + 1) \geq d'!$, where $d' = \lfloor d/2 \rfloor$ for $d \geq 4$. Since $n! \geq (n/e)^n$, it holds that $\Gamma(d/2 + 1) \geq d'! \geq (d'/e)^{d'} \geq (d/(4e))^{d/2}$. This implies

$$\omega_{i,j} = \frac{\mathrm{vol}(B_{i,j})}{\mathrm{vol}(\mathsf{c}_{\mathsf{i,j}})} \leq \frac{\pi^{d/2} (2^{j+1} R)^d}{\Gamma(d/2 + 1)} \left(\frac{\mathsf{b}\,\beta\sqrt{d}}{2^j R \varepsilon} \right)^d$$
$$\leq \frac{\pi^{d/2} (2^{j+1} R)^d}{(d/(4e))^{d/2}} \cdot \frac{(\mathsf{b}\,\beta)^d d^{d/2}}{(2^j R \varepsilon)^d} \leq \left(\frac{2\,\mathsf{b}\,\beta}{\varepsilon} \right)^d \left(\frac{\pi d}{d/(4e)} \right)^{d/2} = \left(\frac{\mathsf{b}'\beta}{\varepsilon} \right)^d,$$

where $\mathsf{b}' = 4\sqrt{\pi e}\,\mathsf{b} < 16\,\mathsf{b}$. Now, the size of $\mathcal{G}$ is

$$|\mathcal{G}| \le \sum_{i,j} \omega_{i,j} \le m(\Phi+1)\left(\frac{\mathsf{b}'\beta}{\varepsilon}\right)^d \le \alpha k\left(\log\frac{7\beta n}{\varepsilon}+1\right)\left(\frac{\mathsf{b}'\beta}{\varepsilon}\right)^d$$

and, thus,

$$\ln|\mathcal{G}| \le \ln(\alpha k) + \ln\left(\log\frac{7\beta n}{\varepsilon}+1\right) + d\ln\frac{\mathsf{b}'\beta}{\varepsilon} = O\left(\log(\alpha k) + \log\log n + d\log\frac{\beta}{\varepsilon}\right),$$

as claimed. □

LEMMA 4.4. *With probability* $\ge 1-\lambda/2$ *for all sets* C' *of at most* k *centers chosen from* $\mathcal{G}$, *it holds that* $|\nu(C',P) - \nu(C',\mathcal{S})| \le (\varepsilon/5)\,\nu(C',P)$.

Proof. The argument follows the proof of Lemma 3.5. As in Lemma 3.5, we need the sample to work for all subsets of size at most k of $\mathcal{G}$, and the number of such subsets is at most $|\mathcal{G}|^k$. Therefore, to achieve confidence $1-\lambda/2$, set $\xi = \varepsilon/(\mathsf{b}\,\beta)$ and $\lambda' = |\mathcal{G}|^{-2k}\lambda/(2m(\phi+1))$. The required sample size is $s' = \lceil \xi^{-2}\ln(2/\lambda')\rceil$. By Claim 4.3, the following holds:

$$s' \le 2\xi^{-2}\ln\frac{2}{\lambda'} = 2\left(\frac{\mathsf{b}\beta}{\varepsilon}\right)^2 \ln\left(\frac{4|\mathcal{G}|^k m(\phi+1)}{\lambda}\right) \le 2\left(\frac{\mathsf{b}\beta}{\varepsilon}\right)^2\left(k\ln|\mathcal{G}| + \ln\frac{4m(\phi+1)}{\lambda}\right)$$
$$= O\left(\frac{\beta^2}{\varepsilon^2}\left(k\log\log n + dk\log\frac{\beta}{\varepsilon} + k\log(\alpha k) + \log\frac{1}{\lambda}\right)\right).$$

This is smaller than s, the number of samples used, if $\mathbf{c}'$ is sufficiently large; see (2). □

CLAIM 4.5. *Let* $\nu_{\mathrm{opt}} = \nu_{\mathrm{opt}}(k,P)$.

(i) *It holds that* $\nu(\mathcal{A},\mathcal{S}) \le 3\nu(\mathcal{A},P) \le 3\beta\nu_{\mathrm{opt}}$.

(ii) *For any set* C *of centers, it holds that* $|\nu(C,P) - \nu(C,\mathcal{S})| \le 6\nu(\mathcal{A},P) \le 6\beta\nu_{\mathrm{opt}}$.

Proof. Consider a ring set $P_{i,j}$ and its corresponding weighted sample set $\mathcal{S}_{i,j}$. Because $\mathbf{w}(\mathcal{S}_{i,j}) = |P_{i,j}|$, there exists a map $f: P_{i,j} \to \mathcal{S}_{i,j}$ such that $|f^{-1}(q)| = \mathbf{w}(q)$ for all $q \in \mathcal{S}_{i,j}$.

(i) By Claim 3.4, we have

$$\nu(\mathcal{A},\mathcal{S}) = \sum_{i,j}\nu(\mathcal{A},\mathcal{S}_{i,j}) = \sum_{i,j}\sum_{p\in P_{i,j}}\mathbf{d}(\mathcal{A},f(p)) \le \sum_{i,j}\sum_{p\in P_{i,j}} 2^j R = \sum_{i,j}|P_{i,j}|\,2^j R$$
$$\le 3\nu(\mathcal{A},P) \le 3\beta\nu_{\mathrm{opt}},$$

since $\mathbf{d}(\mathcal{A},f(p)) \le 2^j R$ for any $p \in P_{i,j}$.

(ii) Let p be an arbitrary point in $P_{i,j}$. By the triangle inequality, it holds that $\mathbf{d}(C,f(p)) + \mathbf{d}(f(p),p) \ge \mathbf{d}(C,p)$ and $\mathbf{d}(C,p) + \mathbf{d}(p,f(p)) \ge \mathbf{d}(C,f(p))$ and, as such, we have that $|\mathbf{d}(C,p) - \mathbf{d}(C,f(p))| \le \mathbf{d}(p,f(p)) \le \mathrm{diam}(P_{i,j})$. Therefore, by Claim 3.4,

$$|\nu(C,P) - \nu(C,\mathcal{S})| \le \sum_{i,j}|\nu(C,P_{i,j}) - \nu(C,\mathcal{S}_{i,j})| = \sum_{i,j}\left|\sum_{p\in P_{i,j}}(\mathbf{d}(C,p) - \mathbf{d}(C,f(p)))\right|$$
$$\le \sum_{i,j}\sum_{p\in P_{i,j}}\mathrm{diam}(P_{i,j}) = \sum_{i,j}|P_{i,j}|\,\mathrm{diam}(P_{i,j})$$
$$\le 6\nu(\mathcal{A},P) \le 6\beta\nu_{\mathrm{opt}}. \qquad \square$$

In the following, let $C \subseteq \mathbb{R}^d$ be an arbitrary set of at most k centers. We need to prove that $|\nu(C,\mathcal{S}) - \nu(C,P)| \le \varepsilon\nu(C,P)$. Recall that $\mathcal{U}$ is a union of balls centered at the points of $\mathcal{A}$; see Definition 4.2.

LEMMA 4.6. *For $0 < \varepsilon < 1$, if there exists a center $c \in C$ and a point $p \in P$ such that c is outside $\mathcal{U}$ and $\mathbf{d}(C,p) = \|cp\|$, then $|\nu(C,\mathcal{S}) - \nu(C,P)| \le \varepsilon\nu(C,P)$.*

Proof. Let a_p be the nearest center to p in $\mathcal{A}$. We have $\|a_p p\| \le \nu(\mathcal{A},P)$. In addition, since c is outside $\mathcal{U}$, it holds that $\|ca_p\| \ge \mathbf{d}(\mathcal{A},c) \ge 2^{\Phi}R \ge 7\beta nR/\varepsilon = 7\nu(\mathcal{A},P)/\varepsilon$; see Definition 4.2. By the triangle inequality, we thus have

$$\nu(C,P) \ge \nu(C,p) = \|cp\| \ge \|ca_p\| - \|a_p p\| \ge \frac{7}{\varepsilon}\nu(\mathcal{A},P) - \nu(\mathcal{A},P) \ge \frac{6}{\varepsilon}\nu(\mathcal{A},P),$$

which implies that $\nu(\mathcal{A},P) \le \varepsilon\nu(C,P)/6$. Now, by Claim 4.5(ii), we have

$$|\nu(C,\mathcal{S}) - \nu(C,P)| \le 6\nu(\mathcal{A},P) \le \varepsilon\nu(C,P). \qquad \square$$

The above lemma implies that we can assume that $C \subseteq \mathcal{U}$ (since otherwise, the set $\mathcal{S}$ is the required coreset). Therefore, suppose that $C = \{c_1, \dots, c_h\}$, where $h \le k$. Let $C' = \{c'_1, \dots, c'_h\}$, where $c'_t \in \mathcal{G}$ is the representative point of the cell containing c_t for $t = 1, \dots, h$.

LEMMA 4.7. *If $C \subseteq \mathcal{U}$ and $|C| \le k$, then*

$$|\nu(C,q) - \nu(C',q)| \le \frac{\varepsilon}{\mathsf{b}\,\beta}(2\nu(C,q) + 2\nu(\mathcal{A},q) + R)$$

for any $q \in P$.

Proof. Let c_i and c'_j be the nearest centers to q in C and C', respectively. Namely, $\nu(C,q) = \|c_i q\|$ and $\nu(C',q) = \left\|c'_j q\right\|$. We consider the case when $\nu(C,q) \le \nu(C',q)$, as the other case is similar. By the triangle inequality, it holds that

$$|\nu(C,q) - \nu(C',q)| = \nu(C',q) - \nu(C,q) = \left\|c'_j q\right\| - \|c_i q\| \le \|c'_i q\| - \|c_i q\| \le \|c'_i c_i\|,$$

since $\left\|c'_j q\right\| \le \|c'_i q\|$. If $\mathbf{d}(\mathcal{A},c_i) \le R$, then

$$\|c_i c'_i\| \le \sqrt{d} \cdot \frac{\varepsilon R}{\mathsf{b}\,\beta\sqrt{d}} = \frac{\varepsilon R}{\mathsf{b}\,\beta},$$

and this implies the required bound. Otherwise, we have $2^{t-1}R < \mathbf{d}(\mathcal{A},c_i) \le 2^t R$ for some $t \ge 1$. Then c_i and c'_i are inside a cell of side length $2^t\varepsilon R/(\mathsf{b}\,\beta\sqrt{d})$ and, as such,

$$\begin{aligned}\|c_i c'_i\| \le \sqrt{d} \cdot \frac{2^t\varepsilon R}{\mathsf{b}\,\beta\sqrt{d}} &= \frac{2\varepsilon}{\mathsf{b}\,\beta} \cdot 2^{t-1}R < \frac{2\varepsilon}{\mathsf{b}\,\beta}\mathbf{d}(\mathcal{A},c_i) \\ &\le \frac{2\varepsilon}{\mathsf{b}\,\beta}(\|c_i q\| + \mathbf{d}(\mathcal{A},q)) = \frac{2\varepsilon}{\mathsf{b}\,\beta}(\nu(C,q) + \nu(\mathcal{A},q)),\end{aligned}$$

by the triangle inequality. $\square$

LEMMA 4.8. *If $C \subseteq \mathcal{U}$ and $|C| \le k$, then*

(i) $|\nu(C,P) - \nu(C',P)| \le (\varepsilon/10)\nu(C,P)$, and

(ii) $|\nu(C,\mathcal{S}) - \nu(C',\mathcal{S})| \le (\varepsilon/2)\nu(C,P)$.

Proof. (i) Recall that $\mathsf{b} = 50$. Summing up the inequality of Lemma 4.7 over all the points of P, we obtain

$$\begin{aligned}|\nu(C,P) - \nu(C',P)| &\le \frac{\varepsilon}{\mathsf{b}\,\beta}(2\nu(C,P) + 2\nu(\mathcal{A},P) + nR) \\ &\le \frac{\varepsilon}{50\beta}(2\nu(C,P) + 2\beta\nu(C,P) + \nu(C,P)) \le \frac{\varepsilon}{10}\nu(C,P),\end{aligned}$$

since $\nu(\mathcal{A}, P) \leq \beta\nu_{\text{opt}} \leq \beta\nu(C, P)$ and $nR \leq \nu_{\text{opt}} \leq \nu(C, P)$, where $\nu_{\text{opt}} = \nu_{\text{opt}}(k, P)$.

(ii) Summing up the inequality of Lemma 4.7 over all the weighted points of $\mathcal{S} \subseteq P$, we obtain

$$\begin{aligned}|\nu(C, \mathcal{S}) - \nu(C', \mathcal{S})| &\leq \frac{\varepsilon}{\mathsf{b}\,\beta}(2\nu(C, \mathcal{S}) + 2\nu(\mathcal{A}, \mathcal{S}) + nR)\\ &\leq \frac{\varepsilon}{\mathsf{b}\,\beta}(2\nu(C, P) + 18\beta\nu_{\text{opt}} + \nu_{\text{opt}})\\ &\leq \frac{\varepsilon}{50\beta}(2\nu(C, P) + 18\beta\nu(C, P) + \nu(C, P)) \leq \frac{\varepsilon}{2}\nu(C, P),\end{aligned}$$

since $\nu(C, \mathcal{S}) \leq \nu(C, P) + 6\beta\nu_{\text{opt}}$, by Claim 4.5(ii), and $\nu(\mathcal{A}, \mathcal{S}) \leq 3\beta\nu_{\text{opt}}$, by Claim 4.5(i). □

LEMMA 4.9. *For $0 < \varepsilon < 1$, with probability $\geq 1 - \lambda/2$, it holds that for every set $C \subseteq \mathcal{U}$ with $|C| \leq k$, we have $|\nu(C, P) - \nu(C, \mathcal{S})| \leq \varepsilon\nu(C, P)$.*

Proof. By Lemma 4.8(i), we have that $\nu(C', P) \leq \nu(C, P) + (\varepsilon/10)\nu(C, P) \leq (11/10)\nu(C, P)$. Therefore, by Lemma 4.4, we have

$$|\nu(C', P) - \nu(C', \mathcal{S})| \leq \frac{\varepsilon}{5}\nu(C', P) \leq \frac{\varepsilon}{5} \cdot \frac{11}{10}\nu(C, P) \leq \frac{\varepsilon}{4}\nu(C, P).$$

Now, by Lemma 4.8, it holds that

$$\begin{aligned}|\nu(C, P) - \nu(C, \mathcal{S})| &\leq |\nu(C, P) - \nu(C', P)| + |\nu(C', P) - \nu(C', \mathcal{S})| + |\nu(C', \mathcal{S}) - \nu(C, \mathcal{S})|\\ &\leq \frac{\varepsilon}{10}\nu(C, P) + \frac{\varepsilon}{4}\nu(C, P) + \frac{\varepsilon}{2}\nu(C, P) < \varepsilon\nu(C, P),\end{aligned}$$

and this holds with probability at least $\geq 1 - \lambda/2$, since Lemma 4.4 holds with probability $\geq 1 - \lambda/2$. □

Putting the above together implies the following.

THEOREM 4.10. *Given a set P of n points in $\mathbb{R}^d$ and parameters $1 > \varepsilon > 0$ and $\lambda > 0$, one can compute a set $\mathcal{S}$ of size*

$$O\left(\frac{k\log n}{\varepsilon^2}\left(dk\log\frac{1}{\varepsilon} + k\log k + k\log\log n + \log\frac{1}{\lambda}\right)\right)$$

in $O(ndk\log(1/\lambda))$ time. The set $\mathcal{S}$ is a (k, ε)-coreset of P for k-median clustering, with probability $\geq 1 - \lambda$.

If P is a weighted point set, with total weight W, then the running time is $O(ndk\log(1/\lambda)\log\log W)$, and the coreset size is

$$O\left(\frac{k\log^2 W}{\varepsilon^2}\left(dk\log\frac{1}{\varepsilon} + k\log k + k\log\log W + \log\frac{1}{\lambda}\right)\right).$$

5. Coreset for *k*-means clustering.

In this section, we present algorithms for computing coresets for k-means clustering.

5.1. Coreset for metric *k*-means clustering.

Assume that a point set $\mathcal{A} = \{a_1, \ldots, a_m\} \subseteq P$ satisfies $\mu(\mathcal{A}, P) \leq \beta\mu_{\text{opt}}(k, P)$, where $m \leq \alpha k$ and $\alpha, \beta \geq 1$. Let $R = \sqrt{\mu(\mathcal{A}, P)/(\beta n)}$ be a lower bound estimate of the average radius of the optimal k-means clustering. Set $\phi = \lceil\log(\beta n)\rceil$. As before, we compute the set $\mathcal{A}$ using the algorithm of Indyk [20]; see Appendix A.

We construct ring sets $P_{i,j}$ and combine the samples $\mathcal{S}_{i,j}$ from all ring sets, as in the metric k-median case. Here, set the sample size (from each ring set)

$$s = \left\lceil \frac{\mathbf{c}\beta^2}{\varepsilon^2}\left(k \ln n + \ln \frac{1}{\lambda}\right)\right\rceil, \tag{3}$$

where $\mathbf{c}$ is a sufficiently large constant.

The correctness proof proceeds similarly as in the k-median case. For the sake of completeness, in the following, we prove the required lemmas that imply the correctness of the algorithm.

Observation 5.1. *$\beta n R^2 = \mu(\mathcal{A}, P) \leq \beta \mu_{\text{opt}}$, where $\mu_{\text{opt}} = \mu_{\text{opt}}(k, P)$.*

Lemma 5.2. *Let V be a set of points in a metric space $(X, \mathbf{d})$, and let $\lambda', \xi > 0$ be given parameters. Let U be a sample of $s' = \lceil 4\xi^{-2} \ln(2/\lambda')\rceil$ points picked from V independently and uniformly, where each point of U is assigned weight $|V| / |U|$ such that $\mathbf{w}(U) = |V|$. For a fixed set C, we have that*

$$|\mu(C, V) - \mu(C, U)| \leq \xi |V| \left[(\mathbf{d}(C, V))^2 + (\operatorname{diam}(V))^2\right],$$

with probability $\geq 1 - \lambda'$.

Proof. Consider the function $h(v) = (\mathbf{d}(C, v))^2$ defined over the points of V. Observe that for a point $v \in V$,

$$0 \leq h(v) = (\mathbf{d}(C, v))^2 \leq \left(\mathbf{d}(C, V) + \operatorname{diam}(V)\right)^2 \leq 2(\mathbf{d}(C, V))^2 + 2(\operatorname{diam}(V))^2.$$

The remainder of the proof is similar to the proof of Lemma 3.3, and we omit the easy modifications. □

Claim 5.3. *Let $\mu_{\text{opt}} = \mu_{\text{opt}}(k, P)$. We have that $\sum_{i,j} |P_{i,j}| (2^j R)^2 \leq 5\mu(\mathcal{A}, P) \leq 5\beta\mu_{\text{opt}}$ and $\sum_{i,j} |P_{i,j}| (\operatorname{diam}(P_{i,j}))^2 \leq 20\mu(\mathcal{A}, P) \leq 20\beta\mu_{\text{opt}}$.*

Proof. Let p be an arbitrary point in $P_{i,j}$. We have $2^j R = R$ if $j = 0$, and $2^j R \leq 2\mathbf{d}(\mathcal{A}, p)$ if $j \geq 1$. Therefore, $(2^j R)^2 \leq \max(4(\mathbf{d}(\mathcal{A}, p))^2, R^2) \leq 4(\mathbf{d}(\mathcal{A}, p))^2 + R^2$. It follows that

$$\begin{aligned}\sum_{i,j} |P_{i,j}| \left(2^j R\right)^2 &= \sum_{i,j} \sum_{p \in P_{i,j}} \left(2^j R\right)^2 \leq \sum_{i,j} \sum_{p \in P_{i,j}} \left(4(\mathbf{d}(\mathcal{A}, p))^2 + R^2\right) \\ &= \sum_{p \in P} \left(4(\mathbf{d}(\mathcal{A}, p))^2 + R^2\right) = 4\mu(\mathcal{A}, P) + |P| R^2 \\ &= 4\mu(\mathcal{A}, P) + nR^2 \leq 5\mu(\mathcal{A}, P) \leq 5\beta\mu_{\text{opt}},\end{aligned}$$

by Observation 5.1. Because $\operatorname{diam}(P_{i,j}) \leq 2\left(2^j R\right)$, the above inequality also implies the second part of the claim. □

Lemma 5.4. *With probability $\geq 1 - \lambda/2$ for all sets $C \subseteq P$ of size at most k, it holds that $|\mu(C, P) - \mu(C, \mathcal{S})| \leq \varepsilon\mu(C, P)$.*

Proof. Fix an arbitrary set C of at most k centers. By Lemma 5.2, setting $\xi = \varepsilon/(21\beta)$ and $\lambda' = n^{-k}\lambda/(2m(\phi + 1))$, it holds that

$$|\mu(C, P_{i,j}) - \mu(C, \mathcal{S}_{i,j})| \leq \frac{\varepsilon}{21\beta} |P_{i,j}| \left[(\mathbf{d}(C, P_{i,j}))^2 + (\operatorname{diam}(P_{i,j}))^2\right],$$

with probability $\geq 1 - \lambda'$ for $i = 1, \ldots, m$ and $j = 0, \ldots, \phi$. Here, the sample required is of size $s' = \lceil \xi^{-2} \ln(2/\lambda')\rceil = \left\lceil (21\beta/\varepsilon)^2 \ln\left(4n^k m(\phi + 1)/\lambda\right)\right\rceil$. This is smaller than

s, the actual number of points drawn from $P_{i,j}$, if $\mathbf{c}$ is sufficiently large; see (3). Now, by Claim 5.3, we have

$$\begin{aligned}
|\mu(C,P) - \mu(C,\mathcal{S})| &\le \sum_{i,j} |\mu(C,P_{i,j}) - \mu(C,\mathcal{S}_{i,j})| \\
&\le \frac{\varepsilon}{21\beta} \sum_{i,j} |P_{i,j}| \big((\mathbf{d}(C,P_{i,j}))^2 + (\mathrm{diam}(P_{i,j}))^2 \big) \\
&\le \frac{\varepsilon}{21\beta} \sum_{i,j} \left(\sum_{p \in P_{i,j}} (\mathbf{d}(C,p))^2 + |P_{i,j}|\,(\mathrm{diam}(P_{i,j}))^2 \right) \\
&= \frac{\varepsilon}{21\beta} \left(\mu(C,P) + \sum_{i,j} |P_{i,j}|\,(\mathrm{diam}(P_{i,j}))^2 \right) \\
&\le \frac{\varepsilon}{21\beta} (\mu(C,P) + 20\beta\mu_{\mathrm{opt}}) \le \frac{\varepsilon}{21\beta} (\mu(C,P) + 20\beta\mu(C,P)) \\
&\le \varepsilon\mu(C,P),
\end{aligned}$$

where the third inequality follows from $|P_{i,j}|\,(\mathbf{d}(C,P_{i,j}))^2 \le \sum_{p\in P_{i,j}} (\mathbf{d}(C,p))^2$. And this holds with probability $\ge 1 - m(\phi+1)\lambda' = 1 - n^{-k}\lambda/2$.

There are at most n^k different ways to select a set C of at most k centers from P. As such, the above inequality holds for every set C of size at most k, with probability $\ge 1 - n^k(n^{-k}\lambda/2) \ge 1 - \lambda/2$. □

Continuing in a fashion similar to that of section 3.2, we get the following result.

Theorem 5.5. *Given a set P of n points in a metric space and parameters $1 > \varepsilon > 0$ and $\lambda > 0$, one can compute a weighted set $\mathcal{S}$ of size $O(k\varepsilon^{-2}\log n(k\log n + \log(1/\lambda)))$ in $O(nk\log(1/\lambda))$ time such that $\mathcal{S}$ is a (k,ε)-coreset of P for k-means clustering, with probability $\ge 1-\lambda$.*

If P is a weighted point set, with total weight W, then the running time is $O(nk\log(1/\lambda)\log\log W)$, and the coreset size is $O(k\varepsilon^{-2}\log^2 W(k\log n + \log(1/\lambda)))$.

5.2. Coreset for Euclidean k-means clustering.

For the Euclidean k-means clustering, the construction of coresets is similar to the construction in the metric k-means case in section 5.1. The only difference is the sample size (from each ring set):

$$s = \left\lceil \frac{\mathbf{c}'\beta^2}{\varepsilon^2} \left(k\ln(\alpha k) + k\ln\ln n + dk\ln\frac{\beta d}{\varepsilon} + \ln\frac{1}{\lambda} \right) \right\rceil, \tag{4}$$

where $\mathbf{c}'$ is a sufficiently large constant.

The correctness proof proceeds similarly as in the Euclidean k-median case in section 4. For the sake of completeness, in the following, we provide the proofs to the required lemmas that imply the correctness of the algorithm.

Definition 5.6. *Let $\mathcal{U}$ be the union of "huge" balls centered at the points of $\mathcal{A}$. Formally, let $\Phi = \lceil \log(12\sqrt{\beta n}/\varepsilon) \rceil$ and let $\mathcal{U} = \bigcup_{i=1}^{m} \mathrm{ball}(a_i, 2^\Phi R)$, where $a_i \in \mathcal{A}$. For $i = 1,\dots,m$ and $j = 0,\dots,\Phi$, let*

$$L_{i,j} = \begin{cases} \mathrm{ball}(a_i, R), & j = 0, \\ \mathrm{ball}(a_i, 2^j R) \setminus \mathrm{ball}(a_i, 2^{j-1}R), & j \ge 1. \end{cases}$$

We use an axis-parallel grid with side length $\varrho_j = 2^j\varepsilon R/(\mathsf{b}\,\beta\sqrt{d})$ to partition $L_{i,j}$ into cells, where $\mathsf{b} = 800$. Inside each grid cell of $L_{i,j}$, pick an arbitrary point (say, the *center of the cell) as its* representative point. *Let $\mathcal{G}_{i,j}$ denote the set of representative points for $L_{i,j}$, and let $\mathcal{G} = \bigcup_{i,j} \mathcal{G}_{i,j}$.*

CLAIM 5.7. *We have* $\ln|\mathcal{G}| = O(\log(\alpha k) + \log\log n + d\log(\beta d/\varepsilon))$.

The following lemma is analogous to Lemma 5.4; we omit the easy but tedious proof.

LEMMA 5.8. *With probability $\geq 1-\lambda/2$ for all sets C' of at most k centers chosen from $\mathcal{G}$, it holds that $|\mu(C',P) - \mu(C',\mathcal{S})| \leq (\varepsilon/5)\,\mu(C',P)$.*

CLAIM 5.9. *Let $\mu_{\text{opt}} = \mu_{\text{opt}}(k,P)$. It holds that*

(i) $\mu(\mathcal{A},\mathcal{S}) \leq 5\beta\mu_{\text{opt}}$;

(ii) $|\mu(C,P) - \mu(C,\mathcal{S})| \leq \mu(C,P) + 40\beta\mu_{\text{opt}}$ *for any set C of centers;*

(iii) $|\mu(C,P) - \mu(C,\mathcal{S})| \leq \frac{\varepsilon}{2}\mu(C,P) + \frac{20(2+\varepsilon)}{\varepsilon}\mu(\mathcal{A},P)$ *for any set C of centers.*

Proof. Consider a ring set $P_{i,j}$ and its corresponding weighted sample set $\mathcal{S}_{i,j}$. Because $\mathbf{w}(\mathcal{S}_{i,j}) = |P_{i,j}|$, there exists a map $f : P_{i,j} \to \mathcal{S}_{i,j}$ such that $\left|f^{-1}(q)\right| = \mathbf{w}(q)$ for all $q \in \mathcal{S}_{i,j}$.

(i) By Claim 5.3, we have

$$\begin{aligned}\mu(\mathcal{A},\mathcal{S}) &= \sum_{i,j}\mu(\mathcal{A},\mathcal{S}_{i,j}) = \sum_{i,j}\sum_{p\in P_{i,j}}(\mathbf{d}(\mathcal{A},f(p)))^2 \leq \sum_{i,j}\sum_{p\in P_{i,j}}(2^jR)^2\\ &= \sum_{i,j}|P_{i,j}|\,(2^jR)^2 \leq 5\beta\mu_{\text{opt}},\end{aligned}$$

since $\mathbf{d}(\mathcal{A},f(p)) \leq 2^jR$ for any $p \in P_{i,j}$.

(ii) Let p be an arbitrary point in $P_{i,j}$. We have $|\mathbf{d}(C,p) - \mathbf{d}(C,f(p))| \leq \mathbf{d}(p,f(p))$ and $|\mathbf{d}(C,p) + \mathbf{d}(C,f(p))| \leq 2\mathbf{d}(C,p) + \mathbf{d}(p,f(p))$, by the triangle inequality. Therefore,

$$\begin{aligned}|\mu(C,p) - \mu(C,f(p))| &= \left|(\mathbf{d}(C,p))^2 - (\mathbf{d}(C,f(p)))^2\right|\\ &= |\mathbf{d}(C,p) - \mathbf{d}(C,f(p))|\cdot|\mathbf{d}(C,p) + \mathbf{d}(C,f(p))|\\ &\leq \mathbf{d}(p,f(p))\cdot(2\mathbf{d}(C,p) + \mathbf{d}(p,f(p)))\\ &\leq 2\text{diam}(P_{i,j})\cdot\mathbf{d}(C,p) + (\text{diam}(P_{i,j}))^2,\end{aligned}$$

since $\mathbf{d}(p,f(p)) \leq \text{diam}(P_{i,j})$. It follows that

$$\begin{aligned}|\mu(C,P) - \mu(C,\mathcal{S})| &= \sum_{i,j}\sum_{p\in P_{i,j}}|\mu(C,p) - \mu(C,f(p))|\\ &\leq \sum_{i,j}\sum_{p\in P_{i,j}}\left(2\text{diam}(P_{i,j})\cdot\mathbf{d}(C,p) + (\text{diam}(P_{i,j}))^2\right). \qquad (5)\end{aligned}$$

Now, It follows from (5) and $2\text{diam}(P_{i,j})\cdot\mathbf{d}(C,p) \leq (\mathbf{d}(C,p))^2 + (\text{diam}(P_{i,j}))^2$ that

$$\begin{aligned}|\mu(C,P) - \mu(C,\mathcal{S})| &\leq \sum_{i,j}\sum_{p\in P_{i,j}}\left((\mathbf{d}(C,p))^2 + 2(\text{diam}(P_{i,j}))^2\right)\\ &= \sum_{i,j}\sum_{p\in P_{i,j}}(\mathbf{d}(C,p))^2 + 2\sum_{i,j}\sum_{p\in P_{i,j}}(\text{diam}(P_{i,j}))^2\\ &= \sum_{p\in P}(\mathbf{d}(C,p))^2 + 2\sum_{i,j}|P_{i,j}|\,(\text{diam}(P_{i,j}))^2 \leq \mu(C,P) + 40\beta\mu_{\text{opt}},\end{aligned}$$

since $\sum_{i,j}|P_{i,j}|\,(\text{diam}(P_{i,j}))^2 \leq 20\beta\mu_{\text{opt}}$, by Claim 5.3.

(iii) It follows from (5) and $2\mathrm{diam}(P_{i,j}) \cdot \mathbf{d}(C,p) \leq \frac{\varepsilon}{2}(\mathbf{d}(C,p))^2 + \frac{2}{\varepsilon}(\mathrm{diam}(P_{i,j}))^2$ that

$$\begin{aligned}|\mu(C,P) - \mu(C,\mathcal{S})| &\leq \sum_{i,j}\sum_{p\in P_{i,j}} \left(\frac{\varepsilon}{2}(\mathbf{d}(C,p))^2 + \frac{2}{\varepsilon}(\mathrm{diam}(P_{i,j}))^2 + (\mathrm{diam}(P_{i,j}))^2\right)\\ &\leq \frac{\varepsilon}{2}\mu(C,P) + \frac{20(2+\varepsilon)}{\varepsilon}\mu(\mathcal{A},P),\end{aligned}$$

since $\sum_{i,j}|P_{i,j}|\,(\mathrm{diam}(P_{i,j}))^2 \leq 20\mu(\mathcal{A},P)$, by Claim 5.3. □

In the following, let $C \subseteq \mathbb{R}^d$ be an arbitrary set of at most k centers. We need to prove that $|\mu(C,\mathcal{S}) - \mu(C,P)| \leq \varepsilon\,\mu(C,P)$. Recall that $\mathcal{U}$ is a union of balls centered at the points of $\mathcal{A}$; see Definition 5.6.

LEMMA 5.10. *For $0 < \varepsilon < 1$, if there exists a center $c \in C$ and a point $p \in P$ such that c is outside $\mathcal{U}$ and $\mathbf{d}(C,p) = \|cp\|$, then $|\mu(C,\mathcal{S}) - \mu(C,P)| \leq \varepsilon\,\mu(C,P)$.*

Proof. Let a_p be the nearest center to p in $\mathcal{A}$. We have $\|a_p p\|^2 \leq \mu(\mathcal{A},P) = \beta n R^2$, which implies that $\|a_p p\| \leq \sqrt{\beta n}R$. In addition, since c is outside $\mathcal{U}$, it holds that $\|ca_p\| \geq \mathbf{d}(\mathcal{A},c) \geq 2^{\Phi}R \geq 12\sqrt{\beta n}R/\varepsilon$; see Definition 5.6. By the triangle inequality, we thus have

$$\|cp\| \geq \|ca_p\| - \|a_p p\| \geq 12\sqrt{\beta n}R/\varepsilon - \sqrt{\beta n}R \geq 11\sqrt{\beta n}R/\varepsilon.$$

It follows that $\mu(C,P) \geq \mu(C,p) = \|cp\|^2 \geq 121\beta n R^2/\varepsilon^2 = 121\mu(\mathcal{A},P)/\varepsilon^2$, which implies

$$\mu(\mathcal{A},P) \leq \frac{\varepsilon^2}{121}\mu(C,P).$$

Now, by Claim 5.9(iii), we have

$$\begin{aligned}|\mu(C,\mathcal{S}) - \mu(C,P)| &\leq \frac{\varepsilon}{2}\mu(C,P) + \frac{20(2+\varepsilon)}{\varepsilon}\mu(\mathcal{A},P) < \frac{\varepsilon}{2}\mu(C,P) + \frac{60}{\varepsilon}\mu(\mathcal{A},P)\\ &\leq \frac{\varepsilon}{2}\mu(C,P) + \frac{60\varepsilon}{121}\mu(C,P) \leq \varepsilon\mu(C,P),\end{aligned}$$

since $20(2+\varepsilon)/\varepsilon < 60/\varepsilon$, implied by $0 < \varepsilon < 1$. □

The above lemma implies that we can assume that $C \subseteq \mathcal{U}$ (since otherwise, the set $\mathcal{S}$ is the required coreset). Therefore, suppose that $C = \{c_1, \ldots, c_h\}$, where $h \leq k$. Let $C' = \{c'_1, \ldots, c'_h\}$, where $c'_t \in \mathcal{G}$ is the representative point of the cell containing c_t for $t = 1, \ldots, h$.

LEMMA 5.11. *For $0 < \varepsilon < 1$, if $C \subseteq \mathcal{U}$ and $|C| \leq k$, then for any $q \in P$,*

$$|\mu(C,q) - \mu(C',q)| \leq \frac{\varepsilon}{\mathsf{b}\,\beta}(8\mu(C,q) + 4\mu(\mathcal{A},q) + R^2/2).$$

Proof. Let c_i and c'_j be the nearest centers to q in C and C', respectively. Namely, $\mu(C,q) = \|c_i q\|^2$ and $\mu(C',q) = \|c'_j q\|^2$. We consider the case when $\mu(C,q) \leq \mu(C',q)$, as the other case is similar. By the triangle inequality, it holds that

$$\begin{aligned}|\mu(C,q) - \mu(C',q)| &= \mu(C',q) - \mu(C,q) = \|c'_j q\|^2 - \|c_i q\|^2 \leq \|c'_i q\|^2 - \|c_i q\|^2\\ &= (\|c'_i q\| + \|c_i q\|)\cdot(\|c'_i q\| - \|c_i q\|) \leq (2\|c_i q\| + \|c'_i c_i\|)\cdot\|c'_i c_i\|\\ &= 2\|c_i q\|\cdot\|c'_i c_i\| + \|c'_i c_i\|^2 \leq \frac{4\varepsilon}{\mathsf{b}\,\beta}\|c_i q\|^2 + \frac{\mathsf{b}\,\beta}{4\varepsilon}\|c'_i c_i\|^2 + \|c'_i c_i\|^2\\ &< \frac{4\varepsilon}{\mathsf{b}\,\beta}\|c_i q\|^2 + \frac{\mathsf{b}\,\beta}{2\varepsilon}\|c'_i c_i\|^2 = \frac{4\varepsilon}{\mathsf{b}\,\beta}\mu(C,q) + \frac{\mathsf{b}\,\beta}{2\varepsilon}\|c'_i c_i\|^2,\end{aligned} \tag{6}$$

since $\left\|c'_j q\right\| \leq \|c'_i q\|$ (recall that c'_j is the nearest center to q in C') and $\mathsf{b}\,\beta/(4\varepsilon)+1 < \mathsf{b}\,\beta/(2\varepsilon)$, because $\mathsf{b}\,\beta > 4 > 4\varepsilon$. There are two cases.

(i) If $\mathbf{d}(\mathcal{A}, c_i) \leq R$, then

$$\|c_i c'_i\| \leq \sqrt{d} \cdot \frac{\varepsilon R}{\mathsf{b}\,\beta\sqrt{d}} = \frac{\varepsilon R}{\mathsf{b}\,\beta}.$$

Combining this and (6) implies the required bound.

(ii) Otherwise, we have $2^{t-1}R < \mathbf{d}(\mathcal{A}, c_i) \leq 2^t R$ for some $t \geq 1$. Then c_i and c'_i are inside a cell of side length $2^t \varepsilon R/(\mathsf{b}\,\beta\sqrt{d})$ and, as such,

$$\|c_i c'_i\| \leq \sqrt{d} \cdot \frac{2^t \varepsilon R}{\mathsf{b}\,\beta\sqrt{d}} = \frac{2\varepsilon}{\mathsf{b}\,\beta} \cdot 2^{t-1}R < \frac{2\varepsilon}{\mathsf{b}\,\beta}\mathbf{d}(\mathcal{A}, c_i) \leq \frac{2\varepsilon}{\mathsf{b}\,\beta}\left(\|c_i q\| + \mathbf{d}(\mathcal{A}, q)\right),$$

by the triangle inequality. Therefore,

$$\begin{aligned}\|c_i c'_i\|^2 &\leq \frac{4\varepsilon^2}{\mathsf{b}^2\beta^2}(\|c_i q\| + \mathbf{d}(\mathcal{A}, q))^2 \leq \frac{4\varepsilon^2}{\mathsf{b}^2\beta^2}\left(2\|c_i q\|^2 + 2(\mathbf{d}(\mathcal{A}, q))^2\right)\\ &\leq \frac{8\varepsilon^2}{\mathsf{b}^2\beta^2}\left(\mu(C, q) + \mu(\mathcal{A}, q)\right).\end{aligned}$$

Combining this and (6) implies the required bound. ∎

Lemma 5.12. *If $C \subseteq \mathcal{U}$ and $|C| \leq k$, then*

(i) $|\mu(C, P) - \mu(C', P)| \leq (\varepsilon/10)\mu(C, P)$, *and*

(ii) $|\mu(C, \mathcal{S}) - \mu(C', \mathcal{S})| \leq (\varepsilon/2)\mu(C, P)$.

Proof. We will sketch the proof only, and omit the tedious computations. Recall that $\mathsf{b} = 800$.

(i) Summing up the inequality of Lemma 5.11 over all the points of P, we obtain

$$|\mu(C, P) - \mu(C', P)| \leq \frac{\varepsilon}{\mathsf{b}\,\beta}(8\mu(C, P) + 4\mu(\mathcal{A}, P) + nR^2/2) < \frac{\varepsilon}{10}\mu(C, P).$$

(ii) Summing up the inequality of Lemma 5.11 over the weighted points of $\mathcal{S} \subseteq P$, we obtain

$$|\mu(C, \mathcal{S}) - \mu(C', \mathcal{S})| \leq \frac{\varepsilon}{\mathsf{b}\,\beta}(8\mu(C, \mathcal{S}) + 4\mu(\mathcal{A}, \mathcal{S}) + nR^2/2) \leq \frac{\varepsilon}{2}\mu(C, P),$$

since $\mu(C, \mathcal{S}) \leq \mu(C, P) + 40\beta\mu_{\mathrm{opt}}$, by Claim 5.9(ii), and $\mu(\mathcal{A}, \mathcal{S}) \leq 5\beta\mu_{\mathrm{opt}}$, by Claim 5.9(i). ∎

The proof of the following lemma is almost verbatim as that of Lemma 4.9, and is omitted.

Lemma 5.13. *For $0 < \varepsilon < 1$, and for any $C \subseteq \mathcal{U}$ such that $|C| \leq k$, it holds that*

$$|\mu(C, P) - \mu(C, \mathcal{S})| \leq \varepsilon\mu(C, P),$$

and this holds with probability $\geq 1 - \lambda/2$.

We summarize with the following theorem.

Theorem 5.14. *Given a set P of n points in $\mathbb{R}^d$ and parameters $1 > \varepsilon > 0$ and $\lambda > 0$, one can compute a weighted set $\mathcal{S}$ of size*

$$O\left(\frac{k\log n}{\varepsilon^2}\left(dk\log\frac{d}{\varepsilon} + k\log k + k\log\log n + \log\frac{1}{\lambda}\right)\right)$$

in $O(ndk\log(1/\lambda))$ time such that $\mathcal{S}$ is a (k,ε)-coreset of P for k-means clustering, with probability $\geq 1-\lambda$.

If P is a weighted point set, with total weight W, then the running time is $O(ndk\log(1/\lambda)\log\log W)$, and the coreset size is

$$O\left(\frac{k\log^2 W}{\varepsilon^2}\left(dk\log\frac{d}{\varepsilon}+k\log k+k\log\log W+\log\frac{1}{\lambda}\right)\right).$$

6. Applications.

In this section, we provide applications for the (k,ε)-coreset constructions described in sections 3, 4, and 5. We can plug the coresets into any k-clustering algorithm that works for a weighted point set.

6.1. Faster clustering algorithms.

In the metric spaces, we plug the local search algorithm of Arya et al. [6] into our machinery. Specifically, we compute a constant factor bicriteria approximation for the optimal solution using `FastCluster` (described in Appendix A). Next, we apply the coreset construction of Theorem 3.6 to compute a (k,ε)-coreset. Now, we plug the coreset into the following theorem.

THEOREM 6.1 (see [6]). *Given a set P of n points in a metric space, one can compute a $(5+\varepsilon)$-approximate k-median clustering of P in time $O(\varepsilon^{-1}n^2k^3\log n)$.*

If P is a weighted point set, with the total weight W, then the time required is $O(\varepsilon^{-1}n^2k^3\log W)$.

Proof. The proof we sketch here follows Har-Peled and Mazumdar [18] and Meyerson, O'Callaghan, and Plotkin [30].

The local search algorithm of Arya et al. [6] works in multiple steps. The algorithm maintains a solution C_i in the step i. It checks whether there exist a point $c\in C_i$ and a point $s\in P\setminus C_i$ such that $C_i'=C_i\setminus\{c\}\cup\{s\}$ satisfies $\nu(C_i',P)<(1-\varepsilon/k)\nu(C_i,P)$. If such a point set C_i' exists, then the algorithm goes to step $(i+1)$ and sets $C_{i+1}=C_i'$; otherwise, the algorithm terminates by returning C_i. Arya et al. prove that the returned solution is a $(5+\varepsilon)$-approximation to the k-median clustering of P. The number of steps required is $O(\varepsilon^{-1}k\log(\nu(C_0,P)/\nu_{\mathrm{opt}}(k,P)))$, where C_0 is the initial solution. Now, to compute C_0, we can use the min-max algorithm for the k-center clustering [15]. It is well known that such an initial solution C_0 satisfies $\nu(C_0,P)=O(n)\cdot\nu_{\mathrm{opt}}(k,P)$. Therefore, $O(\varepsilon^{-1}k\log n)$ steps suffice. In each step, the algorithm needs to consider $O(nk)$ candidate sets C_i', and computing $\nu(C_i',P)$ costs $O(nk)$ time. As such, the overall running time is $O(\varepsilon^{-1}n^2k^3\log n)$.

If P is weighted, with total weight W, the above analysis still holds, and the only difference is that the number of required steps is $O(\varepsilon^{-1}k\log W)$. □

THEOREM 6.2. *Given a set P of n points in a metric space, for $0<\varepsilon<1$, one can compute a $(10+\varepsilon)$-approximate k-median clustering of P in $O(nk+k^7\varepsilon^{-5}\log^5 n)$ time, with constant probability of success.*

Proof. We first compute a $(k,\varepsilon/60)$-coreset Q of P by using Theorem 3.6. Next, we apply Theorem 6.1 to Q, and let C be the returned $(5+\varepsilon/4)$-approximation solution to the k-median clustering of Q. We have

$$|\nu(C,Q)-\nu(C,P)|\leq(\varepsilon/60)\nu(C,P).$$

Let ν_1 denote the cost of the optimal k-median clustering of Q using a subset of Q as centers, let ν_2 denote the cost of the optimal k-median clustering of Q using a subset of P as centers, and let ν_3 denote the cost of the optimal k-median clustering of P using a subset of P as centers, namely $\nu_3=\nu_{\mathrm{opt}}(k,P)$. It is easy to verify that

$$\nu_2\leq(1+\varepsilon/60)\nu_3=(1+\varepsilon/60)\nu_{\mathrm{opt}}(k,P),$$

since Q is a (k, ε)-coreset of P. It follows from Theorem 6.1 that $\nu(C, Q) \leq (5+\varepsilon/4)\nu_1$. On the other hand, it is well known that $\nu_1 \leq 2\nu_2$, implying that

$$\nu(C, Q) \leq (10 + \varepsilon/2)\nu_2.$$

Combining the above inequalities, we have $\nu(C, P) \leq (10 + \varepsilon)\nu_{\mathrm{opt}}(k, P)$, as desired.

By Theorem 3.6, it takes $O(nk)$ time to compute Q. Note that Q is a weighted set with total weight n, and its size is $O\big(\varepsilon^{-2}k^2 \log n\big)$. As such, by Theorem 6.1, it takes $O(\varepsilon^{-1} \cdot (\varepsilon^{-2}k^2 \log^2 n)^2 k^3 \log n) = O\big(k^7\varepsilon^{-5} \log^5 n\big)$ time to compute C. Therefore, the overall running time is $O(nk + k^7\varepsilon^{-5} \log^5 n)$. □

In $\mathbb{R}^d$, we use the same algorithm with the twist that in the final stage we use the $(1+\varepsilon)$-approximate algorithm of Kumar, Sabharwal, and Sen [26, 27] and Sabharwal [32] (instead of the local search algorithm in the metric case).

THEOREM 6.3 (see [32]). *Given a point set Q of n points in $\mathbb{R}^d$ with total weight W, one can compute $(1+\varepsilon)$-approximation solutions to the weighted k-means clustering and weighted k-medians clustering in $O(2^{(k/\varepsilon)^{O(1)}} dn \log^k W)$ time, with constant probability.*

THEOREM 6.4. *Given a set P of n points in $\mathbb{R}^d$, one can compute a $(1 + \varepsilon)$-approximation to the optimal k-median (or k-means) clustering of P in time $O(ndk + 2^{(k/\varepsilon)^{O(1)}} d^2 \log^{k+2} n)$, with constant probability.*

Proof. For the Euclidean k-median clustering, we apply the coreset construction of Theorem 4.10 to compute (with constant probability) a (k, ε)-coreset Q of the set P. Now, we apply the weighted k-median method in Theorem 6.3 to Q. The correctness follows from Theorem 6.3 and the fact that Q is a (k, ε)-coreset of P. Next, we analyze the running time. By Theorem 4.10, it takes $O(ndk)$ time to compute Q. Once the coreset Q has been found, by Theorem 6.3, the weighted k-median clustering algorithm costs time

$$\begin{aligned} T &= O\bigg(2^{(k/\varepsilon)^{O(1)}} d \cdot \bigg(\frac{k \log n}{\varepsilon^2}\bigg(dk \log \frac{1}{\varepsilon} + k \log k + k \log \log n\bigg)\bigg) \cdot \log^k n\bigg) \\ &= O\Big(2^{(k/\varepsilon)^{O(1)}} d^2 \log^{k+2} n\Big). \end{aligned}$$

Therefore, the overall running time is $O(ndk + 2^{(k/\varepsilon)^{O(1)}} d^2 \log^{k+2} n)$.

The Euclidean k-means clustering is similar as above, by using the coreset construction of Theorem 5.14. □

6.2. Streaming. Coresets were used to design approximation algorithms in the streaming model [18, 3]. In particular, Har-Peled and Mazumdar [18] used coresets to develop approximation algorithms for k-clustering in the insertion-only streaming model. The randomized coreset construction described in this paper can also be used in the streaming model using the same techniques. See Appendix B for details. In particular, we have the following theorem.

THEOREM 6.5. *Given a stream P of n points in $\mathbb{R}^d$ and $\varepsilon > 0$, one can maintain a (k, ε)-coreset for k-median (or k-means) clustering efficiently for the points seen so far. The coreset is correct with high probability. The space used is $O(d^2k^2\varepsilon^{-2} \log^8 n)$, and the amortized update time is $O(dk\, \mathrm{polylog}(ndk/\varepsilon))$.*

7. Conclusions. In this paper, we used sampling techniques to extract a small (k, ε)-coreset for k-clustering in both metric spaces and high-dimensional Euclidean spaces. Such a coreset construction for metric spaces was not known before. In high-dimensional Euclidean spaces, this is the first construction with polynomial dependency on the dimension. The coreset can be used to obtain fast approximation algorithms for the k-median and k-means problems. It is especially useful in the streaming model of computation, where the small storage space is desired. In particular, we provide the first streaming clustering algorithm that has space complexity with polynomial dependency on the dimension.

In addition, the small coreset leads to an $O(ndk + 2^{(k/\varepsilon)^{O(1)}} d^2 \log^{k+2} n)$ time $(1+\varepsilon)$-approximation algorithm to the optimal k-clustering in $\mathbb{R}^d$, which succeeds with constant probability. This improves the work of Kumar, Sabharwal, and Sen [26, 27]. This result, together with the low-dimensional result of Har-Peled and Mazumdar [18], indicates, surprisingly, that the expensive part in computing k-clustering in $\mathbb{R}^d$ is answering nearest neighbor queries (this is the $O(ndk)$ term in the running time). In particular, a slight speedup can be achieved by using a fast data structure for approximate nearest neighbor; see [23].

In light of the recent result of Har-Peled and Kushal [17] (see also [11]), which constructed a low-dimensional coreset of size independent of n (but exponential in the dimension), it is natural to ask if one can construct a coreset of size with polynomial dependency on the dimension and with no dependency on n. We leave this as an open problem for further research. A more intriguing possibility is that one can construct coresets of size independent of the dimension altogether, as was done in the min-enclosing ball case [7].

Appendix A. A fast bicriteria approximation algorithm for metric k-clustering.

In this section, we show a bicriteria approximation algorithm for k-clustering of a point set P in a metric space. The new algorithm is a simple extension of the algorithm of Indyk [20], which computes an $[O(1), O(1)]$-bicriteria approximation for the k-median problem in $O(nk \operatorname{polylog}(nk))$ time, where $n = |P|$. We improve the running time to $O(nk)$ when $k = O(\sqrt{n})$, and show that the same algorithm can also compute a similar approximation for the k-means problem.

The required modifications of Indyk's algorithm are easy, and we include the details here only for the sake of completeness.

A.1. Review of the algorithm.

In the following, we assume that $k = O(\sqrt{n}\,)$ and P is unweighted; see Remark 3 below. (In fact, if $k = \Omega(\sqrt{n})$, the coreset computed by our algorithm is of size $\Omega(n)$, which is not an interesting case for our coreset construction.)

Let $D(p, q) = \mathbf{d}(p, q)$ when considering the k-median clustering case, and let $D(p, q) = (\mathbf{d}(p, q))^2$ when considering the k-means clustering case. That is, in either case, the cost of the clustering with respect to a center set C is $\tau(C, P) = \sum_{q \in P} D(C, q)$.

CLAIM A.1. *Given points $q_0, q_1, q_2 \in P$, we have that $D(C, q_0) \leq 3(D(C, q_2) + D(q_2, q_1) + D(q_1, q_0))$ for any $C \subseteq P$.*

Proof. Let c be the nearest point to q_2 in C. Then it suffices to prove that

$$D(c, q_0) \leq 3(D(c, q_2) + D(q_2, q_1) + D(q_1, q_0)),$$

since $D(C, q_0) \leq D(c, q_0)$ and $D(C, q_2) = D(c, q_2)$.

If $D(x, y) = \mathbf{d}(x, y)$, then this holds immediately by the triangle inequality. Otherwise, if $D(x, y) = (\mathbf{d}(x, y))^2$, then observe that $D(c, q_2) + D(q_2, q_1) + D(q_1, q_0)$ is minimized when $D(c, q_2) = D(q_2, q_1) = D(q_1, q_0)$ and $\mathbf{d}(c, q_2) + \mathbf{d}(q_2, q_1) + \mathbf{d}(q_1, q_0) = \mathbf{d}(c, q_0)$. Therefore,

```
FastCluster($k, P$)
1. Sample a set $\mathcal{X}$ of $\mathsf{m} = \left\lceil \mathsf{b}\sqrt{kn \ln k} \right\rceil$ points from $P$ with replacement.
2. $C' \leftarrow$ ApproxSlow($k, \mathcal{X}$).
3. Let $\mathcal{Y}$ be the set of $\mathsf{m}$ points furthest away from $C'$ in $P$.
4. $C'' \leftarrow$ ApproxSlow($k, \mathcal{Y}$).
5. Return $C' \cup C''$.
```

FIG. 2. *The bicriteria approximation algorithm for k-clustering* [20]. *Here b is a sufficiently large constant.*

$$D(c, q_0) = (\mathbf{d}(c, q_0))^2 = 9\left(\frac{\mathbf{d}(c, q_0)}{3}\right)^2 \leq 3(D(c, q_2) + D(q_2, q_1) + D(q_1, q_0)). \qquad \square$$

The algorithm `FastCluster` of Indyk [20] is depicted in Figure 2. It requires a "slow" black-box $[\alpha, \beta]$-bicriteria approximation algorithm `ApproxSlow` for k-clustering to work. Several known algorithms [24, 29] can serve for this purpose.

Let $\tau_{\text{opt}}(k, P)$ be the cost of the optimal k-clustering, and let $C_{\text{opt}} = \{c_1, \ldots, c_k\}$ be the set of centers realizing this optimal k-clustering of P. Let K_i denote the cluster of c_i in P, namely, $p \in \mathrm{K}_i$ if c_i is the nearest center to p in C_{opt}. Let $\mathrm{K}'_i = \mathcal{X} \cap \mathrm{K}_i$, where $\mathcal{X}$ is the random sample computed in step (1) of `FastCluster`. Let $H = \{i \mid |\mathrm{K}_i| \geq \mathsf{m}/k\}$ be the set of indices of the "heavy" clusters. Let $\widehat{\mathrm{K}} = \cup_{i \in H} \mathrm{K}_i$ and $\widehat{\mathrm{K}}' = \cup_{i \in H} \mathrm{K}'_i$. Note that $|\widehat{\mathrm{K}}| > n - \mathsf{m}$ (indeed, each cluster that is not heavy contains less than m/k points, and there are at most k such clusters).

CLAIM A.2 (see [20]). *Let $\mathcal{E}_1$ be the event that $\tau(C_{\text{opt}}, \mathcal{X}) \leq (1 + \varrho)\frac{\mathsf{m}}{n}\tau(C_{\text{opt}}, P)$. We have that $\psi_1 = \mathbf{Pr}[\mathcal{E}_1] \geq \varrho/(1 + \varrho)$.*

Proof. Consider an arbitrary sample $q \in \mathcal{X}$. The expected value of $D(C_{\text{opt}}, q)$ is $\tau(C_{\text{opt}}, P)/n$. It follows that

$$\mathbf{E}[\tau(C_{\text{opt}}, \mathcal{X})] = \frac{|\mathcal{X}|}{n}\tau(C_{\text{opt}}, P) = \frac{\mathsf{m}}{n}\tau(C_{\text{opt}}, P).$$

The claim now follows from the Markov inequality. $\square$

CLAIM A.3 (see [20]). *Let $0 < \gamma < 1$ be an arbitrary parameter, and let $\mathcal{E}_2$ be the event that $\frac{|\mathrm{K}_i|}{|\mathrm{K}'_i|} \leq (1 + \gamma)\frac{n}{\mathsf{m}}$ for all $i \in H$. We have that $\psi_2 = \mathbf{Pr}[\mathcal{E}_2] \geq 1 - k\exp(-\mathsf{m}^2\gamma^2/(8nk))$.*

Proof. Fix an index $i \in H$. It suffices to prove that

$$\mathbf{Pr}\left[\frac{|\mathrm{K}_i|}{|\mathrm{K}'_i|} > (1 + \gamma)\frac{n}{\mathsf{m}}\right] \leq \exp\left(-\frac{\mathsf{m}^2\gamma^2}{8nk}\right).$$

Since $1 + \gamma > \frac{1}{1 - \gamma/2}$, we have

$$\mathbf{Pr}\left[\frac{|\mathrm{K}_i|}{|\mathrm{K}'_i|} > (1 + \gamma)\frac{n}{\mathsf{m}}\right] < \mathbf{Pr}\left[\frac{|\mathrm{K}_i|}{|\mathrm{K}'_i|} > \frac{1}{1 - \gamma/2}\frac{n}{\mathsf{m}}\right] = \mathbf{Pr}\left[\frac{|\mathrm{K}'_i|}{|\mathrm{K}_i|} < \left(1 - \frac{\gamma}{2}\right)\frac{\mathsf{m}}{n}\right].$$

Therefore, it suffices to prove that

$$\mathbf{Pr}\left[|\mathrm{K}'_i| \leq \left(1 - \frac{\gamma}{2}\right)\frac{\mathsf{m}}{n}|\mathrm{K}_i|\right] \leq \exp\left(-\frac{\mathsf{m}^2\gamma^2}{8nk}\right),$$

which follows by Chernoff's inequality, since $|\mathrm{K}_i| \geq \mathsf{m}/k$ (by the definition of H) and $\mathbf{E}[|\mathrm{K}'_i|] = m\,|\mathrm{K}_i|/n$. $\square$

Consider a function $f_i : \mathrm{K}_i \to \mathrm{K}'_i$ for $i \in H$ such that every point of K'_i has at most $\lceil |\mathrm{K}_i| \,/\, |\mathrm{K}'_i| \rceil$ points assigned to it by f_i. For any point $p \in \mathrm{K}_i$, we have that $D(C', p) \le 3(D(C', f_i(p)) + D(f_i(p), c_i) + D(c_i, p))$, by Claim A.1. Recall that $\widehat{\mathrm{K}} = \cup_{i\in H} \mathrm{K}_i$ and $\widehat{\mathrm{K}}' = \cup_{i\in H} \mathrm{K}'_i$, and observe that

$$\begin{aligned}
\tau(C', \widehat{\mathrm{K}}) &= \sum_{i\in H} \sum_{p\in \mathrm{K}_i} D(C', p) \le 3 \sum_{i\in H} \sum_{p\in \mathrm{K}_i} \Big[D(C', f_i(p)) + D(f_i(p), c_i) + D(c_i, p) \Big] \\
&\le 3 \sum_{i\in H} \left\lceil \frac{|\mathrm{K}_i|}{|\mathrm{K}'_i|} \right\rceil \sum_{q\in \mathrm{K}'_i} \Big[D(C', q) + D(q, c_i) \Big] + 3\tau(C_{\mathrm{opt}}, \widehat{\mathrm{K}}) \\
&\le 3(2+\gamma) \frac{n}{\mathsf{m}} \sum_{i\in H} \sum_{q\in \mathrm{K}'_i} \Big[D(C', q) + D(C_{\mathrm{opt}}, q) \Big] + 3\tau(C_{\mathrm{opt}}, \widehat{\mathrm{K}}) \\
&= 3 \Big[(2+\gamma) \frac{n}{\mathsf{m}} \Big(\tau(C', \widehat{\mathrm{K}}') + \tau(C_{\mathrm{opt}}, \widehat{\mathrm{K}}') \Big) + \tau(C_{\mathrm{opt}}, \widehat{\mathrm{K}}) \Big] \\
&\le 3 \Big[(2+\gamma) \frac{n}{\mathsf{m}} \Big(\tau(C', \mathcal{X}) + \tau(C_{\mathrm{opt}}, \mathcal{X}) \Big) + \tau(C_{\mathrm{opt}}, P) \Big],
\end{aligned}$$

where the second inequality holds because for every $q \in \mathrm{K}'_i$ there are at most $\lceil |\mathrm{K}_i| \,/\, |\mathrm{K}'_i| \rceil$ points of K_i assigned to it by f_i; the third inequality holds with probability ψ_2 because

$$\left\lceil \frac{|\mathrm{K}_i|}{|\mathrm{K}'_i|} \right\rceil \le \frac{|\mathrm{K}_i|}{|\mathrm{K}'_i|} + 1 \le (1+\gamma)\frac{n}{\mathsf{m}} + 1 \le (2+\gamma)\frac{n}{\mathsf{m}}$$

by Claim A.3; and the last inequality holds because $\widehat{\mathrm{K}} \subseteq P$ and $\widehat{\mathrm{K}}' \subseteq \mathcal{X}$. Since `ApproxSlow` is an $[\alpha, \beta]$-bicriteria approximation algorithm for k-clustering, we have $\tau(C', \mathcal{X}) \le \beta\, \tau_{\mathrm{opt}}(k, \mathcal{X}) = \beta \tau(C_{\mathrm{opt}}, \mathcal{X})$. It thus follows that

$$\begin{aligned}
\tau(C', \widehat{\mathrm{K}}) &\le 3 \Big[(2+\gamma) \frac{n}{\mathsf{m}} \Big(\tau(C', \mathcal{X}) + \tau(C_{\mathrm{opt}}, \mathcal{X}) \Big) + \tau(C_{\mathrm{opt}}, P) \Big] \\
&\le 3 \Big[(2+\gamma) \frac{n}{\mathsf{m}} (1+\beta) \tau(C_{\mathrm{opt}}, \mathcal{X}) + \tau(C_{\mathrm{opt}}, P) \Big] \\
&\le 3 \Big[(2+\gamma) \frac{n}{\mathsf{m}} (1+\beta)(1+\varrho) \frac{\mathsf{m}}{n} \tau(C_{\mathrm{opt}}, P) + \tau(C_{\mathrm{opt}}, P) \Big] \\
&= 3((2+\gamma)(1+\beta)(1+\varrho) + 1)\, \tau_{\mathrm{opt}}(k, P), \qquad (7)
\end{aligned}$$

where the last inequality holds with probability ψ_1, by Claim A.2.

Because $|\widehat{\mathrm{K}}| \ge n - \mathsf{m}$, the cost of points in $P \setminus \mathcal{Y}$ with respect to center set C' does not exceed $\tau(C', \widehat{\mathrm{K}})$. Indeed, the points of $\mathcal{Y}$ are the m most expensive points in P with respect to C', and as such we have

$$\tau(C', P \setminus \widehat{\mathrm{K}}) = \sum_{p\in P\setminus \widehat{K}} D(C', p) \le \sum_{p\in \mathcal{Y}} D(C', p) = \tau(C', \mathcal{Y}),$$

since $|P \setminus \widehat{\mathrm{K}}| \le \mathsf{m}$. This implies

$$\tau(C', P \setminus \mathcal{Y}) = \tau(C', P) - \tau(C', \mathcal{Y}) \le \tau(C', P) - \tau(C', P \setminus \widehat{\mathrm{K}}) = \tau(C', \widehat{K}).$$

In addition, we have that $\tau(C'', \mathcal{Y}) \le \beta\, \tau_{\mathrm{opt}}(k, \mathcal{Y}) \le \beta\, \tau_{\mathrm{opt}}(k, P)$. Therefore, by (7),

$$\begin{aligned}
\tau(C' \cup C'', P) &\le \tau(C', P \setminus \mathcal{Y}) + \tau(C'', \mathcal{Y}) \le \tau(C', \widehat{K}) + \beta\, \tau_{\mathrm{opt}}(k, P) \\
&\le (3((2+\gamma)(1+\beta)(1+\varrho) + 1) + \beta)\, \tau_{\mathrm{opt}}(k, P) \\
&< 3(2+\gamma)(1+\beta)(2+\varrho)\, \tau_{\mathrm{opt}}(k, P).
\end{aligned}$$

Set $\gamma = 1/4$ and $\varrho = 3$. We have that $\psi_1 \geq \varrho/(1+\varrho) = 3/4$ and

$$\psi_2 \geq 1 - k\exp\left(-\frac{\mathsf{m}^2\gamma^2}{8nk}\right) \geq 1 - k\exp\left(-\frac{(\mathsf{b}\sqrt{kn\ln k}\,)^2\gamma^2}{8nk}\right) = 1 - k\exp\left(-\frac{\mathsf{b}^2\ln k}{128}\right) \geq \frac{3}{4}$$

for $\mathsf{b} \geq 20$, by Claims A.2 and A.3. It follows that the algorithm succeeds with probability $\mathbf{Pr}[\mathcal{E}_1 \cap \mathcal{E}_2] \geq \psi_1 + \psi_2 - 1 \geq 1/2$. (Note that $\mathcal{E}_1$ and $\mathcal{E}_2$ are not necessarily independent.)

Since $|C' \cup C''| \leq 2\alpha k$ and $3(2+\gamma)(1+\beta)(2+\varrho) = \frac{135}{4}(\beta+1)$, it follows that the algorithm `FastCluster` computes a $[2\alpha, \frac{135}{4}(\beta+1)]$-bicriteria approximation for k-clustering, with constant probability. If the black-box algorithm `ApproxSlow` runs in $g(n)$ time, then the new algorithm runs in time $O(nk + g(\sqrt{kn\ln k}\,))$. Note that we can boost the probability of success to be arbitrarily close to 1 by increasing ϱ and b (this of course would result in a worse approximation).

A.2. The result. The algorithm of Jain and Vazirani [24] can be used as the black-box algorithm `ApproxSlow` inside `FastCluster`, and we get a new algorithm, denoted by `IJVAlg`, which runs in time $O(nk\log k\log^2 n)$. (Note that the algorithm of Jain and Vazirani works for both k-median and k-means clustering.) Now, use `IJVAlg` as the black-box algorithm in `FastCluster`. The resulting algorithm returns an $[O(1), O(1)]$-bicriteria approximation, and the overall running time is $O\big(nk + \sqrt{kn\log k}\cdot k\log k\log^2 n\big) = O(nk)$, since by assumption $k = O(\sqrt{n})$. Since the algorithm `IJVAlg` might fail, we boost its success probability to, say, above 0.99 (by increasing ϱ and b as suggested above). It is now easy to verify that the new algorithm succeeds with probability $\geq 1/2$.

Remark 3. If P is weighted, with total weight W, we use the grouping technique of Mettu and Plaxton [29]. We group points with roughly equal weights together, run the unweighted algorithm on each group with confidence parameter set to $O(1/\log W)$, and combine the centers computed for each group. See [29] for details. This yields a constant factor approximation using $O(k\log W)$ centers with constant probability. The overall running time is $O(nk\log\log W)$.

Note that for the above algorithm, we can boost its success probability from constant to $\geq 1 - \lambda/2$ by running it $O(\log(1/\lambda))$ times, and take the best solution computed (that is, the solution with the cheapest cost). We summarize with the following theorem.

THEOREM A.4. *Given a set P of n points in a metric space and a parameter $k = O(\sqrt{n})$, one can compute $O(k)$ centers in $O(nk\log(1/\lambda))$ time, such that the cost of k-median clustering of P using these centers is within a constant factor of the optimal k-median clustering cost. The algorithm succeeds with probability $\geq 1 - \lambda/2$.*

If the input is weighted, with total weight W, the algorithm computes $O(k\log W)$ centers, and the running time is $O(nk\log(1/\lambda)\log\log W)$.

The same result holds verbatim for k-means clustering.

Appendix B. Streaming. In this section, we adapt the algorithm of Har-Peled and Mazumdar [18] to our randomized coresets. The required modifications are straightforward and are included here for the sake of completeness.

The algorithm of Har-Peled and Mazumdar is based on the standard dynamization technique of Bentley and Saxe [8] and the following observation.

OBSERVATION B.1.

(i) *If $\mathcal{S}_1$ and $\mathcal{S}_2$ are the (k,ε)-coresets for disjoint sets P_1 and P_2, respectively, then $\mathcal{S}_1 \cup \mathcal{S}_2$ is a (k,ε)-coreset for $P_1 \cup P_2$.*

(ii) *If $\mathcal{S}_1$ is a (k, ε)-coreset for $\mathcal{S}_2$ and $\mathcal{S}_2$ is a (k, δ)-coreset for $\mathcal{S}_3$, then $\mathcal{S}_1$ is a $(k, (1+\varepsilon)(1+\delta)-1)$-coreset for $\mathcal{S}_3$.*

Suppose that a sequence of points $p_1, p_2, \ldots$ in $\mathbb{R}^d$ arrive one by one in a stream. We want to compute a coreset for the k-clustering of the points that arrived so far, and the result should be correct with probability $\geq 1 - \lambda$, where λ is a prespecified confidence parameter.

We use buckets $B_0, B_1, \ldots$ to store the points. The capacity of bucket B_0 is M, where M is a parameter to be specified shortly, and the capacity of bucket B_i is $2^{i-1}M$ for $i \geq 1$. We will keep the invariant that B_i is either full or empty for $i \geq 1$. When p_m arrives, we insert p_m into B_0. If B_0 has fewer than M points, then we are done. Otherwise, let $t \geq 1$ be the smallest index such that B_t is empty, and merge all the points of $B_0, \ldots, B_{t-1}$ into B_t. Here, B_t is *triggered* by p_m. (After B_t is triggered by p_m, the buckets $B_0, \ldots, B_{t-1}$ become empty and B_t becomes full.)

However, we cannot afford (spacewise) to keep every point in the buckets in the streaming model. Instead, we maintain a coreset Q_i for each bucket B_i. Q_0 is B_0 itself, and whenever B_t is triggered by p_m, let Q_t be a (k, ρ_t)-coreset of $\bigcup_{i=0}^{t-1} Q_i$ with confidence parameter $\lambda_m = \lambda/m^2$, where $\rho_t = \varepsilon/(\mathsf{b}\,(t+1)^2)$ and b is a sufficiently large constant. Let $Q = \bigcup_{i \geq 0} Q_i$.

Claim B.2. *The set Q is a (k, ε)-coreset of the points received so far, with probability $\geq 1 - \lambda$.*

Proof. Recall that $\rho_r = \varepsilon/(\mathsf{b}\,(r+1)^2)$. It is easy to verify that $\prod_{l=0}^{r}(1+\rho_l) \leq 1+\varepsilon$ if b is sufficiently large. On the other hand, Q_r is a $(k, \prod_{l=0}^{r}(1+\rho_l) - 1)$-coreset of B_r, by applying Observation B.1 repeatedly. Therefore, Q_r is a (k, ε)-coreset of the points in B_r, and $Q = \bigcup_{i \geq 0} Q_i$ is a (k, ε)-coreset of the points in $\bigcup_{i \geq 0} B_i$. That is, Q is a (k, ε)-coreset of the points received so far.

When we process the newly arrived point p_m, our computation may fail with probability $\leq \lambda_m = \lambda/m^2$ whenever we compute a coreset with confidence parameter λ_m. When p_m arrives, where $m \geq M$, it may trigger at most one coreset computation. As such, overall, the algorithm may fail with probability $\leq \sum_{i=M}^{n} \lambda_i = \sum_{i=M}^{n} \big(\lambda/i^2\big) \leq \lambda$ for $M \geq 2$. ∎

Set $M = \lceil k^2 \varepsilon^{-2} d \rceil$ and assume that we have received n points so far. Note that $|Q_0| \leq M$. For $i = 1, \ldots, \lceil \log n \rceil$, Q_i has a total weight $2^{i-1}M$ (if it is not empty) and it is generated as a $\big(k, \varepsilon/(\mathsf{b}\,i^2)\big)$-coreset of $\bigcup_{j=0}^{i-1} Q_j$ with confidence parameter at least λ/n^2. By Theorem 4.10, we have that

$$|Q_i| = O\bigg(\frac{ki^4(i+\log M)^2}{\varepsilon^2}\bigg(dk \log \frac{i^2}{\varepsilon} + k \log k + k \log(i + \log M) + \log \frac{n^2}{\lambda}\bigg)\bigg).$$

If $\lambda = 1/\mathrm{poly}(n)$, then the total storage requirement is

$$M + \sum_{i=1}^{\lceil \log n \rceil} |Q_i| = O\big(dk^2 \varepsilon^{-2} \log^8 n\big).$$

Note that we assume $\varepsilon > 1/n$ and $d \leq n$ in the computation above. These assumptions are valid, since otherwise, the total number of points arrived so far, n, is also $O(dk^2\varepsilon^{-2}\log^8 n)$. Now, since each point in $\mathbb{R}^d$ uses $O(d)$ space, the total space required is $O(d \cdot dk^2 \varepsilon^{-2} \log^8 n) = O(d^2 k^2 \varepsilon^{-2} \log^8 n)$.

To analyze the update time of the data structure, observe that the amortized time dealing with Q_0 is constant, and Q_i is constructed after every $2^{i-1}M$ insertions are made for $i = 1, \dots, \lceil \log n \rceil$. Therefore by Theorem 4.10, the amortized time spent for an update is

$$O\left(\sum_{i=1}^{\lceil \log n \rceil} \frac{\sum_{j=0}^{i-1} |Q_j|}{2^{i-1}M} dk \big(\log \log \big(2^{i-1}M \big) \big) \log \frac{n^2}{\lambda} \right) = O\left(dk \big(\log^2 n \big) \operatorname{polylog} \left(\frac{dk}{\varepsilon} \right) \right).$$

This implies Theorem 6.5.

Acknowledgments. The author thanks Sariel Har-Peled for helpful discussions on the problems studied in the paper and his comments on the manuscript. The author also thanks the anonymous referees for their detailed and useful comments.

REFERENCES

[1] M. R. Ackermann and J. Blömer, *Coresets and approximate clustering for Bregman divergences*, in Proceedings of the 20th ACM-SIAM Symposium on Discrete Algorithms, New York, 2009, pp. 1088–1097.

[2] M. R. Ackermann, J. Blömer, and C. Sohler, *Clustering for metric and non-metric distance measures*, in Proceedings of the 19th ACM-SIAM Symposium on Discrete Algorithms, San Francisco, 2008, pp. 799–808.

[3] P. K. Agarwal, S. Har-Peled, and K. R. Varadarajan, *Approximating extent measures of points*, J. ACM, 51 (2004), pp. 606–635.

[4] P. K. Agarwal, S. Har-Peled, and K. R. Varadarajan, *Geometric approximation via coresets*, in Current Trends in Combinatorial and Computational Geometry, J. E. Goodman, J. Pach, and E. Welzl, eds., Cambridge University Press, Cambridge, UK, 2005, pp. 1–30.

[5] S. Arora, P. Raghavan, and S. Rao, *Approximation schemes for Euclidean k-median and related problems*, in Proceedings of the 30th Annual ACM Symposium on Theory of Computing, Dallas, 1998, pp. 106–113.

[6] V. Arya, N. Garg, R. Khandekar, A. Meyerson, K. Munagala, and V. Pandit, *Local search heuristic for k-median and facility location problems*, SIAM J. Comput., 33 (2004), pp. 544–562.

[7] M. Bădoiu and K. L. Clarkson, *Optimal core-sets for balls*, in Proceedings of the 14th ACM-SIAM Symposium on Discrete Algorithms, Baltimore, 2003, pp. 801–802.

[8] J. L. Bentley and J. B. Saxe, *Decomposable searching problems* I: *Static-to-dynamic transformation*, J. Algorithms, 1 (1980), pp. 301–358.

[9] M. Charikar, S. Guha, E. Tardos, and D. B. Shmoys, *A constant-factor approximation algorithm for the k-median problem*, J. Comput. System Sci., 65 (2002), pp. 129–149.

[10] M. Charikar, L. O'Callaghan, and R. Panigrahy, *Better streaming algorithms for clustering problems*, in Proceedings of the 35th Annual ACM Symposium on Theory of Computing, San Diego, 2003, pp. 30–39.

[11] M. Effros and L. J. Schulman, *Deterministic Clustering with Data Nets*, Technical report TR04-050, Electronic Colloquium on Computational Complexity, University of Trier, Tier, Germany, 2004.

[12] D. Feldman, A. Fiat, and M. Sharir, *Coresets for weighted facilities and their applications*, in Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science, Berkeley, CA, 2006, pp. 315–324.

[13] D. Feldman, M. Monemizadeh, and C. Sohler, *A PTAS for k-means clustering based on weak coresets*, in Proceedings of the 23rd Annual ACM Symposium on Computational Geometry, Gyeongju, South Korea, 2007, pp. 11–18.

[14] G. Frahling and C. Sohler, *Coresets in dynamic geometric data streams*, in Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, 2005, pp. 209–217.

[15] T. Gonzalez, *Clustering to minimize the maximum intercluster distance*, Theoret. Comput. Sci., 38 (1985), pp. 293–306.

[16] S. Guha, A. Meyerson, N. Mishra, R. Motwani, and L. O'Callaghan, *Clustering data streams: Theory and practice*, IEEE Trans. Knowl. Data Eng., 15 (2003), pp. 515–528.

[17] S. Har-Peled and A. Kushal, *Smaller coresets for k-median and k-means clustering*, in Proceedings of the 21st Annual ACM Symposium on Computational Geometry, Pisa, 2005, pp. 126–134.

[18] S. Har-Peled and S. Mazumdar, *Coresets for k-means and k-median clustering and their applications*, in Proceedings of the 36th Annual ACM Symposium on Theory of Computing, Chicago, 2004, pp. 291–300.

[19] D. Haussler, *Decision theoretic generalizations of the PAC model for neural net and other learning applications*, Inform. and Comput., 100 (1992), pp. 78–150.

[20] P. Indyk, *Sublinear time algorithms for metric space problems*, in Proceedings of the 31st Annual ACM Symposium on Theory of Computing, Atlanta, 1999, pp. 154–159.

[21] P. Indyk, *High-dimensional computational geometry*, Ph.D. dissertation, Department of Computer Science, Stanford University, Stanford, CA, 2000.

[22] P. Indyk, *Algorithms for dynamic geometric problems over data streams*, in Proceedings of the 36th Annual ACM Symposium on Theory of Computing, Chicago, 2004, pp. 373–380.

[23] P. Indyk and R. Motwani, *Approximate nearest neighbors: Towards removing the curse of dimensionality*, in Proceedings of the 30th Annual ACM Symposium on Theory of Computing, Dallas, 1998, pp. 604–613.

[24] K. Jain and V. V. Vazirani, *Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and Lagrangian relaxation*, J. ACM, 48 (2001), pp. 274–296.

[25] S. G. Kolliopoulos and S. Rao, *A nearly linear-time approximation scheme for the Euclidean k-median problem*, SIAM J. Comput., 37 (2007), pp. 757–782.

[26] A. Kumar, Y. Sabharwal, and S. Sen, *A simple linear time* $(1+\varepsilon)$*-approximation algorithm for k-means clustering in any dimensions*, in Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science, Rome, 2004, pp. 454–462.

[27] A. Kumar, Y. Sabharwal, and S. Sen, *Linear time algorithms for clustering problems in any dimensions*, in Proceedings of the 32nd International Colloquium on Automata, Languages and Programming, Lisboa, Portugal, 2005, pp. 1374–1385.

[28] J. Matoušek, *On approximate geometric k-clustering*, Discrete Comput. Geom., 24 (2000), pp. 61–84.

[29] R. R. Mettu and C. G. Plaxton, *Optimal time bounds for approximate clustering*, Mach. Learn., 56 (2004), pp. 35–60.

[30] A. Meyerson, L. O'Callaghan, and S. Plotkin, *A k-median algorithm with running time independent of data size*, Mach. Learn., 56 (2004), pp. 61–87.

[31] N. Mishra, D. Oblinger, and L. Pitt, *Sublinear time approximate clustering*, in Proceedings of the 12nd ACM-SIAM Symposium on Discrete Algorithms, Washington, DC, 2001, pp. 439–447.

[32] Y. Sabharwal, *Approximation Algorithms for Proximity and Clustering Problems*, Ph.D. dissertation, Department of Computer Science, Indian Institute of Technology Delhi, Delhi, 2006.

[33] M. Thorup, *Quick k-median, k-center, and facility location for sparse graphs*, SIAM J. Comput., 34 (2005), pp. 405–432.