A Proofs of Section 3

Proof of Lemma 2.2. Fix an i and consider a grid $\mathcal{G}_i$. For each center z_j, denote $X_{j,\alpha}$ the indicator random variable for the event that the distance to the boundary in dimension α of grid $\mathcal{G}_i$ is at most $\Delta/(2^{i+1}d)$. Since in each dimension, if the center is close to a boundary, it contributes a factor at most 2 to the total number of close cells. It follows that the number of cells that have distance at most $\Delta/(2^{i+1}d)$ to z_j is at most,

$$N = 2^{\sum_{\alpha=1}^{d} X_{j,\alpha}}.$$

Defining $Y_{j,\alpha} = 2^{X_{j,\alpha}}$, we obtain,

$$E[N] = E\left[\prod_{\alpha=1}^{d} Y_{j,\alpha}\right] = \prod_{\alpha=1}^{d} E[Y_{j,\alpha}].$$

We have that $Pr[X_{j,\alpha} = 1] \leq 1/d$ and so we get,

$$E[Y_{j,\alpha}] \leq E\left[1 + X_{j,\alpha}\right] = 1 + E[X_{j,\alpha}] \leq 1 + 1/d.$$

Thus $E(N) = \prod_{\alpha=1}^{d} E[Y_{j,\alpha}] \leq (1+1/d)^d \leq e$. Thus the expected number of center cells is at most $(1+1/d)^d|Z| \leq e|Z|$. By Markov's inequality, the probability that we have more than $e|Z|(L+1)/\rho$ center cells in each grid is at most $\rho/(L+1)$. By a union bound, the probability that in any grid we have more than $e|Z|(L+1)/\rho$ center cells is at most ρ. □

Proof of Lemma 3.4. Let $L' = L + 1$. Note that for each point $p \in P$, $|d(c_p^i, Z) - d(c_p^{i-1}, Z)| \leq \Delta\sqrt{d}/2^i$. Denote $\hat{A} = \sum_{p\in S}(d(c_p^i, Z) - d(c_p^{i+1}, Z))/\pi_i$ and $A = \sum_{p\in\cup\{\mathcal{C}\in C\}}(d(c_p^i, Z) - d(c_p^{i+1}, Z))$. We have that $E(\hat{A}) = A$. Let

$$X_p := \mathbb{I}_{p\in S}(d(c_p^i, Z) - d(c_p^{i+1}, Z))/\pi_i,$$

where $\mathbb{I}_{p\in S}$ is the indicator function that $p \in S$. Then we have that $Var(X_p) \leq \Delta^2 d/(4^i\pi_i)$ and $b := \max_p |X_p| \leq \Delta\sqrt{d}/(2^i\pi_i)$. By Bernstein's inequality,

$$\begin{aligned} Pr\left[|\hat{A} - A| > t\right] &\leq 2e^{-\frac{t^2}{2|P|\Delta^2 d/(4^i\pi_i)+2bt/3}} \\ &\leq 2e^{-\frac{3\times 2^{i-1}t^2\pi_i}{(\beta\mathsf{OPT}+\frac{t}{3})\Delta\sqrt{d}}}. \end{aligned} \tag{7}$$

By setting $t = \epsilon\mathsf{OPT}/L'$, we have that

$$Pr\left[|\hat{A} - A| > \frac{\epsilon\mathsf{OPT}}{L'}\right] \leq 2e^{-\ln\frac{2L'\Delta^{dk}}{\rho}} \leq \frac{\rho}{L'\Delta^{dk}}. \tag{8}$$

Thus with probability $1-\rho/(L'\Delta^{dk})$, $\hat{A}$ is an $\epsilon\mathsf{OPT}/L'$ additive approximation to the sum $\sum_{p\in P} d(c_p^i, Z)-d(c_p^{i+1}, Z)$. □

B Proof of Theorem 3.6

Before we prove this theorem, we first present Lemma B.1 and Lemma B.2. In Algorithm 1, for each level $i \in [0, L]$, let H_i be the set of cells in $\mathcal{G}_i$ whose frequencies are returned by `HEAVY-HITTER` in the `RetrieveFrequency` procedure. For each $\mathcal{C} \in H_i$, let $\widehat{|\mathcal{C}|}$ be the returned frequency of $\mathcal{C}$. Let H_i' be the set of cells in $\mathcal{G}_i$ whose frequencies are returned by a `K-set` in the `RetrieveFrequency` procedure. Then H_i and H_i' are complements in $\mathcal{G}_i$.

Lemma B.1. *Let $L' = L+1$. Fix $\epsilon, \rho \in (0, 1/2)$. Let $Z^* \subset [\Delta]^d$ be a set of optimal k-centers for the k-median problem of the input point set. For each $i \in [0, L]$, if at most ekL'/ρ cells $\mathcal{C}$ in $\mathcal{G}_i$ satisfy $d(\mathcal{C}, Z^*) \leq \Delta/(2^{i+1}d)$, then with probability $1 - \rho/L'$, the following two statements hold:*

1. $\left|\sum_{\mathcal{C}\in H_i}(\widehat{|\mathcal{C}|} - |\mathcal{C}|)\left(d(c(\mathcal{C}), Z) - d(c(\mathcal{C}^P), Z)\right)\right| \leq \frac{\epsilon\mathsf{OPT}}{2L'}$ *for every* $Z \subset [\Delta]^d$.
2. $\sum_{\mathcal{C}\in H_i'} |\mathcal{C}|diam(\mathcal{C}) \leq \beta\mathsf{OPT}$ *for* $\beta = 3d^{3/2}$

Proof of Lemma B.1. Let $L' = L+1$. Fix a value $i \in [0, L]$ and then $W = \Delta/2^i$ is the width of a cell in $\mathcal{G}_i$. Since at most ekL'/ρ cells in $\mathcal{G}_i$ satisfy $d(\mathcal{C}, Z^*) \leq W/(2d)$, then of the remaining cells, at most $2kL'/\rho$ cells can contain more than $\rho d\mathsf{OPT}/(WkL')$ points. This is because each such cells contribute at least $\frac{\rho d\mathsf{OPT}}{WkL'}\frac{W}{2d} = \frac{\rho\mathsf{OPT}}{2kL'}$ to the cost which sums to OPT. Therefore, at most $(e+2)L'k/\rho$ cells contain more than $\rho d\mathsf{OPT}/(WkL')$ points.

The number of cells in grid $\mathcal{G}_i$ is at most $N = (1+2^i)^d$ (and perhaps as few as 2^{id}, depending on the random vector v), so `HEAVY-HITTER` receives cells of at most N types. Enumerating all cells $C \in \mathcal{G}_i$ such that $|C_j| \geq |C_{j+1}|$, define $f_j = |C_j|$. Algorithm 1 sets $k' = (e+2)L'k/\rho$, and the additive error of the estimator of f_i of `HEAVY-HITTER` is given

by $\epsilon'\sqrt{\sum_{j=k'+1}^{N} f_j^2}$. We know that for all $j > k'$ the value $f_j \le \rho d\mathsf{OPT}/(WkL')$. Moreover, the sum $\sum_{j=k'+1}^{N} f_j \le 2d\mathsf{OPT}/W$ because each point is at distance at least $W/(2d)$ to a point of Z^*. Under these two restraints, the grouping of maximal error is with $f_j = \rho d\mathsf{OPT}/(WkL')$ for $k' < j \le k' + 2kL'/\rho$ and $f_j = 0$ for $j > k' + 2kL'/\rho$. Then the additive error becomes $\epsilon'\sqrt{2\rho/(kL')}d\mathsf{OPT}/W$.

The error from a single cell $\mathcal{C}_j$ is at most $|f_j - \hat{f}_j|\sqrt{d}W$, and HEAVY-HITTER gaurantees with probability $1-\delta$ that $|f_j - \hat{f}_j| \le \epsilon'\sqrt{2\rho/(kL')}d\mathsf{OPT}/W$ for every j. Therefore to ensure total error over all k' cells is bounded by $\epsilon\mathsf{OPT}/(2L')$, we set $\epsilon' \le \epsilon\sqrt{\frac{\rho}{8(2+e)^2kd^3L'^3}}$. Setting $\delta = \rho/L'$, the above bound holds with probability at least $1-\rho/L'$.

For the second claim, we must bound $\sum_{\mathcal{C}\in H_i'}|\mathcal{C}|$. H_i consists of the top k' cells when ordered by value of $\hat{f}_j$. This may differ from the top k' cells when ordered by value of f_j, but if j and j' change orders between these two orderings then $|f_j - f_{j'}| \le 2\epsilon'\sqrt{2\rho/(kL')}d\mathsf{OPT}/W$. Since the sum may swap up to k' indices, the difference is bounded by $2k'\epsilon'\sqrt{2\rho/(kL')}d\mathsf{OPT}/W$. By setting $\epsilon' \le \sqrt{\frac{\rho}{8(2+e)^2dkL'}}$, we can ensure that the difference is at most $d\mathsf{OPT}/W$. We know that $\sum_{j=k'+1}^{N} f_j \le 2d\mathsf{OPT}/W$, and so $\sum_{\mathcal{C}\in H_i'}|\mathcal{C}| \le 3d\mathsf{OPT}/W$. For all cells $\mathcal{C}\in\mathcal{G}_i$, $\mathrm{diam}(\mathcal{C}) = \sqrt{d}W$. Therefore $\sum_{\mathcal{C}\in H_i'}|\mathcal{C}|\mathrm{diam}(\mathcal{C}) \le 3d^{3/2}\mathsf{OPT}$.

□

Lemma B.2. *Let $L' = L+1$. In Algorithm 1, fixing $\epsilon,\rho\in(0,1/2)$, $o\in O$ and $i\in[0,L]$, if $\mathsf{OPT}/2 \le o \le \mathsf{OPT}$, then with probability $1-\rho/(L'\Delta^{kd})$, at most $\frac{(2+e)L'k}{\rho} + \frac{24d^4L'^3k}{\epsilon^2}\ln\frac{1}{\rho}$ cells of $\mathcal{G}_i$ contain a point of $S_{i,o}$.*

Proof. Similar to the proof of Lemma B.1, there are at most $k' = (2+e)L'k/\rho$ cells $\mathcal{C}$ in $\mathcal{G}_i$ that satisfy $|\mathcal{C}| \ge \rho d\mathsf{OPT}/(Wk)$ and/or $d(\mathcal{C}, Z^*) \le W/(2d)$. Considering the other cells, together they contain at most $2d\mathsf{OPT}/W$ points. So by a Chernoff bound, with probability $1-\rho/(L'\Delta^{kd})$ at most $O(2d\pi_{i,o}\mathsf{OPT}/(W\rho)) \le 24d^4L'^3k\ln\frac{1}{\rho}/\epsilon^2$ points are sampled. The claim follows since each non-empty cell must contain at least one point. □

Proof of Theorem 3.6. Let $L' = L+1$. W.l.o.g. we assume $\rho \ge \Delta^{-d}$, since otherwise we store the entire set of points and the theorem is proved. By Lemma 2.2, with probability at least $1-\rho$, for every level $i\in[0,L]$, at most ekL'/ρ cells $\mathcal{C}$ in $\mathcal{G}_i$ satisfy $d(C,Z^*) \le \Delta/(2^{i+1}d)$. Conditioning on this event, we will show 1) in the query phase, if $o^* \le \mathsf{OPT}$, then with probability at least $1-4\rho$, S is the desired coreset; 2) there exists $o \le \mathsf{OPT}$ in the guesses $O = \{1,2,4,\dots,\Delta^{d+1}\}$ such that with probability $1-4\rho$, none of the K-set structures return Nil. 1) and 2) guarantee the correctness of the algorithm. Note that one can always rescale ρ to $\rho/9$ to achieve the correct probability bound. Finally, we will bound the space, update time and query time of the algorithm.

To show 1), we first note that the coreset size is at most $O(KL)$ as desired. Then by Lemma 3.2, we only need to show that with probability at least $1-4\rho$, for any k-set $Z\subset[\Delta]^d$ and any level $i\in[-1,L]$,

$$|\mathrm{cost}(\mathcal{G}_i,Z) - \widehat{\mathrm{cost}}(\mathcal{G}_i,Z)| \le \frac{\epsilon\mathsf{OPT}}{L'},$$

where the value of each $\widehat{|C|}$ is returned by RetrieveFrequency. For each level i, we denote C_i as the set of cells that gets frequency from a HEAVY-HITTER instances in the RetrieveFrequency procedure, and $S_i = \{p\in\mathcal{C} : \mathcal{C}\notin C_i, h_{o^*,i}(p) = 1\}$ be the set of points sampled in the rest of cells. Since $KS_{o^*,i}$ does not return Fail, then for each $\mathcal{C}\in\mathcal{G}_i\backslash C_i$, $\widehat{|\mathcal{C}|} = |S_i\cap\mathcal{C}|/\pi_i(o^*)$. Fix a k-set $Z\subset[\Delta]^d$, we rewrite the cost as,

$$\widehat{\mathrm{cost}}(\mathcal{G}_i,Z) = \sum_{\mathcal{C}\in C_i}\widehat{|\mathcal{C}|}\left(d(c(\mathcal{C}),Z) - d(c(\mathcal{C}^P),Z)\right) + \sum_{p\in S_i}(d(c_p^i,Z) - d(c_p^{i-1},Z)))/\pi_i(o^*),$$

where $\mathcal{C}^P$ is the parent cell of $\mathcal{C}$ in grid $\mathcal{G}_{i-1}$. By Lemma B.1 we have that, with probability at least $1-\rho/L'$, for every $Z\subset[\Delta]^d$,

$$\left|\sum_{\mathcal{C}\in C_i}\widehat{|\mathcal{C}|}\left(d(c(\mathcal{C}),Z) - d(c(\mathcal{C}^P),Z)\right) - \sum_{\mathcal{C}\in C_i}|\mathcal{C}|\left(d(c(\mathcal{C}),Z) - d(c(\mathcal{C}^P),Z)\right)\right| \le \frac{\epsilon\mathsf{OPT}}{2L'},$$

and that, $\sum_{\mathcal{C}\in\mathcal{G}_i\backslash C_i}|\mathcal{C}|\mathrm{diam}(\mathcal{C}) \le 3d^{3/2}\mathsf{OPT}$. Conditioning on this event, by Lemma 3.4, with probability at least $1-\rho/(L'\Delta^{kd})$,

$$\left|\sum_{p\in S_i}(d(c_p^i,Z) - d(c_p^{i-1},Z)))/\pi_i - \sum_{\mathcal{C}\in\mathcal{G}_i\backslash C_i}|\mathcal{C}|\left(d(c(\mathcal{C}),Z) - d(c(\mathcal{C}^P),Z)\right)\right| \le \frac{\epsilon\mathsf{OPT}}{2L'}.$$

By a union bound, we show with probability at least $1-4\rho$, for any k-set $Z \subset [\Delta]^d$ and any level $i \in [-1, L]$,

$$|\mathrm{cost}(\mathcal{G}_i, Z) - \widehat{\mathrm{cost}}(\mathcal{G}_i, Z)| \leq \frac{\epsilon \mathsf{OPT}}{L'},$$

as desired.

To show 2), we will consider some $\mathsf{OPT}/2 \leq o \leq \mathsf{OPT}$. By Lemma B.2 with probablity at least $1-\rho/\Delta^{kd}$, the total number of cells occupied by sample points in each level is upper bounded by $K = \frac{(2+e)L'k}{\rho} + \frac{24d^4L'^3k}{\epsilon^2}\ln\frac{1}{\rho}$. Thus by the guarantee of the `K-Set` structure, with probability at least $1-\rho$, none of the $\mathtt{KS}_{o,0}, \mathtt{KS}_{o,1} \ldots, \mathtt{KS}_{o,L}$ will return `Fail`.

The memory requirement of the algorithm is determined by the L instances of `HEAVY-HITTER` and the dL^2 instances of K-set. By Theorem 3.5, each instance of `HEAVY-HITTER` requires $O\left((k' + \frac{1}{\epsilon'^2})\log\frac{N}{\delta}\log m\right)$ bits of space. Here $N \leq (1+\Delta/W)^d \leq \Delta^d$ and m is the maximum number of elements active in the stream. Since we require that at most one point exists at each location at the same time, we have that $m \leq N$. The parameters are set to $k' = (2+e)Lk/\rho$, $\epsilon' = \left(\epsilon\sqrt{\frac{\rho}{8(2+e)^2kd^3L^3}}\right)$, and $\delta = \rho/L$. This translates to a space bound of $O\left(dL + \log\frac{1}{\rho}\right)\frac{d^4L^5k}{\rho\epsilon^2}$ bits. For each `K-Set` data structure, it requires

$$O(KdL\log(KL/\rho)) = O\left(\frac{d^5L^4k}{\epsilon^2} + \frac{dkL^2}{\rho}\right)\log\frac{dkL}{\epsilon\rho}$$

bits of space. In total, there are $O(dL^2)$ `K-Set` instances and thus all `K-Set` instances cost $O\left(\frac{d^6L^6k}{\epsilon^2} + \frac{d^2kL^4}{\rho}\right)\log\frac{dkL}{\epsilon\rho}$ bits of space. By the same argument as in the offline case, the last paragraph of the proof of Theorem 3.3, the size of the coreset is at most $O((k'+K)L) = O(d^4kL^4\epsilon^{-2} + kL^2/\rho)$ points. Finally, to derandomize the fully random functions, we use Nissan's pseudorandom generator (Nisan, 1992) in a similar way used in (Indyk, 2000b). But our pseudo-random bits only need to fool the sampling part of the algorithm rather than whole algorithm. We consider an augmented streaming algorithm $\mathcal{A}$ that does exactly the same as in `CoreSet` but with all the `HEAVY-HITTER` operations removed. Thus all K-set instances will have identical distribution with the ones in algorithm `CoreSet`. $\mathcal{A}$ uses $O(KdL\log(KL/\rho))$ bits of space. To fool this algorithm, using Nissan's pseudo-random generator, the length of random seed to generate the hash functions we need is of size $O(KdL\log(KL/\rho)\log(|O|\Delta^d)) = O\left(\left(\frac{d^7kL^7}{\epsilon^2} + \frac{d^3kL^5}{\rho}\right)\log\frac{dkL}{\rho\epsilon}\right)$. This random seed is thus sufficient to be used in Algorithm `CoreSet`. Thus the total space used in the algorithm is $O\left(\left(\frac{d^7kL^7}{\epsilon^2} + \frac{d^3kL^5}{\rho}\right)\log\frac{dkL}{\rho\epsilon} + \frac{d^5kL^6}{\epsilon^2\rho}\right)$ bits.

Regarding the update time, for the `HEAVY-HITTER` operations, it requires $O(L\log N) = O(dL^2)$ time. For the K-set operations, it requires $|O|LO(\log(KL/\rho)) = dL^2\log(dkL/(\rho\epsilon))$ time. The de-randomized hash operation takes $O(dL)$ more time per update. The final query time is dominated by the `HEAVY-HITTER` data structure, which requires $\mathrm{poly}(d, k, L, 1/\epsilon)$ time. □

C Full Construction of Positively Weighted Coreset

In this section, we will introduce a modification to our previous coreset construction, which leads to a coreset with all positively weighted points. The high level idea is as follows. When considering the estimate of the number of points in a cell, the estimate is only accurate when it truly contains a large number of points. However, in the construction of the previous section, we sample from each cell of each level, even though some of the cells contain a single point. For those cells, we cannot adjust their weights from negative to positive, since doing so would introduce large error. In this section, we introduce an ending level to each point. In other words, the number of points of a cell is estimated by sampling only if it contains many points. Thus, the estimates will be accurate enough and allow us to rectify the weights to be all positive.

This section is organized as follows. We reformulate the telescope sum in Subsection 4.1, provide a different construction (still with negative weights) in Subsection 4.2, modify our different construction to output non-negative weights in Subsection 4.3, and move this construction into to the streaming setting in Subsection 4.4. For simplicity of presentation, we will use $\lambda_1, \lambda_2, \ldots$ to denote some fixed positive universal constants.

C.1 Reformulation of the Telescope Sum

Definition C.1. *A* heavy cell identification scheme $\mathcal{H}$ *is a map* $\mathcal{H} : \mathcal{G} \to \{$*heavy, non-heavy*$\}$ *such that,* $h(\mathcal{C}_{-1}) =$*heavy and for cell* $\mathcal{C} \in \mathcal{G}_i$ *for* $i \in [0, L]$

1. *if* $|\mathcal{C}| \geq \frac{2^i\rho d\mathsf{OPT}}{k(L+1)\Delta}$ *then* $\mathcal{H}(\mathcal{C}) =$ *heavy;*
2. *If* $\mathcal{H}(\mathcal{C}) =$ *non-heavy, then* $\mathcal{H}(\mathcal{C}') =$ *non-heavy for every subsell* $\mathcal{C}'$ *of* $\mathcal{C}$.
3. *For every cell* $\mathcal{C}$ *in level* L, $\mathcal{H}(\mathcal{C}) =$ *non-heavy.*

4. *For each $i \in [0, L]$, $|\{\mathcal{C} \in \mathcal{G}_i : \mathcal{H}(\mathcal{C}) = \text{heavy}\}| \leq \frac{\lambda_1 kL}{\rho}$, where $\lambda_1 \leq 10$ is a positive universal constant.*

The output for a cell not specified by the above conditions can be arbitrary. We call a cell heavy *if it is identified heavy by $\mathcal{H}$. Note that a heavy cell does not necessarily contain a large number of points, but the total number of these cells is always bounded.*

In the sequel, heavy cells are defined by an arbitrary fixed identification scheme unless otherwise specified.

Definition C.2. *Fix a heavy cell identification scheme $\mathcal{H}$. For level $i \in [-1, L]$, let $\mathcal{C}(p, i) \in \mathcal{G}_i$ be the cell in $\mathcal{G}_i$ containing p. The* ending level $l(p)$ *of a point $p \in P$ is the largest level i such that $\mathcal{H}(\mathcal{C}(p, i))$ =heavy, and $\mathcal{H}(\mathcal{C}(p, i+1))$ =non-heavy.*

Note that the ending level is uniquely defined if a heavy cell identification scheme is fixed. We now rewrite the telescope sum for p as follows,

$$p = \sum_{i=0}^{l(p)} \left(c_p^i - c_p^{i-1}\right) + c_p^L - c_p^{l(p)},$$

where $c_p^{-1} = \mathbf{0}$ and $c_p^L = p$. For arbitrary k-centers $Z \subset [\Delta]^d$, we write,

$d(p, Z) = \sum_{i=0}^{l(p)} \left(d(c_p^i, Z) - d(c_p^{i-1}, Z)\right) + d(c_p^L, Z) - d(c_p^{l(p)}, Z) + d(\mathbf{0}, Z)$

Let P_l be all the points with ending level $l(p) = l$. We now present the following lemmas.

Lemma C.3. *Let P_i be the set of points with ending level i. Let $Z^* \subset [\Delta]^d$ be a set of optimal k-centers for the k-median problem of the input point set. Assume that for each $i \in [-1, L]$, at most $ek(L+1)/\rho$ cells $\mathcal{C}$ in $\mathcal{G}_i$ satisfy $d(\mathcal{C}, Z^*) \leq \Delta/(2^{i+1}d)$. Then*

$$|P_i| \cdot \frac{\Delta\sqrt{d}}{2^i} \leq \lambda_2 d^{3/2} \mathit{OPT},$$

where $\lambda_2 > 0$ is a universal constant.

Before we prove this lemma, we first introduce the following lemmas to bound the cells with a large number of points.

Lemma C.4. *Assume that for each $i \in [0, L]$, at most $ek(L+1)/\rho$ cells $\mathcal{C}$ in $\mathcal{G}_i$ satisfy $d(\mathcal{C}, Z^*) \leq \Delta/(2^{i+1}d)$. Then for any $r > 0$ there are at most $\frac{(e+2r)k(L+1)}{\rho}$ cells that satisfy $|\mathcal{C}| \geq \frac{2^i \rho d \mathsf{OPT}}{rk(L+1)\Delta}$.*

Proof of Lemma C.4. Let $L' = L + 1$. Fix a value $i \in [0, L]$ and then $W = \Delta/2^i$ is the width of a cell in $\mathcal{G}_i$. Since at most ekL/ρ cells in $\mathcal{G}_i$ satisfy $d(\mathcal{C}, Z^*) \leq W/(2d)$, then of the remaining cells, each contribute at least $\frac{\rho d \mathsf{OPT}}{rWkL'} \frac{W}{2d} = \frac{\rho \mathsf{OPT}}{2rkL'}$ to the cost, and the cost of these cells is at most OPT. Therefore there can be at most $2rL'k/\rho$ cells such that $d(\mathcal{C}, Z^*) > W/(2d)$. Along with the at most ekL'/ρ cells (by the assumption) such that $d(\mathcal{C}, Z^*) \leq W/(2d)$, there are at most $(e + 2r)L'k/\rho$ cells that contain at least $\rho d \mathsf{OPT}/(rWkL')$ points. □

Lemma C.5. *Assume that for each $i \in [0, L]$, at most $ek(L+1)/\rho$ cells $\mathcal{C}$ in $\mathcal{G}_i$ satisfy $d(\mathcal{C}, Z^*) \leq \Delta/(2^{i+1}d)$. Then for $i \in [-1, L]$, the points of P_i can be partitioned to at most $k' = \frac{2(e+6)k(L+1)}{\rho}$ groups, $G_1, G_2, \ldots, G_{k'}$, such that for each $j \in [k']$, there exists a $\mathcal{C} \in \mathcal{G}_i$, such that $G_j \in \mathcal{C}$, $|G_j| < 5\frac{2^{i-1}\rho d \mathsf{OPT}}{k(L+1)\Delta}$.*

Proof of Lemma C.5. Let $L' = L + 1$. For each heavy cell in $\mathcal{G}_i$, if the number of points falling into its non-heavy subcells (in $\mathcal{G}_{i+1}$) is less than $\frac{2^{i-1}\rho d \mathsf{OPT}}{kL'\Delta}$, we group all these subcells into a single group. Let the groups formed this way be called type I, and by Property 4 of Definition 4.1 there are at most $(e + 4)kL'/\rho$ type I groups.

For each of the remaining heavy cells in $\mathcal{G}_i$, we group its subcells into groups such that each group contains a number of points in the interval $\left[\frac{2^{i-1}\rho d \mathsf{OPT}}{kL'\Delta}, 5\frac{2^{i-1}\rho d \mathsf{OPT}}{kL'\Delta}\right)$. This can be done since each non-heavy subcell contains less than $\frac{2^{i+1}\rho d \mathsf{OPT}}{kL'\Delta} = 4\frac{2^{i-1}\rho d \mathsf{OPT}}{kL'\Delta}$ points, and the total number of points contained in them is at least $\frac{2^{i-1}\rho d \mathsf{OPT}}{kL'\Delta}$ (otherwise we would have formed a type I group). Let the groups formed this way be called type II. By the assumption of of Lemma C.3, at most $\frac{ekL'}{\rho}$ of these non-heavy subcells are within distance $\frac{\Delta}{2^{i+2}d}$ from an optimal center of Z^*. Since each type II group contains at least $\frac{2^{i-1}\rho d \mathsf{OPT}}{kL'\Delta}$ points, by the same argument as in Lemma C.4, the number of type II groups further than distance $\frac{\Delta}{2^{i+2}d}$ from an optimal center is at most $\frac{8kL'}{\rho}$. We conclude that,

$$k' \leq \frac{(e+4)kL'}{\rho} + \frac{(e+8)kL'}{\rho}.$$

□

Proof of Lemma C.3. Let $L' = L+1$. Fix a value $i \in [-1, L]$ and then $W = \Delta/2^i$ is the upper bound of the width of a cell in $\mathcal{G}_i$. Let $G_1, G_2, \ldots G_{k'}$ be group of points satisfying Lemma C.5. Thus, $\sum_{p\in P_l} \frac{\Delta\sqrt{d}}{2^i} \leq \sum_{j\in[k']} \frac{2^{i+1}\rho d\mathsf{OPT}}{kL'\Delta} \cdot \frac{\Delta\sqrt{d}}{2^i} \leq \frac{\lambda' kL'}{\rho} \cdot \frac{2^{i+1}\rho d\mathsf{OPT}}{kL'\Delta}\frac{\Delta\sqrt{d}}{2^i} \leq \lambda_2 d^{3/2}\mathsf{OPT}$ for some universal constants λ' and λ_2. □

Proof of Proposition C.10. First notice that the weighted set satisfies the about condition is an ϵ-coreset. If we replace each $|\widehat{\mathcal{C}_i}|$ by the exact number of points in $|\mathcal{C}_i|$, then the new weighted set is an $(\epsilon/2)$-coreset. For each $\mathcal{C} \in \mathcal{G}$, let $b_{\mathcal{C}}$ be the new value returned by the algorithm, and b_q is the new value of a point $q \in S$. The error of the cost introduced is at most,

$$A = \sum_{i=0}^{L}\left(\sum_{\mathcal{C}\in\mathcal{G}_i:\text{ heavy}} ||\widehat{\mathcal{C}}| - b_{\mathcal{C}}| + \sum_{p\in S_{i-1}}\left|\left(\frac{1}{\pi_{i-1}} - b_p\right)\right|\right)\frac{\Delta\sqrt{d}}{2^i}.$$

By the procedure, the new value of a cell is always smaller than its original value, thus

$$A = \sum_{i=0}^{L}\left(\sum_{\mathcal{C}\in\mathcal{G}_i:\text{ heavy}} \widehat{|\mathcal{C}|} - b_{\mathcal{C}} + \sum_{p\in S_{i-1}}\left(\frac{1}{\pi_{i-1}} - b_p\right)\right)\frac{\Delta\sqrt{d}}{2^i} = \sum_{i=0}^{L} g_i,$$

where

$$g_i = \left(\sum_{\mathcal{C}\in\mathcal{G}_i:\text{heavy}} \widehat{|\mathcal{C}|} - b_{\mathcal{C}} + \sum_{p\in S_{i-1}}\frac{1}{\pi_{i-1}} - b_p\right)\frac{\Delta\sqrt{d}}{2^i}.$$

Let

$$f_i = \left(\sum_{\mathcal{C}\in\mathcal{G}_i:\text{heavy}}\left||\mathcal{C}| - \widehat{|\mathcal{C}|}\right| + \sum_{\mathcal{C}'\in\mathcal{G}_{i-1}:\text{heavy}}\left|\frac{|S_{i-1}\cap\mathcal{C}'|}{\pi_{i-1}} - |P_{i-1}\cap\mathcal{C}'|\right|\right)\frac{\Delta\sqrt{d}}{2^i}.$$

Thus $f_i \leq \epsilon\mathsf{OPT}/L$ by choosing appropriate λ_6. Now consider heavy cell $\mathcal{C} \in \mathcal{G}_i$, let $s_{\mathcal{C}} = \left|b_{\mathcal{C}} - \sum_{\mathcal{C}\in\mathcal{G}_{i+1}:\text{heavy}}\widehat{|\mathcal{C}|} - \frac{|S_i\cap\mathcal{C}|}{\pi_i}\right|$. Then,

$$\begin{aligned} s_{\mathcal{C}} &= \left|b_{\mathcal{C}} - \widehat{|\mathcal{C}|} + \widehat{|\mathcal{C}|} - |\mathcal{C}| - \sum_{\mathcal{C}'\in\mathcal{G}_{i+1}:\text{heavy}}(\widehat{|\mathcal{C}'|} - |\mathcal{C}'|) - \left(\frac{|S_i\cap\mathcal{C}|}{\pi_i} - |P_i\cap\mathcal{C}|\right)\right| \\ &\leq \left|b_{\mathcal{C}} - \widehat{|\mathcal{C}|}\right| + \left|\widehat{|\mathcal{C}|} - |\mathcal{C}|\right| + \sum_{\mathcal{C}'\in\mathcal{G}_{i+1}:\text{heavy}}\left|\widehat{|\mathcal{C}'|} - |\mathcal{C}'|\right| + \left|\frac{|S_i\cap\mathcal{C}|}{\pi_i} - |P_i\cap\mathcal{C}|\right|. \end{aligned} \tag{9}$$

Then

$$g_i = \sum_{\mathcal{C}\in\mathcal{G}_{i-1}} s_{\mathcal{C}}\frac{\Delta\sqrt{d}}{2^i} + \left(\sum_{p\in S_{i-1}}\frac{1}{\pi_{i-1}} - b_p\right)\frac{\Delta\sqrt{d}}{2^i} \leq \frac{1}{2}g_{i-1} + \frac{1}{2}f_{i-1} + f_i. \tag{10}$$

Since $g_{-1} = f_{-1} = 0$, thus

$$g_i \leq f_i + 3\sum_{j=0}^{i-1} 2^{j-i}f_j, \text{ and } \sum_{i=0}^{L} g_i \leq \sum_{i=0}^{L} f_i(1 + \sum_{j=1}^{i}\frac{3}{2^j}) \leq 4\sum_{i=1}^{L} f_i \leq 4\epsilon\mathsf{OPT}.$$ □

Remark C.6. *The multiset of centers of heavy cells with each assigned a weight of the number of points in the cell is a $O(d^{3/2})$-coreset. This can be easily seen by removing the term of $d(c_p^L, Z) - d(c_p^{l(p)}, Z)$ from Equation (C.1) together with Lemma C.3, which bounds the error introduced by this operation.*

C.2 The New Construction (with arbitrary weights)

For these heavy cells, we use HEAVY-HITTER algorithms to obtain accurate estimates of the number of points in these cells, thus providing a *heavy cell identification scheme*. For the non-heavy cells, we only need to sample points from the bottom level, $\mathcal{G}_L$, but with a different probability for points with different ending levels. We present the following lemma that governs the correctness of sampling from the last level.

Lemma C.7. *If a set of points $P_i \subset P$ satisfies $|P_i|\Delta\sqrt{d}/(2^i) \leq \beta\mathsf{OPT}$ for some $\beta \geq 2\epsilon/(3(L+1))$, let S_i be an*

independent sample from P_i such that $p \in S_i$ with probability

$$\pi_i \geq \min\left(\frac{3a(L+1)^2\Delta\sqrt{d}\beta}{2^i\epsilon^2 o}\ln\frac{2\Delta^{kd}(L+1)}{\rho}, 1\right)$$

where $0 < o \leq a\mathsf{OPT}$ for some $a > 0$. Then for a fixed set $Z \subset [\Delta]^d$, with probability at least $1 - \rho/((L+1)Delta^{kd})$, $|\sum_{p\in S_i}(d(c_p^i, Z) - d(p, Z))/\pi_i - \sum_{p\in P_i}(d(c_p^i, Z) - d(p, Z))| \leq \frac{\epsilon\mathsf{OPT}}{L+1}$.

Proof. The proof is identical to that of Lemma 3.4. □

Lemma C.8. *Consider a set of sets $\{P_i\}_{i=0}^L$ which satisfies $|P_i|\Delta\sqrt{d}/(2^i) \leq \frac{\beta\rho}{k(L+1)}\mathsf{OPT}$ for some $\beta \geq \epsilon/(3(L+1))$. For each $i \in [0, L]$, let S_i be an independent sample from P_i with sampling probability*

$$\pi_i \geq \min\left(\frac{4a\beta k(L+1)^3\Delta\sqrt{d}}{2^i\epsilon^2\rho o}\log\frac{2}{\delta}, 1\right)$$

where $0 < o \leq a\mathsf{OPT}$ for some $a > 0$, then with probability at least $1 - \delta$,

$$\left|\frac{|S_i \cap P_i|}{\pi_i} - |P_i|\right|\frac{\Delta\sqrt{d}}{2^i} \leq \frac{\epsilon\rho\mathsf{OPT}}{k(L+1)^2}.$$

Proof of Lemma C.8. The proof is simply by Bernstein inequality. Let $t = \frac{2^i\epsilon\rho\mathsf{OPT}}{\sqrt{d}k(L+1)^2\Delta}$, $X_p := \mathbb{I}_{p\in S_i}/\pi_i$, then we have that $Var(X_p) \leq 1/\pi_i$ and $b := \max_p |X_p| \leq 1/\pi_i$. By Bernstein's inequality, for any $j \in [k']$,

$$Pr\left[\left|\frac{|P_j \cap S_i|}{\pi_i} - |P_j|\right| > t\right] \leq 2e^{-\frac{t^2}{2|C|/\pi_i + 2bt/3}} \leq \delta.$$ □

We now describe the new construction. This essentially has the same gaurantee as the simpler construction from the previous section, however the benefit here is that (as shown in the next subsection) it can be modified to output only positive weights. In the following paragraph, the estimations $\widehat{|\mathcal{C}|}$ are given as a blackbox. In proposition C.9 we specify the conditions these estimations must satisfy.

Non-Negatively Weighted Construction Fix an arbitrary heavy cell identification scheme $\mathcal{H}$. Let P_l be all the points with ending level $l(p) = l$. For each heavy cell $\mathcal{C}$, let $\widehat{|\mathcal{C}|}$ be an estimation of number of points of $|\mathcal{C}|$, we also call $\widehat{|\mathcal{C}|}$ the *value* of cell $\mathcal{C}$. For each non-heavy cell $\mathcal{C}'$, let $\widehat{|\mathcal{C}'|} = 0$. Let S be a set samples of P constructed as follows: $S = S_{-1} \cup S_0 \cup S_1, \cup \ldots \cup S_L$, where S_l is a set of i.i.d samples from P_l with probability π_l. Here π_l for $l \in [-1, L]$ is redefined as,$\pi_l = \min\left(\frac{\lambda_3 d^2\Delta L^2}{2^l\epsilon^2 o}\log\left(\frac{2L\Delta^{dk}}{\rho}\right) + \frac{\lambda_4 d^2 kL^3\Delta}{2^i\epsilon^2\rho o}\log\frac{30kL^2}{\rho^2}, 1\right)$ where $\lambda_3 > 0$ and $\lambda_4 > 0$ are universal constants. Our coreset $\mathcal{S}$ is composed by all the sampled points in S and the cell centers of heavy cells, with each point p assigned a weight $1/\pi_{l(p)}$ and for each cell center c of a heavy cell $\mathcal{C} \in \mathcal{G}_i$, the weight is,

$$\mathrm{wt}(c) = \widehat{|\mathcal{C}|} - \sum_{\substack{\mathcal{C}':\mathcal{C}'\in\mathcal{G}_{i+1},\mathcal{C}'\subset\mathcal{C},\\ \mathcal{C}' \text{ is heavy}}} \widehat{|\mathcal{C}'|} - \frac{|S_i \cap \mathcal{C}|}{\pi_i}. \tag{11}$$

For each non-heavy cell $\mathcal{C}$ except for those in the bottom level, $\mathrm{wt}(c(\mathcal{C})) = 0$. The weight of each point from S is the value of the corresponding cell in the bottom level.

We now state the following proposition for a coreset construction, which immediately serves as an offline coreset construction.

Proposition C.9. *Let $\mathcal{H}$ be an arbitrary heavy cell identification scheme. Fix $\Omega(\Delta^{-d}) \leq \rho < 1$ and for each heavy $\mathcal{C} \in \mathcal{G}_i$ in level i, $\widehat{|\mathcal{C}|}$ is an estimation of number of points in $\mathcal{C}$ with additive error at most $\frac{\epsilon}{\lambda_5 Ld^{3/2}} \cdot \frac{2^i\rho d\mathsf{OPT}}{kL\Delta}$, where $\lambda_5 > 0$ is a universal constant. Let S_l be the set of i.i.d. samples of P_l with probability $\pi_l(o)$. If $0 < o \leq \mathsf{OPT}$, then with probability at least $1 - 4\rho$, for every k-set $Z \subset [\Delta]^d$,*

$$\left|\sum_{q\in\mathcal{S}}\mathrm{wt}(q)d(q, Z) - \sum_{p\in P} d(p, Z)\right| \leq \epsilon\mathsf{OPT}.$$

And the coreset size $|\mathcal{S}|$ is

$$O\left[\frac{d^3L^4k}{\epsilon^2}\left(d+\frac{1}{\rho}\log\frac{kL}{\rho}\right)\frac{\mathsf{OPT}}{o}\right].$$

Proof of Proposition C.9. Fix a k-set $Z\subset[\Delta]^d$. First notice that,

$$\begin{aligned}\widehat{\mathrm{cost}}(Z)&=\sum_{q\in\mathcal{S}}\mathrm{wt}(q)d(q,Z)\\&=\sum_{i=-1}^{L-1}\left[\sum_{\mathcal{C}\in\mathcal{G}_i:\mathcal{C}\text{ heavy}}\left(\widehat{|\mathcal{C}|}-\sum_{\substack{\mathcal{C}':\mathcal{C}'\in\mathcal{G}_{i+1},\mathcal{C}'\subset\mathcal{C},\\ \mathcal{C}'\text{ is heavy}}}\widehat{|\mathcal{C}'|}-\frac{|S_i\cap\mathcal{C}|}{\pi_i}\right)d(c(\mathcal{C}),Z)+\sum_{p\in S_i}\frac{d(p,Z)}{\pi_i}\right]\\&=\sum_{i=-1}^{L-1}\left[\sum_{\mathcal{C}\in\mathcal{G}_i:\mathcal{C}\text{ heavy}}\widehat{|\mathcal{C}|}(d(c(\mathcal{C}),Z)-d(c(\mathcal{C}^P),Z))+\sum_{p\in S_i}\frac{d(p,Z)-d(c_p^i,Z)}{\pi_i}\right],\end{aligned}\tag{12}$$

where we denote $d(c(\mathcal{C}^P_{-1}),Z)=0$ for convenience. Let $\mathrm{cost}(Z)=\sum_{p\in P}d(p,Z)$. Note that we can also write the true cost of Z as

$$\mathrm{cost}(Z)=\sum_{i=-1}^{L-1}\left[\sum_{\mathcal{C}\in\mathcal{G}_i:\mathcal{C}\text{ heavy}}|\mathcal{C}|(d(c(\mathcal{C}),Z)-d(c(\mathcal{C}^P,Z)))+\sum_{p\in P_i}d(p,Z)-d(c_p^i,Z)\right].$$

We have that,

$$\widehat{\mathrm{cost}}(Z)-\mathrm{cost}(Z)=A_1+A_2,$$

where

$$A_1=\sum_{i=-1}^{L-1}\left[\sum_{\mathcal{C}\in\mathcal{G}_i:\mathcal{C}\text{ heavy}}(\widehat{|\mathcal{C}|}-|\mathcal{C}|)(d(c(\mathcal{C}),Z)-d(c(\mathcal{C}^P,Z)))\right]$$

and

$$A_2=\sum_{i=-1}^{L-1}\left(\sum_{p\in S_i}\frac{d(p,Z)-d(c_p^i,Z)}{\pi_i}-\sum_{p\in P_i}d(p,Z)-d(c_p^i,Z)\right).$$

Let $Z^*\subset[\Delta]^d$ be a set of optimal k-centers for the k-median problem of the input point set. By Lemma 2.2, with probability at most $1-\rho$, for each $i\in[0,L]$, if at most $ek(L+1)/\rho$ cells $\mathcal{C}$ in $\mathcal{G}_i$ satisfy $d(\mathcal{C},Z^*)\le\Delta/(2^{i+1}d)$. Conditioning on this event, we have that, by Lemma C.4 there are at most $k'=O\left(\frac{kL}{\rho}\right)$ heavy cells per level. Since for each $\mathcal{C}\in\mathcal{G}_i$, $\left|\widehat{|\mathcal{C}|}-|\mathcal{C}|\right|\le\frac{\epsilon}{\lambda_5Ld^{3/2}}\cdot\frac{2^i\rho d\mathsf{OPT}}{kL\Delta}$, by choosing appropriate constant $\lambda_5>0$ we have

$$\begin{aligned}|A_1|&\le\sum_{i=-1}^{L-1}\left|\sum_{\mathcal{C}\in\mathcal{G}_i:\mathcal{C}\text{ heavy}}(\widehat{|\mathcal{C}|}-|\mathcal{C}|)(d(c(\mathcal{C}),Z)-d(c(\mathcal{C}^P,Z)))\right|\\&\le L\cdot k'\cdot\frac{\epsilon}{\lambda_5Ld^{3/2}}\cdot\frac{2^i\rho d\mathsf{OPT}}{kL\Delta}\cdot\frac{\Delta\sqrt{d}}{2^i}\le\frac{\epsilon\mathsf{OPT}}{2}.\end{aligned}\tag{13}$$

For A_2, let

$$A_{2i}=\left(\sum_{p\in S_i}\frac{d(p,Z)-d(c_p^i,Z)}{\pi_i}-\sum_{p\in P_i}d(p,Z)-d(c_p^i,Z)\right).$$

By Lemma C.3, for each $i\in[-1,L-1]$, $|P_i|\Delta\sqrt{d}/(2^i)=\lambda_2(d^{3/2}\mathsf{OPT})$. Thus by Lemma C.7, and choosing appropriate constants, with probability at least $1-\rho/(L+1)\Delta^{dk}$, $|A_{2i}|\le\frac{\epsilon\mathsf{OPT}}{2(L+1)}$. By the union bound, with probability at least $1-\rho$, for every level i, and every k-set $Z\subset[\Delta]^d$, $|A_{2i}|\le\frac{\epsilon\mathsf{OPT}}{2(L+1)}$. Thus $|A_2|\le\epsilon\mathsf{OPT}/2$. In total, with probability at least $1-3\rho$, $|A_1+A_2|\le\epsilon\mathsf{OPT}$ for any k-set $Z\subset[\Delta]^d$.

The coreset size is the number of heavy cells plus the number of sampled points. The number of heavy cells is $O(kL^2/\rho)$.

The expected number of sampled points per level is at most,

$$|S_i| = O\left(\frac{d^4L^3k}{\epsilon^2} + \frac{d^3L^2k}{\epsilon^2\rho}\log\left(\frac{kL}{\rho}\right)\right)\frac{\mathsf{OPT}}{o}.$$

By a Chernoff bound, with probability at least $1-\rho/\Delta^{dk}$, for every level $i\in[0,L]$, the number of sampled points is,

$$|S_i| \le O\left(\frac{d^4L^3k}{\epsilon^2} + \frac{d^3L^3k}{\rho\epsilon^2}\log\left(\frac{kL}{\rho}\right)\right)\frac{\mathsf{OPT}}{o}.$$

Thus the size of the coreset $\mathcal{S}$ is,

$$|\mathcal{S}| \le O\left[\frac{d^3L^4k}{\epsilon^2}\left(d+\frac{1}{\rho}\log\frac{kL}{\rho}\right)\frac{\mathsf{OPT}}{o}\right].$$

□

C.3 Ensuring Non-Negative Weights

In this section, we will provide a procedure to rectify all the weights for the coreset constructed in the last sub-section. The idea is similar to the method used in (Indyk & Price, 2011). The procedure is shown in Algorithm 4.3.

Proposition C.10. *Let $\mathcal{S}$ be a weighted set constructed using the Non-Negatively Weighted Construction, i.e. each heavy cell $\mathcal{C}$ has value $\widehat{|\mathcal{C}|}$ and the set of sampled points $S = S_{-1}\cup S_0\ldots\cup S_L$ with each point in S_l has weight $1/\pi_l$. If for each heavy cell $\mathcal{C}\in\mathcal{G}_i$, $||\widehat{\mathcal{C}}|-|\mathcal{C}|| \le \frac{\epsilon}{\lambda_6 Ld^{3/2}}\cdot\frac{2^i\rho d\mathsf{OPT}}{kL\Delta}$ for some universal constant $\lambda_6>0$ and for each $i\in[-1,L]$ and any k-set $Z\subset[\Delta]^d$,*

$$\left|\sum_{p\in S}\mathrm{wt}(p)(d(c_p^i,Z)-d(p,Z)) - \sum_{p\in P_i}(d(c_p^i,Z)-d(p,Z))\right| \le \frac{\epsilon\mathsf{OPT}}{2L},$$

and

$$\sum_{\mathcal{C}\in\mathcal{G}_i:\ heavy}\left|\frac{|S_i\cap\mathcal{C}|}{\pi_i}-|P_i\cap\mathcal{C}|\right|\frac{\Delta\sqrt{d}}{2^i} \le \frac{\epsilon\mathsf{OPT}}{L}.$$

Then on input $\widehat{|\mathcal{C}_1|},\widehat{|\mathcal{C}_2|},\ldots,\widehat{|\mathcal{C}_{k'}|}$ and S, where k' is the number of heavy cells, the coreset output by Algorithm 4.3 is a (4ϵ)-coreset.

C.4 The Streaming Algorithm

C.4.1 Sampling From Sparse Cells

For the streaming algorithm, we can still use `HEAVY-HITTER` algorithms to find the heavy cells. The major challenge is to do the sampling for each point from its ending level. We do this using a combination of hash functions and `K-Set`. In Algorithm C.4.1, we provide a procedure that recovers the set of points from cells with a small number of points and ignore all the heavy cells. The guarantee is,

Theorem C.11. *Given as input a set of dynamically updating streaming points $P\subset[N]$, a set of mutually disjoint cells $C\subset[M]$, whose union covers the region of P. Algorithm C.4.1 outputs all the points in cells with less than β points or output `Fail`. If with the promise that at most α cells from C contain a point of P, then the algorithm outputs `Fail` with probability at most δ. The algorithm uses $O(\alpha\beta(\log(M\beta)+\log N)\log N\log(\log N\alpha\beta/\delta)\log(\alpha\beta/\delta))$ bits in the worst case.*

The high level idea of this algorithm is to hash the original set of points to a universe of smaller size. For cells with less points, the collision rate is much smaller than cells with more points. To recover one bit of a point, we update that bit and the cell ID and also its hash tag to the `K-Set`-data structure. If there are no other points with hash values colliding with this point, the count of that point is simply 1. If this is the case, we immediately recover the bit. By repeating the above procedure once for each bit, we can successfully recover the set of points with no colliding hash tags. For those points with colliding hash tags, we simply ignore them. Each point has a constant probability to collide with another point, thus not be in the output. By running the whole procedure $O(\log(\alpha\beta/\epsilon))$ times in parallel, we reduce the probability to roughly ϵ for each point in cells with less than β points. To formally prove Theorem C.11, we first prove the following lemma, which is the guarantee of Algorithm C.4.1.

Lemma C.12. *Given input a set of dynamically updating streaming points $P\subset[N]$, a set of mutually disjoint cells $C\subset[M]$, whose union covers the region of P. Algorithm C.4.1 outputs a set of points in cells with less than β points*

or output `Fail`*. If with the promise that at most α cells from C contain a point of P, then the algorithm outputs* `Fail` *with probability at most δ. Conditioning on the event that the algorithm does not output* `Fail`*, each point p from cell with less than β points is in the output with marginal probability at least* 0.9*. The algorithm uses $O(\alpha\beta(\log(M\beta) + \log N)\log N\log(\log N\alpha\beta/\delta))$ bits in the worst case.*

Proof. We prove this lemma by showing that (a) if a point $p \in P$ contained in cell $\mathcal{C}$, with $|\mathcal{C} \cap P| \leq \beta$, then with probability at least 0.99, there are no other points $p' \in \mathcal{C} \cap P$ with $H(p) = H(p')$, (b) conditioning on the event that the algorithm does not output `Fail`, then for any cell $\mathcal{C} \in C$, if a point $p \in \mathcal{C}$ such that no other points in $\mathcal{C} \cap P$ has the same hash value $H(p)$, then p is in the output and (c) the algorithm outputs `Fail` with probability at most δ. The correctness of the algorithm follows by (a), (b) and (c).

To show (a), consider any cell $\mathcal{C} \in C$ with $|\mathcal{C}| \leq \beta$, let $p \in \mathcal{C}$ with hash value $H(p)$. Since H is 2-wise independent, the expected number of other points hashed to the same hash value $H(p)$ is at most $\beta/U = 1/100$. By Markov's inequality, with probability at least 0.99, no other point in $\mathcal{C}$ is hashed to $H(p)$.

To show (b), notice that if the algorithm does not output `Fail`, then for a given cell $\mathcal{C}$, let c be its ID, and p_j be the j-th bit of point p. Then (c, h, p_j) has 1 count and $(c, h, 1 - p_j)$ has 0 count for each $j \in [t]$, where $t = \lceil \log N \rceil$. Thus we can uniquely recover each bit of point p, hence the point p.

For (c), since there are at most α cells, there are at most $2\alpha U = O(\alpha\beta)$ many different updates for each `KS` structure. Therefore, with probability at most $\frac{\delta}{t}$, a single `KS` instance outputs `Fail`. By the union bound, with probability at least $1 - \delta$, no `KS` instance outputs `Fail`.

Finally, the space usage is dominated by the `KS` data structures. Since the input data to `KS` is from universe $[M] \times [U] \times \{0, 1\}$, each `KS` structure uses space $O(\alpha\beta(\log(M\beta) + \log N)\log(t\alpha\beta/\delta))$ bits of memory, the total space is $O(\alpha\beta(\log(M\beta) + \log N)\log N\log(t\alpha\beta/\delta))$. □

Proof of Theorem C.11. Each instance of `SparseCellsSingle` fails with probability at most $\delta/(4A)$, where A is the number independent `SparseCellsSingle` instances. By the union bound, with probability at least $1 - \delta/4$, none of them output `Fail`. Conditioning on this event, the random bits of the hash functions of each `SparseCellsSingle` instance are independent, thus by Lemma C.12 a fixed point $p \in \mathcal{C}$ with $|\mathcal{C}| \leq \beta$ is in the output with probability at least $10^{-\log \frac{4\alpha\beta}{\delta}} \leq \delta/(4\alpha\beta)$. Since there are at most $\alpha\beta$ points in cells with less than β points, by the union bound we conclude that with probability at least $1 - \delta/4$, every point in cells with less than β points is in the output set S. In sum, with probability at least $1 - \delta/2$, S contains all the desired points.

The other `KS` instance outputs `Fail` with probability at most $\delta/2$. Thus if T is not `Fail`, then T contains the exact number of points of each cell. If any desired point is not in S, then $|\mathcal{C}| > |\mathcal{C} \cap S|$, we output `Fail`. This happens with probability at most δ under the gaurantee of the `KS`structure.

Since each `SparseCellsSingle` instance uses $O(\alpha\beta(\log(M\beta) + \log N)\log N\log(tA\alpha\beta/\delta))$ bits of space, the final space of the algorithm is $O(\alpha\beta(\log(M\beta) + \log N)\log N\log(t\alpha\beta/\delta)\log(\alpha\beta/\delta))$. □

Algorithm 4 `SparseCells`$(N, M, \alpha, \beta, \delta)$: input the point sets $P \subset [N]$ and set of cells $C \subset [M]$ such that at most α cells containing a point, output the set of points in cells with less than β points.

Let $A \leftarrow \log \frac{4\alpha\beta}{\delta}$;
Let $R_1, R_2, \ldots, R_A$ be the results of independent instances of `SparseCellsSingle`$(N, M, \alpha, \beta, \delta/(4A))$ running in parallel;
Let T be the results of another parallel `KS` structure with space parameter α and error $\delta/2$ and with input as the cell IDs of points in P; /*T returns the exact counts of each cell*/
if *any of the data structures returns* `Fail`**:**
  return `Fail`;
Let $S \leftarrow R_1 \cup R_2 \cup \ldots R_A$;
if $\exists$ *set* $\mathcal{C} \in T$ *with* $|\mathcal{C}| \leq \beta$ *and* $|\mathcal{C}| \neq |\mathcal{C} \cap S|$**:**
  return `Fail`;
return S;

C.4.2 The Algorithm

With the construction of algorithm `SparseCells`, we now have all the tools for the streaming coreset construction. The streaming algorithm is composed by $O(L)$ levels of `HEAVY-HITTER` instances, which serve as a heavy cell identifier and

Algorithm 5 `SparseCellsSingle`$(N, M, \alpha, \beta, \delta)$: input the point sets $P \subset [N]$ and set of cells $C \subset [M]$ such that at most α cells containing a point, output the set of points in cells with less than β points.

Initization:
$U \leftarrow 100\beta$.
$t \leftarrow \lceil \log N \rceil$;
$H : [N] \to [U]$, 2-wise independent;
`K-Set` structures $\texttt{KS}_1, \texttt{KS}_2, \ldots \texttt{KS}_t$ with space parameter $2\alpha U$ and probability $\frac{\delta}{t}$;
Update(p, op): /*$op \in \{\texttt{Insert}, \texttt{Delete}\}$*/
$c \leftarrow$ cell ID of p;
for $i \in [t]$**:**
  /*A point p is represented as $(p_1, p_2, \ldots, p_t)$*/;
  $p_i \leftarrow$ the i-th bit of point p;
  $\texttt{KS}_i$.update($(c, H(p), p_i), op$);
Query:
if *for any* $i \in [t]$, $\texttt{KS}_i$ *returns* `Fail`**:**
  return `Fail`;
$S \leftarrow \emptyset$;
for *each* (c, h, p_1) *in the output of* $\texttt{KS}_1$**:**
  if $(c, h, p_j) \notin \texttt{KS}_j$ *for some* $j \in [t]$**:**
    /*A checking step, may not happen at all*/;
    return `Fail`;
  Let $s(c, h, p_j)$ be the counts of (c, h, p_j) in $\texttt{KS}_j$;
  if $s(c, h, p_j) = 1$ *and* $s(c, h, 1 - p_j) = 0$ *for each* $j \in [t]$**:**
    $p \leftarrow (p_1, p_2, \ldots, p_t)$;
    $S \leftarrow S \cup \{p\}$;
return S

by $O(L)$ levels of `SparseCells` instances, which sample the points from their ending levels. The full algorithm is stated in Algorithm 6. The guarantee of the algorithm is stated in the following theorem.

Theorem C.13. *Fix $\epsilon, \rho \in (0, 1/2)$, positive integers k and Δ, Algorithm 6 makes a single pass over the streaming point set $P \subset [\Delta]^d$, outputs a weighted set S with* non-negative weights *for each point, such that with probability at least* 0.99*, S is an ϵ-coreset for k-median of size $O\left[\frac{d^3L^4k}{\epsilon^2}\left(d + \frac{1}{\rho}\log\frac{kL}{\rho}\right)\right]$, where $L = \log\Delta$. The algorithm uses $O\left[\frac{d^7L^7k}{\epsilon^2}\left(\rho dL + \frac{1}{\rho}\log^2\frac{dkL}{\rho\epsilon}(\log\log\frac{dkL}{\rho\epsilon} + L)\right)\log^2\frac{dkL}{\rho\epsilon}\right]$ bits in the worst case. For each update of the input, the algorithm needs* $\mathrm{poly}\,(d, 1/\epsilon, L, \log k)$ *time to process and outputs the coreset in time* $\mathrm{poly}(d, k, L, 1/\epsilon, 1/\rho, \log k)$ *after one pass of the stream.*

Proof. W.l.o.g. assume $\rho \geq \Delta^{-d}$, since otherwise we can store the entire set of points. In the sequel, we will prove the theorem with parameter $O(\rho)$ and $O(\epsilon)$. It translates to ρ and ϵ directly by scaling and with losing at most a constant factor in space and time bounds. By Lemma 2.2, with probability at least $1 - \rho$, for every level $i \in [0, L]$, at most ekL/ρ cells $\mathcal{C}$ in $\mathcal{G}_i$ satisfy $d(C, Z^*) \leq \Delta/(2^{i+1}d)$. We condition on this event for the following proof.

We first show that the `HEAVY-HITTER` instances faithfully implement a *heavy cell identification scheme*. First note that with probability at least $1 - \rho$, all `HEAVY-HITTER` instances succeed. Conditioning on this event for the following proof. As shown in the proof of Lemma B.1, by setting $\epsilon' = \epsilon\sqrt{\frac{\rho}{\lambda_7 k d^3 L^3}}$ and $k' = \lambda_8 kL/\rho$, for appropriate positive universal constants λ_7, λ_8, then the additive error to each cell is at most $\frac{\epsilon}{\lambda_9 d^{3/2}L} \cdot \frac{2^i d\mathsf{OPT}}{kL\Delta}$ for some universal constant λ_9, which matches the requirement of Proposition C.10. For each cell $\mathcal{C}$ with at least $2^i\rho d\mathsf{OPT}/(k(L+1)\Delta)$ points, by Lemma C.4 it must be in the top $(e+2)k(L+1)/\rho$ cells. For each cell $\mathcal{C}'$ with at least $2^{i-1}\rho d\mathsf{OPT}/(k(L+1)\Delta)$ points, it must be in the top $(e+4)k(L+1)/\rho$ cells. Since the additive error is $\frac{\epsilon}{\lambda_7 d^{3/2}(L+1)} \cdot \frac{2^i d\mathsf{OPT}}{k(L+1)\Delta} \ll \frac{1}{2}\frac{2^i d\mathsf{OPT}}{k(L+1)\Delta}$. Thus $\mathcal{C}$ is in the output of the `HEAVY-HITTER` instances, since otherwise $\widehat{|\mathcal{C}|} \leq \frac{1}{2}\frac{2^i d\mathsf{OPT}}{k(L+1)\Delta} + \frac{\epsilon}{\lambda_7 d^{3/2}(L+1)} \cdot \frac{2^i d\mathsf{OPT}}{k(L+1)\Delta}$ contradicts the error bound

(by choosing sufficiently large λ_7). Thus the algorithm faithfully implements a heavy cell identification scheme.

Now we show that if there exists an $o \leq \mathsf{OPT}$ such that no instance of `SparseCells` outputs `Fail`, then the result is a desired $O(\epsilon)$-coreset. This follows by Proposition C.9 and Proposition C.10. Then we note that the coreset size is upper bounded by $O\left[\frac{d^3L^4k}{\epsilon^2}\left(d+\frac{1}{\rho}\log\frac{kL}{\rho}\right)\right]$ as desired.

Next we show that there exists an $\mathsf{OPT}/2 \leq o^* \leq \mathsf{OPT}$ that with probability at least $1-\rho$, no `SparseCells` instance $\texttt{SC}_{o^*,i}$ outputs `Fail`. By Chernoff bound, with probability at least $1-O(\rho)$, as also shown in the proof of Proposition C.9, per level at most $O\left[\frac{d^3L^4k}{\epsilon^2}\left(d+\frac{1}{\rho}\log\frac{kL}{\rho}\right)\frac{\mathsf{OPT}}{o}\right]$ cells is occupied by a point. And at most $O\left[\frac{d^3L^2}{\epsilon^2}\left(\rho d+\log\frac{kL}{\rho}+\frac{\rho}{kL}\log\frac{L}{\rho}\right)\right]$ points is sampled per light cell. Conditioned on this fact and that each instances fails with probability at most $O(\rho/(dL))$, with probability at least $1-O(\rho)$, no nstance $\texttt{SC}_{o^*,i}$ fails.

Lastly, we bound the space usage and update/query time. For the `HEAVY-HITTER` instances, the total space used is $O\left(dL+\log\frac{1}{\rho}\right)\frac{d^4L^5k}{\rho\epsilon^2}$ bits, analogous to the proof of Theorem 3.6. Each `SparseCells` instance uses space $O\left[\frac{d^5L^4r^2k}{\epsilon^2}\left(\rho dL+\frac{r^2(\log r+L)}{\rho}\right)\right]$, where $r=\log\frac{dkL}{\rho\epsilon}$. The total space bound is $O\left[\frac{d^6L^6r^2k}{\epsilon^2}\left(\rho dL+\frac{r^2(\log r+L)}{\rho}\right)\right]$ bits. As a same argument in the proof of Theorem 3.6, the cost of de-randomization introduce an additional dL factor. Thus, the final space bound is $O\left[\frac{d^7L^7r^2k}{\epsilon^2}\left(\rho dL+\frac{r^2(\log r+L)}{\rho}\right)\right]$ bits. The query time and update time is similar to that of Theorem 3.6 thus $\text{poly}\left(d,L,\frac{1}{\epsilon},\frac{1}{\rho},k\right)$ and $\text{poly}\left(d,L,\frac{1}{\epsilon},\log k\right)$.

□

D Synthetic Dataset

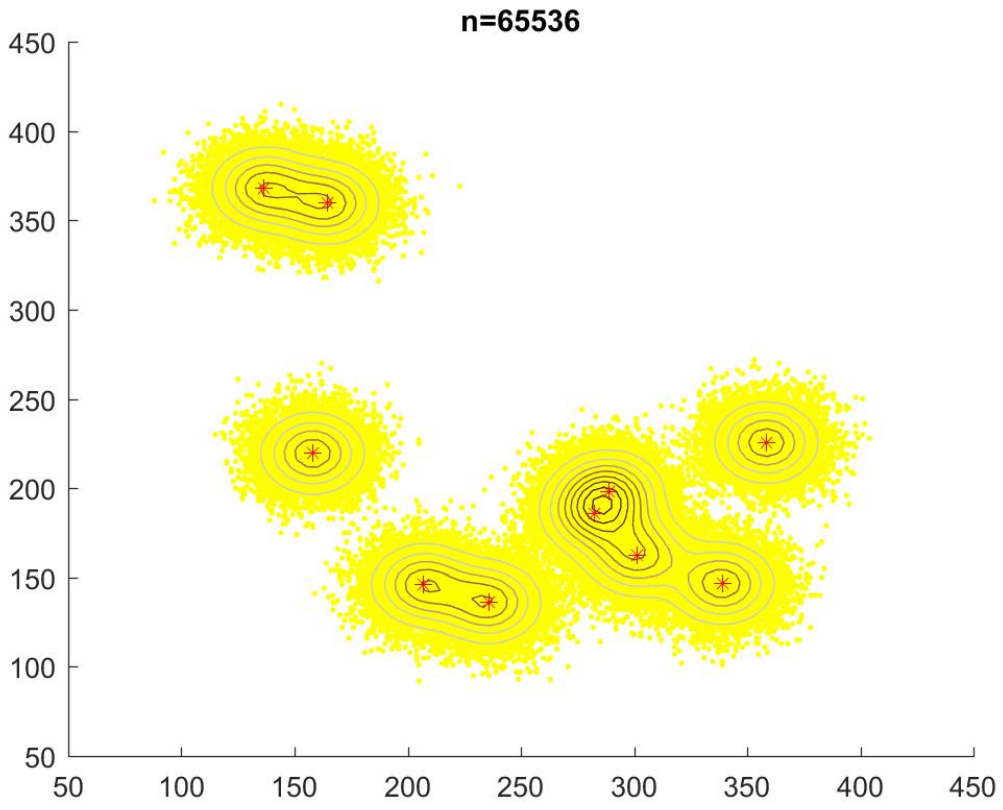

Figure 2. 65536 points are drawn from a Gaussian Mixture distribution. The contours illustrate the PDF function.

Algorithm 6 `PositiveCoreSet`(S, k, ρ, ϵ): construct a ϵ-coreset for dynamic stream S.

Initization:
Initialize a grid structure;
$O \leftarrow \{1, 2, 4, \ldots, \sqrt{d}\Delta^{d+1}\}$; $L \leftarrow \lceil \log \Delta \rceil$; $\pi_i(o) \leftarrow \min\left(\frac{\lambda_3 d^2 \Delta L^2}{2^i \epsilon^2 o} \log\left(\frac{2L\Delta^{dk}}{\rho}\right) + \frac{\lambda_4 d^2 k L^3 \Delta}{2^i \epsilon^2 \rho o} \log \frac{30kL^2}{\rho^2}, 1\right)$;
$\alpha \leftarrow O\left[\frac{d^3 L^4 k}{\epsilon^2}\left(d + \frac{1}{\rho}\log\frac{kL}{\rho}\right)\right]$, $\beta \leftarrow O\left[\frac{d^3 L^2}{\epsilon^2}\left(\rho d + \log\frac{kL}{\rho} + \frac{\rho}{kL}\log\frac{L}{\rho}\right)\right]$, $\epsilon' \leftarrow \epsilon\sqrt{\frac{\rho}{\lambda_7 k d^3 L^3}}$; $m \leftarrow 0$;
For each $o \in O$ and $i \in [0, L]$, construct fully independent hash function $h_{o,i} : [\Delta]^d \rightarrow \{0, 1\}$ with $Pr_{h_{o,i}}(h_{o,i}[q] = 1) = \pi_i(o)$; initialize `SparseCells`$(\Delta^d, (1+2^i)^d, \alpha, \beta, O(\rho/(dL)))$ instances $\mathtt{SC}_{o,i}$;
Initialize `HEAVY-HITTER`$(\Delta^d, 10Lk/\rho, \epsilon', \rho/L)$ instances, $\mathtt{HH}_0, \mathtt{HH}_1, \ldots, \mathtt{HH}_{L-1}$, one for a level;

Update (S):
for *each update* $(op, q) \in S$**:**
    /*$op \in \{$`Insert`, `Delete`$\}$*/
    $m \leftarrow m \pm 1$; /*`Insert`: $+1$, `Delete`:-1*/
    for *each* $i \in [0, L]$**:**
        $c_q^i \leftarrow$ the center of the cell contains q at level i;
        $\mathtt{HH}_i$.update(op, c_q^i);
        for *each* $o \in [O]$**:**
            if $h_{o,i}(q) == 1$**:**
                $\mathtt{SC}_{o,i}$.update(op, c_q^i);
Query:
Let o^* be the smallest o such that no instance of $\mathtt{SC}_{o,0}, \mathtt{SC}_{o,1}, \ldots, \mathtt{SC}_{o,L}$ returns `Fail`;
$S \leftarrow \{\}$;
$\mathcal{C}_{-1} \leftarrow$ the cell of the entire space $[\Delta]^d$; $\widehat{|\mathcal{C}_{-1}|} \leftarrow m$;
for $i \in [0, L-1]$**:**
    $C_i \leftarrow \mathtt{HH}_i$.query().top$((e+4)(L+1)k/\rho)$;
    Remove cells $\mathcal{C}$ from C_i if $\mathcal{C}^P(\mathcal{C}) \notin C_{i-1}$, where $\mathcal{C}^P(\mathcal{C})$ is the parent cell of $\mathcal{C}$ in level $i-1$;
    $B_i \leftarrow \mathtt{SC}_{o^*,i}$.query();
    $S_i \leftarrow \{p \in B_i : \mathcal{C}(p, i-1) \in C_{i-1}$ AND $\mathcal{C}(p, i) \notin C_i\}$;
    Each point in $\mathcal{S}_i$ receives weight $1/\pi_i(o^*)$;
    $S \leftarrow S \cup S_i$;
$k' \leftarrow \sum_{i \in [0,L]} |C_i|$;
Let $\{\mathcal{C}_1, \mathcal{C}_2, \ldots \mathcal{C}_{k'}\} = \cup_{i \in [0,L]} C_i \cup \{\mathcal{C}_{-1}\}$ be the set of heavy cells;
Let $\left\{\widehat{|\mathcal{C}_1|}, \widehat{|\mathcal{C}_2|}, \ldots \widehat{|\mathcal{C}_{k'}|}\right\}$ be the estimated frequency of each cell;
$R \leftarrow$ `RectifyWeights`$(\widehat{|\mathcal{C}_1|}, \widehat{|\mathcal{C}_2|}, \ldots \widehat{|\mathcal{C}_{k'}|}, S)$;
return R.