Efficient Coreset Constructions via Sensitivity Sampling

Vladimir Braverman VOVA@CS.JHU.EDU
Johns Hopkins University

Dan Feldman DANNYF.POST@GMAIL.COM
University of Haifa

Harry Lang HLANG08@GMAIL.COM
MIT

Adiel Statman STATMAN.ADIEL@GMAIL.COM
University of Haifa

Samson Zhou SAMSONZHOU@GMAIL.COM
Carnegie Mellon University

Editors: Vineeth N Balasubramanian and Ivor Tsang

Abstract

A coreset for a set of points is a small subset of weighted points that approximately preserves important properties of the original set. Specifically, if P is a set of points, Q is a set of queries, and $f : P \times Q \to \mathbb{R}$ is a cost function, then a set $S \subseteq P$ with weights $w : P \to [0,\infty)$ is an ϵ-coreset for some parameter $\epsilon > 0$ if $\sum_{s\in S} w(s)f(s,q)$ is a $(1+\epsilon)$ multiplicative approximation to $\sum_{p\in P} f(p,q)$ for all $q \in Q$. Coresets are used to solve fundamental problems in machine learning under various big data models of computation. Many of the suggested coresets in the recent decade used, or could have used a general framework for constructing coresets whose size depends quadratically on the total sensitivity t.

In this paper we improve this bound from $O(t^2)$ to $O(t\log t)$. Thus our results imply more space efficient solutions to a number of problems, including projective clustering, k-line clustering, and subspace approximation. The main technical result is a generic reduction to the sample complexity of learning a class of functions with bounded VC dimension. We show that obtaining an (ν,α)-sample for this class of functions with appropriate parameters ν and α suffices to achieve space efficient ϵ-coresets.

Our result implies more efficient coreset constructions for a number of interesting problems in machine learning; we show applications to k-median/k-means, k-line clustering, j-subspace approximation, and the integer (j,k)-projective clustering problem.

Keywords: Dimensionality reduction, coresets, sensitivity sampling

1. Introduction

Coresets are an important technique in machine learning, data sciences, and statistics for representing a large dataset with a much smaller amount of memory. Coresets are often used as a pre-processing dimensionality technique to improve the downstream efficiency of algorithms, both space and time. Informally speaking, a coreset S of an input set P of underlying points $p_1,\dots,p_n$ is a smaller number of weighted representatives of P that can be used to approximate the cost of any query from a set of a given queries. For example, in the common k-means clustering problem, the coreset must approximate $\sum_{i=1}^{n} d(p_i,C)^2$ for every query C, where C is a set of k points and $d(p_i,C)$

is taken to be the smallest Euclidean distance from p_i to any point in C. Thus to use a coreset S to approximately solve the k-means clustering problem, it suffices to find the optimal clustering on S rather than find the optimal clustering on P. Because the size of S is much smaller than the size of P, i.e., $|S| \ll |P|$, then finding an optimal clustering on S instead of P will be much more efficient.

More generally, coreset is a set of points P' with corresponding weight function $w(\cdot)$ such that $\sum_{p'_i \in P'} w(p') d(p'_i, C)^2$ is a $(1 \pm \epsilon)$ approximation to $\sum_{i=1}^{n} d(p_i, C)^2$. Coresets have been extensively studied in k-means clustering Badoiu et al. (2002); Har-Peled and Mazumdar (2004); Frahling and Sohler (2005, 2008); Feldman and Langberg (2011); Feldman and Schulman (2012); Braverman et al. (2019); Huang and Vishnoi (2020); Feldman et al. (2020), subspace approximation Deshpande et al. (2006); Deshpande and Varadarajan (2007); Feldman and Langberg (2011); Feldman et al. (2010a, 2013); Clarkson and Woodruff (2015); Sohler and Woodruff (2018), and a number of other geometric problems and applications Agarwal et al. (2006); Feldman et al. (2006); Clarkson (2008); Dasgupta et al. (2008); Ackermann and Blömer (2009); Phillips and Tai (2018); Huang et al. (2018); Assadi et al. (2019); Munteanu et al. (2018); Braverman et al. (2018); Mussay et al. (2020), due to the increasing availability of big data and the necessity for scalable methods to process this information.

The most common algorithmic procedure to designing a coreset is the following simple template. An algorithm first approximately evaluates the *sensitivity* of each point in the dataset. Informally, the sensitivity of a point quantities how important or distinct that point is, with respect to the given objective function on which we would like to optimize. Approximating the sensitivity of each point can often be done efficiently, so that the time to construct a coreset is often a lower order term compared to the runtime of the post-processing algorithm. The template then samples a fixed number of points, so that each point in the dataset with probability proportional to the sensitivity of the point. This approach is called sensitivity sampling and the fixed number of points is often a monotonically increasing function of the total sensitivity, defined to be the sum of the sensitivities of each point. Hence, if the numbered of sampled points is much smaller than the number of input points, this approach allows for compact dimensionality reduction, leading to improved performance of post-processing algorithms.

Since the total sensitivity is a central quantity to coreset techniques, the total sensitivity for various objective functions has been well-studied and completely characterized in some cases. However, it is not quite known what the optimal dependency between the total sensitivity and the size of the coreset should be; that is, what is optimal monotonically increasing function of the total sensitivity that governs the number of sampled points? Clearly smaller functions lead to smaller coresets, which lead to more efficient post-processing functions. Many recent coreset constructions in the past decade require constructing coresets whose size depends quadratically on the total sensitivity. In this paper, we show this dependency is not optimal; we introduce in Theorem 1 a generic construction whose dependency on the total sensitivity t is only $O(t \log t)$ rather than $O(t^2)$ (Feldman and Langberg, 2011). Because the dependency is already black-boxed into the design of many coreset constructions, our results automatically improve many existing coreset algorithms simply by lowering the number of required samples, without modifying any other property of the algorithm; we are only showing that the worst-case theoretical guarantee of these algorithms is significantly and universally better than previously thought.

1.1. Our Contributions

We show that the common sensitivity sampling framework only needs to sample $O(t\log t)$ points, where t is the total sensitivity.

Theorem 1 *Let d be the dimension of a query space (P,w,Q,f). For each point p, let $m(p)$ be an upper bound on the sensitivity of point p. Let $t=\sum_{p\in P}m(p)$, and $\varepsilon,\delta\in(0,1)$. Then by sampling $O\left(\frac{t}{\varepsilon^2}\left(d\log t+\log\left(\frac{1}{\delta}\right)\right)\right)$ i.i.d. points from P and rescaling each sampled point by $\frac{1}{m(p)}$, the resulting sample is an ϵ-coreset for P.*

In contrast, previous analysis showed that the sensitivity sampling framework required $O(t^2)$ points Feldman and Langberg (2011). We emphasize that our results are purely theoretical; we show that any worst-case guarantee that could previously be achieved with $O(t^2)$ samples can actually be achieved with only $O(t\log t)$ samples. Hence our results can be universally plugged into any existing coreset construction algorithm simply by requiring a lower number of samples. Moreover, our results are optimal, since it can be shown by standard coupon-collector arguments that $\Omega(t\log t)$ samples are necessary in some cases.

Our analysis is simple and uses results from the sample complexity of learning functions from unknown distributions Li et al. (2001). Namely, let X be a domain with an unknown underlying probability distribution P and let $\mathcal{F}$ be a possibly infinite set of real-valued functionswith domain X. Suppose an algorithm is given independent samples $x=(x_1,...,x_m)$ from P and oracle access to any $f\in\mathcal{F}$. Then how many samples must an algorithm observe before the sample average $\hat{\mathbb{E}}_x(f):=\frac{1}{m}\sum_{i=1}^m f(x_i)$ is a rough approximation to $\mathbb{E}_P(f)=\underset{x\sim P}{\mathbb{E}} f(x)$? Note that since $\hat{\mathbb{E}}_x(f)$ is an unbiased estimator for $\mathbb{E}_P(f)$, then standard concentration inequalities or variance bounding techniques demonstrate that the empirical average is roughly equal to the actual expectation. Surprisingly, Li et al. (2001) proved that if $\mathcal{F}$ has pseudo-dimension d, then $O\left(\frac{1}{\alpha^2\nu}\log\frac{1}{\nu}\right)$ samples can *simultaneously* obtain a good estimate to the expectation of all functions in $\mathcal{F}$ with constant probability.

We show that the results of Li et al. (2001) also succeed if the VC-dimension of $\mathcal{F}$ is d, rather than the pseudo-dimension. This implies that the algorithm outputs a (p,ϵ)-approximation of all functions in $\mathcal{F}$, which means if the function is too small, then the resulting data structure can only provide an additive error guarantee rather than a multiplicative relative error guarantee, but if the function is adequately large, then the resulting data structure provides a multiplicative error guarantee.

Fortunately, we show this guarantee suffices to obtain an ε-coreset. We break the query space into partitions, based on how much a point contributes to a query, compared to the total contribution to a query across all the points. If the contribution of a partition is large, then our (p,ϵ)-approximation guarantees a good approximation to this partition. Now if the contribution of a partition is small, then two things can happen. Either it is possible that the sum of the contributions of all of the “small” partitions is large, in which case our (p,ϵ)-approximation again guarantees a good approximation, or the sum of the contributions remains insignificant. In this case, we only have an additive approximation of the contributions for these points, but because the sum of the contributions is insignificant, an additive error on these points translates to a small relative error on the entire objective function.

Applications, Generalizations, and Empirical Evaluations. Theorem 1 has applications to many problems in machine learning. In Section 3, we describe applications to model fitting problems. Specifically, we consider the (j,k)-projective clustering problems such as k-median/k-means, k-line clustering, j-subspace approximation, and the integer (j,k)-projective clustering problem.

Informally, the goal is to find a model F in a restricted family $\mathcal{F}$ of set of k-tuples of affine j-subspaces that minimizes $\sum_{p\in P}d(p,F)$, where P is a set of input points. Here, F is the union of k j-flats so that if $j=0$, then each j-flat reduces to a point and the (j,k)-projective clustering problem becomes the k-median problem with the appropriate metric. On the other hand, with an alternate distance function (the squared Euclidean distance), $d(\cdot,\cdot)$, the (j,k)-projective clustering objective becomes the k-means problem. When $j=1$ and k is fixed, the objective becomes the k-line clustering problem but if j is fixed and $k=1$, then the objective instead becomes the subspace approximation problem. Finally, in the integer (j,k)-projective clustering problem, all points in P are assumed to have integer coordinates from some predetermined range. Our results subsume earlier versions online that have not received independent verification and are summarized in Figure 1.

Problem	Coreset Size (Theorem 17)	Previous Coreset Size	Reference
Integer (j,k)-projective clustering	$\tilde{O}\left(\frac{d}{\varepsilon^2}g(d,j,k)(\log n)^{g(d,j,k)}\right)$	$\tilde{O}\left(\frac{d}{\varepsilon^2}g(d,j,k)(\log n)^{2\cdot g(d,j,k)}\right)$ Feldman and Langberg (2011); Varadarajan and Xiao (2012a)	Thm. 20
k-line center	$\tilde{O}\left(\frac{d}{\varepsilon^2}f(k)k^{f(k)}\log n\right)$	$\tilde{O}\left(\frac{d}{\varepsilon^2}k^{2\cdot f(k)}\log^2 n\right)$ Feldman and Langberg (2011); Varadarajan and Xiao (2012b)	Thm. 23
k-median	$O\left(\frac{d}{\varepsilon^2}k\log k\right)$	$O\left(\frac{d}{\varepsilon^2}k^2\right)$ Feldman and Langberg (2011); Varadarajan and Xiao (2012b)	Thm. 25
k-means	$O\left(\frac{d}{\varepsilon^2}k\log k\right)$	$O\left(\frac{d}{\varepsilon^2}k^2\right)$ Feldman and Langberg (2011); Varadarajan and Xiao (2012b)	Thm. 25
j-subspace fitting	$\tilde{O}\left(\frac{ds(j,d)}{\varepsilon^2}\right)$	$\tilde{O}\left(\frac{ds^2(j,d)}{\varepsilon^2}\right)$ Feldman and Langberg (2011); Varadarajan and Xiao (2012b)	Thm. 27

Figure 1: Coreset sizes achieved by Theorem 17 for various unsupervised learning problems. f,g,s are functions independent of the input size n, specific to the problem setting. $\tilde{O}(\cdot)$ omits lower order terms.

Although our primary contribution is theoretical, we complement our worst-case guarantees with empirical evaluations on both small and large-scale datasets, which we describe in Section 4. Finally, we give a number of tighter bounds for k-clustering in the appendix, which may be skipped for general applications.

Subsequent work. Following our work, a number of subsequent coreset constructions have been proposed. For k-median, Sohler and Woodruff (2018) showed a coreset construction with size $O(\varepsilon^{-4}k^2\log k)$. Their techniques used a new dimensionality reduction result in combination with existing coreset constructions to show that the optimal coreset size of k-median can be independent of the dimension d. Similarly, Becchetti et al. (2019) gave coreset constructions with size $O(\varepsilon^{-4}k^2\log k)$ for k-means clustering using by introducing a data-dependent random projection and showing that the random projection is a terminal embedding. Finally, Huang and Vishnoi (2020) showed that importance sampling is essentially optimal, by giving coresets with size $O(\min\{\varepsilon^{-2z-2},2^{2z}\varepsilon^{-4}k\}\cdot k\log k\log(k/\varepsilon))$ for (k,z)-clustering, where $z=1$ for k-median and $z=2$ for k-means. It should be noted that since these methods focus on reducing the dimension d of the input space, their methods can be used in combination with our constructions to obtain coresets with size independent of d.

1.2. Preliminaries

For an integer $d\geq 1$, we denote by $\mathbb{R}^d=\mathbb{R}^{d\times 1}$ the set of column vectors in $\mathbb{R}^d$, and $[d]=\{1,\cdots,d\}$. For $\varepsilon>0$, we denote by $1\pm\varepsilon$ the interval $[1-\varepsilon,1+\varepsilon]$. A multiplication of a real number $c\in\mathbb{R}$ by a set $X\subseteq\mathbb{R}^d$ is denoted by $cX=\left\{cx\,|\,x\in\mathbb{R}^d\right\}$. A sum or minimum over an empty set is defined to be 0 in this paper. We say a set of random variables are i.i.d. if the random variables are independent and identically distributed. We use $\tilde{O}(\cdot)$ to suppress lower order terms, e.g. $\tilde{O}\left(\frac{1}{\varepsilon^2}\log n\right)$ omits $\mathrm{polylog}\left(\frac{1}{\varepsilon}\right)$ and $\mathrm{polyloglog} n$ terms.

Definition 2 (Weighted Set) *Let X be called a* ground set*. Let $P \subseteq X$ be a (possibly ordered) multi-set and $w: P \to [0,\infty)$ be a function that maps every $p \in P$ to a weight $w(p) \geq 0$. The pair (P,w) is called a* weighted set *in X. If $w(p)=1$ for every $p \in P$ then the (un)weighted set (P,w) may be denoted by P for short.*

The order of the points in P can be arbitrary in this paper. However, even if P contains only a single copy of each point, the corresponding coreset may contain multiple instances of some points. Hence, we consider coresets as multi-sets, although duplicated points can usually be replaced by a single weighted point without changing the claimed results. The union and intersection are also implied to be over multi-sets in this paper.

Definition 3 (Query and Range Space) *Let X be a ground set and (P,w) be a weighted set in X called the* input set*. Let Q be a* query function *that maps every set $S \subseteq X$ to a corresponding $Q(S)$, such that $Q(T) \subseteq Q(S)$ for every $T \subseteq S$. Let $f: X \times Q(X) \to \mathbb{R}$ be called a* loss function*. The tuple (P,w,Q,f) is called a* query space*. For a collection $\mathcal{R}$ of subsets of X, we call $(X,\mathcal{R})$ a* range space*. For every weighted set $C' = (C,u)$ in X, and every $q \in Q$ we define the overall fitting error of C' to q by $f(C',q) = \sum_{p \in C} u(p) f(p,q)$.*

For example, in the k-means clustering problem on $\mathbb{R}^n$, the ground set X is the domain $\mathbb{R}^n$ that contains is the set P of input points. The query function for k-means clustering restricts queries of $\mathbb{R}^n$ to k points and the loss function is the squared Euclidean distances, so that when the arguments of the loss function are a point from P and a point from the query function, the loss function is the squared distance from the input point to the closest center.

Usually the loss function has specific properties such as being a pseudo distance function D, as will be defined later. However, it may also be more complicated such as a subtraction between pseudo distance functions, which will also be used in this paper. This is also why it may return a negative number. In general, we will be interested in approximating $\sum_{p \in P} w(p) f(p,q)$ for every query q in the query space up to an additive error of ε.

For a set X, a query function Q, and a cost function f, we define the VC-dimension of the range space that it induced, as defined below. The classic VC-dimension was defined for sets and subset and here we generalize it to query spaces, following Feldman and Langberg (2011).

Definition 4 (VC-dimension) *Vapnik and Chervonenkis (1971); Feldman and Langberg (2011) For a ground set X and a set* ranges *of subsets of X, the VC-dimension of (X, ranges) is the size $|C|$ of the largest subset $C \subseteq X$ such that $|\{C \cap \text{range} \mid \text{range} \in \text{ranges}\}| = 2^{|C|}$. Let Q be a query function and $f: X \times Q(X) \to \mathbb{R}$. For every $q \in Q(X)$, and $r \in \mathbb{R}$ we define the sets*

$$\begin{aligned} \text{range}_{P,f}(q,r) &:= \{p \in P \mid f(p,q) \leq r\}, \\ \text{ranges}(P,Q,f) &:= \{C \cap \text{range}_{P,f}(q,r) \mid C \subseteq P, q \in Q(C), r \in \mathbb{R}\}. \end{aligned}$$

The dimension *of (P,Q,f) is the VC-dimension of $(P, \text{ranges}(P,Q,f))$.*

The following definition of sensitivity is central to our paper, as we shall show that the coreset size of our algorithm is proportional to the total sensitivity of the input set.

Definition 5 (Sensitivity) *Let (P,w,Q,f) be a query space over a ground set X, where $w:P\to[0,\infty)$ and $f:P\times Q(P)\to[0,\infty)$. Then we define the* sensitivity *of a point $p\in P$ by $s(p)=\sup_{C\in Q}w(p)|f(p,C)|$.*

Then the total sensitivity of an input set is the natural definition:

Definition 6 (Total Sensitivity) *Let (P,w,Q,f) be a query space over a ground set X, where $w:P\to[0,\infty)$ and $f:P\times Q(P)\to[0,\infty)$. We define the* total sensitivity *of P by $\sum_{p\in P}s(p)$, where $s(p)$ is the sensitivity of p.*

We next define two related concepts, the (ν,α)-samples and relative (p,ε)-approximations.

Definition 7 ((ν,α)-Sample) *(Li et al., 2001) Let $\alpha,\nu>0$. For every $a,b\geq 0$, we define the distance function $d_\nu(a,b)=\frac{|a-b|}{a+b+\nu}$. Let (P,w,Q,f) be a query space over a ground set X, where $w:P\to[0,\infty)$ and $f:P\times Q(P)\to[0,\infty)$. Then the weighted set (S,u) is called a (ν,α)-*sample *for (P,w,Q,f) if (S,u,Q,f) is a query space, and for every $q\subseteq Q(S)$, $d_\nu(\overline{f}(P,w,q),\overline{f}(S,u,q))\leq\alpha$, where $\overline{f}(P,w,q)=\frac{\sum_{p\in P}|w(p)\cdot f(p,q)|}{\sum_{p\in P}w(p)}$.*

Definition 8 ((p,ε)-Approximation) *(Har-Peled and Sharir, 2011) Let $0<p,\varepsilon<1$. Let (P,w,Q,f) be a query space over a ground set X, where $w:P\to[0,\infty)$ and $f:P\times Q(P)\to[0,\infty)$. Then the weighted set (S,u) is called a (ν,α)-*sample *for (P,w,Q,f) if (S,u,Q,f) is a query space, and for every $q\subseteq Q(S)$,*

1. *$(1-\varepsilon)\overline{f}(P,w,q)\leq\overline{f}(S,u,q)\leq(1+\varepsilon)\overline{f}(P,w,q)$, for $\overline{f}(P,w,q)\geq p$*
2. *$\overline{f}(P,w,q)-\varepsilon p\leq\overline{f}(S,u,q)\leq\overline{f}(P,w,q)+\varepsilon p$, for $\overline{f}(P,w,q)\leq p$,*

where $\overline{f}(P,w,q):=\frac{\sum_{p\in P}|w(p)\cdot f(p,q)|}{\sum_{p\in P}w(p)}$.

We assume throughout, without loss of generality, that $\sum_{p\in P}w(p)=1$ and recall the equivalence between (ν,α)-samples and relative (p,ε)-approximations:

Theorem 9 *(Har-Peled and Sharir, 2011) Let $(X,\mathcal{R})$ be a range space. If $(Z,\mathcal{R})$ is a (ν,α)-sample for $(X,\mathcal{R})$ with $0<\alpha<\frac{1}{4}$ and $\nu>0$, then Z is a relative $(\nu,4\alpha)$-approximation for $(X,\mathcal{R})$. Moreover, if $d_\nu(a,b)\leq\frac{\varepsilon}{4}$ and $\nu\leq\frac{1}{4}$, then $|a-b|\leq\varepsilon$.*

Definition 10 (ε-coreset) *Let $P'=(P,w)$ be a weighted set in X, and $\varepsilon>0$ be an approximation error. The weighted set $C'=(C,u)$ in X is an ε-*coreset *for a query space (P',Q,f) if for every $q\in Q(C)$ we have $(1-\varepsilon)\sum_{p\in C}u(p)f(p,q)\leq\sum_{p\in P}w(p)f(p,q)\leq(1+\varepsilon)\sum_{p\in C}u(p)f(p,q)$.*

2. Sensitivity Sampling

In this section, we show that provable worst-case guarantees for constant factor approximation can be achieved using the sensitivity sampling framework to construct coresets of size $O(t\log t)$, where t is the total sensitivity[1] This improves on previous analysis that the sensitivity sampling framework

[1] We also achieve optimal dependence on $\frac{1}{\epsilon}$ for a $(1+\epsilon)$-approximation, but we omit these factors for ease of discussion.

to sample $O(t^2)$ points to construct coresets that guaranteed constant factor approximation (Feldman and Langberg, 2011). We remark that our result is purely theoretical and does not require novel algorithmic implementation. Instead, our result shows that the parameters in existing coreset construction algorithms can be improved while still guaranteeing worst-case performance.

Recall that the sensitivity of a point $p \in P$ is defined by $s(p) = \sup_{C \in Q(P)} w(p)|f(p,C)|$, where $w : P \to [0,\infty)$ is a weight function p and $f : P \times Q(P) \to [0,\infty)$ is a loss function between an input point and a query set. However, determining the exact sensitivity of a point can be time-consuming, so we instead define $m(p) \geq s(p)$ to be an upper bound on the sensitivity. It turns out that upper bounds $m(p)$ that are within a constant factor approximation of the exact sensitivity of a point are often efficiently computable. Then $t := t(P) = \sum_{p \in P} m(p)$ is an upper bound on the total sensitivity, which is the sum of the sensitivities of all points in P.

We now formalize the sensitivity sampling framework broadly used in algorithmic design. We form a sample S by picking the first point of S to be $p \in P$ with probability $\frac{m(p)}{t(P)}$ and reweighting the sampled point with the inverse of the sampling probability. We show that repeatedly sampling points from P with replacement until S has $\frac{ct}{\varepsilon^2}\left(d\log t + \log\left(\frac{1}{\delta}\right)\right)$ points suffices to obtain an ε-coreset for P with probability $1-\delta$ if the underlying query space has VC dimension d. The sensitivity sampling framework appears in full in Algorithm 0.

```
Algorithm 0: Basic_Coreset($P,d,\varepsilon,\delta$)
Input: Set of $n$ points $P$ in a query space with VC dimension $d$, approximation parameter $\varepsilon > 0$,
failure probability $\delta > 0$, oracle access to upper bound $m(\cdot)$ on sensitivities.
Output: $\varepsilon$-coreset of $P$. $t \leftarrow 0$
for every $p \in P$ do
    Let $m(p)$ be an upper bound on the sensitivity $s(p)$ of $p$.
    $t \leftarrow t + m(p)$
end
$S \leftarrow \emptyset$
$N \leftarrow \frac{ct}{\varepsilon^2}\left(d\log t + \log\left(\frac{1}{\delta}\right)\right)$ for sufficiently large constant $c > 0$ that can be determined from the proof.
for $i = 1$ to $N$ do
    With probability $\frac{m(p)}{t}$, set $x = p$ for $p \in P$ with weight $\frac{t}{m(p)}$.
    $S \leftarrow S \cup \{x\}$
end
return $S$
```

2.1. Sample Complexity of Learning

We first consider the problem of relating the sample complexity of learning a class of functions to its VC dimension. We then show this implies an ε-coreset construction under certain parameters. Let X be a domain with probability distribution μ and let $\mathcal{F}$ be a possibly infinite set of real-valued functions defined on X. Given access to samples $x = (x_1,\ldots,x_N)$ independently drawn from μ and oracle access to any $f \in \mathcal{F}$, the sample average $\hat{\mathbb{E}}_x(f) := \frac{1}{N}\sum_{i=1}^{N} f(x_i)$ serves as an unbiased estimator to the expectation of f, denoted $\mathbb{E}_\mu(f) = \mathop{\mathbb{E}}_{x \sim \mu} f(x)$. For sufficiently large N, standard concentration inequalities can show that the sample average $\hat{\mathbb{E}}_x(f)$ is an α-approximation to the expectation $\mathbb{E}_\mu(f)$, where $0 < \alpha \leq 1$. However, it is not obvious whether there exists a value of N that allows simultaneous estimation of the expectation of *all* functions in $\mathcal{F}$.

We first recall the following definition of pseudo-dimension:

Definition 11 (Pseudo-dimension) *The* pseudo-dimension *of a class $\mathcal{F}$ of functions from a domain X to $[0,1]$ is defined to be the largest d such that there exists a sequence $x_1,...,x_d \in X$ and a sequence $r_1,...,r_d \in \mathbb{R}$ of thresholds such that for all 2^d combinations of $b_1,...,b_d \in \{0,1\}$, there exists an $f \in \mathcal{F}$ such that for all $i \in [d]$, $f(x_i) \geq r_i$ if and only if $b_i = 1$.*

Li et al. (2001) show that if $\mathcal{F}$ has pseudo-dimension d, then $O\left(\frac{1}{\alpha^2\nu}\log\frac{1}{\nu}\right)$ samples suffices to simultaneously obtain an (ν,α)-sample to expectation of all functions in $\mathcal{F}$ with constant probability. Namely, Li et al. (2001) show the following two lemmas:

Lemma 12 (Lemma 6 in Li et al. (2001)) *Let $\mathcal{F}$ be a set of functions from X to $[0,1]$, μ be a probability distribution over X and $\nu > 0$, $0 < \alpha < 1$, and $N \geq \frac{2}{\alpha^2\nu}$. For any integer $N > 0$, let Γ_N denote the set of all permutations of $\{1,...,2N\}$ so that for each $i \leq N$, either i and $N+i$ are fixed, or i and $N+i$ are swapped. Let U be the uniform distribution over Γ_m. Then*

$$\mathop{\mathbf{Pr}}_{N}\left[x : \exists f \in \mathcal{F}, d_\nu(\hat{\mathbb{E}}_x(f), \mathbb{E}_\mu(f)) > \alpha\right] \leq 2 \sup_{x \in \mathbb{X}^{2N}} U\Big\{\sigma :$$

$$\exists f \in \mathcal{F}, d_\nu\left(\frac{1}{N}\sum_{i=1}^{N} f(x_{\sigma(i)}), \frac{1}{N}\sum_{i=1}^{N} f(x_{\sigma(Nn+i)})\right) > \frac{\alpha}{2}\Big\}.$$

Lemma 13 (Lemma 10 in Li et al. (2001)) *Let d be the pseudo-dimension of $F \subseteq [0,1]^{2N}$, where $N \geq \frac{125(2d+1)}{\alpha^2\nu}$ for any $\alpha,\nu > 0$. Let U be the uniform distribution over Γ_N. Then*

$$U\left\{\sigma : \exists f \in F, d_\nu\left(\frac{1}{N}\sum_{i=1}^{N} f(x_{\sigma(i)}), \frac{1}{N}\sum_{i=1}^{N} f(x_{\sigma(N+i)})\right) > \alpha\right\} \leq 6\left(\frac{2624}{\nu}\right)^d e^{-\frac{\alpha^2\nu N}{90}}.$$

Observe that combining Lemma 12 and Lemma 13 and solving for N recovers the bound from Li et al. (2001) of $O\left(\frac{1}{\alpha^2\nu}\left(d\log\frac{1}{\nu} + \log\frac{1}{\delta}\right)\right)$. We need an analog of their sampling result for VC-dimension rather than pseudo-dimension. As it turns out, the only place Li et al. (2001) uses pseudo-dimension in Lemma 13 is a black-box reduction from the following lemma to bound the size of $\mathcal{F}$:

Lemma 14 *(Haussler, 1995; Li et al., 2001) For $v,w \in \mathbb{R}^k$, let $\ell_1(v,w) = \frac{1}{k}|v_i - w_i|$. Let $k > 0$ be an integer and $0 < \alpha \leq 1$. Suppose each pair of distinct elements $f,g \in F \subseteq [0,1]^k$ has $\ell_1(f,g) \leq \alpha$. Then $|F| \leq \left(\frac{41}{\alpha}\right)^d$, where d is the pseudo-dimension of F.*

Moreover, Haussler (1995) proved the exact same statement when d is the VC-dimension of F, rather than the pseudo-dimension.

Lemma 15 *(Haussler, 1995) For $v,w \in \mathbb{R}^k$, let $\ell_1(v,w) = \frac{1}{k}\sum_{i=1}^{k}|v_i - w_i|$. Let $k > 0$ be an integer and $0 < \alpha \leq 1$. Suppose each pair of distinct elements $f,g \in F \subseteq [0,1]^k$ has $\ell_1(f,g) \leq \alpha$. Then $|F| \leq \left(\frac{41}{\alpha}\right)^d$, where d is the VC-dimension of F.*

Specifically, Lemma 15 follows from Corollary 1 in Haussler (1995) because $e(d+1)(2e/\varepsilon)^d \leq (2e^3/\varepsilon)^d < (41/\varepsilon)^d$.

Thus by using Lemma 15 rather than Lemma 14, we can recover Lemma 13 using VC-dimension rather than pseudo-dimension in the following formulation of Theorem 16. Hence, we can relate the sampling complexity of learning a class of functions to their VC-dimension:

Theorem 16 *Let $\delta \in (0,1)$ be a failure probability and $\mathcal{F} : X \to [0,1]$ be a family of functions with VC dimension d. Then with probability at least $1-\delta$, $N = O\left(\frac{1}{\alpha^2 \nu}\left(d \log \frac{1}{\nu} + \log \frac{1}{\delta}\right)\right)$ samples suffice to simultaneously obtain an (ν,α)-sample to any $f \in \mathcal{F}$, i.e., $d_\nu(\hat{\mathbb{E}}_x(f), \mathbb{E}_\mu(f)) < \alpha$.*

2.2. Reduction to ε-Coresets

We now show that an (ν,α)-sample to a class of functions $\mathcal{F}$ suffices to achieve an ε-coreset under the appropriate parameters. The proof partitions the points in an input set P by their contribution to $\sum_{p \in P} w(p) f(p,X)$ for some X in the query space. A subset S_i that contributes a large fraction towards $\sum_{p \in P} w(p) f(p,X)$ will be well-estimated by the (ν,α)-sample. On the other hand, if S_i is not well-estimated by the (ν,α)-sample, then its contribution towards $\sum_{p \in P} w(p) f(p,X)$ must be small, so that intuitively, the additive error from the (ν,α)-sample is also small. Thus we can show that the sample is actually an ε-coreset.

Theorem 17 (Theorem 1, Restated) *Let d be the dimension of a query space (P,w,Q,f). Suppose that $m : P \to [0,\infty)$ such that $m(p) \geq \sup_{C \in Q(P)} w(p)|f(p,C)|$. Let $t \geq \sum_{p \in P} m(p)$, and $\varepsilon, \delta \in (0,1)$. Let $c \geq 1$ be a sufficiently large constant, and let S be a sample of*

$$|S| \geq \frac{ct}{\varepsilon^2}\left(d \log t + \log\left(\frac{1}{\delta}\right)\right)$$

i.i.d. points from P, where for every $p \in P$ and $s \in S$ we have $\mathrm{Prob}(p = s) \geq \frac{m(p)}{t}$. Then, with probability at least $1-\delta$, then simultaneously for all $C \in Q(S)$,

$$\left|\sum_{p \in P} w(p) f(p,C) - \sum_{p \in S} \frac{w(q)}{|S| \mathrm{Prob}(q)} \cdot f(p,C)\right| \leq \varepsilon \left|\sum_{p \in P} w(p) f(p,C)\right|.$$

3. Applications

Our theoretical worst-case guarantee has a wide range of applications due to the prevalence of the coreset technique and how well-studied the total sensitivity is of various optimization problems. Note that for any problem whose sensitivity is known to be t, Theorem 1 gives an improvement on dependency of t from $O(t^2)$ to $O(t \log t)$ for the number of sampled points. Varadarajan and Xiao (2012b) considers sensitivity sampling for shape fitting problems, focusing on (j,k)-projective clustering problems, such as k-median/k-means, k-line clustering, j-subspace approximation, and the integer (j,k)-projective clustering problem. We show that our results imply more efficient coreset constructions using the total sensitivity bounds on these problems obtained by Varadarajan and Xiao (2012b).

Definition 18 ((j,k)-projective clustering problem) *Given integers $j,k \geq 0$, a distance function $\mathrm{dist} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, and a set of points $P \subseteq \mathbb{R}^d$, the goal is to find a shape F in the family of shapes $\mathcal{F}$ that minimizes $\sum_{p \in P} \mathrm{dist}(p,F)$, where the family of shapes $\mathcal{F}$ is the set of k-tuples of affine j-subspaces. Hence, F is the union of k j-flats.*

In particular if $j = 0$, then each j-flat is just a point, so the (j,k)-projective clustering problem with the distance function $\mathrm{dist}(\cdot,\cdot)$ being the Euclidean distance reduces to the k-median problem, while if $\mathrm{dist}(\cdot,\cdot)$ is the squared Euclidean distance, then the objective becomes the k-means problem. If $j=1$, then each j-flat becomes a line, so that the objective becomes the k-line clustering problem. On the other hand, if j is fixed and $k=1$, the (j,k)-projective clustering problem becomes the subspace approximation problem, which can be parametrized by the distance function. Finally, in the *integer* (j,k)-projective clustering problem, all points in P are assumed to have integer coordinates from the range $[-n^\gamma,n^\gamma]$ for some constant $\gamma>0$.

For the k-line center problem and the integer general (j,k)-projective clustering problem, Varadarajan and Xiao (2012a) showed the following upper bounds on the total sensitivity:

Theorem 19 *(Varadarajan and Xiao, 2012a) Let the distance function* dist *used in projective clustering be the z-th power of the Euclidean distance for some fixed $z\in(0,\infty)$. Then the total sensitivity t satisfies $t=O(k^{f(d,k)}\log n)$ for $j=1$, i.e., the k-line center problem and $t=O((\log n)^{g(d,j,k)})$ for the integer (j,k)-projective clustering problem, where f and g are fixed functions that only depend on d,k and d,j,k, respectively.*

Then Theorem 19 and Theorem 1 together imply efficient coresets for both the k-line center problem and the integer (j,k)-projective clustering problem.

Theorem 20 *There exists an algorithm that outputs a set of $\tilde{O}\left(\frac{d}{\varepsilon^2}g(d,j,k)(\log n)^{g(d,j,k)}\right)$ weighted points that is an ϵ-coreset for the integer (j,k)-projective clustering problem with probability at least $\frac{2}{3}$.*

Theorem 21 *There exists an algorithm that outputs an ϵ-coreset of $\tilde{O}\left(\frac{d}{\varepsilon^2}f(d,k)k^{f(d,k)}\log n\right)$ weighted points for the k-line center problem with probability at least $\frac{2}{3}$.*

We remark that Theorem 21 is subsumed by Theorem 23 below, as Varadarajan and Xiao (2012b) tighten the total sensitivity upper bound for the k-line center problem by showing that $f(d,k)$ in Theorem 19 is independent of d.

Theorem 22 *(Varadarajan and Xiao, 2012b) Let the distance function* dist *used in projective clustering be the z-th power of the Euclidean distance for some fixed $z\in(0,\infty)$. Then the total sensitivity t satisfies $t=O(k^{f(k)}\log n)$ for $j=1$, where f is a fixed function that only depends on k, respectively.*

From Theorem 22 and Theorem 1, we have

Theorem 23 *There exists an algorithm that outputs a set of $\tilde{O}\left(\frac{d}{\varepsilon^2}f(k)k^{f(k)}\log n\right)$ weighted points that is an ϵ-coreset for the k-line center problem with probability at least $\frac{2}{3}$.*

Varadarajan and Xiao (2012b) bounded the total sensitivity for the $(0,k)$-projective clustering problem, which includes k-median and k-means.

Theorem 24 *(Varadarajan and Xiao, 2012b) Let the distance function* dist *used in projective clustering be the z-th power of the Euclidean distance for some fixed $z\in(0,\infty)$. For $z\geq 1$, the total sensitivity t satisfies $t\leq 2^{2z-1}k+2^{z-1}$, whereas for $z\in(0,1)$, we have $t\leq 2k+1$.*

Thus by Theorem 24 and Theorem 1:

Theorem 25 *For either the k-median problem or the k-means problem, there exists an algorithm that outputs a set of $O\left(\frac{d}{\varepsilon^2}k\log k\right)$ weighted points that is an ϵ-coreset, with probability at least $\frac{2}{3}$.*

Varadarajan and Xiao (2012b) also bounded the total sensitivity for the j-subspace fitting problem.

Theorem 26 *(Varadarajan and Xiao, 2012b) Let $s = \min(j,d)$ and let the distance function* dist *used in projective clustering be the z-th power of the Euclidean distance for some fixed $z \in (0,\infty)$. Then total sensitivity of any set of n points in $\mathbb{R}^d$ for the j-subspace fitting problem is $O\left(s^{1+\frac{z}{2}}\right)$ for $1 \le z < 2$, $O(s)$ for $z = 2$, and $O(s^z)$ for $z > 2$.*

By Theorem 26 and Theorem 1, we conclude that:

Theorem 27 *Let $s = \min(j,d)$ and let the distance function* dist *used in projective clustering be the z-th power of the Euclidean distance for some fixed $z \in (0,\infty)$. Let $t = O\left(s^{1+\frac{z}{2}}\right)$ for $1 \le z < 2$, $O(s)$ for $z = 2$, and $O(s^z)$ for $z > 2$. Then there exists an algorithm that outputs a set of $\tilde{O}\left(\frac{dt}{\varepsilon^2}\right)$ weighted points that is an ϵ-coreset for the j-subspace fitting problem with probability at least $\frac{2}{3}$.*

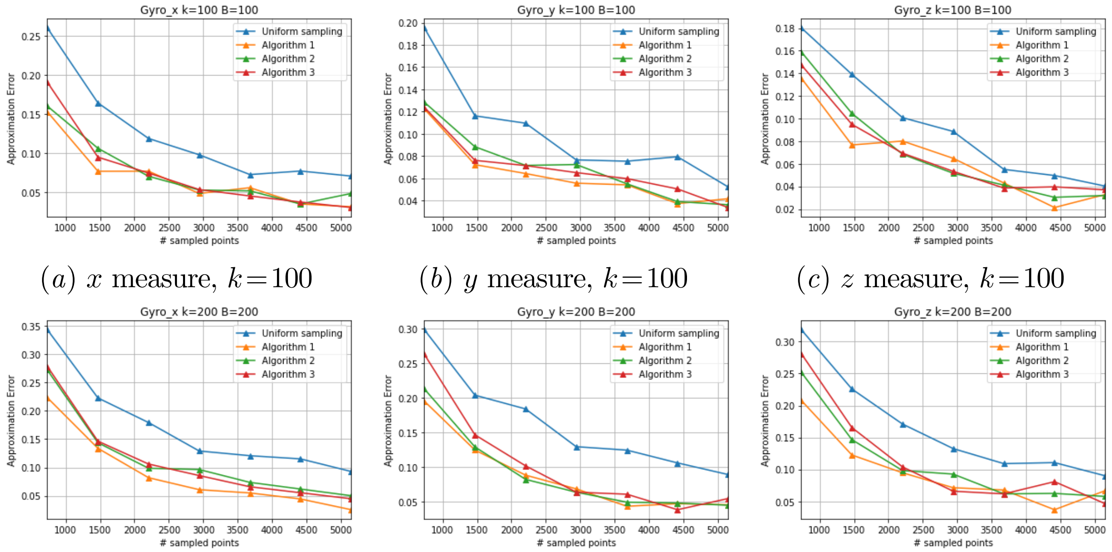
4. Empirical Evaluations

This concludes our discussion of the general sensitivity sampling framework. Although our contribution is primarily theoretical, we nevertheless performed empirical evaluations in Python 3.6 via the Numpy and Scipy.sparse libraries on a desktop machine with an Intel i7-6850K CPU @ 3.60GHZ, 64GB RAM. We consider coreset constructions based on sensitivity sampling for bicriteria algorithms (Algorithm 1), general loss functions that satisfy the weak triangle inequality (Algorithm 2), and the conditional normalized distance (Algorithm 3). We compared Algorithms 1, 2, and 3 to uniform sampling on k-means clustering on both relatively small offline data and large-scale streaming data that cannot fit into memory. Algorithms 1, 2, and 3 each require a bicriteria algorithm to approximate the importance of each point; we use `kmeans++` with $\alpha = O(\log k)$ and $\beta = 1$ to approximate the importances, so that the runtime is linear.

4.1. Evaluations on Offline Data

For experiments on small offline datasets, we compared our coreset constructions for k-means clustering in Algorithms 1-3 vs. uniform sampling on the datasets: (i) Gyroscope data and (ii) Accelerometer data. Collected by Anguita et al. (2013b), and can be found in Anguita et al. (2013a), the experiments have been carried out with a group of 30 volunteers within an age bracket of 19-48 years. Each person performed six activities (walking, walking upstairs, walking downstairs, sitting, standing, laying) while wearing a Samsung Galaxy S II smartphone on the waist. Using its embedded gyroscope (resp. accelerometer), 3-axial angular velocity (resp. linear acceleration) were captured at a constant rate of 50Hz. The experiments have been video-recorded to label the data manually. Data was collected from $n = 7352$ measurements; each instance consists of measurements from $d = 3$ dimensions: x, y, z, each in a size of 128.

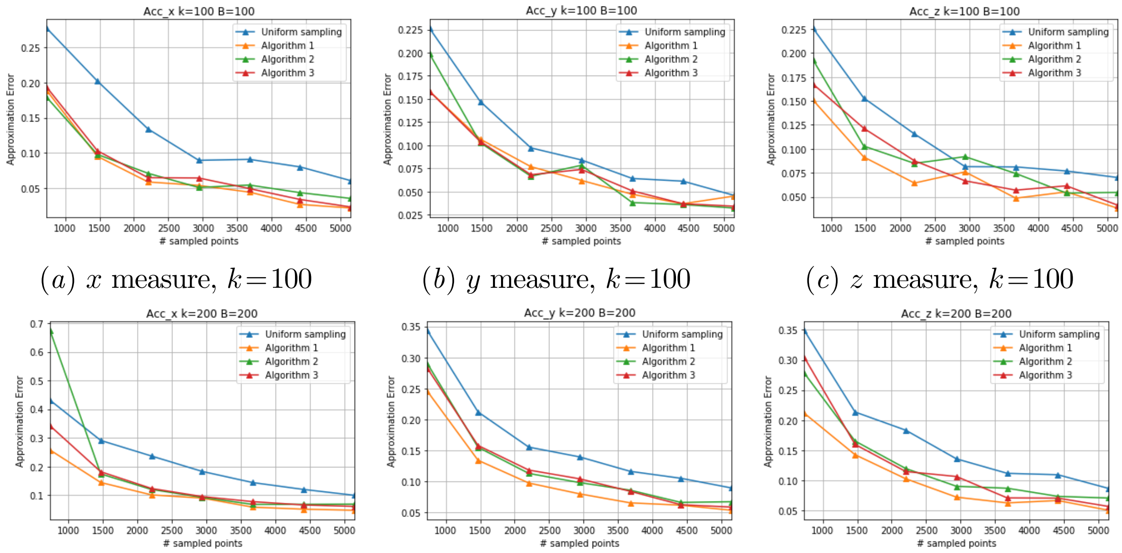
We ran Algorithms 1-3 and uniform sampling on the above six datasets with different sample/coreset size, between 1000 to 7000, with $k = 100$ and $k = 200$. The multiplicative approximation error (empirical ε) was calculated by $\varepsilon := \frac{\mathsf{COST}(A,Q_C) - \mathsf{COST}(A,Q_A)}{\mathsf{COST}(A,Q_A)}$, where A is the matrix whose rows correspond to the n input points, Q_A corresponds to the k centers of the whole data (two Lloyd's iterations after `kmeans++` initialization) and Q_C is the clustering of the coreset. Our results show a significant improvement of our algorithms over uniform sampling; we present the gyroscope data evaluations in Figure 2 and the accelerometer data evaluations in Figure 3.



(a) x measure, $k = 100$ (b) y measure, $k = 100$ (c) z measure, $k = 100$

(d) x measure, $k = 200$ (e) y measure, $k = 200$ (f) z measure, $k = 200$

Figure 2: Experimental results on gyroscope data for uniform sample compared to our 3 algorithms.



(a) x measure, $k = 100$ (b) y measure, $k = 100$ (c) z measure, $k = 100$

(d) x measure, $k = 200$ (e) y measure, $k = 200$ (f) z measure, $k = 200$

Figure 3: Experimental results on accelerometer data for uniform sample compared to our 3 algorithms.

4.2. Evaluations on Streaming Data

To handle large-scale streaming data that cannot fit into memory, our system separates the n points of the data into chunks of a desired size of coreset, called m. We use a merge-and-reduce framework on a binary tree, e.g. Feldman et al. (2010b), where each node is a coreset of the union of the data represented by its children nodes and the bottom layer of the tree consists of consecutive chunks of the data of size 4516. Thus the root of the tree is a coreset of the whole data. We build a tree of height 10 for our data, dividing the $n=4624611$ input points across 1024 chunks of size 4516.

Wikipedia Dataset. We compared uniform sampling to Algorithms 1 and 3 for k-means clustering on a created document-term matrix of Wikipedia (parsed enwiki-latest-pages-articles.xml.bz2-rss.xml from wic19), i.e. sparse matrix with 4624611 rows and 100k columns where each cell (i,j) equals the value of how many appearances the word number j has in article number i. We use a standard dictionary of the 100k most common words in Wikipedia (Dic12). We concatenated the coreset received in each floor and compared the received error in each floor. The error we determined was calculated by the formula $\frac{\mathsf{COST}(A,Q_C)-\mathsf{COST}(A,Q_A)}{\mathsf{COST}(A,Q_A)}$, where A is the original data matrix, Q_A is the clustering of the whole data (Lloyd's iterations until 1% convergence, after ++ initialization) and Q_C is the clustering of the coreset. We used two values of k, 100 and 200. We present our results in Figure 4. We concatenated the coreset received in each floor and compared the received error in each floor. The error we determined was calculated by the formula $\frac{\mathsf{COST}(A,Q_C)-\mathsf{COST}(A,Q_A)}{\mathsf{COST}(A,Q_A)}$, where A is the original data matrix, Q_A is the clustering of the whole data (Lloyd's iterations until 1% convergence, after ++ initialization) and Q_C is the clustering of the coreset. We present our results in Figure 4 for $k=100$ and $k=200$. Similar to the offline evaluations, we obtain better results for our algorithms than uniform sampling. However, unlike than the offline data, here the conditional normalized algorithm gets much better results than the general sensitivity sampling algorithm.

Algorithms. The algorithms we compared are uniform sampling, and our Algorithm 1 and 3.

Results. We concatenated the coreset received in each floor and compared the received error in each floor. The error we determined was calculated by the formula $\frac{\mathsf{COST}(A,Q_A)-\mathsf{COST}(A,Q_C)}{\mathsf{COST}(A,Q_A)}$, where A is the original data matrix, Q_A is the clustering of the whole data (Lloyd's iterations until 1% convergence, after ++ initialization) and Q_C is the clustering of the coreset. We used two values of k, 100 and 200. We present our results in Figure 4.

Discussion. Indeed also for this dataset we got better results for our algorithm than uniform sampling. However, unlike than in Section 4, here Algorithm 2 gets much better results than Algorithm 1.

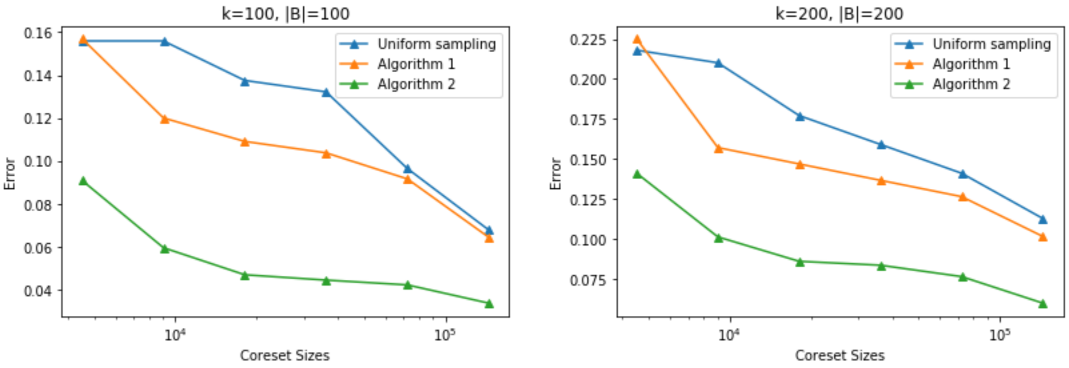
References

Marcel R. Ackermann and Johannes Blömer. Coresets and approximate clustering for bregman divergences. In *Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1088–1097, 2009.

Pankaj K. Agarwal, Sariel Har-Peled, and Hai Yu. Robust shape fitting via peeling and grating coresets. In *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 182–191, 2006.

Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, and Jorge L. Reyes-Ortiz. A public domain dataset for human activity recognition using smartphones, 2013a. URL https://archive.ics.uci.edu/ml/datasets/human+activity+recognition+using+smartphones.

Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, and Jorge Luis Reyes-Ortiz. A public domain dataset for human activity recognition using smartphones., 2013b.



(a) (a) $k\!=\!100$

(b) (b) $k\!=\!200$

Figure 4: Experimental results for k-means clustering on a data stream of a created document-term matrix of Wikipedia, with $k = 100$ in (a) and $k = 200$ in (b). Algorithms 1 and 2 in figures based on general sensitivity sampling for bicriteria algorithms and conditional normalized distances, respectively.

Sepehr Assadi, MohammadHossein Bateni, Aaron Bernstein, Vahab S. Mirrokni, and Cliff Stein. Coresets meet EDCS: algorithms for matching and vertex cover on massive graphs. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1616–1635, 2019.

Mihai Badoiu, Sariel Har-Peled, and Piotr Indyk. Approximate clustering via core-sets. In *Proceedings on 34th Annual ACM Symposium on Theory of Computing*, pages 250–257, 2002.

Luca Becchetti, Marc Bury, Vincent Cohen-Addad, Fabrizio Grandoni, and Chris Schwiegelshohn. 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*, pages 1039–1050, 2019.

Vladimir Braverman, Petros Drineas, Cameron Musco, Christopher Musco, Jalaj Upadhyay, David P. Woodruff, and Samson Zhou. Near optimal linear algebra in the online and sliding window models. *CoRR*, abs/1805.03765, 2018.

Vladimir Braverman, Harry Lang, Enayat Ullah, and Samson Zhou. Improved algorithms for time decay streams. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM*, pages 27:1–27:17, 2019.

Kenneth L. Clarkson. Coresets, sparse greedy approximation, and the frank-wolfe algorithm. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 922–931, 2008.

Kenneth L. Clarkson and David P. Woodruff. Input sparsity and hardness for robust subspace approximation. In *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS*, pages 310–329, 2015.

Anirban Dasgupta, Petros Drineas, Boulos Harb, Ravi Kumar, and Michael W. Mahoney. Sampling algorithms and coresets for ℓ_p regression. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 932–941, 2008.

Amit Deshpande and Kasturi R. Varadarajan. Sampling-based dimension reduction for subspace approximation. In *Proceedings of the 39th Annual ACM Symposium on Theory of Computing*, pages 641–650, 2007.

Amit Deshpande, Luis Rademacher, Santosh S. Vempala, and Grant Wang. Matrix approximation and projective clustering via volume sampling. *Theory of Computing*, 2(12):225–247, 2006.

Dic12. https://gist.github.com/h3xx/1976236, 2012.

Dan Feldman and Michael Langberg. A unified framework for approximating and clustering data. In *Proceedings of the 43rd ACM Symposium on Theory of Computing, STOC*, pages 569–578, 2011.

Dan Feldman and Leonard J. Schulman. Data reduction for weighted and outlier-resistant clustering. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2012, Kyoto, Japan, January 17-19, 2012*, pages 1343–1354, 2012.

Dan Feldman, Amos Fiat, and Micha Sharir. Coresets for weighted facilities and their applications. In *47th Annual IEEE Symposium on Foundations of Computer Science, FOCS Proceedings*, pages 315–324, 2006.

Dan Feldman, Morteza Monemizadeh, Christian Sohler, and David P. Woodruff. Coresets and sketches for high dimensional subspace approximation problems. In *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 630–649, 2010a.

Dan Feldman, Morteza Monemizadeh, Christian Sohler, and David P Woodruff. Coresets and sketches for high dimensional subspace approximation problems. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*, pages 630–649. Society for Industrial and Applied Mathematics, 2010b.

Dan Feldman, Melanie Schmidt, and Christian Sohler. Turning big data into tiny data: Constant-size coresets for k-means, PCA and projective clustering. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1434–1453, 2013.

Dan Feldman, Melanie Schmidt, and Christian Sohler. Turning big data into tiny data: Constant-size coresets for k-means, pca, and projective clustering. *SIAM J. Comput.*, 49(3):601–657, 2020.

Gereon Frahling and Christian Sohler. Coresets in dynamic geometric data streams. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 209–217, 2005.

Gereon Frahling and Christian Sohler. A fast k-means implementation using coresets. *Int. J. Comput. Geometry Appl.*, 18(6):605–625, 2008.

Sariel Har-Peled and Soham Mazumdar. 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*, pages 291–300, 2004.

Sariel Har-Peled and Micha Sharir. Relative (p, ϵ)-approximations in geometry. *Discrete & Computational Geometry*, 45(3):462–496, 2011.

David Haussler. Sphere packing numbers for subsets of the boolean n-cube with bounded vapnik-chervonenkis dimension. *J. Comb. Theory, Ser. A*, 69(2):217–232, 1995.

Lingxiao Huang and Nisheeth K. Vishnoi. 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*, pages 1416–1429, 2020.

Lingxiao Huang, Shaofeng H.-C. Jiang, Jian Li, and Xuan Wu. Epsilon-coresets for clustering (with outliers) in doubling metrics. In *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS*, pages 814–825, 2018.

Yi Li, Philip M. Long, and Aravind Srinivasan. Improved bounds on the sample complexity of learning. *J. Comput. Syst. Sci.*, 62(3):516–527, 2001.

Alexander Munteanu, Chris Schwiegelshohn, Christian Sohler, and David P. Woodruff. On coresets for logistic regression. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems, NeurIPS*, pages 6562–6571, 2018.

Ben Mussay, Margarita Osadchy, Vladimir Braverman, Samson Zhou, and Dan Feldman. Data-independent neural pruning via coresets. In *8th International Conference on Learning Representations, ICLR*, 2020.

Jeff M. Phillips and Wai Ming Tai. Improved coresets for kernel density estimates. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 2718–2727, 2018.

Christian Sohler and David P. Woodruff. Strong coresets for k-median and subspace approximation: Goodbye dimension. In *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS*, pages 802–813, 2018.

Michel Talagrand. Sharper bounds for gaussian and empirical processes. *The Annals of Probability*, pages 28–76, 1994.

VN Vapnik and A Ya Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability & Its Applications*, 16(2):264–280, 1971.

Kasturi R. Varadarajan and Xin Xiao. A near-linear algorithm for projective clustering integer points. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1329–1342, 2012a.

Kasturi R. Varadarajan and Xin Xiao. On the sensitivity of shape fitting problems. In *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS*, pages 486–497, 2012b.

wic19. https://dumps.wikimedia.org/enwiki/latest/, 2019.

Appendix A. Proof of Theorem 17

Theorem 28 (Theorem 17, Restated) *Let d be the dimension of a query space (P,w,Q,f). Suppose that $m:P\to[0,\infty)$ such that $m(p)\geq \sup_{C\in Q(P)} w(p)|f(p,C)|$. Let $t\geq \sum_{p\in P}m(p)$, and $\varepsilon,\delta\in(0,1)$. Let $c\geq 1$ be a sufficiently large constant, and let S be a sample of*

$$|S|\geq \frac{ct}{\varepsilon^2}\left(d\log t+\log\left(\frac{1}{\delta}\right)\right)$$

i.i.d. points from P, where for every $p\in P$ and $s\in S$ we have $\mathrm{Prob}(p=s)\geq\frac{m(p)}{t}$. Then, with probability at least $1-\delta$, then simultaneously for all $C\in Q(S)$,

$$\left|\sum_{p\in P}w(p)f(p,C)-\sum_{p\in S}\frac{w(q)}{|S|\mathrm{Prob}(q)}\cdot f(p,C)\right|\leq\varepsilon\left|\sum_{p\in P}w(p)f(p,C)\right|.$$

Proof : Let $g(S,w',C)=\sum_{p\in S}\frac{w'(p)w(p)}{s(p)}\frac{f(p,C)}{\sum_{p\in P}w(p)f(p,C)}$, where $w'(p)=\frac{s(p)}{t}$. Note that $g(P,w',C)\in[0,1]$ so the conditions of Theorem 16 hold. Thus by Theorem 16, we obtain some $\left(\frac{1}{4t},\frac{\varepsilon}{4}\right)$-sample $(Z,\mathcal{R})$ for $(X,\mathcal{R})$ with respect to the function $g(P,w',C)$. By definition of (ν,α)-sample, it follows that $d_{1/4t}(g(P,w',C),g(S,u',C))\leq\frac{\varepsilon}{4}$. Therefore,

$$d_{1/4t}\left(\sum_{p\in P}\frac{w'(p)w(p)}{s(p)}\frac{f(p,C)}{\sum_{p\in P}w(p)f(p,C)}-\sum_{p\in S}\frac{u'(p)w(p)}{s(p)}\frac{f(p,C)}{\sum_{p\in P}w(p)f(p,C)}\right)\leq\frac{\varepsilon}{4}.$$

Because $w'(p)=\frac{s(p)}{t}$, then

$$d_{1/4t}\left(\sum_{p\in P}\frac{w(p)}{t}\frac{f(p,C)}{\sum_{p\in P}w(p)f(p,C)}-\sum_{p\in S}\frac{u'(p)w(p)}{s(p)}\frac{f(p,C)}{\sum_{p\in P}w(p)f(p,C)}\right)\leq\frac{\varepsilon}{4}.$$

Since $d_x(a,b)=\frac{|a-b|}{a+b+x}=\frac{|ta-tb|}{ta+tb+tx}=d_{tx}(ta,tb)$, then

$$d_{1/4}\left(\sum_{p\in P}w(p)\frac{f(p,C)}{\sum_{p\in P}w(p)f(p,C)}-t\sum_{p\in S}\frac{u'(p)w(p)}{s(p)}\frac{f(p,C)}{\sum_{p\in P}w(p)f(p,C)}\right)\leq\frac{\varepsilon}{4}.$$

Let $u(p)=\frac{t\cdot u'(p)w(p)}{s(p)}$ and $\overline{g}(P,w,C)=\sum_{p\in P}w(p)g(p,C)$, where $g(p,C)=\frac{f(p,C)}{\sum_{p\in P}w(p)f(p,C)}$. Then $d_{1/4}(\overline{g}(P,w,C),\overline{g}(S,u,C))\leq\frac{\varepsilon}{4}$.

$$\overline{g}(P,w,C)=\sum_{p\in P}w(p)g(p,C)=\frac{\sum_{p\in P}w(p)f(p,C)}{\sum_{p\in P}w(p)f(p,C)}=1$$

and

$$\overline{g}(S,u,C)=\sum_{p\in S}u(p)g(p,C)=\frac{\sum_{p\in S}u(p)f(p,C)}{\sum_{p\in P}w(p)f(p,C)}=\frac{\overline{f}(S,u,C)}{\overline{f}(P,w,C)},$$

where $\overline{f}(S,w,C)=\sum_{p\in S}w(p)f(p,C)$. Thus from $d_{1/4t}(\overline{g}(P,w,C),\overline{g}(S,u,C))\le\varepsilon/4$, we have

$$d_{1/4t}\left(1,\frac{\overline{f}(S,u,C)}{\overline{f}(P,w,C)}\right)\le\frac{\varepsilon}{4}.$$

By Theorem 9, we thus have

$$\left|1-\frac{\overline{f}(S,u,C)}{\overline{f}(P,w,C)}\right|\le\varepsilon.$$

Hence,

$$|\overline{f}(P,w,C)-\overline{f}(S,u,C)|\le\varepsilon|\overline{f}(P,w,C)|.$$

Since we defined $\overline{f}(S,w,C)=\sum_{p\in S}w(p)f(p,C)$, then it follows that

$$\left|\sum_{p\in P}w(p)f(p,C)-\sum_{p\in S}u(p)f(p,C)\right|\le\varepsilon\left|\sum_{p\in P}w(p)f(p,C)\right|,$$

as desired. □

Intuition for Sample Complexity of Learning and Coresets. The intuition is that the class of functions $\mathcal{F}$ represents the objective in the query space, so that a particular $f\in\mathcal{F}$ represents the objective for a particular query in the query space. For objectives like k-means or k-median clustering, each f represents the objective for a separate set of k centers. Then the goal is to learn $\mathcal{F}$ simultaneously with a small number of samples.

The domain X for the class of functions $\mathcal{F}$ translates exactly to the ground set X, which is the input points for objectives like k-means or k-medians. We first note that sampling a point of X and then rescaling by the (inverse of the) sampling probability provides an unbiased estimator to the objective. Li et al. (2001) then states that if we sample uniformly over X, we can obtain a $(1+\varepsilon)$-approximation to the objective by bounding the variance through a small number of samples. Then the idea of sensitivity sampling is that instead of uniformly sampling points from X, we sample each point of X according to its sensitivity, but still rescale by the (inverse of the) sampling probability. Now the expectation of the samples is still the objective, but the variance is much smaller and so we require a smaller number of samples.

The real workhorse in this bound is Lemma 12 by Li et al. (2001), which uses the chaining technique of Kolmogorov and refined by Talagrand Talagrand (1994). Crucially, the usage of chaining by Li et al. (2001) manages to simultaneously learn a large number of functions in a class $\mathcal{F}$ without needing to union bound over a net over the functions in $\mathcal{F}$. It is precisely this technique that avoids a quadratic dependency on t from the union bound.

Appendix B. Coreset for k-clustering

This section considers tighter bounds for k-clustering and may be skipped for general applications. We introduce ρ-pseudo distances and define the importance of a point as a generalization of the sensitivity. Using the importance, we then give an analog of Theorem 1 with sharper bounds for ρ-pseudo distances. As a result, we obtain stronger bounds for coreset constructions for k-clustering.

We reiterate that our results in this section mirror those of Section 2. We only provide theoretical guarantees on the number of samples required by the sensitivity sampling framework for k-clustering. We specify the framework in full in Algorithm 1.

We first require the following definition of (α,β)-assignment for bicriteria algorithms.

Definition 29 **((α,β)-assignment.)** *Let X be a ground set and (P,w,Q,g) be a query space where $g:X^2\to[0,\infty)$. Let $Q^*\in Q(P)$ be a query that minimizes the loss of P over every query,*

$$\mathrm{opt}(P)=\arg\inf_{Q\in Q(P)}\sum_{p\in P}w(p)g(p,Q)=\sum_{p\in P}w(p)g(p,Q^*).$$

Let $\alpha,\beta>0$, and $B\subseteq X$ such that $|B|\leq\beta|Q^*|$. A function $\mathcal{B}:P\to B$ is an (α,β)-assignment for (P,w,Q,g) if

$$\sum_{p\in P}w(p)g(p,\mathcal{B}(p))\leq\alpha\cdot\mathrm{opt}(P).$$

Every $b\in B$ is called a center *and its* cluster *in P is $\mathcal{B}^{-1}(b)=\{p\in P\mid\mathcal{B}(p)=b\}$.*

Intuitively, an (α,β)-assignment is just a bicriteria clustering with approximation factor α and an overselection of centers by a factor of β. Thus for k-means clustering, any α-approximation algorithm that chooses βk centers can be used to determine an (α,β)-assignment for each of the points.

The following definition is especially useful for k-means clustering.

Definition 30 **(ρ-pseudo distance)** *Let X be a ground set and $\rho\geq1$. A symmetric function $g:X^2\to[0,\infty)$ is* ρ-pseudo distance *over X if for every $(p,q,x)\in X^3$*

$$g(p,x)\leq\rho(g(p,q)+g(q,x)).$$

For a finite set $Q\subseteq X$, we denote $g(p,Q):=\min_{x\in Q}g(p,x)$.

Note that the above definition does not assume that $g(p,p)=0$ for every $p\in P$. The inequality in Definition 30 is sometimes called "weak triangle inequality".

Lemma 31 *Let g be a ρ-pseudo distance over a ground set X. For every pair of points $p,q\in X$ and a finite set $Q\subseteq X$,*

$$g(p,Q)\leq\rho(g(q,p)+g(p,Q)).$$

Proof : For every $p,q\in X$, and a center $x_p\in Q$ that is closest to p, i.e. $g(p,Q)=g(p,x_p)$, we have

$$g(q,Q)=\min_{x\in Q}g(q,x)\leq g(q,x_p)\leq\rho(g(q,p)+g(p,x_p))=\rho(g(q,p)+g(p,Q)).$$

□

We now give a generalization of the notion of sensitivity in the form of importance.

Definition 32 **(The function f and its importance m)** *Let (P,w,Q,g) be a query space where $w:P\to[0,\infty)$ and g is a ρ-pseudo distance. Let $\mathcal{B}:P\to B$ be an (α,β)-assignment for (P,w,Q,g). For every $p\in P$ and $Q\in Q(P)$, we define*

$$f(p,Q)=\frac{g(p,Q)}{\sum_{q\in P}w(q)g(q,Q)}. \tag{1}$$

```
Algorithm 1: Coreset($P$,$w$,$\mathcal{B}$,$s$); see Theorem 34
Input: A weighted set $(P,w)$ where $w:P\to[0,\infty)$, an $(\alpha,\beta)$-assignment $\mathcal{B}:P\to B$ for a query
space $(P,w,Q,g)$, and sample size (integer) $s\geq 1$.
Output: A weighted set $(C,u)$. for every center $b\in B$ and a point in its cluster $p\in\mathcal{B}^{-1}(b)$ do
    Set $\text{Prob}(p):=\frac{w(p)g(p,\mathcal{B}(p))}{2\sum_{q\in P}w(q)g(q,\mathcal{B}(q))}+\frac{w(p)}{2|B|\sum_{q\in\mathcal{B}^{-1}(b)}w(q)}$.
        // Note that $\sum_{p\in P}\text{Prob}(p)=1$
end
$C:=\emptyset$
for $s$ iterations do
    Sample a point $q$ from $P$, where every $p\in P$ is chosen with probability $\text{Prob}(p)$.
        $C:=C\cup\{q\}$// add $q$ to the multi-set $C$
    $u(q):=\frac{w(q)}{|C|\cdot\text{Prob}(q)}$
end
return $(C,u)$
```

For every center $b\in B$ and a point $p\in\mathcal{B}^{-1}(b)$ in its cluster we define

$$m(p)=\frac{\rho\alpha\cdot w(p)g(p,\mathcal{B}(p))}{\sum_{q\in P}w(q)g(q,\mathcal{B}(p))}+\frac{\rho^2(\alpha+1)w(p)}{\sum_{q\in\mathcal{B}^{-1}(b)}w(q)}.$$

We now show that the importance m satisfies a similar function as the notion of sensitivity.

Lemma 33 *Let $\mathcal{B}:P\to B$, m and f be defined as in Definition 32. Then for every $p\in P$ we have*

$$m(p)\geq\sup_{Q\in Q(P)}w(p)f(p,Q).$$

Proof : For simplicity, we denote $p'=\mathcal{B}(p)$ and $q'=\mathcal{B}(q)$ for every $p,q\in P$, and $P_b=\mathcal{B}^{-1}(p)$ for every $b\in B$. We prove the claim for a point $p\in P_b$ in the cluster of some center $b\in B$, as in Definition 32. Let $Q\in Q(P)$ and assume $w(p)f(p,Q)>0$, otherwise the lemma trivially holds. We need to upper bound

$$\begin{aligned}w(p)f(p,Q)&=\frac{w(p)g(p,Q)}{\sum_{q\in P}w(q)g(q,Q)}\leq\frac{\rho w(p)g(p,p')}{\sum_{q\in P}w(q)g(q,Q)}+\frac{\rho w(p)g(p',Q)}{\sum_{q\in P}w(q)g(q,Q)}\\&\leq\frac{\alpha\rho w(p)g(p,p')}{\sum_{q\in P}w(q)g(q,q')}+\frac{\rho w(p)g(p',Q)}{\sum_{q\in P}w(q)g(q,Q)},\end{aligned}\tag{2}$$

where the first inequality holds by Lemma 31, and the second inequality holds since $\mathcal{B}$ is an (α,β)-assignment. To bound the last term, note that

$$\begin{aligned}g(p',Q)\sum_{q\in P_b}w(q)&=\sum_{q\in P_b}w(q)g(p',Q)\leq\sum_{q\in P_b}w(q)\cdot\rho(g(p',q)+g(q,Q))\\&=\rho\sum_{q\in P_b}w(q)g(q,b)+\rho\sum_{q\in P_b}w(q)g(q,Q)&(3)\\&\leq\rho\alpha\sum_{q\in P_b}w(q)g(q,Q)+\rho\sum_{q\in P_b}w(q)g(q,Q)=\rho(\alpha+1)\sum_{q\in P_b}w(q)g(q,Q),&(4)\end{aligned}$$

where the first inequality is by Lemma 31, (3) holds since $p'=b$ and since g is symmetric by definition, and (4) holds since $\mathcal{B}$ is an (α,β)-assignment.

Dividing by $\sum_{q\in P_b} w(q)\cdot\sum_{q\in P} w(q)g(q,Q)$ yields

$$\frac{g(p',Q)}{\sum_{q\in P} w(q)g(q,Q)} \le \frac{\rho(\alpha+1)}{\sum_{q\in P_b} w(q)}.$$

Substituting this in (2) yields the desired result

$$\begin{aligned} w(p)f(p,Q) &\le \frac{\rho\alpha w(p)g(p,p')}{\sum_{q\in P} w(q)g(q,q')} + \frac{\rho^2(\alpha+1)w(p)}{\sum_{q\in P_b} w(q)} \\ &= m(p). \end{aligned}$$

□

As a warm-up, we now prove an analog of Theorem 1 for ρ-pseudo distances that provides tighter bounds, due to the tighter setting of t from the (α,β) assignment.

Theorem 34 *Let*

- *(P,w,Q,g) be a query space, where g is a ρ-pseudo distance and $w: P \to [0,\infty)$.*
- *$\mathcal{B}: P \to B$ be an (α,β) assignment for (P,w,Q,g).*
- *d be the VC-dimension of (P,w,Q,f), where f was defined in (1).*
- *$t = 2(\rho\alpha + \rho^2(\alpha+1)|B|)$.*
- *$c \ge 1$ be a sufficiently large constant, $\varepsilon,\delta \in (0,1)$, and*

$$s \ge \frac{ct}{\varepsilon^2}\left(d\log t + \log\left(\frac{1}{\delta}\right)\right).$$

- *(C,u) be the output of a call to* Coreset$(P,w,\mathcal{B},s)$*; see Algorithm 1.*

Then, $C \subseteq P$, $u: C \to [0,\infty)$ and, with probability at least $1-\delta$, (C,u) is an ε-coreset of size $|C| = s$ for (P,w,Q,g).

Proof : Let $p \in P$. By Lemma 33, for every $Q \in Q(P)$

$$m(p) \ge w(p)f(p,Q). \tag{5}$$

The probability of choosing p to be, say, the first point in C, is

$$\begin{aligned} \text{Prob}(p) &= \frac{w(p)g(p,\mathcal{B}(p))}{2\sum_{q\in P} w(q)g(q,\mathcal{B}(q))} + \frac{w(p)}{2|B|\sum_{q\in\mathcal{B}^{-1}(b)} w(q)} \\ &\ge \frac{1}{2(\rho\alpha+\rho^2(\alpha+1)|B|)}\cdot\left(\frac{\rho\alpha w(p)g(p,\mathcal{B}(p))}{\sum_{q\in P} w(q)g(q,\mathcal{B}(q))} + \frac{\rho^2(\alpha+1)w(p)}{\sum_{q\in\mathcal{B}^{-1}(b)} w(q)}\right) \ge \frac{m(p)}{t}. \end{aligned}$$

Using the last inequality and (5), we apply Theorem 17 to obtain that, with probability at least $1-\delta$, we have that for all $Q \in Q(C)$,

$$\left|\sum_{p\in P} w(p) f(p,Q) - \sum_{q\in C} \frac{w(q)}{\text{Prob}(q)|C|} \cdot f(q,Q)\right| \leq \varepsilon.$$

Multiplying this by $\sum_{q\in P} w(q)g(q,Q)$, and substituting $u(q) = \frac{w(q)}{\text{Prob}(q)|C|}$ yields that for all $Q \in Q(C)$,

$$\left|\sum_{p\in P} w(p)g(p,Q) - \sum_{q\in C} u(q)\cdot g(q,Q)\right| \leq \varepsilon \sum_{q\in P} w(q)g(q,Q)$$

implies that (C,u) is an ε-coreset as desired. □

B.1. Smaller coreset

In this section, we define a (ρ,ψ,ϕ)-pseudo distance function, which serves as a generalization of ρ-pseudo distances, and give smaller coreset constructions for (ρ,ψ,ϕ)-pseudo distance functions. Our algorithms appear in Algorithm 2 and Algorithm 3.

```
Algorithm 2: SMALLER-CORESET$(P,w,\mathcal{B},s)$; See Theorem 38
Input: Ā weighted set $(P,w)$ where $w: P \to [0,\infty)$, $(\alpha,\beta)$-assignment $\mathcal{B}: P \to B$ for
  $(P,w,Q,g)$, and sample size (integer) $s \geq 1$.
  Output: A weighted set $(C \cup B,u)$.
  $(C,u) :=$ CORESET$(P,w,\mathcal{B},s)$  // see Algorithm 1
for every $b \in B$ do
    Set $u(b) := \sum_{p\in\mathcal{B}^{-1}(b)} w(p) - \sum_{q\in C\cap\mathcal{B}^{-1}(b)} u(q)$
end
return $(C \cup B,u)$//   $C \cup B$ is a multi-set
```

We define the following generalization of distance to handle k-clustering.

Definition 35 *Let g be a ρ-pseudo distance over X as in Definition 30. For $\phi > 0$ and $\psi \geq 0$, g is also a (ρ,ψ,ϕ)-*pseudo distance function *if for every $(p,q,x) \in X^3$ we have*

$$|g(p,x) - g(q,x)| \leq \phi g(p,q) + \psi g(q,x).$$

Intuitively, the ρ-pseudo distance handles loss functions such as the squared Euclidean distance that do not satisfy the triangle inequality but rather a generalized version of the triangle inequality.

Lemma 36 *Let $g: X^2 \to [0,\infty)$ be a (ρ,ψ,ϕ)-pseudo distance function. Then for every finite set $M \subseteq X$ and $p,q \in M$ we have*

$$|g(p,Q) - g(q,Q)| \leq \phi g(p,q) + \psi g(q,Q).$$

Proof : We assume that M is non-empty, otherwise the claim trivially holds, as we define the minimum of an empty set to be 0 in the notation. Let $x_q \in \operatorname{argmin}_{x\in Q} g(q,x)$ and $x_p \in \operatorname{argmin}_{x\in Q} g(p,x)$. The proof is by case analysis: (i) $g(p,Q) > g(q,Q)$, and (ii) $g(p,Q) \leq g(q,Q)$ as follows.
Case (i): $g(p,Q) > g(q,Q)$. We have

$$\begin{aligned}|g(p,Q)-g(q,Q)| &= g(p,Q)-g(q,Q) = g(p,Q)-g(q,x_q)\\ &\leq g(p,x_q)-g(q,x_q) \leq \phi g(p,q)+\psi g(q,x_q)\\ &= \phi g(p,q)+\psi g(q,Q),\end{aligned} \tag{6}$$

where the first inequality is by the definition of $g(p,Q)=\min_{x\in Q} g(p,Q)$, and the second inequality is by Definition 35.
Case (ii): $g(p,Q) \leq g(q,Q)$. We have

$$\begin{aligned}|g(p,Q)-g(q,Q)| &= g(q,Q)-g(p,Q) = g(q,x_q)-g(p,x_p)\\ &\leq g(q,x_p)-g(p,x_p) \leq \phi g(q,p)+\psi g(p,x_p)\\ &= \phi g(q,p)+\psi g(p,Q) \leq \phi g(p,q)+\psi g(q,Q),\end{aligned} \tag{7}$$

where the last inequality is by the assumption of this case. Combining (6) and (7) yields that (in both cases)

$$|g(p,Q)-g(q,Q)| \leq \phi g(p,q)+\psi g(q,Q).$$

□

Definition 37 (Conditional normalized distance h.) *Let (P,w,Q,g) be a query space where $w: P \to [0,\infty)$, and $g: X^2 \to [0,\infty)$ is a (ρ,ψ,ϕ)-pseudo distance function. Let $\mathcal{B}: P \to B$ be an (α,β)-assignment for (P,w,Q,g). Let $\varepsilon \in (0,1)$ such that $\psi < \varepsilon/(4\rho(\alpha+1))$, and*

$$\varepsilon' = \frac{\varepsilon}{4\phi\rho(\alpha+1)} - \frac{\psi}{\phi}. \tag{8}$$

For every $Q \in Q(P)$ define

$$F(Q) = \left\{ p \in P \mid g(\mathcal{B}(p),Q) > \frac{g(p,\mathcal{B}(p))}{\varepsilon'} \right\}, \tag{9}$$

and $h: P \times Q(P) \to \mathbb{R}$ such that for every $p \in P$ and $Q \in Q(P)$,

$$h(p,Q) = \begin{cases} \frac{g(p,Q)-g(\mathcal{B}(p),Q)}{\sum_{q\in P} w(q) g(q,Q)} & p \in P \setminus F(Q)\\ 0 & p \in F(Q). \end{cases} \tag{10}$$

Theorem 38 *Consider the variables in Definition 37. Let*

$$t = 1 + 2\alpha(\phi + \psi/\varepsilon')$$

and d be the VC-dimension of (P,w,Q,h). Let c' be a sufficiently large constant,

$$s \geq \frac{c' t}{\varepsilon^2}\left(d \log t + \log\left(\frac{1}{\delta}\right)\right) + c'|B|\left(\log|B| + \log\left(\frac{1}{\delta}\right)\right),$$

and $(C\cup B,u)$ *be the output of a call to algorithm* Smaller-Coreset$(P,w,\mathcal{B},s)$*; see Algorithm* 2*. Then,* $C\subseteq P$*,* $u:C\to[0,\infty)$*, and with probability at least* $1-\delta$*, we have that for all* $X\in Q(C)$*,*

$$\left|\sum_{p\in P}w(p)g(p,X)-\sum_{q\in C\cup B}u(q)g(q,X)\right|\leq\varepsilon\sum_{p\in P}w(p)g(p,X).$$

Proof : Let $Q\in Q(C)$ and extend the function u as defined in Algorithm 2 to be $u(p)=0$ for every $p\in P\setminus C$. Also define $v(p)=w(p)-u(p)$ and $p'=\mathcal{B}(p)$ for every $p\in P$. The difference in the loss between taking the original points or its coreset $C\cup B$ is

$$\begin{aligned}|\sum_{p\in P}w(p)g(p,Q)-\sum_{q\in C\cup B}u(q)g(q,Q)|&=|\sum_{p\in P}v(p)g(p,Q)-\sum_{b\in B}u(b)g(b,Q)|\\&\leq\left|\sum_{p\in P}v(p)(g(p,Q)-g(p',Q))\right|+\left|\sum_{p\in P}v(p)g(p',Q)-\sum_{b\in B}u(b)g(b,Q)\right| \qquad (11)\\&=\left|\sum_{p\in P}v(p)(g(p,Q)-g(p',Q))\right|, \qquad (12)\end{aligned}$$

where (11) is by the triangle inequality, and (12) holds since

$$\begin{aligned}\sum_{p\in P}v(p)g(p',Q)&=\sum_{b\in B}\sum_{p\in\mathcal{B}^{-1}(b)}v(p)g(b,Q)=\sum_{b\in B}g(b,Q)\sum_{p\in\mathcal{B}^{-1}(b)}(w(p)-u(p))\\&=\sum_{b\in B}g(b,Q)\left(\sum_{p\in\mathcal{B}^{-1}(b)}w(p)-\sum_{q\in C\cap\mathcal{B}^{-1}(b)}u(q)\right)\\&=\sum_{b\in B}u(b)g(b,Q),\end{aligned}$$

where the last equality is by the definition of u in Line 2 of Algorithm 2.

By letting $F=F(Q)$ as defined in (9), (12) is bounded by $\left|\sum_{p\in P}v(p)(g(p,Q)-g(p',Q))\right|$, which equals

$$\begin{aligned}&\left|\sum_{p\in P\setminus F}v(p)(g(p,Q)-g(p',Q))+\sum_{p\in F}v(p)(g(p,Q)-g(p',Q))\right|\\&\qquad\leq\left|\sum_{p\in P\setminus F}v(p)(g(p,Q)-g(p',Q))\right| \qquad (13)\\&\qquad+\left|\sum_{p\in F}v(p)(g(p,Q)-g(p',Q))\right|, \qquad (14)\end{aligned}$$

where the last inequality is by the triangle inequality. We now bound each of the last terms.

Bound on (13): Let $h:P\times Q(M)\to\mathbb{R}$ be as in Definition 37. Let $p\in P\setminus F$ and recall that $\mathrm{Prob}(p)$ was defined to be the probability of choosing p in Line 1 of Algorithm 1. We then have

$$\begin{aligned}
w(p)|h(p,Q)| &= \frac{w(p)|g(p,Q)-g(p',Q)|}{\sum_{q\in P}w(q)g(q,Q)} \\
&\leq \frac{\alpha\cdot w(p)|g(p,Q)-g(p',Q)|}{\sum_{q\in P}w(q)g(q,q')} && (15)\\
&\leq \frac{\alpha\cdot w(p)(\phi g(p,p')+\psi g(p',Q))}{\sum_{q\in P}w(q)g(q,q')} && (16)\\
&\leq \frac{\alpha\cdot w(p)g(p,p')(\phi+\psi/\varepsilon')}{\sum_{q\in P}w(q)g(q,q')} && (17)\\
&\leq 2\alpha(\phi+\psi/\varepsilon')\mathrm{Prob}(p),
\end{aligned}$$

where (15) holds since $\mathcal{B}$ is an (α,β)-assignment, i.e.,

$$\sum_{q\in P}w(q)g(q,q')\leq\alpha\sum_{q\in P}w(q)g(q,Q), \tag{18}$$

(16) is by Lemma 36, and (17) is by (9) and the assumption $p\in P\setminus F$.

Hence, $\sup_{Q\in Q(M)}w(p)|h(p,Q)|$ is bounded by $m(p)=2\alpha(\phi+\psi/\varepsilon')\mathrm{Prob}(p)$,

$$\mathrm{Prob}(p)=\frac{m(p)}{2\alpha(\phi+\psi/\varepsilon')}\geq\frac{m(p)}{t},$$

$t\geq 2$, and for every constant $c>0$ there is a sufficiently large constant c' such that

$$|C|=s\geq\frac{c't}{\varepsilon^2}\left(d\log t+\log\left(\frac{1}{\delta}\right)\right)\geq\frac{16ct}{\varepsilon^2}\left(d\log t+\log\left(\frac{2}{\delta}\right)\right).$$

Plugging these bound in Theorem 17 with $\varepsilon/4$, and the query space (P,w,Q,h) yields that, with probability at least $1-\delta/2$, we have that for all $Q\in Q(C)$,

$$\left|\sum_{p\in P}w(p)h(p,Q)-\sum_{q\in C}\frac{w(q)}{|C|\mathrm{Prob}(q)}\cdot h(q,Q)\right|\leq\frac{\varepsilon}{4}.$$

Substituting $u(q)=w(q)/(|C|\mathrm{Prob}(q))$, $v(q)=w(q)-u(q)$ for every $q\in P$, and removing the points $p\in F(Q)$ whose loss is $h(p,Q)=0$ simplify the last expression to

$$\forall Q\in Q(C):\left|\sum_{p\in P\setminus F}v(p)h(p,Q)\right|\leq\frac{\varepsilon}{4}.$$

Assume that the last equation indeed holds (which happens with probability at least $1-\delta/2$). By this and the definition of g, for every $Q\in Q(C)$, (13) is bounded by

$$\begin{aligned}
\left|\sum_{p\in P\setminus F}v(p)(g(p,Q)-g(p',Q))\right| &= \left|\sum_{p\in P\setminus F}v(p)h(p,Q)\right|\cdot\sum_{q\in P}w(q)g(q,Q)\\
&\leq\frac{\varepsilon}{4}\sum_{q\in P}w(q)g(q,Q). && (19)
\end{aligned}$$

Bound on (14)**:** Since $|v(p)|=|w(p)-u(p)|\le w(p)+u(p)$, and using the triangle inequality

$$\left|\sum_{p\in F}\frac{v(p)(g(p,Q)-g(p',Q))}{\sum_{q\in P}w(q)g(q,Q)}\right|\le\sum_{p\in F}\frac{|v(p)|\cdot|g(p,Q)-g(p',Q)|}{\sum_{q\in P}w(q)g(q,Q)}$$

$$\le\sum_{p\in F}\frac{(w(p)+u(p))|g(p,Q)-g(p',Q)|}{\sum_{q\in P}w(q)g(q,Q)}. \tag{20}$$

To bound the numerator,

$$|g(p,Q)-g(p',Q)|\le\phi g(p,p')+\psi g(p',Q)\le(\phi\varepsilon'+\psi)g(p',Q)$$

$$\le\frac{\varepsilon g(p',Q)}{4\rho(\alpha+1)}, \tag{21}$$

where the first inequality is by Lemma 36, the second holds since $p\in F$, and the last inequality is by (8).

Our (α,β)-assignment approximates the sum of distances to a query up to an additive error as follows.

$$\sum_{q\in P}w(q)g(q',Q)\le\sum_{q\in P}w(q)\rho(g(q',q)+g(q,Q)) \tag{22}$$

$$=\rho\sum_{q\in P}w(q)g(q',q)+\rho\sum_{q\in P}w(q)g(q,Q)$$

$$\le\rho(\alpha+1)\sum_{q\in P}w(q)g(q,Q), \tag{23}$$

where (22) is by Lemma 31, and (23) is by (18). Hence,

$$\left|\sum_{p\in F}\frac{v(p)(g(p,Q)-g(p',Q))}{\sum_{q\in P}w(q)g(q,Q)}\right|\le\sum_{p\in F}\frac{(w(p)+u(p))|g(p,Q)-g(p',Q)|}{\sum_{q\in P}w(q)g(q,Q)} \tag{24}$$

$$\le\sum_{p\in F}\frac{(\rho\alpha+1)(w(p)+u(p))|g(p,Q)-g(p',Q)|}{\sum_{q\in P}w(q)g(q',Q)} \tag{25}$$

$$\le\frac{\varepsilon}{4}\sum_{p\in F}\frac{(w(p)+u(p))g(p',Q)}{\sum_{q\in P}w(q)g(q',Q)} \tag{26}$$

$$\le\frac{\varepsilon}{4}\sum_{p\in P}\frac{w(p)g(p',Q)}{\sum_{q\in P}w(q)g(q',Q)}+\frac{\varepsilon}{4}\sum_{p\in P}\frac{u(p)g(p',Q)}{\sum_{q\in P}w(q)g(q',Q)}$$

$$=\frac{\varepsilon}{4}+\frac{\varepsilon}{4}\sum_{p\in P}\frac{u(p)g(p',Q)}{\sum_{q\in P}w(q)g(q',Q)}, \tag{27}$$

where (24) is by (20), (25) holds by (22), (26) holds by (21), and (27) holds since $F\subseteq P$.

It is left to bound the rightmost term in (27). Let $z:P\times B\to[0,\infty)$ such that for every $b\in B$ and $p\in P_b$

$$z(p,b)=\begin{cases}\frac{1}{\sum_{q\in P_b}w(q)} & p\in P_b\\ 0 & p\in P\setminus P_b.\end{cases} \tag{28}$$

The last term in (27) is then bounded by

$$\begin{aligned}
\sum_{p\in P}\frac{u(p)g(p',Q)}{\sum_{q\in P}w(q)g(q',Q)} &= \sum_{b\in B}\sum_{p\in P_b}\frac{u(p)g(p',Q)}{\sum_{q\in P}w(q)g(q',Q)} \qquad (29)\\
&= \sum_{b\in B}\frac{\sum_{q\in P_b}w(q)g(q',Q)}{\sum_{q\in P}w(q)g(q',Q)}\sum_{p\in P_b}\frac{u(p)g(p',Q)}{\sum_{q\in P_b}w(q)g(q',Q)} \qquad (30)\\
&= \sum_{b\in B}\frac{\sum_{q\in P_b}w(q)g(q',Q)}{\sum_{q\in P}w(q)g(q',Q)}\sum_{p\in P_b}\frac{u(p)}{\sum_{q\in P_b}w(q)} \qquad (31)\\
&= \sum_{b\in B}\frac{\sum_{q\in P_b}w(q)g(q',Q)}{\sum_{q\in P}w(q)g(q',Q)}\sum_{p\in P}u(p)z(p,b),
\end{aligned}$$

where (29) holds since $P=\bigcup_{b\in B}P_b$, in (30) we simply multiplied and divided by $\sum_{q\in P_b}w(q)g(q',Q)$, and (31) holds since $p'=q'$ for every $p,q\in P_b$.

For every $b\in B$, we have

$$\begin{aligned}
\sum_{p\in P}u(p)z(p,b) &= \sum_{p\in P}w(p)z(p,b)-\sum_{p\in P}v(p)z(p,b) \qquad (32)\\
&= \sum_{p\in P_b}w(p)z(p,b)-\sum_{p\in P}v(p)z(p,b)\\
&= 1-\sum_{p\in P}v(p)z(p,b)\leq 1+\left|\sum_{p\in P}v(p)z(p,b)\right|, \qquad (33)
\end{aligned}$$

where (32) holds since $v(p)=w(p)-u(p)$, and (33) is by definition (28) of z.

Let $b\in B$, $t'=2|B|$, and for every $p\in P$, let $m(p)=w(p)z(p,b)$. Hence, for every $p\in P$,

$$\begin{aligned}
\mathrm{Prob}(p) &\geq \frac{w(p)}{2|B|\sum_{q\in\mathcal{B}^{-1}(p)}w(q)} = \frac{w(p)z(p,\mathcal{B}(p))}{2|B|}\\
&\geq \frac{w(p)z(p,b)}{2|B|} = \frac{m(p)}{t'},
\end{aligned}$$

and for every constant $c\geq 1$ there is a sufficiently large c' such that

$$|C|=s\geq 4ct'\left(\log t'+\log\left(\frac{|B|}{\delta}\right)\right)\in O(|B|)\left(\log|B|+\log\left(\frac{2|B|}{\delta}\right)\right)$$

Substituting the query space $(P,w,\{b\},z)$, $\varepsilon=1/2$, $d=1$, and $\delta/|B|$ instead of δ in Theorem 17, yields that with probability at least $1-\delta/(2|B|)$, we have

$$\left|\sum_{p\in P}w(p)z(p,b)-\sum_{q\in C}u(q)z(q,b)\right|\leq\frac{1}{2}. \qquad (34)$$

Assume the event that (34) holds for every $b \in B$ occurs, which happens with probability at least $\delta/2$, by the union bound[2]. Plugging (34) in (33) yields

$$\sum_{p\in P} u(p)z(p,b) \le 1 + \left|\sum_{p\in P} v(p)z(p,b)\right| \le 2. \tag{35}$$

Combining the last inequalities bounds (14) with probability at least $1-\delta/2$, as

$$\left|\sum_{p\in F} \frac{v(p)(g(p,Q)-g(p',Q))}{\sum_{q\in P} w(q)g(q,Q)}\right| \le \frac{\varepsilon}{4} + \frac{\varepsilon}{4}\sum_{p\in P}\frac{u(p)g(p',Q)}{\sum_{q\in P} w(q)g(q',Q)} \tag{36}$$

$$= \frac{\varepsilon}{4} + \frac{\varepsilon}{4}\sum_{b\in B}\frac{\sum_{q\in P_b} w(q)g(q',Q)}{\sum_{q\in P} w(q)g(q',Q)}\sum_{p\in P} u(p)z(p,b) \tag{37}$$

$$\le \frac{\varepsilon}{4} + \frac{\varepsilon}{2} = \frac{3\varepsilon}{4}, \tag{38}$$

where (36) holds by (27), (37) by (29), and (38) by (35).

Finally, replacing (13) and (14) with (19) and (38) respectively, proves that, with probability at least $1-\delta/2-\delta/2 = 1-\delta$ we have

$$\begin{aligned}\left|\sum_{p\in P} v(p)(g(p,Q)-g(p',Q))\right| &\le \left|\sum_{p\in P\setminus F} v(p)(g(p,Q)-g(p',Q))\right| + \left|\sum_{p\in F} v(p)(g(p,Q)-g(p',Q))\right| \\ &\le \frac{\varepsilon}{4}\sum_{q\in P} w(q)g(q,Q) + \frac{3\varepsilon}{4}\sum_{q\in P} w(q)g(q,Q) \\ &= \varepsilon \sum_{q\in P} w(q)g(q,Q).\end{aligned}$$

By this and (12), it follows that (C,u) approximates X as desired. □

We now handle the specific case where $g: X^2 \to [0,\infty)$ is a (ρ,ψ,ϕ)-pseudo distance function.

Theorem 39 *Consider the variables in Theorem 38, where s is replaced by*

$$s \ge \frac{c' t(1+\rho(\alpha+1))^2}{\varepsilon^2}\left(d\log t + \log\left(\frac{1}{\delta}\right)\right) + \frac{c'|B|(1+\rho(\alpha+1))^2}{\varepsilon^2}\left(\log|B| + \log\left(\frac{1}{\delta}\right)\right).$$

Let (C,u) be the output of a call to algorithm Coreset$(P,w,\mathcal{B},s)$*; see Algorithm 2.*

Then, $C \subseteq P$, $u: C \to [0,\infty)$, and with probability at least $1-\delta$, (C,u) is an ε-coreset of size s for (P,w,Q,g).

Proof : Let $\varepsilon' = \varepsilon/(1+\rho(\alpha+1))$. After replacing δ with $\delta/2$ and ε with ε' in Theorem 38, we obtain that with probability at least $1-\delta/2$,

$$\forall X \in Q(C): \left|\sum_{p\in P} w(p)g(p,X) - \sum_{q\in C\cup B} u(q)g(q,X)\right| \le \varepsilon' \sum_{p\in P} w(p)g(p,X). \tag{39}$$

[2] Instead of using the union bound, we could simply choose B as the set of queries, δ instead of $\delta/(2|B|)$ and $d = \log|B|$. However, in this would introduce a term of $d\log t = O(\log^2|B|)$ in the coreset size compared to the current $\log|B|$ term.

Assume that this event indeed occurs and the inequality holds, and let $Q\in Q(C)$.

We will bound the error by excluding B from this coreset, i.e.,

$$\begin{aligned}
&\left|\sum_{p\in P}w(p)g(p,X)-\sum_{q\in C}u(q)g(q,X)\right|\\
&\le\left|\sum_{p\in P}w(p)g(p,X)-\sum_{q\in C\cup B}u(q)g(q,X)\right|+\left|\sum_{q\in C\cup B}u(q)g(q,X)-\sum_{p\in C}u(p)g(p,X)\right| \qquad (40)\\
&\le\varepsilon'\sum_{p\in P}w(p)g(p,X)+\left|\sum_{q\in C\cup B}u(q)g(q,X)-\sum_{p\in C}u(p)g(p,X)\right|, \qquad (41)
\end{aligned}$$

where (40) is by the triangle inequality, and (41) is by (39). The rightmost term is

$$\begin{aligned}
\left|\sum_{q\in C\cup B}u(q)g(q,X)-\sum_{p\in C}u(p)g(p,X)\right| &=\left|\sum_{q\in B}u(q)g(q,X)\right|\\
&=\left|\sum_{b\in B}g(b,X)\left(\sum_{p\in P_b}w(p)-\sum_{p\in C\cap P_b}u(p)\right)\right| \qquad (42)\\
&\le\sum_{b\in B}g(b,X)\left|\sum_{p\in P_b}w(p)-\sum_{p\in C\cap P_b}u(p)\right|,
\end{aligned}$$

where (42) is by the definition of u in Line 2 of Algorithm 2.

The bound on the rightmost term is similar to (34), after replacing the bound $1/2$ with ε', which is the reason for the largest size s of the coreset. Specifically, let $b\in B$, $t'=2|B|$, and for every $p\in P$, let $m(p)=w(p)z(p,b)$, where z is defined in (28). Hence, for every $p\in P$,

$$\text{Prob}(p)\ge\frac{w(p)}{2|B|\sum_{q\in\mathcal{B}^{-1}(p)}w(q)}=\frac{w(p)z(p,\mathcal{B}(p))}{2|B|}\ge\frac{w(p)z(p,b)}{2|B|}=\frac{m(p)}{t'},$$

and for every constant $c\ge1$ there is a sufficiently large c' such that

$$|C|=s\ge\frac{4ct'}{\varepsilon'^2}\left(\log t'+\log\left(\frac{|B|}{\delta}\right)\right)\in\frac{O(|B|)}{\varepsilon'^2}\left(\log|B|+\log\left(\frac{1}{\delta}\right)\right).$$

Substituting the query space $(P,w,\{b\},z)$, $d=1$, and $\delta/|B|$ instead of δ in Corollary 17, yields that with probability at least $1-\delta/(2|B|)$, we have

$$\left|\sum_{p\in P}w(p)z(p,b)-\sum_{q\in C}u(q)z(q,b)\right|\le\varepsilon'. \qquad (43)$$

Assume the event that (34) holds for every $b\in B$ indeed occurs, which happens with probability at least $\delta/2$. Substituting the value of $z(p,b)$ from (34) and multiplying by $\sum_{q\in P_b}w(q)$ yields

$$\forall b\in B:\left|\sum_{p\in P_b}w(p)-\sum_{q\in C\cap P_b}u(q)\right|\le\varepsilon'\sum_{q\in P_b}w(q). \qquad (44)$$

Hence,

$$\sum_{b\in B} g(b,X)\left|\sum_{p\in P_b} w(p) - \sum_{p\in C\cap P_b} u(p)\right| \le \varepsilon' \sum_{b\in B} g(b,X) \sum_{q\in P_b} w(q) \tag{45}$$

$$\le \varepsilon' \sum_{b\in B} \rho(\alpha+1) \sum_{q\in P_b} w(q)g(q,X) \tag{46}$$

$$= \varepsilon' \rho(\alpha+1) \sum_{q\in P} w(q)g(q,X),$$

where (45) is by (44), and (46) is by the property of (α,β)-assignment in (23).

Combining the previous inequalities all together yields the desired result

$$\left|\sum_{p\in P} w(p)g(p,X) - \sum_{q\in C} u(q)g(q,X)\right| \le \varepsilon' \sum_{p\in P} w(p)g(p,X) + \left|\sum_{q\in C\cup B} u(q)g(q,X) - \sum_{q\in C} u(q)g(p,X)\right| \tag{47}$$

$$\le \varepsilon' \sum_{p\in P} w(p)g(p,X) + \sum_{b\in B} g(b,X) \Big|\sum_{p\in P_b} w(p) - \sum_{p\in C\cap P_b} u(p)\Big|$$

$$\le \varepsilon'(1+\rho(\alpha+1)) \sum_{q\in P} w(q)g(q,X) \tag{48}$$

$$\le \varepsilon \sum_{q\in P} w(q)g(q,X),$$

where (47) is by (41), and (48) is by (46).

Using the union bound on previous assumptions, this holds with probability at least $1-\delta/2-\delta/2=1-\delta$. □

B.2. Positively weighted coresets

In this section, we give a construction for a coreset that is guaranteed to output positive weights associated with each sampled point.

```
Algorithm 3: Conditional-Coreset($P,w,\mathcal{B},s,\varepsilon'$); See Theorem 40
Input: Ā weighted set $(P,w)$ where $w: P \to [0,\infty)$, $(\alpha,\beta)$-assignment $\mathcal{B}: P \to B$ for $(P,w,Q,g)$, and sample size (integer) $s \ge 1$.
Output: A set $C \subseteq P$ and $u: C \times Q(C) \to [0,\infty)$.
$(C\cup B,u) :=$ Smaller-Coreset($P,w,\mathcal{B},s$)   // see Algorithm 2
for every $p \in C$ and $Q \in Q(C)$ do
    $u'(p,Q) := \begin{cases} u(p,Q) & \text{if } g(\mathcal{B}(p),Q) \le \frac{g(p,\mathcal{B}(p))}{\varepsilon'} \\ 0 & \text{otherwise} \end{cases}$.
end
return $(C\cup B,u')$   // $C\cup B$ is a multi-set
```

Theorem 40 *Consider the variables in Theorem 38. Let*

$$s \geq \frac{c't}{\varepsilon^2}\left(d\log t + \log\left(\frac{1}{\delta}\right)\right)$$

and let $(C \cup B, u')$ be the output of a call to algorithm Conditional-Coreset$(P, w, \mathcal{B}, s, \varepsilon')$*; see Algorithm 3. Then, $C \subseteq P$, $u : C \times Q(C) \to [0, \infty)$, and with probability at least $1-\delta$, we have that for all $X \in Q(C)$,*

$$\left|\sum_{p \in P} w(p)g(p,X) - \sum_{q \in C \cup B} u'(q,X)g(q,X)\right| \leq \varepsilon \sum_{p \in P} w(p)g(p,X).$$

Proof : The proof is the same as the proof of Theorem 38 except for replacing $u(p)$ with $u'(p,Q)$ everywhere, and replacing the bound on (24) by

$$\begin{aligned}\left|\sum_{p \in F} \frac{v(p)(g(p,Q) - g(p',Q))}{\sum_{q \in P} w(q)g(q,Q)}\right| &\leq \frac{\varepsilon}{4} \sum_{p \in F} \frac{(w(p) + u'(p,Q))g(p',Q)}{\sum_{q \in P} w(q)g(q',Q)} \\ &= \frac{\varepsilon}{4} \sum_{p \in F} \frac{w(p)g(p',Q)}{\sum_{q \in P} w(q)g(q',Q)} \leq \frac{\varepsilon}{4}.\end{aligned}$$

where the first inequality is similar to (26), and the equality is since $u'(p,Q) = 0$ for every

$$p \in F = \left\{ p \in P \mid g(\mathcal{B}(p), Q) > \frac{g(p, \mathcal{B}(p))}{\varepsilon'} \right\}.$$

□