Improved Coresets and Sublinear Algorithms for Power Means in Euclidean Spaces

Vincent Cohen-Addad[*]
Google Research, Zurich.

David Saulpic[*]
Sorbonne Université, Paris

Chris Schwiegelshohn[*]
Aarhus University

Abstract

In this paper, we consider the problem of finding high dimensional power means: given a set A of n points in $\mathbb{R}^d$, find the point m that minimizes the sum of Euclidean distance, raised to the power z, over all input points. Special cases of problem include the well-known Fermat-Weber problem – or geometric median problem – where $z = 1$, the mean or centroid where $z = 2$, and the Minimum Enclosing Ball problem, where $z = \infty$.

We consider these problem in the big data regime. Here, we are interested in sampling as few points as possible such that we can accurately estimate m. More specifically, we consider sublinear algorithms as well as coresets for these problems. Sublinear algorithms have a random query access to the set A and the goal is to minimize the number of queries. Here, we show that $\widetilde{O}\left(\varepsilon^{-z-3}\right)$ samples are sufficient to achieve a $(1+\varepsilon)$-approximation, generalizing the results from Cohen, Lee, Miller, Pachocki, and Sidford [STOC '16] and Inaba, Katoh, and Imai [SoCG '94] to arbitrary z. Moreover, we show that this bound is nearly optimal, as any algorithm requires at least $\Omega\left(\varepsilon^{-z+1}\right)$ queries to achieve said approximation.

The second contribution are coresets for these problems, where we aim to find find a small, weighted subset of the points which approximates cost of every candidate point $c \in \mathbb{R}^d$ up to a $(1 \pm \varepsilon)$ factor. Here, we show that $\widetilde{O}\left(\varepsilon^{-2}\right)$ points are sufficient, improving on the $\widetilde{O}\left(d\varepsilon^{-2}\right)$ bound by Feldman and Langberg [STOC '11] and the $\widetilde{O}\left(\varepsilon^{-4}\right)$ bound by Braverman, Jiang, Krauthgamer, and Wu [SODA 21].

1 Introduction

Large data sets have shifted the focus of algorithm design. In the past, an algorithm might have been deemed feasible if its running time was polynomial in the input size and so a textbook *fast* algorithm can have time complexity for example quadratic. For truly gargantuan data sets, even linear time or nearly linear time algorithms could be considered too slow or requiring too much memory. This led to the emergence of the field of sublinear algorithms: How well can we solve a problem without reading the entire input?

Except for trivial problems, deterministic time sublinear algorithms do not exist. Our primary tool in designing sublinear algorithms is thus the following basic approach:

- Take a uniform sample of the input.
- Run an algorithm on the sample.

Hence, the performance of a sublinear algorithm is often measured in terms of its *query complexity*, i.e. the number of samples required such that we can extract a high quality solution in the second

[*]Equal contribution.

35th Conference on Neural Information Processing Systems (NeurIPS 2021).

step above that generalizes to the entire input. Sublinear algorithms have close ties to questions in learning theory and estimation theory, where we are similarly interested in a quality to sample size tradeoff.

A perhaps very fundamental problem of primary importance in machine learning and data analysis is to efficiently estimate the parameters of a distribution. For example, given a distribution $\mathcal{D}$, how many samples do we need to estimate the mean? Even such a simple and basic question has surprisingly involved answers and are still subject to ongoing research (Lugosi & Mendelson (2019); Lee & Valiant (2021)).

In this paper, we investigate the possibility of estimating power means in high dimensional Euclidean spaces. Specifically, given an arbitrary set of points A, we wish to determine the number of uniform queries S such that we can extract a power mean m with

$$\text{cost}(m) := \sum_{p \in A} \|p - m\|^z \leq (1+\varepsilon) \cdot \min_{\mu} \sum_{p \in A} \|p - \mu\|^z,$$

where $\|p\|$ denotes the Euclidean norm of a vector p.

The power mean problem captures a number of important problems in computational geometry and multivariate statistics. For example, for $z = 1$, this corresponds to the Fermat-Weber problem also known as the geometric median. For $z = 2$, the problem is to determine the mean or centroid of the data set. Letting $z \to \infty$, we have the *Minimum Enclosing Ball* (MEB), where one needs to find the Euclidean sphere of smallest radius containing all input points.

For $z > 2$, the problem is not as well studied, but it still has many applications. First, higher powers allows us to interpolate between $z = 2$ and $z \to \infty$, which is interesting as the latter admits no sublinear algorithms[2]. Skewness (a measure of the asymmetry of the probability distribution of a real-valued random variable about its mean) and kurtosis (a measure of the "tailedness" of the probability distribution) are the centralized moments with respect to the three and the four norms and are frequently used in statistics. The power mean is a way of estimating these values for multivariate distributions.

Another application is when dealing with non-Euclidean distances, such as the Hamming metric, coreset constructions for powers of z can be reduced to coreset constructions for powers $2z$. So for example if we want the mean in Hamming space, we can reduce it to the $z = 4$ case /in squared Euclidean spaces Huang & Vishnoi (2020).

These problems are convex and thus can be approximated in the near-linear time efficiently via convex optimization techniques. However, aside from the mean ($z = 2$), doing so in a sublinear setting is challenging and to the best of our knowledge, only the mean and the geometric median ($z = 1$) are currently known to admit nearly linear time algorithms.

Our main result is:

Theorem 1. *There exists an algorithm that, with query complexity $O(\varepsilon^{-z-3} \cdot \text{polylog}(\varepsilon^{-1}) \log^2 1/\delta)$, computes a $(1+\varepsilon)$ approximate solution to the high dimensional power mean problem with probability at least $1-\delta$.*

A key component in designing this algorithm is a novel analysis for *coresets* for these problems. Coresets are succinct summaries that approximate cost for any center solution c.

Theorem 2. *For any set of points in d-dimensional Euclidean space, there exists an ε-coreset for the high dimensional power mean problem of size $\tilde{O}(\varepsilon^{-2} \cdot 2^{O(z)})$.*

With the exception of the mean, previous coresets for these problems achieved bounds $O(\varepsilon^{-2} \cdot 2^{O(z)} \cdot \min(d, \varepsilon^{-2}))$ (Cohen-Addad et al. (2021); Braverman et al. (2021); Feldman & Langberg (2011)), or had weaker guarantees such as merely approximating an optimal solution or requiring removal of outliers from the data set.

Comparing the bounds in Theorem 1 and Theorem 2, one may question whether the exponential dependency in z is necessary for computing an approximation. Indeed, previous sublinear algorithms

[2]To see this, we place $n-1$ points at 0 and one point with probability $1/2$ at 1 and with probability $1/2$ at 0. Any $2-\varepsilon$ approximation can distinguish between the two cases, but this clearly requires us to query $\Omega(n)$ points.

for both the geometric median and the mean had a query complexity of $O(\varepsilon^{-2})$, and thus matched our coresets bounds. Unfortunately, we show that the exponential dependency in the power is indeed necessary even in a single dimension:

Theorem 3. *For any $\varepsilon > 0$ and z, any algorithm that computes with probability more than $4/5$ a $(1+\varepsilon)$-approximation for a one-dimensional power mean has query complexity $\Omega(\varepsilon^{-z+1})$.*

Hence, up to constants in the exponent, our sublinear algorithm is tight. Moreover, the algorithm is very simple to implement and performs well empircally.

1.1 Techniques

While stochastic gradient descent has been used for a variety of center-based problems (Clarkson et al. (2012); Cohen et al. (2016)), it is difficult to apply it for higher powers. Indeed, Cohen et al. (2016) remark in their paper that even for the mean[3] ($z = 2$) their analysis does not work as the objective function is neither Lipschitz, nor strictly convex.

Hence, one needs to use new tools. A natural starting point is to use techniques from coresets, as they allow us to preserve most of the relevant information, using a substantially smaller number of points. Unfortunately, coresets have a drawback: the sampling distributions used to construct coresets is non-uniform and therefore difficult to use in a sublinear setting. The first step is therefore to design coresets from uniform sampling. To do this, we use and improve upon a technique originally introduced by Chen (2009). Chen showed that, given a sufficiently good initial solution q, one can partition the points into rings exponentially increasing radii such that the points cost the same, up to constant factors. Thereafter, taking a uniform sample of size $\tilde{O}(d \cdot \varepsilon^{-2})$ from each ring produces a coreset. Since there are at most $O(\log n)$ rings in the worst case, this yields a coreset of size $\tilde{O}(d \cdot \varepsilon^{-2} \cdot \log n)$.

To realize these ideas in a sublinear setting, we are now faced with a number of challenges. First, rings that are particularly far from q may contain very few points. This makes is difficult for a sublinear algorithm to access them. Second, partitioning the points into rings depends on the cost of q. It is very simple to construct examples where estimating the cost of an optimal power center requires $\Omega(n)$ many queries. Finally, this analysis loses both factors $\log n$ and d, which we aim to avoid.

We improve and extend this framework in two ways. First, we show that it is sufficient to only consider $O(\log \varepsilon^{-1})$ many rings, which, in of itself, already removes the dependency on $\log n$. Moreover, we show that it is possible to simply ignore any ring containing too few points, i.e. any ring with less than $\varepsilon^{z+O(1)} \cdot n$ points may be discarded. The intuition is that, while rings with few point may contribute significantly to the cost, these points do not influence the position of the optimal center by much. Thus, using a number of carefully chosen pruning steps, we show how to reduce both the problem of obtaining a sublinear algorithm as well as obtaining coresets to sampling from a select few rings containing many points.

The second improvement directly considers an improved analysis of sampling from rings. The standard way to prove a coreset guarantee is to show that using $s \log 1/\delta$ samples we preserve the cost of a single solution with probability $1 - delta$. If there exist T solutions then we set $\delta = 1/T$ and have obtained a coreset, which we call the "naive" union bound. This works in certain cases such as finite metrics, but is insufficient if we have infinitely many solutions such as Euclidean spaces. The simplest way to improve over the naive union bound is to discretize the space and then apply the naive union bound on the discritization. In literature this is sometimes called an ε-net bound. This can be optimal or close to optimal for certain metrics, but so far these arguments have only lead to $\tilde{O}(\varepsilon^{-2} \min(d, \varepsilon^{-2}))$ bounds for coresets in Euclidean spaces.

Instead of applying a union bound over the discretization in "one shot", we apply a union bounded over a nested sequence of increasingly better discretizations. Essentially, instead of only using a set of centers C_ε as a substitute for all centers is $\mathbb{R}^d$, we use centers $c^h \in C_h$ for different values of $h \in \{1, \ldots, \log 1/\varepsilon\}$. In literature, this is known as chaining, see Nelson (2016) for a survey. We can then write, for any input point set P, $\sum_{p \in P} \text{cost}(p, c) = \sum_{p \in P} \sum_{h \geq 0} \text{cost}(p, c^{h+1}) - \text{cost}(p, c^h)$, where $\text{cost}(p, c^0)$ is defined to be 0 and $|\text{cost}(p, c^h) - \text{cost}(p, c)| \leq 2^{-h}\text{cost}(p, c)$. We now only

[3]For the special case of the mean, Inaba et al. Inaba et al. (1994) observed that $O(\varepsilon^{-2})$ samples are nevertheless sufficient.

apply the naive union bound for successive summands, i.e. we approximate $\sum_{p\in P} \text{cost}(p, c^{h+1}) - \text{cost}(p, c^h)$ up to an error of $\varepsilon \cdot \sum_{p\in P} \text{cost}(p, c)$.

Essentially, the idea is to use a sequence of solutions c_h that approximate a candidate solution c with increasing accuracy as $h \to \infty$. Approximating the cost of a candidate center c can then be written as a telescoping sum of solutions c_h, i.e. $\text{cost}(c) = \sum_{h=0}^{\infty} \text{cost}(c_{h+1}) - \text{cost}(c_h)$, where c_h is a solution that has *distance* to c of the order 2^{-h} and $\text{cost}(c_0) := 0$. The notion of distance between candidate solutions is perhaps most easily understood by imagining a solution to be a cost vector v^c where the ith entry corresponds to the cost of the ith point of A when using c as a candidate center, i.e., $v_i^c = \text{cost}(p_i, c)$. Informally, we consider two solutions c and c' to have distance at most 2^{-h} if $|\|p_i - c\|^z - \|p_i - c'\|^z| \leq 2^{-h} \cdot (\|p_i - c\|^z + \|p_i - c'\|^z)$ for all points $p_i \in A$.

1.2 Related Work

Sublinear Approximation for Clustering: A number of sublinear algorithm are known for clustering problems. For k-Median, under the constraint that the input space has a small diameter, a constant factor approximation is known Czumaj & Sohler (2007). Ben-David (2007) proposed a different set of conditions under which a sublinear algorithm for k-median and k-means exists. Other approximations, with different constraint are also known: for instance, Meyerson et al. (2004) give an algorithm achieving a $O(1)$-approximation in time poly(k/ε) for discrete metrics, when each cluster has size $\Omega(n\varepsilon/k)$. For the 1-median problem, this assumption is always satisfied, and their algorithm gives a constant factor approximation in constant running time. The algorithm by Cohen et al. (2016) produces a $(1+\varepsilon)$-approximation in time $\min(nd\log^3(n/\varepsilon), d/\varepsilon^2)$ for Euclidean spaces of dimension d. Ding (2020, 2021) and Clarkson et al. (2012) showed how to obtain sublinear algorithms for the minimum enclosing ball problem assuming that either the algorithm is allowed to drop a fraction of the points, or with an additive error. For the unconstrained version of the problem, no sublinear time algorithm is possible. For the k-means problem, Bachem et al. (2016) showed how to approximate the k-means++ algorithm in sublinear time. To our knowledge, no algorithm is known for higher distance powers z.

Coreset Constructions: A coreset is a weighted subset of the input points, such that the cost of any clustering is the same on the coreset than on the input points, up to a $(1 \pm \varepsilon)$ factor. The arguably most widely studied problem is coresets for the k clustering problems with powers of distances. Following a long line of work Bachem et al. (2018); Becchetti et al. (2019); Braverman et al. (2021); Chen (2009); Cohen-Addad et al. (2021); Feldman & Langberg (2011); Feldman et al. (2020); Feng et al. (2021); Fichtenberger et al. (2013); Har-Peled & Kushal (2007); Har-Peled & Mazumdar (2004); Huang & Vishnoi (2020); Langberg & Schulman (2010); Sohler & Woodruff (2018), we now know that coreset of size $\tilde{O}(k\varepsilon^{-4} \cdot 2^{O(z)} \cdot \min(k, \varepsilon^{-\max z-2,0)}))$ exist. In a low-dimensional regime, this may be improved to $\tilde{O}(k^2 d\varepsilon^{-2} 2^{O(z)})$ and $\tilde{O}(kd\varepsilon^{-2z})$ Feldman & Langberg (2011). For the special case of clustering with a single center, this yields the state of the art $\tilde{O}(\varepsilon^{-4})$ bound due to Braverman et al. (2021). For the Minimum Enclosing Ball problem and its generalizations, these algorithms yield no space saving, although sketches of weaker gaurantees of size $O(\varepsilon^{-1})$ exist (Badoiu & Clarkson (2008)). Given a coreset, it is easy to compute a $(1+\varepsilon)$-approximation in time independent of n: one just needs to find a $(1+\varepsilon)$-approximation on the coreset points.

Coresets have also been studied for many other problems: we cite non-comprehensively fair clustering Cohen-Addad & Li (2019); Huang et al. (2019); Schmidt et al. (2019) determinant maximization Indyk et al. (2020), diversity maximization Ceccarello et al. (2018); Indyk et al. (2014) logistic regression Huggins et al. (2016); Munteanu et al. (2018), dependency networks Molina et al. (2018), or low-rank approximation Maalouf et al. (2019). The interested reader is referred to the recent surveys for more information Feldman (2020); Munteanu & Schwiegelshohn (2018).

1.3 Organization

We prove our key sampling result in Section 2. We follow up on that result by applying it to the sublinear setting (Section 3) and to coresets (Section 4). We conclude with a short experimental evaluation. Due to space constraints, all proofs are included in the supplementary material.

1.4 Preliminaries

We denote the Euclidean distance of a vector x by $\|x\| := \sqrt{\sum_{i=1}^d x_i^2}$. Similarly, we define the Hamming norm $\|x\|_1 = \sum_{i=1}^d |x_i|$. For a set of points A, we say that $\|A\|_0$ is the distinct number

of points. Note that if A contains multiplicities $|A| \neq \|A\|_0$. We write $\tilde{O}(x)$ to denote $O(x \cdot \log^a x)$, where a is any constant. For any set A and candidate solution c, we defined $\text{cost}(A, c) = \sum_{p \in A} \|p - c\|^z$. If the set is clear from context, we simply write $\text{cost}(c)$. We frequently use the following generalized triangle inequality, see Cohen-Addad & Schwiegelshohn (2017); Makarychev et al. (2019) for proofs and similar statements.

Lemma 1 (Triangle Inequality for Powers)**.** *Let a, b, c be an arbitrary set of points in a metric space with distance function d and let z be a positive integer. Then for any $\varepsilon > 0$*

$$d(a,b)^z \leq (1+\varepsilon)^{z-1} d(a,c)^z + \left(\frac{1+\varepsilon}{\varepsilon}\right)^{z-1} d(b,c)^z$$

$$|d(a,b)^z - d(a,c)^z| \leq \varepsilon \cdot d(a,c)^z + \left(\frac{z+\varepsilon}{\varepsilon}\right)^{z-1} d(b,c)^z.$$

We also use the fact that uniform sampling is efficient to approximate the density: we review details on sampling in bounded VC dimension in the supplementary material.

Lemma 2 (Li et al. (2001))**.** *Given a range space $(X, \mathcal{R})$ with VC-dimension d, an constants $\varepsilon, \delta, \eta$, and a uniform sample $S \subset X$ of size at least $O(\frac{1}{\eta \cdot \varepsilon^{-2}}(d \log 1/\eta + \log 1/\delta))$, we have for all ranges $R \in \mathcal{R}$ with $|X \cap R| \geq \eta \cdot |X|$*

$$\left| \frac{|X \cap R|}{|X|} - \frac{|S \cap R|}{|S|} \right| \leq \varepsilon \cdot \frac{|X \cap R|}{|X|}$$

and for all ranges $R \in \mathcal{R}$ with $|X \cap R| \leq \eta \cdot |X|$

$$\left| \frac{|X \cap R|}{|X|} - \frac{|S \cap R|}{|S|} \right| \leq \varepsilon \cdot \eta$$

with probability at least $1 - \delta$.

The only range space we will consider is the one induced by Euclidean spheres centered around a single fixed point. The VC dimension induced of this range space is 2, which seems to be a well known fact, although we could not find a reference. For completeness, we added a short proof in the supplementary material.

First, we recall the commonly used coreset definition for clustering problems in Euclidean spaces.

Definition 1. *Let A be a set of points in $\mathbb{R}^d$. Then a set Ω is a strong (ε, z)-coreset if there exists a weight function $w : \Omega \to \mathbb{R}^+$ and a constant κ such that for any point c*

$$\left| \sum_{p \in A} \|p - c\|^z - (\sum_{p \in S} w_p \|p - c\|^z + \kappa) \right| \leq \varepsilon \cdot \sum_{p \in A'} \|p - c\|^z.$$

We say that Ω is a weak (ε, z)-coreset if for some $\alpha \in [0, 1]$ any point satisfying $\sum_{p \in \Omega} w_p \cdot \|p - c'\|^z \leq (1 + \alpha \cdot \varepsilon) \underset{c \in \mathbb{R}^d}{\text{argmin}} \sum_{p \in \Omega} w_p \cdot \|p - c\|^z$ also satisfies $\sum_{p \in A} \|p - c'\|^z \leq (1 + \varepsilon) \underset{c \in \mathbb{R}^d}{\text{argmin}} \sum_{p \in A} \|p - c\|^z$.

The difference between the two notions is that strong coresets give a guarantee for all candidate centers, whereas the weak coreset guarantee only applies for the optimum. In an offline setting, there is no difference in the size for our construction for either guarantee. In the sublinear setting, we will be satisfied with a weak coreset as it can be obtained with a nearly optimal query complexity.

2 Uniform Sampling Routine

In this section, we outline the proof of our basic sampling subroutine. We assume that we are given a point q, and a set of points R such that for all $p, p' \in R$

$$\|p - q\|^z \leq 2 \cdot \|p' - q\|^z.$$

In the following sections, we refer to R as a ring. Under this assumption, the following claim holds.

Theorem 4. *Let q and R be defined as above and let Ω be a uniform random sample consisting of $s \in \tilde{O}(\varepsilon^{-2} \cdot \log 1/\delta)$ points. Then with probability at least $1-\delta$, we have for all candidate centers c*

$$\left| \mathit{cost}(R,c) - \sum_{p \in \Omega} \frac{|R|}{s} \cdot \|p-c\|^z \right| \quad \leq \quad \varepsilon \cdot (\mathit{cost}(R,q) + \mathit{cost}(R,c)).$$

We prove this theorem, first by proving the following slightly simpler result and then showing how to apply the result recursively to remove dependency in $\|R\|_0$:

Lemma 3. *Let q and A be defined as above and let Ω be a uniform random sample consisting of $s \in \tilde{O}(\varepsilon^{-2} \cdot \log \|R\|_0 \cdot \log 1/\delta)$ points. Then with probability at least $1-\delta$, we have for all candidate centers c that $\left| \mathit{cost}(R,c) - \sum_{p \in \Omega} \frac{|R|}{s} \cdot \|p-c\|^z \right| \leq \varepsilon \cdot (\mathit{cost}(R,q) + \mathit{cost}(R,c))$.*

To set up a chaining analysis, we require two ingredients: (1) a notion of nets and (2) a Gaussian process. We focus on the latter first. For any point c, let v^c be the $|R|$-dimensional vector, henceforth called a *cost vector*, such that $v_{p_i} = \|p_i - c\|^z$ for some arbitrary but fixed ordering of the points in R. Note that $\|v\|_1 = \text{cost}(R,c)$. Let $p_j \in \Omega$ with $j \in \{1, \ldots, s\}$ be the jth point of the sample. Observe for any cost vector v induced by some center c, we have

$$\mathbb{E}_\Omega \left[\frac{n}{s} v_{p_j} \right] = \sum_{p \in R} \|p-c\|^z = \text{cost}(R,c).$$

We now symmetrize the expectation. Let $g_1, \ldots g_s$ be standard Gaussian random variables, i.e. Gaussians with mean 0 and variance 1. Then we have for any collection of cost vectors $\mathbb{N}$ (see Appendix B.3 of Rudra & Wootters (2014))

$$\mathbb{E}_\Omega \sup_{v \in \mathbb{N}} \left| \frac{\sum_{j=1}^s \frac{|R|}{s} v_{p_j} - \|v\|_1}{\text{cost}(R,q) + \|v\|_1} \right| \quad \leq \quad \sqrt{2\pi} \mathbb{E}_\Omega \mathbb{E}_g \sup_{v \in \mathbb{N}} \left| \frac{\sum_{j=1}^s \frac{|R|}{s} v_{p_j} \cdot g_j}{\text{cost}(R,q) + \|v\|_1} \right| \tag{1}$$

Note that the first term in Equation 1 is the expected maximum deviation from the (normalized) expected cost of the sample, if the cost vectors are induced by centers. In other words, if $\sup_{c \in \mathbb{R}^d} \left| \frac{\sum_{j=1}^s \frac{|R|}{s} \|p_j - c\|^z - \text{cost}(R,c)}{\text{cost}(R,q) + \text{cost}(R,c)} \right| \leq \varepsilon$, we have the desired coreset guarantee. Our cost vectors will not be induced by centers for technical reasons, but are in a well-defined sense close enough such that it will be enough to bound the deviation for these approximate cost vectors to obtain a bound for all center induced cost vectors. Introducing Gaussians is standard in this type of analysis as concentration bounds typically used for weighted Bernoulli random variables such as Hoeffding or Bernstein are too weak. Our goal is therefore to show that $\mathbb{E}_\Omega \mathbb{E}_g \sup_{c \in \mathbb{R}^d} \left| \frac{\sum_{j=1}^s \frac{|R|}{s} \|p_j - c\|^z \cdot g_j}{\text{cost}(R,q) + \text{cost}(R,c)} \right| \leq \frac{\varepsilon}{\sqrt{2\pi}}$.

We now define the nets we will use.

Definition 2. *Let R be a set of points and let q be a candidate solution. For $\beta > 0$, we say that a set of vectors $\mathbb{N}_\beta$ is a β-net of R, if there exists a vector $v' \in \mathbb{N}_\beta$ such that for every point $p \in R$ with $\|p-c\| \leq \frac{8z}{\varepsilon} \cdot \|p-q\|$, we have*

$$\left| \|p-c\|^z - v'_p \right| \leq \beta \cdot \left(\|p-c\|^z + \|p-q\|^z \right).$$

We need to show the following three properties:

1. By bounding $\mathbb{E}_\Omega \sup_{v \in \mathbb{N}_{\varepsilon/10}} \left| \frac{\sum_{j=1}^s \frac{|R|}{s} v_{p_j} - \sum_{p \in R_i} v_p}{\text{cost}(R,q) + \text{cost}(R,c)} \right|$, we can also bound $\mathbb{E}_\Omega \sup_{c \in \mathbb{R}^d} \left| \frac{\sum_{j=1}^s \frac{|R|}{s} \|p_j - c\|^z - \text{cost}(R,c)}{\text{cost}(R,q) + \text{cost}(R,c)} \right|$

2. There exist β-nets of size $\exp\left(z^2 \log \|R\|_0 \cdot \beta^{-2} \cdot \log \frac{1}{\varepsilon \cdot \beta} \right)$.

3. Let v be a cost vector written as a telescoping sum $v = \sum_{i=1} \infty v_i - v_{i-1}$ of cost vectors from 2^i-nets. Let v', v'' be cost vectors from successive β and $\beta/2$ nets $\mathbb{N}_\beta$ and $\mathbb{N}_{\beta/2}$, respectively. By bounding the term $\mathbb{E}_\Omega \mathbb{E}_g \sup_{v'-v''} \left| \sum_{j=1}^{s} \frac{|R|}{s} |v'_{p_j} - v''_{p_j}| \cdot g_j \right|$, we can also bound $\mathbb{E}_\Omega \mathbb{E}_g \sup_{v \in \mathbb{N}_\varepsilon} \left| \sum_{j=1}^{s} \frac{|R|}{s} v_{p_j} \cdot g_j \right|$

We start with item 1 via the following lemma.

Lemma 4. *Let $\mathbb{N}_{\varepsilon/10}$ be an $\varepsilon/10$-net of R. Then if $\sup_{v \in \mathbb{N}_{\varepsilon/10}} \frac{\left|\sum_{j=1}^{s} \frac{|R|}{s} v_{p_j} - \|v\|_1\right|}{cost(R,q)+cost(R,c)} \leq \frac{\varepsilon}{10}$ we have $\sup_{c \in \mathbb{R}^d} \frac{\left|\sum_{j=1}^{s} \frac{|R|}{s} \|p_j - c\|^z - cost(R,c)\right|}{cost(R,q)+cost(R,c)} \leq \varepsilon$ for all $c \in \mathbb{R}^d$.*

To prove item 2, we use terminal embeddings Elkin et al. (2017); Mahabadi et al. (2018); Narayanan & Nelson (2019), defined as follows.

Definition 3 (Terminal Embeddings)**.** *Let A be a set of points in $\mathbb{R}^d$. A mapping $f : \mathbb{R}^d \to R^k$ is a terminal embedding if for all $p \in A$ and all $c \in \mathbb{R}^d$*

$$(1-\varepsilon) \cdot \|p-c\|_2 \leq \|f(p) - f(c)\|_2 \leq (1+\varepsilon) \cdot \|p-c\|_2.$$

The guarantee of a terminal embedding is very similar to the guarantee of the famous Johnson Lindenstrauss lemma, but stronger in one crucial detail. A terminal embedding preserves not only the distances between the points of A but also the distance between an arbitrary point in $\mathbb{R}^d$ and any point of A. Despite this stronger guarantee, the target dimension of terminal embedding is in fact no worse than that of the Johnson Lindenstrauss lemma. Specifically:

Theorem 5 (Narayanan & Nelson (2019))**.** *For any n point-set $A \subset \mathbb{R}^d$, there exists a terminal embedding $f : \mathbb{R}^d \to \mathbb{R}^k$ with $k \in \gamma \cdot \varepsilon^{-2} \log n$ for some absolute constant γ.*

We now use the terminal embeddings to show that small nets exist.

Lemma 5. *Let R and q be defined as above. Then for every $\beta > 0$, there exists a β-net of R of size at most $\exp(\gamma \cdot z^3 \beta^2 \log \|R\|_0 \cdot \log \varepsilon^{-1})$, where γ is an absolute constant.*

We now move onto item 3.

Lemma 6. *Let R and q be defined as above and let Ω be a uniform sample consisting of s points. Then for any point c and $s \geq \eta \cdot z^3 2^{8z} \cdot \varepsilon^{-2} \cdot \log \|R\|_0 \cdot \log^3 \varepsilon^{-1}$ for some absolute constant η, we have $\mathbb{E}_\Omega \mathbb{E}_g \sup_{c \in \mathbb{R}^d} \left| \frac{\sum_{j=1}^{s} \frac{|R|}{s} \|p_j - c\|^z \cdot g_j}{cost(R,q)+cost(R,c)} \right| \leq \varepsilon$. Moreover, if $s \geq \eta \cdot z^3 2^{8z} \cdot \varepsilon^{-2} \cdot \log \|R\|_0 \cdot \log^4 \varepsilon^{-1} \log 1/\delta$ for some absolute constant η, then we have $\sup_{c \in \mathbb{R}^d} \left| \frac{\sum_{j=1}^{s} \frac{|R|}{s} \|p_j - c\|^z \cdot g_j}{cost(R,q)+cost(R,c)} \right| \leq \varepsilon$, with probability at least $1 - \delta$.*

The proof of Lemma 3 is now a direct consquence of Lemma 6 and Equation 1. To finally prove Theorem 4, we apply Lemma 6 recursively. The result is essentially a special case of Theorem 3.1 from Braverman et al. (2021).

Lemma 7. *Let R and q be defined as above. Suppose a uniform sample of size $s \in \tilde{O}(\Gamma \cdot \log \|R\|_0 \cdot \log 1/\delta)$ satisfies with probability at least $1 - \delta$ for all candidate centers c*

$$|cost(R,c) - \sum_{p \in \Omega} \frac{|R|}{s} \cdot \|p-c\|^z| \quad \leq \quad \varepsilon \cdot (cost(R,q) + cost(R,c)).$$

Then a uniform sample of size $\tilde{O}(\Gamma \cdot \log 1/\delta)$ achieves the same guarantee.

3 Sublinear Algorithm and Analysis

We first describe an algorithm that succeeds with constant probability. This probability can be amplified (non-trivially) using independent repetition. In the following we will use parameters β and η that depends on ε which we will specify later. We let A be the set of input points.

Algorithm 1 Sublinear Algorithm for Power Means

1. Sample a random point q.
2. Sample a set S of $O(\varepsilon^{-z-3} \cdot \text{poly}(\varepsilon^{-1}))$ points uniformly at random.
3. Compute the maximum distance d such that there exist $2/3 \cdot \varepsilon \cdot \eta \cdot |S|$ points with distance at least d from q. Discard all points at distance greater than d.
4. Define rings R_i such that $R_i \cap S$ contains all the points at distance $(d \cdot 2^{-i}, d \cdot 2^{-i+1}]$ from q, with $i = \{1, \ldots, \beta\}$.
5. If $|R_i \cap S| < \varepsilon\eta \cdot |S|$, remove all points in $R_i \cap S$ from S.
6. Define $\hat{R}_i = n \cdot \frac{|R_i \cap S|}{|S|}$. Weigh the points $R_i \cap S$ by $\frac{\hat{R}_i}{|R_i \cap S|}$.
7. Solve the problem on the (weighted) set S.

Our goal is to show that S satisfies a weak coreset guarantee. Specifically, we will show that S has the property that for all points $A' \subseteq A$ at distance at most d (defined in Algorithm 1) from q there exists weight function $w : S \to \mathbb{R}^+$ such that S is a strong coreset for A'. We then show that a strong coreset for A' is also weak coreset for A, i.e. we preserve the cost of all points in A', and we preserve the optimum for A.

In order to call Theorem 4 on only few rings, we now show that the loss incurred by only considering A' is indeed negligible.

Pruning Lemmas We first show that we can safely discard points that are sufficiently far away, as parameterized by γ. We recall the meaning of parameters: α is the approximation-factor of the initial solution, β is such that points at distance closer than $2^\beta d$ can be merged to the center (see Lemma 9), η is such that rings with less than an $\varepsilon\eta$-fraction of the points can be discarded (see Lemma 10)

Lemma 8. *Suppose we are given an α-approximate center q. Let $B(q, r)$ be the ball centered at q with radius $r = 4 \cdot \left(\frac{2\alpha \cdot OPT}{n}\right)^{1/z}$. Then the following two statements hold.*

1. *Any α-approximate center is in $B(q, r)$.*
2. *For any two points $c, c' \in B(q, r)$ and for any point p with $\|p - q\|^z > \gamma \cdot r^z$ with $\gamma > \varepsilon^{-z} \cdot (12z)^z$, we have $\|p - c\|^z \leq (1 + \varepsilon) \cdot \|p - c'\|^z$.*

Unfortunately, we are not given a knowledge of OPT a priori. We therefore have to describe how to implement this lemma in a sublinear fashion. For this, we observe the following. First, the point q is an $2^z\alpha$ approximation with probability at least $1 - 1/\alpha$. Second, the number of points that cost more than $\gamma \cdot 2^z\alpha \cdot \frac{OPT}{n}$ is at most $1/\gamma \cdot n$. Combining this observation with Lemma 2 ensures that we are not considering any points that are too far away. Since it is difficult to determine $\frac{OPT}{n}$ in a sublinear fashion, we will rely on additional pruning arguments for points that are close to q.

We proceed here in two steps. First, we consider the points that are very close to q. Second, we will show that the rings which contain too few points to be efficiently sampled have an overall negligible contribution to the cost.

Lemma 9. *Suppose that q is an α-approximate solution. Let $A_{near} \subset A$ be a set of points with cost at most $(\varepsilon/(\alpha 5z))^z \cdot \frac{OPT}{n}$. Let $\hat{A} = (1 \pm \varepsilon)|A_{near}|$ Then for any candidate solution c we have*

$$\left| \hat{A} \cdot \|q - c\|^z - \sum_{p \in A_{near}} \|p - c\|^z \right| \leq \varepsilon/\alpha \cdot \left(\sum_{p \in A_{near}} \|p - c\|^z + OPT \right).$$

For rings with few points, we have the following.

Lemma 10. *Suppose that q that is an α-approximate solution. Let R_{cheap} the union of rings with $|R_i \cap A| < \varepsilon \cdot \eta \cdot n$ and with radius at most $4(\gamma \cdot \frac{\alpha \cdot OPT}{n})^{1/z}$, where γ is given by Lemma 8. Then, for any candidate solution c*

$$\sum_{p \in R_{cheap} \cap A} \|p - c\|^z \leq \varepsilon \cdot 4^{z+1} \cdot \beta \cdot \eta \cdot \gamma \cdot \alpha \cdot \sum_{p \in A} \|p - c\|^z.$$

While we will defer the exact parameterization to later, observe that for $\beta \in O(\log 1/\varepsilon)$, $\alpha \in 2^{O(z)}$ and $\eta \in O(\varepsilon^z)$ this entire sum can be bounded by $O(\varepsilon) \cdot OPT$.

The next lemma states that we can use these pruning results in a sublinear fashion.

Lemma 11. *Let q be a point that is an α-approximation and let S be a uniform sample consisting of $O\left(\alpha \cdot \eta^{-1} \cdot \varepsilon^{-3} polylog(\varepsilon^{-1} \cdot \delta^{-1})\right)$ points. Then with probability at least $1-\delta$ for all rings R_i,*

$$|R_i \cap A| - \varepsilon \cdot \max\left(n \cdot \eta, |R_i \cap A|\right) \leq \frac{|R_i \cap S| \cdot n}{|S|} \leq |R_i \cap A| + \varepsilon \cdot \max\left(n \cdot \eta,\ |R_i \cap A|\right).$$

Furthermore, let d as in the algorithm, i.e., such that $\frac{2}{3} \cdot \varepsilon \cdot \eta \cdot |S| \leq \left|S \setminus (B(q,d) \cap S)\right|$. Then $d < \left(3(\varepsilon\eta)^{-1} \cdot \frac{\alpha \cdot OPT}{n}\right)^{1/z}$

Since we do not know r, we also do not know which of the rings with $i < 0$, if any, satisfy $2^{i+1} \cdot d > (\varepsilon/\alpha 5z) \cdot (\eta/3\alpha)^{1/z}$. However, the maximum number of these rings is at most $\log(\frac{5\alpha z}{\varepsilon} \cdot \left(\frac{3\alpha}{\eta}\right)^{1/z})$, which will turn out to be of the order $O(\log \varepsilon^{-1})$. For all of the rings that are not light, i.e. we cannot discard or snap to q, we now use Theorem 4. To ensure that we can call Theorem 4, we invoke Lemma 2 for every ring.

Probability Amplification While the aforementioned algorithm is guaranteed to produce a $(1+\varepsilon)$-approximation with constant probability, amplifying this is non-trivial. Indeed, when running the algorithm multiple times, it is not clear how to distinguish a successful run from an unsuccessful one. The main issue in amplifying the probability lies in the initial solution q, as any invocation of Lemma 2 or Theorem 4 allows us to control the failure probability. The simplest way to achieve a success probability $1-\delta$ is to condition on $\|q-c\|^z \leq \delta \cdot \frac{OPT}{n}$. Unfortunately, this makes ε dependent on δ, which significantly increases the sampling complexity.

Instead, we use the following algorithm. We sample $m \in O(\log 1/\delta)$ points $q_1, \ldots q_m$ uniformly at random. For each point, we additionally sample $O(\varepsilon^{-2}(\log 1/\varepsilon + \log 1/\delta))$ points S_{q_i}. For q_i, let d_i denote the minimum radius such that the points in $\frac{n}{|S_{q_i}|} \cdot |B(q_i, d_i) \cap S_{q_i}| > \frac{n}{2}$. We output the point with minimal d_i.

The following lemma shows that this point is, with probability at least $1-\delta$, a 8^z approximation.

Lemma 12. *Given query access to A, we can identify with probability $1-\delta$ a 8^z-approximate solution using $O(\varepsilon^{-2}(\log 1/\varepsilon + \log 1/\delta)\log 1/\delta)$ samples.*

To achieve an overall success probability of $1-\delta$, we only need to sample from non-cheap rings. Thereafter, a high probability bound can be obtained by applying Theorem 4 applied to all R_i. The range space induced by rings centered around a single point q has constant VC dimension. Hence, Lemma 2 guarantees that a constant size sample will allow us to distinguish cheap rings from non-cheap ones.

4 Improved Coreset Constructions

Our algorithm is as follows. First, we compute a point q that is a reasonably good approximation to the optimum[4]. In the following, let $\alpha = \frac{\text{cost}(q)}{\text{OPT}}$. Let $\Delta = \frac{\text{cost}(q)}{n}$ be the average cost of the input points when clustering them to q. We now partition the points into rings, defined as follow: $R_i = \{p \in A \mid \left(\frac{\varepsilon}{2z}\right)^z \cdot \frac{1}{\alpha} \cdot 2^i \cdot \Delta \leq \|p-q\|^z \leq \left(\frac{\varepsilon}{2z}\right)^z \cdot \frac{1}{\alpha} \cdot 2^{i+1} \cdot \Delta\}$. Let $R_M = \bigcup_{i=1}^{\log\left(\frac{2z}{\varepsilon}\right)^{3z}} R_i$. For each ring R_i with $1 \leq i \leq \log\left(\frac{2z}{\varepsilon}\right)^{3z}$, we sample a subset S_i of s points uniformly at random, where each point is weighted by $\frac{|R_i|}{s}$. We weigh q by the number of points in $A \setminus R_M$. Finally, we set $\kappa = \sum_{i > \log\left(\frac{2z}{\varepsilon}\right)^{3z}} \sum_{p \in R_i} \|p-q\|^z$. We claim, for $s \in \tilde{O}(\varepsilon^{-2})$ that for the weights defined above, $\bigcup_{i=1}^{\log\left(\frac{2z}{\varepsilon}\right)^{3z}} S_i \cup \{q\}$ together with the constant κ is a coreset.

The analysis for every ring $R_i \in R_M$ is merely an application of Theorem 4. For the remaining points, we use the following lemma.

[4]We described an option in detail for the sublinear algorithm. In the interest of keeping the presentation succinct, we defer to that part of the paper and omit further discussion.

Lemma 13. *Let q, R_M and κ be defined as above. Then for any point $c \in \mathbb{R}^d$, we have*

$$|cost(A, c) - (cost(R_M, c) + \kappa + |A \setminus R_M| \cdot \|q - c\|^2)| \leq \varepsilon \cdot cost(A, c).$$

5 Experimental Evaluation

While we can prove that the algorithm can compute a good solution in constant time for every constant ε and z, even for moderately small ε (e.g. $\varepsilon = 1/2$) the sampling complexity becomes quite large for even small values of z, as indeed our lower bound shows is necessarily the case. Our experiments therefore aim at evaluating the performance of the sublinear algorithm on realistic, not necessarily worst case data sets.

As baseline algorithm, we implemented a simple version of a batched gradient descent. Since all considered objectives are convex, we can expect such an algorithm to find a good solution in a reasonable time. The sublinear algorithm ran Algorithm 1 before calling the batched gradient descent. The code can be found at `https://github.com/DaSau/power-mean` We selected two data sets from the UCI repository Dua & Graff (2017), both of which are under the Creative Commons license. The first data set is the 3D Road Network data set from Kaul et al. (2013). It consists of elevation information with the attributes longitude, latitude and altitude. The total number of points is 434,874. For this data set, we considered all powers from $z = 3$ to 7. The second data set is the USCensus data set, consisting of the records from a 1990 census. The total size of the data set was 2,458,285 samples, each with 68 attributes. For this data set, we considered the powers $z = 3, 4, 5$.

The results essentially confirmed that the sublinear algorithm succeeded in finding a good candidate solution in a fraction of the time as batch gradient descent for essentially all considered problems.

A more extensive discussion can be found in the supplementary material.

6 Conclusion and Future Work

We gave sublinear algorithms and coresets for any power of means. Our bounds are nearly tight for the sublinear algorithms and we conjecture the coreset bound to be optimal, up to polylog factors.

The most immediate open question is whether our results generalize to coresets for k-clustering objectives. It seems likely that coresets of size $O(k^2/\varepsilon^2)$ are achievable using our techniques. Improving on either this bound or the $O(k\varepsilon^{-2-z})$ from Cohen-Addad et al. (2021) is arguable the most important open problem in coresets.

In terms of sublinear algorithms, there is still a sub-optimal dependency on the ε by a factor $\varepsilon^{-O(1)}$. Obtaining tight bounds would be interesting. Finally, it is also an interesting open question whether sublinear algorithms for ℓ_p with $p > 2$ exist. It is known that for these spaces, no coreset that is independent of d can exist, even for the mean or the median. Is it nevertheless possible to obtain a sublinear algorithm that is independent of d?

Acknowledgments and Disclosure of Funding

The work of David Saulpic is [partially] funded by the grant ANR-19-CE48-0016 from the French National Research Agency (ANR).

References

Bachem, O., Lucic, M., Hassani, S. H., and Krause, A. Approximate k-means++ in sublinear time. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2016.

Bachem, O., Lucic, M., and Krause, A. Scalable k -means clustering via lightweight coresets. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018*, pp. 1119–1127, 2018. doi: 10.1145/3219819.3219973. URL https://doi.org/10.1145/3219819.3219973.

Badoiu, M. and Clarkson, K. L. Optimal core-sets for balls. *Comput. Geom.*, 40(1):14–22, 2008. doi: 10.1016/j.comgeo.2007.04.002. URL http://dx.doi.org/10.1016/j.comgeo.2007.04.002.

Becchetti, L., Bury, M., Cohen-Addad, V., Grandoni, F., and Schwiegelshohn, C. Oblivious dimension reduction for *k*-means: beyond subspaces and the johnson-lindenstrauss lemma. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC*, pp. 1039–1050, 2019. URL https://doi.org/10.1145/3313276.3316318.

Ben-David, S. A framework for statistical clustering with constant time approximation algorithms for K-median and K-means clustering. *Mach. Learn.*, 66(2-3):243–257, 2007. doi: 10.1007/s10994-006-0587-3. URL https://doi.org/10.1007/s10994-006-0587-3.

Blumer, A., Ehrenfeucht, A., Haussler, D., and Warmuth, M. K. Learnability and the vapnik-chervonenkis dimension. *J. ACM*, 36(4):929–965, 1989. URL https://doi.org/10.1145/76359.76371.

Braverman, V., Jiang, S. H., Krauthgamer, R., and Wu, X. Coresets for clustering in excluded-minor graphs and beyond. In Marx, D. (ed.), *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms, SODA 2021, Virtual Conference, January 10 - 13, 2021*, pp. 2679–2696. SIAM, 2021. doi: 10.1137/1.9781611976465.159. URL https://doi.org/10.1137/1.9781611976465.159.

Ceccarello, M., Pietracaprina, A., and Pucci, G. Fast coreset-based diversity maximization under matroid constraints. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pp. 81–89, 2018.

Chen, K. On coresets for k-median and k-means clustering in metric and Euclidean spaces and their applications. *SIAM J. Comput.*, 39(3):923–947, 2009.

Clarkson, K. L., Hazan, E., and Woodruff, D. P. Sublinear optimization for machine learning. *J. ACM*, 59(5):23:1–23:49, 2012. doi: 10.1145/2371656.2371658. URL https://doi.org/10.1145/2371656.2371658.

Cohen, M. B., Lee, Y. T., Miller, G. L., Pachocki, J., and Sidford, A. Geometric median in nearly linear time. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC*, 2016.

Cohen-Addad, V. and Li, J. On the fixed-parameter tractability of capacitated clustering. In *46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece*, pp. 41:1–41:14, 2019. doi: 10.4230/LIPIcs.ICALP.2019.41. URL https://doi.org/10.4230/LIPIcs.ICALP.2019.41.

Cohen-Addad, V. and Schwiegelshohn, C. On the local structure of stable clustering instances. In *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2017, Berkeley, CA, USA, October 15-17, 2017*, pp. 49–60, 2017. doi: 10.1109/FOCS.2017.14. URL https://doi.org/10.1109/FOCS.2017.14.

Cohen-Addad, V., Saulpic, D., and Schwiegelshohn, C. A new coreset framework for clustering. In Khuller, S. and Williams, V. V. (eds.), *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021*. ACM, 2021.

Czumaj, A. and Sohler, C. Sublinear-time approximation algorithms for clustering via random sampling. *Random Structures & Algorithms*, 30(1-2):226–256, 2007.

Ding, H. A sub-linear time framework for geometric optimization with outliers in high dimensions. In *28th Annual European Symposium on Algorithms, ESA 2020, September 7-9, 2020, Pisa, Italy (Virtual Conference)*, pp. 38:1–38:21, 2020. doi: 10.4230/LIPIcs.ESA.2020.38. URL `https://doi.org/10.4230/LIPIcs.ESA.2020.38`.

Ding, H. Stability yields sublinear time algorithms for geometric optimization in machine learning. In Mutzel, P., Pagh, R., and Herman, G. (eds.), *29th Annual European Symposium on Algorithms, ESA 2021, September 6-8, 2021, Lisbon, Portugal (Virtual Conference)*, volume 204 of *LIPIcs*, pp. 38:1–38:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

Dua, D. and Graff, C. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`.

Elkin, M., Filtser, A., and Neiman, O. Terminal embeddings. *Theor. Comput. Sci.*, 697:1–36, 2017. doi: 10.1016/j.tcs.2017.06.021. URL `https://doi.org/10.1016/j.tcs.2017.06.021`.

Feldman, D. Core-sets: An updated survey. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, 10 (1), 2020. doi: 10.1002/widm.1335. URL `https://doi.org/10.1002/widm.1335`.

Feldman, D. and Langberg, M. A unified framework for approximating and clustering data. In *Proceedings of the 43rd ACM Symposium on Theory of Computing, STOC 2011, San Jose, CA, USA, 6-8 June 2011*, pp. 569–578, 2011.

Feldman, D., Schmidt, M., and Sohler, C. Turning big data into tiny data: Constant-size coresets for k-means, pca, and projective clustering. *SIAM J. Comput.*, 49(3):601–657, 2020. doi: 10.1137/18M1209854. URL `https://doi.org/10.1137/18M1209854`.

Feng, Z., Kacham, P., and Woodruff, D. P. Dimensionality reduction for the sum-of-distances metric. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 3220–3229. PMLR, 2021. URL `http://proceedings.mlr.press/v139/feng21a.html`.

Fichtenberger, H., Gillé, M., Schmidt, M., Schwiegelshohn, C., and Sohler, C. BICO: BIRCH meets coresets for k-means clustering. In *Algorithms - ESA 2013 - 21st Annual European Symposium, Sophia Antipolis, France, September 2-4, 2013. Proceedings*, pp. 481–492, 2013.

Har-Peled, S. and Kushal, A. Smaller coresets for k-median and k-means clustering. *Discrete & Computational Geometry*, 37(1):3–19, 2007.

Har-Peled, S. and Mazumdar, S. On coresets for k-means and k-median clustering. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing, Chicago, IL, USA, June 13-16, 2004*, pp. 291–300, 2004.

Huang, L. and Vishnoi, N. K. Coresets for clustering in euclidean spaces: importance sampling is nearly optimal. In *Proccedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020*, 2020. doi: 10.1145/3357713.3384296. URL `https://doi.org/10.1145/3357713.3384296`.

Huang, L., Jiang, S. H., and Vishnoi, N. K. Coresets for clustering with fairness constraints. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pp. 7587–7598, 2019.

Huggins, J., Campbell, T., and Broderick, T. Coresets for scalable bayesian logistic regression. In *Advances in Neural Information Processing Systems*, pp. 4080–4088, 2016.

Inaba, M., Katoh, N., and Imai, H. Applications of weighted voronoi diagrams and randomization to variance-based k-clustering (extended abstract). In *Proceedings of the Tenth Annual Symposium on Computational Geometry, Stony Brook, New York, USA, June 6-8, 1994*, pp. 332–339, 1994. doi: 10.1145/177424.178042. URL https://doi.org/10.1145/177424.178042.

Indyk, P., Mahabadi, S., Mahdian, M., and Mirrokni, V. S. Composable core-sets for diversity and coverage maximization. In Hull, R. and Grohe, M. (eds.), *Proceedings of the 33rd ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS'14, Snowbird, UT, USA, June 22-27, 2014*, pp. 100–108. ACM, 2014. doi: 10.1145/2594538.2594560. URL https://doi.org/10.1145/2594538.2594560.

Indyk, P., Mahabadi, S., Gharan, S. O., and Rezaei, A. Composable core-sets for determinant maximization problems via spectral spanners. In Chawla, S. (ed.), *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pp. 1675–1694. SIAM, 2020. doi: 10.1137/1.9781611975994.103. URL https://doi.org/10.1137/1.9781611975994.103.

Kaul, M., Yang, B., and Jensen, C. S. Building accurate 3d spatial networks to enable next generation intelligent transportation systems. In *2013 IEEE 14th International Conference on Mobile Data Management, Milan, Italy, June 3-6, 2013 - Volume 1*, pp. 137–146, 2013. doi: 10.1109/MDM.2013.24. URL https://doi.org/10.1109/MDM.2013.24.

Langberg, M. and Schulman, L. J. Universal ε-approximators for integrals. In *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2010, Austin, Texas, USA, January 17-19, 2010*, pp. 598–607, 2010.

Lee, J. C. H. and Valiant, P. Optimal sub-gaussian mean estimation in R. *To appear at FOCS'21*, 2021.

Li, Y., Long, P. M., and Srinivasan, A. Improved bounds on the sample complexity of learning. *J. Comput. Syst. Sci.*, 62(3):516–527, 2001. doi: 10.1006/jcss.2000.1741. URL https://doi.org/10.1006/jcss.2000.1741.

Lugosi, G. and Mendelson, S. Mean estimation and regression under heavy-tailed distributions: A survey. *Found. Comput. Math.*, 19(5):1145–1190, 2019. doi: 10.1007/s10208-019-09427-x. URL https://doi.org/10.1007/s10208-019-09427-x.

Maalouf, A., Jubran, I., and Feldman, D. Fast and accurate least-mean-squares solvers. In *Advances in Neural Information Processing Systems*, pp. 8307–8318, 2019.

Mahabadi, S., Makarychev, K., Makarychev, Y., and Razenshteyn, I. P. Nonlinear dimension reduction via outer bi-lipschitz extensions. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*, pp. 1088–1101, 2018.

Makarychev, K., Makarychev, Y., and Razenshteyn, I. P. Performance of johnson-lindenstrauss transform for k-means and k-medians clustering. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC*, pp. 1027–1038, 2019.

Meyerson, A., O'callaghan, L., and Plotkin, S. A k-median algorithm with running time independent of data size. *Machine Learning*, 56(1):61–87, 2004.

Molina, A., Munteanu, A., and Kersting, K. Core dependency networks. In McIlraith, S. A. and Weinberger, K. Q. (eds.), *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pp. 3820–3827. AAAI Press, 2018. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16847.

Munteanu, A. and Schwiegelshohn, C. Coresets-methods and history: A theoreticians design pattern for approximation and streaming algorithms. *Künstliche Intell.*, 32(1):37–53, 2018.

Munteanu, A., Schwiegelshohn, C., Sohler, C., and Woodruff, D. P. On coresets for logistic regression. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 6562–6571, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/63bfd6e8f26d1d3537f4c5038264ef36-Abstract.html.

Narayanan, S. and Nelson, J. Optimal terminal dimensionality reduction in euclidean space. In Charikar, M. and Cohen, E. (eds.), *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019*, pp. 1064–1069. ACM, 2019. doi: 10.1145/3313276.3316307. URL https://doi.org/10.1145/3313276.3316307.

Nelson, J. Chaining introduction with some computer science applications. *Bull. EATCS*, 120, 2016. URL http://eatcs.org/beatcs/index.php/beatcs/article/view/450.

Rudra, A. and Wootters, M. Every list-decodable code for high noise has abundant near-optimal rate puncturings. In Shmoys, D. B. (ed.), *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pp. 764–773. ACM, 2014. doi: 10.1145/2591796.2591797. URL https://doi.org/10.1145/2591796.2591797.

Schmidt, M., Schwiegelshohn, C., and Sohler, C. Fair coresets and streaming algorithms for fair k-means. In *Approximation and Online Algorithms - 17th International Workshop, WAOA 2019, Munich, Germany, September 12-13, 2019, Revised Selected Papers*, pp. 232–251, 2019. doi: 10.1007/978-3-030-39479-0_16. URL https://doi.org/10.1007/978-3-030-39479-0_16.

Sohler, C. and Woodruff, D. P. Strong coresets for k-median and subspace approximation: Goodbye dimension. In *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pp. 802–813, 2018. doi: 10.1109/FOCS.2018.00081. URL https://doi.org/10.1109/FOCS.2018.00081.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [No] The potential societal impacts are the ones of any clustering algorithm; since this would be a paper by itself, we did not discussed it.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [No]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Additional Definitions and Properties

Definition 4. *Let X be a ground set, and $\mathcal{R} \subset \mathcal{P}(X)$. We say that $(X, \mathcal{R})$ is a range space.*

The VC-dimension of a range space $(X, \mathcal{R})$ is the largest d such that, for some $S \subseteq X$ with $|S| = d$, $|\{R \cap S \mid R \in \mathcal{R}\}| = 2^d$.

Let us consider the range space induced by Euclidean balls centered around a single point p. A range R is induced by a ball of radius r is the set of all points at distance r or less from p, i.e. $R = \{q \in X \mid \|p - q\| \leq r\}$. Without loss of generality, assume that q is the origin. We show that for any set of two points, we cannot generate all possible dichotomies, i.e. for any point set S we

have $|\{R \cap S \mid R \in \mathcal{R}\}| < 4$. If both points have the same distance from p, then it is not possible to define a range that contains one point and not the other. If both points have different distance from p, it is not possible to define a range that contains the furthest point, but not the closest.

B Uniform Sampling Routine

Lemma 4. *Let $\mathbb{N}_{\varepsilon/10}$ be an $\varepsilon/10$-net of R. Then if $\sup_{v \in \mathbb{N}_{\varepsilon/10}} \frac{\left|\sum_{j=1}^{s} \frac{|R|}{s} v_{p_j} - \|v\|_1\right|}{cost(R,q)+cost(R,c)} \leq \frac{\varepsilon}{10}$ we have $\sup_{c \in \mathbb{R}^d} \frac{\left|\sum_{j=1}^{s} \frac{|R|}{s} \|p_j - c\|^z - cost(R,c)\right|}{cost(R,q)+cost(R,c)} \leq \varepsilon$ for all $c \in \mathbb{R}^d$.*

Proof. Let $c \in \mathbb{R}^d$ be an arbitrary point. We first deal with the case where there is some point $p' \in R_i$ with $\|p' - c\|^z \geq \left(\frac{8z}{\varepsilon}\right)^z \cdot \|p' - q\|^z$. Then, for any point $p'' \in R$

$$\begin{aligned}
& \|p'' - c\|^z \\
(Lemma\ 1) \quad \leq \quad & (1+\varepsilon) \cdot \|p' - c\|^z + \left(\frac{2z+\varepsilon}{\varepsilon}\right)^{z-1} \cdot \|p' - p''\|^z \\
\leq \quad & (1+\varepsilon) \cdot \|p' - c\|^z + \left(\frac{2z+\varepsilon}{\varepsilon}\right)^{z-1} \cdot 2^{z+1} \|p' - q\|^z \\
\leq \quad & (1+\varepsilon) \cdot \|p' - c\|^z + \left(\frac{2z+\varepsilon}{\varepsilon}\right)^{z-1} \cdot 2^{z+1} \cdot \left(\frac{\varepsilon}{8z}\right)^z \cdot \|p' - c\|^z \\
\Rightarrow \frac{\|p'' - c\|^z}{\|p' - c\|^z} \quad \leq \quad & (1+\varepsilon)
\end{aligned}$$

Using an analogous calculation, one can also show

$$\frac{\|p' - c\|^z}{\|p'' - c\|^z} \leq (1+\varepsilon)$$

which implies that $(1-\varepsilon) \cdot \|p' - c\|^z \leq \|p'' - c\|^z \leq (1+\varepsilon) \cdot \|p' - c\|^z$, ..e., any point in R costs the same up to a $(1 \pm \varepsilon)$ factor. Since the coreset weights, by construction, sum up to $|R|$, we therefore have

$$|\text{cost}(R, c) - \text{cost}(\Omega, c)| \leq \varepsilon \cdot \text{cost}(R, c) \tag{2}$$

Now, we focus on the case where $\|p - c\|^z \leq \left(\frac{8z}{\varepsilon}\right)^z \cdot \|p - q\|^z$ for all $p \in R$. We will assume

$$\sup_{v \in N_\varepsilon} \frac{\left|\sum_{j=1}^{s} \frac{|R|}{s} v_{p_j} - \|v\|_1\right|}{\text{cost}(R, q) + \text{cost}(R, c)} \leq \varepsilon$$

and rescale ε by a factor of 10 at the end. We know that there exists net vector $v \in N_\varepsilon$ such that

$$|\|p - c\|^z - v_p| \leq \varepsilon \cdot \left(\|p - c\|^z + \|p - q\|^z\right).$$

Summing this over all points in R, we therefore have

$$\begin{aligned}
\left|\sum_{p \in R} \left(\|p - c\|^z - v_p\right)\right| \quad \leq \quad & \sum_{p \in R} |\|p - c\|^z - v_p| \\
\leq \quad & \varepsilon \cdot \sum_{p \in R} \|p - c\|^z + \|p - q\|^z \\
\leq \quad & \varepsilon \cdot (\text{cost}(R, c) + \text{cost}(R, q)).
\end{aligned} \tag{3}$$

We similarly show that $\sum_{j=1}^{s} \frac{|R|}{s} \|p_j - c\|^z$ and v_{p_j} are close. First observe that if $\left|\sum_{p \in R} v_p - \sum_{j=1}^{s} \frac{|R|}{s} v_{p_j}\right| \leq \varepsilon \cdot \sum_{p \in R} v_p$, as assumed in the lemma, then $\sum_{j=1}^{s} \frac{|R|}{s} v_{p_j} \leq$

$2\left|\sum_{p\in R} v_p\right|$. Therefore we have

$$\begin{aligned}
\left|\sum_{j=1}^{s}\frac{|R|}{s}\cdot\left(\|p_j-c\|^z-v_{p_j}\right)\right| &\leq \sum_{j=1}^{s}\frac{|R|}{s}\cdot\left|\|p_j-c\|^z-v_{p_j}\right| \\
&\leq \varepsilon\cdot\sum_{j=1}^{s}\frac{|R|}{s}\left(\|p_j-c\|^z+\|p_j-q\|^z\right) \\
&\leq 2\varepsilon\cdot\sum_{j=1}^{s}\frac{|R|}{s}\left(v_{p_j}+2\cdot\frac{\text{cost}(R,q)}{|R|}\right) \\
&\leq 2\varepsilon\cdot(2\text{cost}(R,q)+\sum_{p\in R}v_p) \\
&\leq 4\varepsilon\cdot(\text{cost}(R,c)+\text{cost}(R,q)). \qquad (4)
\end{aligned}$$

Combining equations 3 and 4, we therefore obtain

$$\begin{aligned}
\left|\sum_{p\in R}\|p-c\|^z-\sum_{j=1}^{s}\frac{|R|}{s}\cdot\|p_j-c\|^z\right| &\leq \left|\sum_{p\in R}\|p-c\|^z-v_p\right| \\
&\quad+\left|\sum_{j=1}^{s}\frac{|R|}{s}\cdot\|p_j-c\|^z-\sum_{j=1}^{s}\frac{|R|}{s}\cdot v_{p_j}\right| \\
&\quad+\left|\sum_{p\in R}v_p-\sum_{j=1}^{s}\frac{|R|}{s}\cdot v_{p_j}\right| \\
&\leq 10\varepsilon\cdot(\text{cost}(R,c)+\text{cost}(R,q)).
\end{aligned}$$

Together with Equation 2 and rescaling ε by a factor of 10 yields the claim. □

Lemma 5. *Let R and q be defined as above. Then for every $\beta>0$, there exists a β-net of R of size at most $\exp(\gamma\cdot z^3\beta^2\log\|R\|_0\cdot\log\varepsilon^{-1})$, where γ is an absolute constant.*

Proof. Let $f:\mathbb{R}^d\to\mathbb{R}^k$ with $k\in O(z^2\beta^{-2}\log R_i)$ be a terminal embedding satisfying for all $c\in\mathbb{R}^d$ and all $p\in R_i$

$$(1-\beta/2z)\cdot\|p-c\|\leq\|f(p)-f(c)\|\leq(1+\beta/2z)\cdot\|p-c\|.$$

Note that this also implies

$$(1-\beta)\cdot\|p-c\|^z\leq\|f(p)-f(c)\|^z\leq(1+\beta)\cdot\|p-c\|^z.$$

We now discretise $\mathbb{R}^k$ as follows. We cover the entire k-sphere centred around q with radius $\left(\frac{8z}{\varepsilon}\right)\cdot 2\cdot\Delta_i^{\frac{1}{z}}$ with k-dimensional balls of radius at most $\frac{\varepsilon}{3z}\cdot\Delta_i^{\frac{1}{z}}$. Let B be the minimal set of balls required. In k-dimensional Euclidean spaces, such a cover has size at most

$$\left(1+2\cdot\left(\frac{8z}{\varepsilon}\right)^z/\varepsilon\right)^k.$$

For every center c' of some ball in B, we add the vector v' with entries $v'_p=\|f(p)-c'\|^z$ to the net N. We claim that N is an $O(\beta)$-net. The lemma then follows by rescaling β.

For every point c with $\|p-c\| \leq \frac{8z}{\varepsilon} \cdot \|p-q\|$, let $f(c)$ be the image of this point under the terminal embedding. Moreover, let v' be the vector induced by the point c' in B closest to $f(c)$. We have

$$\begin{aligned}
& |\|p-c\|^z - v_p| \\
= \; & |\|p-c\|^z - \|f(p)-f(c)\|^z + \|f(p)-f(c)\|^z - v_p| \\
= \; & |\|p-c\|^z - \|f(p)-f(c)\|^z + \|f(p)-f(c)\|^z - v_p| \\
(Lemma\ 1) \quad \leq \; & \beta \cdot \|p-c\|^z + \varepsilon \cdot \|f(p)-f(c)\|^z + \left(\frac{2z+1}{\varepsilon}\right)^{z-1} \|f(c)-c'\|^z \\
\leq \; & 3\beta \|f(p)-f(c)\|^z + \left(\frac{3z}{\varepsilon}\right)^{z-1} \cdot \left(\frac{\varepsilon}{3z}\right)^z \cdot \Delta_i \\
\leq \; & 3\beta \|f(p)-f(c)\|^z + \varepsilon \|p-q\|^z \\
\leq \; & 3\beta (\|f(p)-f(c)\|^z + \|p-q\|^z).
\end{aligned}$$

Rescaling β completes the proof. □

Lemma 6. *Let R and q be defined as above and let Ω be a uniform sample consisting of s points. Then for any point c and $s \geq \eta \cdot z^3 2^{8z} \cdot \varepsilon^{-2} \cdot \log \|R\|_0 \cdot \log^3 \varepsilon^{-1}$ for some absolute constant η, we have $\mathbb{E}_\Omega \mathbb{E}_g \sup_{c \in \mathbb{R}^d} \left| \frac{\sum_{j=1}^s \frac{|R|}{s} \|p_j - c\|^z \cdot g_j}{cost(R,q) + cost(R,c)} \right| \leq \varepsilon$. Moreover, if $s \geq \eta \cdot z^3 2^{8z} \cdot \varepsilon^{-2} \cdot \log \|R\|_0 \cdot \log^4 \varepsilon^{-1} \log 1/\delta$ for some absolute constant η, then we have $\sup_{c \in \mathbb{R}^d} \left| \frac{\sum_{j=1}^s \frac{|R|}{s} \|p_j - c\|^z \cdot g_j}{cost(R,q) + cost(R,c)} \right| \leq \varepsilon$, with probability at least $1-\delta$.*

Proof. Lemma 4 shows that in order to bound $\mathbb{E}_\Omega \mathbb{E}_g \sup_{c \in \mathbb{R}^d} \left| \frac{\sum_{j=1}^s \frac{|R|}{s} \|p_j - c\|^z \cdot g_j}{\text{cost}(R,q) + \text{cost}(R,c)} \right|$, it is enough to take the supremum only over vectors of $N_{\varepsilon/10}$.

For a center $c \in \mathbb{R}^d$, let v be the cost vector induced by c and let therefore $v' \in N_{\varepsilon/10}$ be the approximation of v given by the net. Furthermore, let $v'^0 = 0, v'^1, v'^2, \ldots$ be a sequence of cost vectors such that $v' = \sum_{k=1}^{\log \varepsilon/10} v'^{k+1} - v'^k$ and $v'^k \in \mathbb{N}_{2^{-k}}$.

$$\begin{aligned}
& \mathbb{E}_\Omega \mathbb{E}_g \sup_{v' \in N_{\varepsilon/10}} \left| \frac{\sum_{j=1}^s \frac{|R|}{s} v'_{p_j} \cdot g_j}{\text{cost}(R,q) + \text{cost}(R,c)} \right| \\
= \; & \mathbb{E}_\Omega \mathbb{E}_g \sup_{v' \in N_{\varepsilon/10}} \sum_{k=1}^{\log 10\varepsilon^{-1}} \left| \frac{\sum_{j=1}^s \frac{|R|}{s} (v'^{k+1}_{p_j} - v'^k_{p_j}) \cdot g_j}{\text{cost}(R,q) + \text{cost}(R,c)} \right| \\
\leq \; & \mathbb{E}_\Omega \sum_{k=1}^{\log \varepsilon/10} \mathbb{E}_g E'^{k+1}
\end{aligned}$$

with $E'^{k+1} = \sup_{v'^{i+1}, v'^i \in \mathbb{N}_{k+1} \times \mathbb{N}_k} \left| \frac{\sum_{j=1}^s \frac{|R|}{s} (v'^{i+1}_{p_j} - v'^i_{p_j}) \cdot g_j}{\text{cost}(R,q) + \text{cost}(R,c)} \right|$ so we now focus on bounding the supremum over the E'^{k+1}. For every i, $\sum_{j=1}^s \frac{|R|}{s} \frac{(v'^{i+1}_{p_j} - v'^i_{p_j}) \cdot g_j}{\text{cost}(R,q) + \text{cost}(R,c)}$ is Gaussian distributed with zero mean and variance

$$\begin{aligned}
& \sum_{j=1}^s \left(\frac{|R|}{s} \frac{(v'^{i+1}_{p_j} - v'^i_{p_j})}{\text{cost}(R,q) + \text{cost}(R,c)} \right)^2 \\
\leq \; & \sum_{j=1}^s \left(\frac{|R|}{s} \frac{(v'^{i+1}_{p_j} - v_{p_j} + v_{p_j} - v'^i_{p_j})}{\text{cost}(R,q) + \text{cost}(R,c)} \right)^2 \\
\leq \; & \sum_{j=1}^s \frac{|R|^2}{s^2} \cdot \frac{2^{-2k} \cdot 2 \cdot v_{p_j}^2}{(\text{cost}(R,q) + \text{cost}(R,c))^2}
\end{aligned}$$

We distinguish between two cases. If $v_{p_j} = \text{cost}(p_j, c) \geq 8^z \cdot \text{cost}(p_j, q)$, then for any point p' $\text{dist}(p, c) \leq \text{dist}(p', c) + \text{dist}(p_j, p') \leq \text{dist}(p', c) + 3\text{dist}(p_j, q) \leq \text{dist}(p', c) + \frac{3}{8}\text{dist}(p_j, c)$, and hence $\text{cost}(p_j, c) \leq 8^z \text{cost}(p', c)$, and by averaging $v_{p_j} \leq 8^z \cdot \frac{\text{cost}(R,c)}{|R|}$. Otherwise $v_{p_j} = \text{cost}(p, c) \leq 8^z \text{cost}(p, q) \leq 2 \cdot 8^z \cdot \frac{\text{cost}(R,q)}{|R|}$. Combining both in the aforementioned variance bound, we have:

$$\begin{aligned}
& \frac{1}{s} \sum_{j=1}^{s} \frac{2^{-2k} \cdot 2 \cdot v_{p_j}^2}{(\text{cost}(R, q) + \text{cost}(R, c))^2} \cdot \frac{|R|^2}{s} \\
\leq \quad & \frac{1}{s} \sum_{j=1}^{s} \frac{2^{-2k+3} \cdot 8^{2z} \cdot \max\left(\frac{\text{cost}(R,c)}{|R|}, \frac{\text{cost}(R,q)}{|R|}\right)^2}{(\text{cost}(R, q) + \text{cost}(R, c))^2} \cdot \frac{|R|^2}{s} \\
\leq \quad & \frac{1}{s} 8^{2z+1} \cdot 2^{-2k}
\end{aligned}$$

Now since E_k is the supremum of at most $|\mathbb{N}_{k+1}| \cdot |\mathbb{N}_k|$ many Gaussians, and the expected maximum over n Gaussians with varaince at most σ^2 is at most $\sqrt{2 \log n \cdot \sigma^2}$, we have

$$\begin{aligned}
& \mathbb{E}_g \sup E'^{k+1} \leq \sqrt{2 \log\left(|\mathbb{N}_{k+1}|^2\right) \cdot \frac{1}{s} \cdot 8^{2z+1} \cdot 2^{-2k}} \\
\leq \quad & \sqrt{2\gamma \cdot z^3 \cdot 2^{2k} \log |R| \cdot \log \varepsilon^{-1} \cdot \frac{1}{s} \cdot 8^{2z+1} 2^{-2k}} \\
\leq \quad & \varepsilon / \log 10\varepsilon^{-1},
\end{aligned}$$

where the first inequality follows from Lemma 5 and the second inequality holds by our choice of s. Therefore, $\mathbb{E}_g E'^{k+1} \leq \varepsilon / \log 1/\varepsilon$ and consequently

$$\begin{aligned}
& \mathbb{E}_\Omega \mathbb{E}_g \sup_{v' \in N_{\varepsilon/10}} \sum_{k=1}^{\log 10\varepsilon^{-1}} \left| \frac{\sum_{j=1}^{s} \frac{|R|}{s} v'_{p_j} \cdot g_j}{\text{cost}(R, q) + \text{cost}(R, c)} \right| \\
\leq \quad & \mathbb{E}_\Omega \sum_{k=1}^{\log 10\varepsilon^{-1}} \mathbb{E}_g E'^{k+1} \\
\leq \quad & \log 10\varepsilon^{-1} \cdot \varepsilon / \log 10\varepsilon^{-1} \leq \varepsilon.
\end{aligned}$$

The proof now follows since Lemma 4 states that is is sufficient to get a bound for all cost vectors in $N_{\varepsilon/10}$ in order to get a coreset for all $c \in \mathbb{R}^d$, up to a rescaling of ε by a factor 10.

To obtain a high probability bound, we now merely observe that for a zero mean Gaussian g with variance σ^2, we have $\mathbb{P}[g > t] \leq \frac{1}{2\pi} \exp(-t^2/(2\sigma^2))$. Hence, we can simply take a union bound over all steps of the chain an obtain

$$\begin{aligned}
& \mathbb{P}[\exists v'^{i+1}, v'^i \in \mathbb{N}_{i+1} \times \mathbb{N}_i \mid E_i > \varepsilon / \log 10\varepsilon^{-1}] \\
\leq \quad & |\mathbb{N}_{i+1}| \cdot |\mathbb{N}_i| \cdot \frac{1}{2\pi} \cdot \exp\left(\frac{\varepsilon^2}{\log^2 10\varepsilon^{-1}} \cdot \frac{8^{2z+2} \cdot 2^{-2k}}{s}\right).
\end{aligned}$$

The claim now follows by the second choice of s, and taking a union bound over all elements of the chain. □

Lemma 7. *Let R and q be defined as above. Suppose a uniform sample of size $s \in \tilde{O}(\Gamma \cdot \log \|R\|_0 \cdot \log 1/\delta)$ satisfies with probability at least $1 - \delta$ for all candidate centers c*

$$|\textit{cost}(R, c) - \sum_{p \in \Omega} \frac{|R|}{s} \cdot \|p - c\|^z| \quad \leq \quad \varepsilon \cdot (\textit{cost}(R, q) + \textit{cost}(R, c)).$$

Then a uniform sample of size $\tilde{O}(\Gamma \cdot \log 1/\delta)$ achieves the same guarantee.

Proof. We start by briefly outlining the key arguments from Theorem 3.1 of Braverman, Jiang, Krauthgamer, and Wu Braverman et al. (2021). Denote by $\log^{(i)} n$ the i-fold logarithm of n, i.e. $\log^{(2)} n = \log\log n$. Suppose that the initial summary has size $\Gamma \cdot \log \|R\|_0$. We call the sampling algorithm recursively. In iteration i, let $\|R_i\|_0$ be the distinct number of points left in iteration i, let ε_i be the precision parameter used in iteration i and let δ_i be the failure probability. We choose the parameters $\varepsilon_i := \varepsilon/(\log^{(i)} \|R\|_0)^{\frac{1}{2}}$ and $\delta_i := \delta/\|R_{i-1}\|_0$.

BJKW show the invariants $\|R_i\|_0 \leq 20\Gamma \log \delta^{-1}(\log^{(i)} \|R\|_0)^3$, $\prod_{i=1}^{t}(1+\varepsilon_i) \leq \exp(2\varepsilon_t)$ and $\sum_{i=1}^{t} \delta_i \leq \delta \cdot \left(\frac{1}{\|R\|_0} + \frac{1}{\log \|R\|_0} + \cdot + \frac{1}{\log^{(t-1)} \|R\|_0}\right) \in O(\delta)$. The first invariant shows that after $O(\Gamma^4)$ iterations, $\log^{(t)} \|R\|_0 \leq 20\Gamma$. The second invariants bounds the overall error, which, by choice of ε_t and the maximum number of iterations, is less than $(1 + O(\varepsilon))$. The final invariant shows that the overall failure increases only by constant factors.

We now argue why it is sufficient for an algorithm that only uses uniform sampling to target the final bound after the iterative size reduction. Consider the sampling distributions D_0 and D_t, where D_0 is the sampling distribution before and and D_t is the sampling distribution after the iterative size reduction. Note that since uniform sampling assigns the exact same weight to every point, the sampling itself remains weight oblivious.

Therefore the sampling distribution of every application of Lemma 3 remains uniform sampling, i.e. the probability that a point p is in the output of D_t is equal for all points. The same holds for the output of D_0, therefore the distributions of both algorithms are identical. Since D_t achieves a the desired guarantee by sampling $\tilde{O}(\varepsilon^{-2} \cdot 2^{O(z)})$ many distinct points, D_0 must do so as well. □

C Pruning Lemmas

Lemma 8. *Suppose we are given an α-approximate center q. Let $B(q,r)$ be the ball centered at q with radius $r = 4 \cdot \left(\frac{2\alpha \cdot OPT}{n}\right)^{1/z}$. Then the following two statements hold.*

1. *Any α-approximate center is in $B(q,r)$.*
2. *For any two points $c, c' \in B(q,r)$ and for any point p with $\|p-q\|^z > \gamma \cdot r^z$ with $\gamma > \varepsilon^{-z} \cdot (12z)^z$, we have $\|p-c\|^z \leq (1+\varepsilon) \cdot \|p-c'\|^z$.*

Proof. For the first claim we consider a point c not in $B(q,r)$ and show that c cannot be an α-approximate center.

The average cost of the points when using q as a center is $\alpha \cdot \frac{OPT}{n}$. Hence, by Markov's inequality, at least half of the points of A lie in $B(q, r/4)$. Furthermore, by choice of c and the triangle inequality, we have $\|p-c\| > 2 \cdot \|p-q\|$ for any point $p \in B(q, r/4)$. Hence, the cost of clustering all the points in $A \cap B(q, r/4)$ to c is at least $n/2 \cdot (2 \cdot r)^z \geq \alpha \cdot OPT$.

For the second claim, let $c, c' \in B(q,r)$ and p with $\|p-q\|^z \geq \gamma \cdot r^z$. We first note that $\|p-c'\| \geq \|p-q\| - \|q-c'\| \geq \gamma^{1/z} \cdot r - 2r = (\gamma^{1/z} - 2)r$, which yields the inequality

$$r \leq \|p-c'\| \cdot \frac{1}{\gamma^{1/z} - 2} \tag{5}$$

We then have

$$
\begin{aligned}
& \|p-c\|^z \\
(Lem.\ 1) \quad \leq \quad & (1+\varepsilon/2z)^{z-1}\|p-c'\|^z + \left(\frac{\varepsilon+2z}{\varepsilon}\right)^{z-1} \cdot \|c-c'\|^z \\
\leq \quad & (1+\varepsilon/2)\cdot\|p-c'\|^z + \left(\frac{3z}{\varepsilon}\right)^{z-1}(2r)^z \\
(Eq.\ 5) \quad \leq \quad & (1+\varepsilon/2)\cdot\|p-c'\|^z + \left(\frac{3z}{\varepsilon}\right)^{z-1} 2^z\cdot\|p-c'\|^z\left(\frac{1}{\gamma^{1/z}-2}\right)^z \\
\leq \quad & (1+\varepsilon/2)\cdot\|p-c'\|^z + \left(\frac{3z}{\varepsilon}\right)^{z-1} 4^z\cdot\|p-c'\|^z\cdot\gamma^{-1} \\
(\text{Choice of } \gamma) \quad \leq \quad & (1+\varepsilon/2)\cdot\|p-c'\|^z + \varepsilon/2\cdot\|p-c'\|^z \\
\leq \quad & (1+\varepsilon)\cdot\|p-c'\|^z
\end{aligned}
$$

□

Lemma 9. *Suppose that q is an α-approximate solution. Let $A_{near} \subset A$ be a set of points with cost at most $(\varepsilon/(\alpha 5z))^z \cdot \frac{OPT}{n}$. Let $\hat{A} = (1\pm\varepsilon)|A_{near}|$ Then for any candidate solution c we have*

$$
\left|\hat{A}\cdot\|q-c\|^z - \sum_{p\in A_{near}}\|p-c\|^z\right| \leq \varepsilon/\alpha\cdot\left(\sum_{p\in A_{near}}\|p-c\|^z + OPT\right).
$$

Proof. We first prove the result for $\hat{A} = |A_{near}|$, the claim for an estimation of $|A_{near}|$ is a simple corollary. We have, using Lemma 1:

$$
\begin{aligned}
& \left|\sum_{p\in A_{near}}(\|p-c\|^z - \|q-c\|^z)\right| \\
\leq \quad & \sum_{p\in A_{near}}|\|p-c\|^z - \|q-c\|^z| \\
\leq \quad & \sum_{p\in A_{near}}\left(\frac{\varepsilon}{\alpha 2}\cdot\|p-c\|^z + \left(\frac{\alpha 5z}{\varepsilon}\right)^{z-1}\|p-q\|^z\right) \\
\leq \quad & \sum_{p\in A_{near}}\left(\frac{\varepsilon}{\alpha 2}\cdot\|p-c\|^z + \left(\frac{\alpha 5z}{\varepsilon}\right)^{z-1}\left(\frac{\varepsilon}{\alpha 5z}\right)^z\frac{OPT}{n}\right) \\
\leq \quad & \varepsilon/\alpha\cdot\left(\sum_{p\in A_{near}}\|p-c\|^z + \text{OPT}\right)
\end{aligned}
$$

For an approximation to $|A_{near}|$ we now merely add an additional additive error $\varepsilon\cdot\sum_{p\in A_{near}}\|p-c\|^z$ to the difference of the two terms. □

Lemma 10. *Suppose that q that is an α-approximate solution. Let R_{cheap} the union of rings with $|R_i \cap A| < \varepsilon\cdot\eta\cdot n$ and with radius at most $4(\gamma\cdot\frac{\alpha\cdot OPT}{n})^{1/z}$, where γ is given by Lemma 8. Then, for any candidate solution c*

$$
\sum_{p\in R_{cheap}\cap A}\|p-c\|^z \leq \varepsilon\cdot 4^{z+1}\cdot\beta\cdot\eta\cdot\gamma\cdot\alpha\cdot\sum_{p\in A}\|p-c\|^z.
$$

Proof. We first require a bound on $\|q-c\|^z$. Using Lemma 1, we have

$$\begin{aligned} n \cdot \|q-c\|^z &\leq 2^z \sum_{p \in A} \|p-q\|^z + \|p-c\|^z \\ &\leq \alpha \cdot 2^{z+1} \cdot \sum_{p \in A} \|p-c\|^z \\ \Rightarrow \|p-c\|^z &\leq \alpha \cdot 2^{z+1} \cdot \frac{\sum_{p \in A} \|p-c\|^z}{n} \end{aligned} \tag{6}$$

Therefore with another application of Lemma 1

$$\begin{aligned} & \sum_{p \in R_{cheap} \cap A} \|p-c\|^z \\ \leq & \sum_{p \in R_{cheap} \cap A} 2^z \cdot (\|p-q\|^z + \|q-c\|^z) \\ (Eq.\ 6) \quad \leq & \beta \cdot \varepsilon \cdot \eta \cdot n \cdot 2^z \cdot \Bigg(\gamma \cdot \frac{\alpha \cdot OPT}{n} \\ & + \alpha \cdot 2^{z+1} \cdot \frac{\sum_{p \in A} \|p-c\|^z}{n} \Bigg) \\ \leq & \beta \cdot \varepsilon \cdot \eta \cdot \alpha \cdot \gamma 4^{z+1} \cdot \sum_{p \in A} \|p-c\|^z. \end{aligned}$$

□

Lemma 11. *Let q be a point that is an α-approximation and let S be a uniform sample consisting of $O\left(\alpha \cdot \eta^{-1} \cdot \varepsilon^{-3} polylog(\varepsilon^{-1} \cdot \delta^{-1})\right)$ points. Then with probability at least $1-\delta$ for all rings R_i,*

$$|R_i \cap A| - \varepsilon \cdot \max\left(n \cdot \eta, |R_i \cap A|\right) \leq \frac{|R_i \cap S| \cdot n}{|S|} \leq |R_i \cap A| + \varepsilon \cdot \max\left(n \cdot \eta,\ |R_i \cap A|\right).$$

Furthermore, let d as in the algorithm, i.e., such that $\frac{2}{3} \cdot \varepsilon \cdot \eta \cdot |S| \leq \left|S \setminus (B(q,d) \cap S)\right|$. Then $d < \left(3(\varepsilon\eta)^{-1} \cdot \frac{\alpha \cdot OPT}{n}\right)^{1/z}$

Proof. We consider the range space induced by Euclidean balls centered around q. This range space has VC dimension of exactly 2. The VC dimension induced by the intersection of two Euclidean balls centered around q is still constant Blumer et al. (1989), hence for our choice of $|S|$, Lemma 2 ensures that we have approximated the cardinality of all rings up to the additive error $\varepsilon \cdot \max(\eta \cdot n, |R_i \cap A|)$ with probability at least $1-\delta$, which proves the first claim.

For the second claim, let d as in the algorithm. By Lemma 2, we have $\left|S \setminus (B(q,d) \cap S)\right| \cdot \frac{n}{|S|} \leq (1+\varepsilon)\left|A \setminus (B(q,d) \cap A)\right|$. Hence, we have

$$\left|A \setminus (B(q,d) \cap A)\right| \geq \frac{2}{3} \cdot \varepsilon \cdot \eta \cdot |S| \cdot \frac{n}{|S|(1+\varepsilon)} \geq \frac{\varepsilon \cdot \eta \cdot n}{3}$$

Using Markov's inequality, we now know that the number of points with cost $3(\varepsilon\eta)^{-1} \cdot \frac{\alpha \cdot OPT}{n}$ is at most $\frac{\varepsilon \cdot \eta \cdot n}{3}$. This implies $d \leq \left(3(\varepsilon\eta)^{-1} \cdot \frac{\alpha \cdot OPT}{n}\right)^{1/z}$. □

D Proof of Theorem 1

We start by specifying our parameters: the approximation is set to be $\alpha = 20^z$. To prune the far points, we set $\gamma = (12z/\varepsilon)^z$. Finally, we pick β and η such that $\eta = \frac{1}{2^{z-1}\alpha \cdot \beta \cdot \gamma}$ and $3 \cdot 2^{-z\beta+z} \cdot \frac{\alpha}{\varepsilon\eta} \leq \left(\frac{\varepsilon}{5\alpha}\right)^z$. This is possible for $\beta = O_z(\log(1/\varepsilon))$ and $\eta \in O_z(\varepsilon^{-z}\text{polylog}(1/\varepsilon))$

First, note that Lemma 8 shows that it is enough to compute an approximate solution for the set A' consisting of points that are at distance less than $O\left(\left(\gamma \frac{\text{OPT}}{n}\right)^{1/z}\right)$.

Let c be the optimal $(1, z)$-center. First, let us consider the initial sampled point q. With probability at least $9/10$, we have $\|q-c\|^z \leq 10 \cdot \frac{OPT}{n}$. Hence due to the triangle inequality $\sum_{p\in A} \|p-q\|^z \leq 20^z \cdot OPT$, and q is an α-approximation. Let S be the set of points sampled and pruned by the algorithm.

For each ring R_i at distance $(2^{-i}d, 2^{-i+1}d]$, we denote by $\hat{R}_i$ either a $(1\pm\varepsilon)$-estimate of the size of $|R_i|$ via $\frac{n\cdot|R_i\cap S|}{|S|}$, if $|R_i \cap S| \geq \varepsilon\cdot\eta\cdot|S|$, or we set $\hat{R}_i = 0$ if $|R_i\cap S| < \varepsilon\cdot\eta\cdot|S|$. Similarly, define R_β to be a $(1\pm\varepsilon)$-estimate of the size of points at distance less than $2^{-\beta}d$ from q, if $R_\beta \geq \varepsilon\cdot\eta\cdot|S|$, or 0 if $R_\beta < \varepsilon\cdot\eta\cdot|S|$.

Our goal to show that for any candidate solution c', we have

$$\left|\sum_{p\in A'}\|p-c'\|^z - \left(\sum_{i\leq\beta-1}\frac{\hat{R}_i}{|R_i\cap S|}\sum_{p\in R_i\cap S}\|p-c'\|^z + \frac{\hat{R}_\beta}{|R_\beta\cap S|}\|q-c'\|^z\right)\right| \leq \varepsilon\cdot\sum_{p\in A'}\|p-c'\|^z. \tag{7}$$

Hence, computing a $(1+\varepsilon)$-approximate solution on the set S will give $(1+\varepsilon)$-approximate solution for A', which is also one for A following Lemma 8.

We now consider all rings R_i centered around q with radius $(2^{-i}d, 2^{-i+1}d]$. First, for $i = \beta \in O(\log 1/\varepsilon)$, we have using Lemma 11 that the cost of points in R_i is, by choice of β, $\left(2^{-\beta+1}d\right)^z \leq 2^{-z\beta+z}\frac{3}{\varepsilon\eta}\frac{OPT}{n} \leq (\varepsilon/(\alpha 5z))^z\,\frac{OPT}{n}$, hence we can use Lemma 9 to bound

$$\begin{aligned}\left|\sum_{p\in A'\cap R_\beta}\|p-c'\|^z - \hat{R}_\beta\|q-c'\|^z\right| &\leq \varepsilon/\alpha\cdot\sum_{p\in A_{near}}\|p-c'\|^z + \|p-q\|^z\\ &\leq \varepsilon\cdot\sum_{p\in A_{near}}\|p-c\|^z + \|p-c'\|^z.\end{aligned}$$

Having dealt with the points close to q, we now deal with those far away. Since A' results in the pruning of A, rings with radius more than $4(\gamma\cdot\frac{\alpha\cdot OPT}{n})^{1/z}$ are empty in A'. Moreover, due to Lemma 11, those rings must have $\hat{R}_i = 0$; hence, their contribution to Eq. (7) is 0.

We now turn our attention to the remaining rings. First, we consider the cheap rings, i.e. all rings with $\hat{R}_i = 0$. Note that, by choice of d; this includes all rings with $i \leq 1$. We have, due to Lemma 10:

$$\sum_{p\in R_{cheap}\cap A}\|p-c\|^z \leq 4^{z+1}\beta\cdot\varepsilon\cdot\eta\cdot\alpha\cdot\gamma\cdot\left(\sum_{p\in A}\|p-c\|^z\right) \leq \varepsilon\left(\sum_{p\in A}\|p-c\|^z\right)$$

Recall that $\hat{R}_i = 0$ in this case. We therefore obtain

$$\left|\sum_{p\in R_{cheap}\cap A}\|p-c\|^z - \frac{\hat{R}_i}{|R_i\cap S|}\sum_{p\in R_i\cap S}\|p-c\|^z\right| \leq \varepsilon\sum_{p\in A}\|p-c\|^z.$$

Finally, we consider rings with $|R_i\cap S| > \varepsilon\eta|S|$, with $\beta-1 \leq i \leq 1$. With our choice of $|S| = \frac{\alpha\cdot\text{polylog}(\varepsilon^{-1}\delta^{-1})}{\eta\cdot\varepsilon^3}$ and Lemma 11, we have therefore $|R_i\cap S| > \varepsilon^{-4}\text{polylog}(\varepsilon^{-1})$. Theorem 4 guarantee us that with $\tilde{O}(\varepsilon^{-2})$ many samples, we have

$$\left|\sum_{p\in R_i}\|p-c\|^z - \frac{|R_i|}{m}\sum_{p\in S}\|p-c\|^z\right| \leq \varepsilon/\beta\cdot\left(\sum_{p\in R_i}\|p-c\|^z + \|p-q\|^z\right)$$

Summing up the error for all rings yields a total error of $O(\varepsilon)\cdot\sum_{p\in A'}\|p-c\|^z + \|p-c'\|^z \in O(\varepsilon)\sum_{p\in A'}\|p-c'\|^z$.

Subsequently, we can use any desired optimization algorithm to compute a $(1+\varepsilon)$-approximate solution c' on S, with weights $\frac{\hat{R}_i}{|R_i\cap S|}$ on points of R_i. Rescaling ε according to degree of precision of the optimization procedure and the precision of the coreset completes the proof.

E Probability Amplification

Lemma 12. *Given query access to A, we can identify with probability $1-\delta$ a 8^z-approximate solution using $O(\varepsilon^{-2}(\log 1/\varepsilon + \log 1/\delta)\log 1/\delta)$ samples.*

Proof. Let c be the optimal center. In the following we will assume no knowledge of OPT, or even an estimate of OPT, but we assume to know n.

With probability at least $1/2$, a random point q_i satisfies

$$\|q_i - c\|^z \leq 2^z \frac{OPT}{n}.$$

Therefore, when sampling $\log 1/\delta$ points, we will have sampled a 2^z approximate solution with probability at least $1-\delta$.

Furthermore

$$\sum_{p\in A} \|p - q_i\|^z \leq \sum_{p\in A} 2^{z-1} \cdot (\|p-c\|^z + \|q_i - c\|^z) \leq 2^z \cdot OPT.$$

Now since the range space induced by unit Euclidean balls centered around q has VC dimension 2, we can estimate the number of points for any given radius up to an additive error of $\varepsilon \cdot n$. Hence, with probability $1-\delta$, for every 2^z-approximate solution q_i, the estimated number of points in $B(q, 2\left(4\frac{OPT}{n}\right)^{1/z})$ will be at least $n/2$.

Conversely, if q_j is not 8^z approximate, the estimated number of points in $B(q_j, 2\left(4\frac{OPT}{n}\right)^{1/z})$ is small. We have

$$\begin{aligned}
& \sum_p \|p - q_j\|^z > 8^z \sum_p \|p - c\|^z \\
\Rightarrow \quad & \left(\sum_p \|p - q_j\|^z\right)^{1/z} > 8\left(\sum_p \|p - c\|^z\right)^{1/z} \\
\Rightarrow \quad & \left(\sum_p \|c - q_j\|^z\right)^{1/z} > 7\left(\sum_p \|p - c\|^z\right)^{1/z} \\
\Rightarrow \quad & \|c - q_j\| > 7\left(\frac{OPT}{n}\right)^{1/z}
\end{aligned}$$

Therefore, the intersection of $B(q_j, 2\left(4\frac{OPT}{n}\right)^{1/z})$ with $B(c, \left(2\frac{OPT}{n}\right)^{1/z})$ is empty. Since at least $\frac{3n}{4}$ points lie in $B(c, 2\left(4\frac{OPT}{n}\right)^{1/z})$, we know that with probability at least $1-\delta$ the estimated number of points in $B(q_i, 3\left(\frac{OPT}{n}\right)^{1/z})$ will be larger than the estimated number of points in $B(q_j, 2\left(4\frac{OPT}{n}\right)^{1/z})$. Conditioned on having a 2^z approximate solution in our sample, the returned point is therefore no worse than a 8^z approximation. □

F Lower bound

The goal of that section is to prove Theorem 3, i.e., that any $(1+\varepsilon)$-approximation algorithm must sample ε^{-z+1} points.

Proof of Theorem 3. Consider the instance $\mathcal{I}$ on the 1-dimensional line where n points are located at 0 and $\varepsilon^{z-1}n$ points are located at 1. Intuitively, we show that any approximation algorithm on $\mathcal{I}$ must sample at least a point at 1, and so must sample at least $\varepsilon^{-z+1}n$ points.

For simplicity, we rescale the instance so that $n = 1$. The optimal solution is $\mathrm{OPT}_{\mathcal{I}} = \inf x^z + \varepsilon^{z-1}(1-x)^z$, and the optimal center is such that the derivative of the objective function is zero:

$$\frac{\partial}{\partial x}\left(x^z + \varepsilon^{z-1}(1-x)^z\right) = (z-1)\left(x^{z-1} - (\varepsilon - \varepsilon x)^{z-1}\right)$$

so the optimal value is for x_{OPT} such that $(z-1)\left(x_{\text{OPT}}^{z-1} - (\varepsilon - \varepsilon x_{\text{OPT}})^{z-1}\right) = 0$, which is $x_{\text{OPT}} = \frac{\varepsilon}{\varepsilon+1}$. Hence,

$$\begin{aligned}\text{OPT} &= \left(\frac{\varepsilon}{\varepsilon+1}\right)^z + \varepsilon^{z-1}\left(1 - \frac{\varepsilon}{\varepsilon+1}\right)^z \\ &= \frac{\varepsilon^{z-1}}{(\varepsilon+1)^z}(\varepsilon+1) = \left(\frac{\epsilon}{1+\varepsilon}\right)^{z-1}.\end{aligned}$$

Since the cost of the solution having a center at 0 is ε^{z-1}, is it bigger than $(1+\varepsilon)$OPT: indeed,

$$(1+\varepsilon)\left(\frac{\epsilon}{1+\varepsilon}\right)^{z-1} < \varepsilon^{z-1}. \tag{8}$$

Now, consider the instance $\mathcal{I}$ and the instance $\mathcal{I}'$ that has n points located at 0. let $\mathcal{A}$ be an algorithm that, with probability more than $4/5$, computes a $(1+\varepsilon)$-approximation for $(1,z)$-clustering.

Assume by contradiction that $\mathcal{A}$ samples less than $\varepsilon^{-z+1}/10$ points. Let X be the random variable counting the number of points located at 1 in that sample: we have $\Pr[X > 0] \leq \mathbb{E}[X] \leq 1/10$. So with probability at least $9/10$, $\mathcal{A}$ samples only point located at 0: even when that event occures, $\mathcal{A}$ must output a center at a position different than 0 (following Equation 8) with some probability p.

Since $\mathcal{A}$ succeeds with probability $4/5$ and $X = 0$ with probability at least $9/10$, we must have have $\frac{9}{1}0p + \frac{1}{1}0 \geq 4/5$, and so $p \geq \frac{7}{9}$.

Hence, when $\mathcal{A}$ samples only points located at 0, it must output a center different from 0 with probability at least $7/9$. In particular, on instance $\mathcal{I}'$, $\mathcal{A}$ fails with probability at least $7/9$, a contradiction.

So, any algorithm that computes a $(1+\varepsilon)$-approximation for $(1,z)$-clustering with probability more than $4/5$ must sample more than $\varepsilon^{-z+1}/10$ points. □

G Improved Coreset Construction

Lemma 13. *Let q, R_M and κ be defined as above. Then for any point $c \in \mathbb{R}^d$, we have*

$$|\mathit{cost}(A,c) - (\mathit{cost}(R_M,c) + \kappa + |A \setminus R_M| \cdot \|q-c\|^2)| \leq \varepsilon \cdot \mathit{cost}(A,c).$$

Proof. First, we bound the difference in cost for the points that are close to q, i.e. the points in R_i with $i \leq -1$. We have for any such point p

$$\begin{aligned} & |\|p-c\|^z - \|q-c\|^z| \\ \leq\ & \varepsilon \cdot \|p-c\|^z + \left(\frac{z+\varepsilon}{\varepsilon}\right)^{z-1}\|p-q\|^z \\ \leq\ & \varepsilon \cdot \|p-c\|^z + \left(\frac{z+\varepsilon}{\varepsilon}\right)^{z-1}\left(\frac{\varepsilon}{2z}\right)^z \cdot \frac{1}{\alpha} \cdot \Delta \\ \leq\ & \varepsilon \cdot \|p-c\|^z + \left(\frac{z+\varepsilon}{\varepsilon}\right)^{z-1}\left(\frac{\varepsilon}{2z}\right)^z \cdot \frac{1}{\alpha} \cdot \alpha \cdot \frac{\text{cost}(c)}{n} \\ \leq\ & \varepsilon \cdot \|p-c\|^z + \varepsilon \cdot \frac{\text{cost}(c)}{n}. \end{aligned}$$

Since there are at most n in $\cup_{i \leq -1} R_i$, we therefore have

$$\begin{aligned} & |\sum_{i \leq -1}\sum_{p \in R_i} \|p-c\|^z - \|q-c\|^2| \\ \leq\ & \varepsilon \cdot \sum_{i \leq -1}\sum_{p \in R_i} \|p-c\|^z + \varepsilon \cdot \text{cost}(c) \\ \leq\ & 2 \cdot \varepsilon \cdot \text{cost}(c). \end{aligned} \tag{9}$$

Now we focus on the points in R_i with $i \geq \log \varepsilon^{-z}$. We distinguish between two cases. The first case will assume that $\|q-c\|^z \leq \Delta \cdot \left(\frac{\varepsilon}{4z}\right)^{-z}$. Here, the intuition is that since these points are close to

q (at least with respect to the points in R_i, $i \geq \log \varepsilon^{-z}$) κ is a good approximation to their cost. The second case assumes that $\|q-c\|^z \geq \Delta \cdot \left(\frac{\varepsilon}{4z}\right)^{-z}$. Here, the intuition is that $\sum_{i \geq \log \varepsilon^{-z}} \text{cost}(R_i, c)$ is very small compared to $\text{cost}(A, c)$.

In the first case, we have for any point $p \in R_i$ with $i \geq \varepsilon^{-3z}$

$$\begin{aligned}
|\|p-c\|^z - \|p-q\|^z| &\leq \varepsilon \cdot \|p-c\|^z + \left(\frac{z+\varepsilon}{\varepsilon}\right)^{z-1} \|c-q\|^z \\
&\leq \varepsilon \cdot \|p-c\|^z + \left(\frac{z+\varepsilon}{\varepsilon}\right)^{z-1} \cdot \Delta \cdot \left(\frac{\varepsilon}{4z}\right)^{-z} \\
&\leq \varepsilon \cdot \|p-c\|^z + \left(\frac{4z}{\varepsilon}\right)^{2z-1} \cdot \Delta \\
&\leq \varepsilon \cdot \|p-c\|^z + \varepsilon \cdot \frac{1}{\alpha} \cdot \|p-q\|^z.
\end{aligned}$$

Again, we sum this over all points in $\cup_{i \geq \log \varepsilon^{-2z}} R_i$. We then have

$$\begin{aligned}
&\left| \sum_{i \geq \log \varepsilon^{-2z}} \sum_{p \in R_i} (\|p-c\|^z - \|q-c\|^z) - \kappa \right| \\
= \; & |\sum_{i \geq \log \varepsilon^{-2z}} \sum_{p \in R_i} \|p-c\|^z - \|p-q\|^z| + \sum_{i \geq \log \varepsilon^{-2z}} \sum_{p \in R_i} \|q-c\|^z \\
\leq \; & \varepsilon \sum_{i \geq \log \varepsilon^{-2z}} \sum_{p \in R_i} \|p-c\|^z + \sum_{i \geq \log \varepsilon^{-2z}} \sum_{p \in R_i} \varepsilon \cdot \frac{2}{\alpha} \cdot \|p-q\|^z \\
\leq \; & \varepsilon \cdot \text{cost}(A, c) + \varepsilon \cdot \frac{2}{\alpha} \cdot \text{cost}(A, q) \\
\leq \; & 3\varepsilon \cdot \text{cost}(A, c) \qquad (10)
\end{aligned}$$

We now focus on the second case. Let A_2 be the set of points with $\|p-q\|^z \leq 2\Delta$. Due to Markov's inequality, we have $|A_2| \geq \frac{n}{2}$. Also due to Markov's inequality, we have $|\bigcup_{i \geq \log \varepsilon^{-2z}} R_i| \leq \varepsilon^{2z} \cdot n$. We now give a lower bound on the cost of the points in A_2. We start by showing that the difference in cost between any point in A_2 and q when clustering to c is negligible. Since $\|q-c\| \geq \frac{4z}{\varepsilon} \cdot \Delta^{1/z}$ and $\|p-q\| \leq 2^{1/z} \cdot \Delta^{1/z}$, we have $\|p-c\|^z \geq (1-\varepsilon)\|q-c\|^z$. This implies

$$\text{cost}(A_2, c) \geq |A_2| \cdot (1-\varepsilon)\|q-c\|^z$$

Therefore

$$\begin{aligned}
\sum_{i \geq \log \varepsilon^{-2z}} \text{cost}(R_i, c) &\leq \sum_{i \geq \log \varepsilon^{-2z}} (1+\varepsilon) \cdot \text{cost}(R_i, q) + \left(\frac{\varepsilon}{4z}\right)^{2z} \cdot n \left(\frac{z+\varepsilon}{\varepsilon}\right)^{z-1} \cdot \|q-c\|^z \\
&\leq (1+\varepsilon) \cdot \text{cost}(A, q) + \left(\frac{\varepsilon}{4z}\right)^{z+1} \cdot n \cdot \|q-c\|^z \\
&\leq (1+\varepsilon) \cdot \text{cost}(A, q) + \left(\frac{\varepsilon}{4z}\right)^{z+1} 2|A_2| \cdot \|q-c\|^z \\
&\leq (1+\varepsilon) \cdot 2|A_2| \cdot \Delta + \left(\frac{\varepsilon}{4z}\right)^{z+1} 2|A_2| \cdot \|q-c\|^z \\
&\leq (1+\varepsilon) \cdot 2|A_2| \cdot \left(\frac{\varepsilon}{4z}\right)^{z} \cdot \|q-c\|^z \\
&\leq \varepsilon \cdot \text{cost}(A_2, c) \leq \varepsilon \cdot \text{cost}(A, c). \qquad (11)
\end{aligned}$$

Similarly

$$\begin{aligned}
\kappa + \sum_{i \geq \log \varepsilon^{-2z}} |R_i| \cdot \|q-c\|^z &= \sum_{i \geq \log \varepsilon^{-2z}} \sum_{p \in R_i} \|p-q\|^z + \|q-c\|^z \\
&\leq n \cdot \Delta + \varepsilon^{2z} \cdot n \cdot \|q-c\|^z \\
&\leq n \cdot \left(\frac{\varepsilon}{4z}\right)^{z} \cdot \|q-c\|^z \|q-c\|^z + \varepsilon^{2z} \cdot n \cdot \|q-c\|^z \\
&\leq \varepsilon \cdot \text{cost}(A_2, c) \leq \varepsilon \cdot \text{cost}(A, c). \qquad (12)
\end{aligned}$$

Combining Equations 9, 10, 11, and 12 and rescaling ε now yields the claim. $\square$

Proof of Theorem 2. Let c be an arbitrary solution. For the points in $A \setminus R_M$, Lemma 13 deterministically allows us to bound the error by at most $\varepsilon \cdot \text{cost}(A, c)$. So we now turn our attention to the rings in R_M. We have for all $c \in \mathbb{R}^d$

$$\begin{aligned}\left|\sum_{p\in R_M} \|p-c\|^z - \sum_{p\in\Omega} w_p \cdot \|p-c\|^z\right| &\leq \sum_{R_i\in R_M} |\sum_{p\in R_i} \|p-c\|^z - \sum_{p\in\Omega_i} w_p \cdot \|p-c\|^z| \\ &\leq \sum_{R_i\in R_M} \varepsilon \cdot (\text{cost}(R_i, q) + \text{cost}(R_i, c)) \\ &= \varepsilon \cdot (\text{cost}(R_M, q) + \text{cost}(R_M, c))\end{aligned}$$

where the second inequality uses Theorem 4. Thus, taking a union bound over all $R_i \in R_M$, we have with probability at least $1 - O(\log 1/\varepsilon)\delta$ for all points $c \in \mathbb{R}^d$

$$\left|\sum_{p\in A} \|p-c\|^z - (\sum_{p\in S} w_p \|p-c\|^z + \kappa)\right| \leq 2\varepsilon \cdot (\text{cost}(A, c) + \text{cost}(A, q))$$

Rescaling ε by a factor $4/\alpha$ and rescaling δ by a factor $1/\log 1/\varepsilon$ yields the desired bounds. $\square$

H Experiments

Implementation: We used a variant of Algorithm 1 which now describe. Instead of specifying a desired accuracy, the algorithm is access to m samples picked uniformly at random from the data set. As an α-approximate solution q, the algorithm merely selects a random point.

We also estimate $\frac{OPT}{n}$, by sampling another point q' and using $\|q - q'\|^z$ as a (coarse) estimate. We then apply the pruning procedures. Our algorithms chose $\{100, 200, \ldots, 1000\}$ samples. For each sample size, we repeated the algorithm 10 times and outputted the best center we could find.

Since the objective function is convex, we use a (simple) stochastic gradient descent on both the sample and the full data set to compute a desired center. We iterated over the data set a total of 10 times. In every iteration, we partitioned the data into random chunks of size $\min(m, 2000)$, and used chunk to perform a gradient step. We did not attempt to optimize the stochastic gradient descent; as our focus is less on solving the problem in the fastest way possible and more on showcasing how the sublinear algorithm can be used to potentially speed up any baseline algorithm.

The algorithms were coded in Python and run on a Intel Core i7-8665U processor with four 2 GHz cores and 32 GByte RAM.

Results Tables with exact figures are given below. Here, we report and interpret the results.

On the Road Network data set, all samples sizes found a nearly optimal solution in at least one of the 10 repetitions, with the largest deviation from the optimum of 4% occurring for 500 samples and the $z = 4$ problem. In addition, the sublinear algorithms all required only a very small amount of time compared to the baseline optimal solution (e.g. a factor of at least 400 quicker for the largest sample size). What is notable is that starting with $z = 5$, the variance in cost of any given sample size increased significantly. Since this occurred regardless of sample size, we attribute this effect to quality of the seeding solution (q in Algorithm 1). The approximation factor of q directly impacts the quality of the subsequent coreset construction, meaning that even with large sample sizes, the algorithm has difficulty to recover. This means that good seeding solution q, for example using Lemma 12 is essential.

Processing the USCensus data set was in its entirety was time consuming, running for more than 90 minutes. Constructing the coreset and optimizing it never took more than 14 seconds, however the algorithm did not compute near optimal solutions as was the case for Road Networks data set. For $z = 3$ and $z = 5$ the approximation was still rather small, and tightly concentrated. For to the authors very unclear reasons, there exists a larger gap at $z = 4$. While a gap of that magnitude is consistent with the lower bound, the data set does not seem to have a structure similar to said lower bound.

Samples	$z = 3$ Cost			Time	$z = 4$ Cost			Time	$z = 5$ Cost			Time
	Min	Avg	Var/Avg^2	Avg	Min	Avg	Var/Avg^2	Avg	Min	Avg	Var/Avg^2	Avg
100	4,90	6,67	0,08	0,62	1,79	2,83	0,12	0,62	7,47	10,13	0,35	0,61
200	4,80	6,36	0,05	1,16	1,86	3,85	0,24	1,29	7,37	8,29	0,30	1,16
300	4,79	7,40	0,21	1,74	1,78	2,48	0,16	1,76	7,39	12,44	0,15	1,76
400	4,86	6,89	0,11	2,27	1,79	2,82	0,13	2,37	7,37	12,44	0,51	2,41
500	4,95	7,80	0,04	2,88	1,84	2,88	0,47	2,90	7,39	11,11	0,61	2,89
600	4,84	10,35	0,47	3,42	1,78	2,91	0,76	3,46	7,55	21,52	0,23	3,48
700	4,79	6,67	0,22	3,97	1,80	3,12	0,31	4,10	7,37	16,22	0,60	4,06
800	4,79	6,92	0,19	4,60	1,78	6,20	0,67	4,62	7,38	15,89	0,57	4,71
900	4,79	9,27	0,58	5,12	1,78	3,43	0,69	5,23	7,37	25,93	0,20	5,26
1k	4,81	10,97	0,30	5,65	1,78	4,91	0,62	5,90	7,47	29,68	0,41	5,81
OPT	4,79	-	-	871	1,78	-	-	875	7,37	-	-	877

Samples	$z = 6$ Cost			Time	$z = 7$ Cost			Time
	Min	Avg	Var/Avg^2	Avg	Min	Avg	Var/Avg^2	Avg
100	3,40	5,38	0,20	0,63	1,59	2,28	0,32	0,61
200	3,32	6,22	0,41	1,20	1,60	2,72	0,46	1,19
300	3,51	11,47	0,91	1,77	1,59	6,20	1,21	1,82
400	3,35	6,54	0,71	2,33	1,59	8,70	4,12	2,36
500	3,54	8,22	1,43	2,89	1,61	2,42	0,33	2,92
600	3,32	6,33	0,85	3,47	1,61	14,01	2,25	3,50
700	3,32	7,86	0,93	4,06	1,60	6,53	2,87	4,10
800	3,35	11,10	2,25	4,63	1,61	5,06	0,76	4,70
900	3,31	8,46	0,29	5,21	1,59	7,49	0,54	5,28
1k	3,34	7,89	0,57	5,98	1,59	2,35	0,16	5,78
OPT	3,31	-	-	885	1,59	-	-	882

Figure 1: Overview of cost and running time for the sublinear algorithm on the Road Networks data set. Costs scaled by a factor 10^9 for $z = 3$, 10^{11} for $z = 4$ and $z = 5$, 10^{14} for $z = 6$ and 10^{16} for $z = 7$. The variance was extremely small for running times, so we omit it. Running time is given in seconds. The running time for the sampling algorithms only considers the time required to sample the points, prune the data set, and run the optimization, i.e. the time required to evaluate the computed solution on the entire data set is not included.

Samples	$z = 3$ Cost		Time	$z = 4$ Cost		Time	$z = 5$ Cost		Time
	Min	Avg	Avg	Min	Avg	Avg	Min	Avg	Avg
100	1,306	1,310	6,03	1,907	1,910	7,03	1,527	1,560	7,22
200	1,306	1,311	6,03	1,907	1,911	7,04	1,518	1,554	7,81
300	1,305	1,310	6,30	1,907	1,908	7,59	1,523	1,560	8,65
400	1,305	1,310	6,48	1,907	1,909	7,74	1,521	1,544	9,06
500	1,306	1,309	6,74	1,906	1,908	7,92	1,512	1,547	9,84
600	1,305	1,308	7,06	1,907	1,908	8,21	1,516	1,553	10,39
700	1,307	1,309	7,20	1,907	1,908	8,35	1,516	1,545	11,14
800	1,306	1,309	7,53	1,907	1,908	8,82	1,528	1,547	11,86
900	1,306	1,310	7,60	1,907	1,909	8,98	1,523	1,553	12,43
1k	1,307	1,312	8,04	1,906	1,909	9,6	1,526	1,550	13,26
OPT	1,125	-	5544	1,296	-	5586	1,499	-	5934

Figure 2: Overview of cost and running time for the sublinear algorithm on the USCensus data set. Costs scaled by a factor 10^{12} for $z = 3$, 10^{14} for $z = 4$ and 10^{16} for $z = 5$. The variance was extremely small for all values (cost and running time), as indicated by the small gaps between minimum and average. We therefore omitted it from the table. The largest variance (relative to the squared cost) we encountered was for $z = 5$ and 600 samples, where it was still below 0.0005. Running time is given in seconds. The running for the sampling algorithms only considers the time required to sample the points, prune the data set, and run the optimization, i.e. the time required to evaluate the computed solution on the entire data set (which vastly exceeds the given time bounds) is not included.