Layered Sampling for Robust Optimization Problems

Hu Ding [1] **Zixiu Wang** [1]

Abstract

In real world, our datasets often contain outliers. Most existing algorithms for handling outliers take high time complexities (*e.g.* quadratic or cubic complexity). *Coreset* is a popular approach for compressing data so as to speed up the optimization algorithms. However, the current coreset methods cannot be easily extended to handle the case with outliers. In this paper, we propose a new variant of coreset technique, *layered sampling*, to deal with two fundamental robust optimization problems: *k-median/means clustering with outliers* and *linear regression with outliers*. This new coreset method is in particular suitable to speed up the iterative algorithms (which often improve the solution within a local range) for those robust optimization problems.

1. Introduction

Coreset is a widely studied technique for solving many optimization problems (Phillips, 2016; Bachem et al., 2017; Munteanu et al., 2018; Feldman, 2020). The (informal) definition is as follows. Given an optimization problem with the objective function Δ, denote by $\Delta(P, C)$ the objective value determined by a dataset P and a solution C; a small set S is called a coreset if

$$\Delta(P, C) \approx \Delta(S, C) \tag{1}$$

for any feasible solution C. Roughly speaking, the coreset is a small set of data approximately representing a much larger dataset, and therefore existing algorithm can run on the coreset (instead of the original dataset) so as to reduce the complexity measures like running time, space, and communication. In the past years, the coreset techniques have been successfully applied to solve many optimization problems, such as clustering (Chen, 2009; Feldman & Langberg, 2011; Huang et al., 2018), logistic regression (Huggins et al., 2016; Munteanu et al., 2018), linear regression (Dasgupta et al., 2009; Drineas et al., 2006), and Gaussian mixture model (Lucic et al., 2017; Karnin & Liberty, 2019).

A large part of existing coreset construction methods are based on the theory of *sensitivity* which was proposed by (Langberg & Schulman, 2010). Informally, each data point $p \in P$ has the sensitivity $\phi(p)$ (in fact, we just need to compute an appropriate upper bound of $\phi(p)$) to measure its importance to the whole instance P over all possible solutions, and $\Phi(P) = \sum_{p \in P} \phi(p)$ is called the total sensitivity. The coreset construction is a simple sampling procedure where each point p is drawn *i.i.d.* from P proportional to $\frac{\phi(p)}{\Phi(P)}$; each sampled point p is assigned a weight $w(p) = \frac{\Phi(P)}{m\phi(p)}$ where m is the sample size depending on the "pseudo-dimension" of the objective function Δ ((Feldman & Langberg, 2011; Li et al., 2001)); eventually, the set of weighted sampled points form the desired coreset S.

In real world, datasets are noisy and contain outliers. Moreover, outliers could seriously affect the final results in data analysis (Chandola et al., 2009; Goodfellow et al., 2018). However, the sensitivity based coreset approach is not appropriate to handle robust optimization problems involving outliers (*e.g.*, k-means clustering with outliers). For example, it is not easy to compute the sensitivity $\phi(p)$ because the point p could be inlier or outlier for different solutions; moreover, it is challenging to build the relation, such as (1), between the original instance P and the coreset S (*e.g.*, how to determine the number of outliers for the instance S?).

1.1. Our Contributions

In this paper, we consider two important robust optimization problems: *k-median/means clustering with outliers* and *linear regression with outliers*. Their quality guaranteed algorithms exist but often have high complexities that seriously limit their applications in real scenarios (see Section 1.2 for more details). We observe that these problems can be often efficiently solved by some heuristic algorithms in practice, though they only guarantee local optimums in theory. For example, (Chawla & Gionis, 2013) proposed the algorithm *k-means-* - to solve the problem of k-means clustering with outliers, where the main idea is an *alternating minimization* strategy. The algorithm is an iterative procedure, where it

[1]School of Computer Science and Technology, University of Science and Technology of China. Correspondence to: Hu Ding <huding@ustc.edu.cn, http://staff.ustc.edu.cn/~huding/ >.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

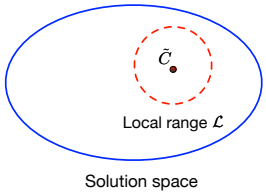

Figure 1. The red point represents the initial solution $\tilde{C}$, and our goal is to guarantee (2) for a local range around $\tilde{C}$.

alternatively updates the outliers and the k cluster centers in each iteration; eventually the solution converges to a local optimum. The alternating minimization strategy is also widely used for solving the problem of linear regression with outliers, *e.g.,* (Shen & Sanghavi, 2019). A common feature of these methods is that they usually start from an initial solution and then locally improve the solution round by round. Therefore, a natural question is

can we construct a "coreset" only for a local range in the solution space?

Using such a coreset, we can substantially speed up those iterative algorithms. Motivated by this question, we introduce a new variant of coreset method called *layered sampling*. Given an initial solution $\tilde{C}$, we partition the given data set P into a consecutive sequence of "layers" surrounding $\tilde{C}$ and conduct the random sampling in each layer; the union of the samples, together with the points located in the outermost layer, form the coreset S. Actually, our method is partly inspired by the coreset construction method of k-median/means clustering (without outliers) proposed by (Chen, 2009). However, we need to develop significantly new idea in theory to prove its correctness for the case with outliers. The purpose of layered sampling is not to guarantee the approximation quality (as (1)) for any solution C, instead, it only guarantees the quality for the solutions in a local range $\mathcal{L}$ in the solution space (the formal definition is given in Section 1.3). Informally, we need to prove the following result to replace (1):

$$\forall C \in \mathcal{L}, \Delta(P, C) \approx \Delta(S, C) \tag{2}$$

See Figure 1 for an illustration. In other words, the new method can help us to find a local optimum faster. Our main results are shown in Theorem 1 and 2. The construction algorithms are easy to implement.

1.2. Related Works

k-median/means clustering (with outliers). k-median/means clustering are two popular center-based clustering problems (Awasthi & Balcan, 2014). It has been extensively studied for using coreset techniques to reduce the complexities of k-median/means clustering algorithms (Chen, 2009; Har-Peled & Kushal, 2007; Fichtenberger et al., 2013; Feldman et al., 2013); in particular, (Feldman & Langberg, 2011) proposed a unified coreset framework for a set of clustering problems. However, the research on using coreset to handle outliers is still quite limited. Recently, (Huang et al., 2018) showed that a uniform independent sample can serve as a coreset for clustering with outliers in Euclidean space; however, such uniform sampling based method often misses some important points and therefore introduces an unavoidable error on the number of outliers. (Gupta, 2018) also studied the uniform random sampling idea but under the assumption that each optimal cluster should be large enough. Partly inspired by the method of (Mettu & Plaxton, 2004), (Chen et al., 2018) proposed a novel summary construction algorithm to reduce input data size which guarantees an $O(1)$ factor of distortion on the clustering cost.

In theory, the algorithms with provable guarantees for k-median/means clustering with outliers (Chen, 2008; Krishnaswamy et al., 2018; Friggstad et al., 2018) have high complexities and are difficult to be implemented in practice. The heuristic but practical algorithms have also been studied before (Chawla & Gionis, 2013; Ott et al., 2014). By using the local search method, (Gupta et al., 2017) provided a 274-approximation algorithm of k-means clustering with outliers but needing to discard more than the desired number of outliers; to improve the running time, they also used k-means++ (Arthur & Vassilvitskii, 2007) to seed the "coreset" that yields an $O(1)$ factor approximation. Based on the idea of k-means++, (Bhaskara et al., 2019) proposed an $O(\log k)$-approximation algorithm.

Linear regression (with outliers). Several coreset methods for ordinary linear regression (without outliers) have been proposed (Drineas et al., 2006; Dasgupta et al., 2009; Boutsidis et al., 2013). For the case with outliers, which is also called "Least Trimmed Squares linear estimator (LTS)", a uniform sampling approach was studied by (Mount et al., 2014; Ding & Xu, 2014). But similar to the scenario of clustering with outliers, such uniform sampling approach introduces an unavoidable error on the number of outliers.

(Mount et al., 2014) also proved that it is impossible to achieve even an approximate solution for LTS within polynomial time under the conjecture of *the hardness of affine degeneracy* (Erickson & Seidel, 1995), if the dimensionality d is not fixed. Despite of its high complexity, several practical algorithms were proposed before and most of them are based on the idea of alternating minimization that improves the solution within a local range, such as (Rousseeuw, 1984; Rousseeuw & van Driessen, 2006; Hawkins, 1994; Mount et al., 2016; Bhatia et al., 2015; Shen & Sanghavi, 2019). (Klivans et al., 2018) provided another approach based on the *sum-of-squares* method.

1.3. Preliminaries

Below, we introduce several important definitions.

i. k-Median/Means Clustering with Outliers. Suppose P is a set of n points in $\mathbb{R}^d$. Given two integers $1 \leq z, k < n$, the problem of k-median clustering with z outliers is to find a set of k points $C = \{c_1, \cdots, c_k\} \subset \mathbb{R}^d$ and a subset $P' \subset P$ with $|P'| = n - z$, such that the following objective function

$$\mathcal{K}_1^{-z}(P, C) = \frac{1}{n-z} \sum_{p \in P'} \min_{1 \leq j \leq k} ||p - c_j|| \tag{3}$$

is minimized. Similarly, we have the objective function

$$\mathcal{K}_2^{-z}(P, C) = \frac{1}{n-z} \sum_{p \in P'} \min_{1 \leq j \leq k} ||p - c_j||^2, \tag{4}$$

for k-means clustering with outliers. The set C is also called a solution of the instance P. Roughly speaking, given a solution C, the farthest z points to C are discarded, and the remaining subset P' is partitioned into k clusters where each point is assigned to its nearest neighbor of C.

ii. Linear Regression with Outliers. Given a vector $h = (h_1, h_2, \cdots, h_d) \in \mathbb{R}^d$, the linear function defined by h is $y = \sum_{j=1}^{d-1} h_j x_j + h_d$ for $d-1$ real variables $x_1, x_2, \cdots, x_{d-1}$. Thus the linear function can be represented by the vector h. From geometric perspective, the linear function can be viewed as a $(d-1)$-dimensional hyperplane in the space. Let z be an integer between 1 and n, and $P = \{p_1, p_2, \cdots, p_n\}$ be a set of n points in $\mathbb{R}^d$, where each $p_i = (x_{i,1}, x_{i,2}, \cdots, x_{i,d-1}, y_i)$ for $1 \leq i \leq n$; the objective is to find a subset $P' \subset P$ with $|P'| = n - z$ and a $(d-1)$-dimensional hyperplane, represented as a coefficient vector $h = (h_1, h_2, \cdots, h_d) \in \mathbb{R}^d$, such that

$$\mathcal{LR}_1^{-z}(P', h) = \frac{1}{n-z} \sum_{p_i \in P'} \left| Res(p_i, h) \right| \tag{5}$$

$$\text{or} \quad \mathcal{LR}_2^{-z}(P', h) = \frac{1}{n-z} \sum_{p_i \in P'} \left(Res(p_i, h) \right)^2 \tag{6}$$

is minimized. $Res(p_i, h) = y_i - \sum_{j=1}^{d-1} h_j x_{i,j} - h_d$ is the "residual" of p_i to h. The objective functions (5) and (6) are called the "least absolute error" and "least squared error", respectively.

Remark 1. *All the above problems can be extended to weighted case. Suppose each point p has a non-negative weight $w(p)$, then the (squared) distance $||p - c_j||$ ($||p - c_j||^2$) is replaced by $w(p) \cdot ||p - c_j||$ ($w(p) \cdot ||p - c_j||^2$); we can perform the similar modification on $\left| Res(p_i, h) \right|$ and $\left(Res(p_i, h) \right)^2$ for the problem of linear regression with outliers. Moreover, the total weights of the outliers should be equal to z. Namely, we can view each point p as $w(p)$ unit-weight overlapping points.*

Solution range. To analyze the performance of our layered sampling method, we also need to define the "solution range" for the clustering and linear regression problems. Consider the clustering problems first. Given a clustering solution $\tilde{C} = \{\tilde{c}_1, \cdots, \tilde{c}_k\} \subset \mathbb{R}^d$ and $L > 0$, we use "$\tilde{C} \pm L$" to denote the range of solutions $\mathcal{L} =$

$$\Big\{ C = \{c_1, \cdots, c_k\} \mid ||\tilde{c}_j - c_j|| \leq L, \forall 1 \leq j \leq k \Big\}. \tag{7}$$

Next, we define the solution range for linear regression with outliers. Given an instance P, we often normalize the values in each of the first $d-1$ dimensions as the preprocessing step; without loss of generality, we can assume that $x_{i,j} \in [0, D]$ with some $D > 0$ for any $1 \leq i \leq n$ and $1 \leq j \leq d-1$. For convenience, denote by $\mathcal{R}_D$ the region $\{(s_1, s_2, \cdots, s_d) \mid 0 \leq s_j \leq D, \forall 1 \leq j \leq d-1\}$ and thus $P \subset \mathcal{R}_D$ after the normalization. It is easy to see that the region $\mathcal{R}_D$ actually is a vertical square cylinder in the space. Given a coefficient vector (hyperplane) $\tilde{h} = (\tilde{h}_1, \tilde{h}_2, \cdots, \tilde{h}_d) \in \mathbb{R}^d$ and $L > 0$, we use "$\tilde{h} \pm L$" to denote the range of hyperplanes $\mathcal{L} =$

$$\Big\{ h = (h_1, h_2, \cdots, h_d) \mid |Res(p, \tilde{h}) - Res(p, h)| \leq L, \forall p \in \mathcal{R}_D \Big\}. \tag{8}$$

To understand the range defined in (8), we can imagine two linear functions $\tilde{h}^+ = (\tilde{h}_1, \tilde{h}_2, \cdots, \tilde{h}_d + L)$ and $\tilde{h}^- = (\tilde{h}_1, \tilde{h}_2, \cdots, \tilde{h}_d - L)$; if we only consider the region $\mathcal{R}_D$, the range (8) contains all the linear functions "sandwiched" by $\tilde{h}^+$ and $\tilde{h}^-$.

For both the clustering and regression problems, we also say that **the size of the solution range** $\mathcal{L}$ is $|\mathcal{L}| = L$.

2. The Layered Sampling Framework

We present the overview of our layered sampling framework. For the sake of completeness, we first introduce the coreset construction method for the ordinary k-median/means clustering proposed by (Chen, 2009).

Suppose α and $\beta \geq 1$. A "bi-criteria (α, β)-approximation" means that it contains αk cluster centers, and the induced clustering cost is at most β times the optimum. Usually, finding a bi-criteria approximation is much easier than achieving a single-criterion approximation. For example, one can obtain a bi-criteria approximation for k-median/means clustering in linear time with $\alpha = O(1)$ and $\beta = O(1)$ (Chen, 2009). Let $T = \{t_1, t_2, \cdots, t_{\alpha k}\} \subset \mathbb{R}^d$ be the obtained (α, β)-approximate solution of the input instance P. For convenience, we use $\mathbb{B}(c, r)$ to denote the ball centered at a point c with radius $r > 0$. At the beginning of Chen's coreset construction algorithm, it takes two carefully designed values $r > 0$ and $N = O(\log n)$, and partitions the space into $N + 1$ layers $H_0, H_1, \cdots, H_N$, where $H_0 = \cup_{j=1}^{\alpha k} \mathbb{B}(t_j, r)$ and $H_i = \left(\cup_{j=1}^{\alpha k} \mathbb{B}(t_j, 2^i r) \right) \setminus \left(\cup_{j=1}^{\alpha k} \mathbb{B}(t_j, 2^{i-1} r) \right)$ for $1 \leq i \leq N$. It can be proved that P is

covered by $\cup_{i=0}^{N} H_i$; then the algorithm takes a random sample S_i from each layer $P \cap H_i$, and the union $\cup_{i=0}^{N} S_i$ forms the desired coreset S satisfying the condition (1).

However, this approach cannot directly solve the case with outliers. First, it is not easy to obtain a bi-criteria approximation for the problem of k-median/means clustering with outliers (*e.g.,* in linear time). Moreover, it is challenging to guarantee the condition (1) for any feasible solution C, because the set of outliers could change when C changes (this is also the major challenge for proving the correctness of our method later on). We propose a modified version of Chen's coreset construction method and aim to guarantee (2) for a local range of solutions. We take the k-median clustering with outliers problem as an example. Let $\tilde{C} = \{\tilde{c}_1, \cdots, \tilde{c}_k\} \subset \mathbb{R}^d$ be a given solution. Assume $\epsilon > 0$ and $N \in \mathbb{Z}^+$ are two pre-specified parameters. With a slight abuse of notations, we still use $H_0, H_1, \cdots, H_N$ to denote the layers surrounding $\tilde{C}$, *i.e.,*

$$H_0 = \cup_{j=1}^{k} \mathbb{B}(\tilde{c}_j, r); \tag{9}$$

$$H_i = \left(\cup_{j=1}^{k} \mathbb{B}(\tilde{c}_j, 2^i r)\right) \setminus \left(\cup_{j=1}^{k} \mathbb{B}(\tilde{c}_j, 2^{i-1} r)\right) \text{ for } 1 \le i \le N. \tag{10}$$

In addition, let

$$H_{out} = \mathbb{R}^d \setminus \left(\cup_{j=1}^{k} \mathbb{B}(\tilde{c}_j, 2^N r)\right). \tag{11}$$

Here, we set the value r to satisfy the following condition:

$$\left|P \cap H_{out}\right| = (1 + \frac{1}{\epsilon})z. \tag{12}$$

That is, the union of the layers $\cup_{i=0}^{N} H_i$ covers $n - (1 + \frac{1}{\epsilon})z$ points of P and excludes the farthest $(1 + \frac{1}{\epsilon})z$. Obviously, such a value r always exists. Suppose P' is the set of $n - z$ inliers induced by $\tilde{C}$, and then we have

$$2^N r \le \frac{\epsilon}{z} \sum_{p \in P'} \min_{1 \le j \le k} ||p - \tilde{c}_j|| = \frac{\epsilon}{z}(n - z)\mathcal{K}_1^{-z}(P, \tilde{C}) \tag{13}$$

via the Markov's inequality. Our new coreset contains the following $N + 2$ parts:

$$S = S_0 \cup S_1 \cup \cdots \cup S_N \cup S_{out}, \tag{14}$$

where S_i is still a random sample from $P \cap H_i$ for $0 \le i \le N$, and S_{out} contains all the $(1 + \frac{1}{\epsilon})z$ points in H_{out}. In Section 3, we will show that the coreset S of (14) satisfies (2) for the k-median clustering with outliers problem (and similarly for the k-means clustering with outliers problem).

For the linear regression with outliers problem, we apply the similar layered sampling framework. Define $\mathbb{S}(h, r)$ to be the slab centered at a $(d-1)$-dimensional hyperplane h with $r > 0$, *i.e.,* $\mathbb{S}(h, r) = \{p \in \mathbb{R}^d \mid -r \le Res(p, h) \le r\}$. Let P be an instance, and $\tilde{h} = (\tilde{h}_1, \cdots, \tilde{h}_d) \in \mathbb{R}^d$ be a given hyperplane. We divide the space into $N + 2$ layers $H_0, H_1, \cdots, H_N, H_{out}$, where

$$H_0 = \mathbb{S}(\tilde{h}, r); \tag{15}$$

$$H_i = \mathbb{S}(\tilde{h}, 2^i r) \setminus \mathbb{S}(\tilde{h}, 2^{i-1} r) \text{ for } 1 \le i \le N; \tag{16}$$

$$H_{out} = \mathbb{R}^d \setminus \mathbb{S}(\tilde{h}, 2^N r). \tag{17}$$

Similar to (12), we also require the value r to satisfy the following condition:

$$\left|P \cap H_{out}\right| = \left|P \setminus \mathbb{S}(\tilde{h}_N, 2^N r)\right| = (1 + \frac{1}{\epsilon})z. \tag{18}$$

And consequently, we have

$$2^N r \le \frac{\epsilon}{z}(n - z)\mathcal{LR}_1^{-z}(P, \tilde{h}). \tag{19}$$

Then, we construct the coreset for linear regression with outliers by the same manner of (14).

Algorithm 1 LAYERED SAMPLING FOR k-MED-OUTLIER

Input: An instance $P \subset \mathbb{R}^d$ of k-median clustering with z outliers, a solution $\tilde{C} = \{\tilde{c}_1, \cdots, \tilde{c}_k\}$, and two parameters $\epsilon, \eta \in (0, 1)$.

1. Let $\gamma = z/(n - z)$ and $N = \lceil \log \frac{1}{\gamma} \rceil$. Compute the value r satisfying (12).
2. As described in (9), (10), and (11), the space is partitioned into $N + 2$ layers $H_0, H_1, \cdots, H_N$ and H_{out}.
3. Randomly sample $\min\left\{O(\frac{1}{\epsilon^2} kd \log \frac{d}{\epsilon} \log \frac{N}{\eta}), |P \cap H_i|\right\}$ points, denoted by S_i, from $P \cap H_i$ for $0 \le i \le N$.
4. For each point $p \in S_i$, set its weight to be $|P \cap H_i|/|S_i|$; let $S_H = \cup_{i=0}^{N} S_i$.

Output $S = S_H \cup (P \cap H_{out})$.

3. k-Median/Means Clustering with Outliers

In this section, we provide the details on applying our layered sampling framework to the problem of k-median clustering with outliers. See Algorithm 1. The algorithm and analysis can be easily modified to handle k-means clustering with outliers, where the only difference is that we need to replace (13) by "$2^N r \le \sqrt{\frac{\epsilon}{z}(n - z)\mathcal{K}_2^{-z}(P, \tilde{C})}$".

Theorem 1. *Algorithm 1 returns a point set S having the size $|S| = \tilde{O}^{[1]}(\frac{1}{\epsilon^2} kd) + (1 + \frac{1}{\epsilon})z$. Moreover, with probability*

[1]The asymptotic notation $\tilde{O}(f) = O\big(f \cdot polylog(\frac{d}{\gamma\epsilon\eta})\big)$.

at least $1-\eta$, for any $L>0$ and any solution $C \in \tilde{C} \pm L$, we have

$$\mathcal{K}_1^{-z}(S,C) \in \mathcal{K}_1^{-z}(P,C) \pm \epsilon\big(\mathcal{K}_1^{-z}(P,\tilde{C})+L\big). \quad (20)$$

Here, S is a weighted instance of k-median clustering with outliers, and the total weight of outliers is z (see Remark 1).

Remark 2. *(1) The running time of Algorithm 1 is $O(knd)$. For each point $p \in P$, we compute its shortest distance to $\tilde{C}$, $\min_{1\le j\le k}||p-\tilde{c}_j||$; then select the farthest $(1+1/\epsilon)z$ points and compute the value r by running the linear time selection algorithm (Blum et al., 1973); finally, we obtain the $N+1$ layers H_i with $0 \le i \le N$ and take the samples $S_0, S_1, \cdots, S_N$ from them.*

(2) Comparing with the standard coreset (1), our result contains an additive error $\epsilon\big(\mathcal{K}_1^{-z}(P,\tilde{C})+L\big)$ in (20) that depends on the initial objective value $\mathcal{K}_1^{-z}(P,\tilde{C})$ and the size L of the solution range. In particular, the smaller the range size L, the lower the error of our coreset.

(3) The algorithm of (Chen et al., 2018) also returns a summary for compressing the input data. But there are two major differences comparing with our result. First, their summary guarantees a constant factor of distortion on the clustering cost, while our error approaches 0 if ϵ is small enough. Second, their construction algorithm (called "successive sampling" from (Mettu & Plaxton, 2004)) needs to scan the data multiple passes, while our Algorithm 1 is much simpler and only needs to read the data in one-pass. We also compare these two methods in our experiments.

To prove Theorem 1, we first show that S_H is a good approximation of $P \setminus H_{out}$. Fixing a solution $C \in \tilde{C} \pm L$, we view the distance from each point $p \in P$ to C, *i.e.*, $\min_{1\le j\le k}||p-c_j||$, as a random variable x_p. For any point $p \in P \cap H_i$ with $0 \le i \le N$, we have the following bounds for x_p. Suppose p is covered by $\mathbb{B}(\tilde{c}_{j_1}, 2^i r)$. Let the nearest neighbor of p in C be c_{j_2}. Then, we have the upper bound

$$\begin{aligned} x_p &= ||p-c_{j_2}|| \le ||p-c_{j_1}|| \\ &\le ||p-\tilde{c}_{j_1}|| + ||\tilde{c}_{j_1}-c_{j_1}|| \\ &\le 2^i r + L. \end{aligned} \quad (21)$$

Similarly, we have the lower bound

$$\left.\begin{array}{ll} x_p \ge \max\{2^{i-1}r - L, 0\} & \text{if } i \ge 1; \\ x_p \ge 0 & \text{if } i = 0. \end{array}\right\} \quad (22)$$

Therefore, we can take a sufficiently large random sample $\hat{S}_i$ from $P \cap H_i$, such that $\frac{1}{|\hat{S}_i|}\sum_{p\in\hat{S}_i} x_p \approx \frac{1}{|P\cap H_i|}\sum_{p\in P\cap H_i} x_p$ with certain probability. Specifically, combining (21) and (22), we have the following lemma through the Hoeffding's inequality.

Lemma 1. *Let $\eta \in (0,1)$. If we randomly sample $O(\frac{1}{\epsilon^2}\log\frac{1}{\eta})$ points, denote by $\hat{S}_i$, from $P \cap H_i$, then with probability $1-\eta$,*

$$\Big|\frac{1}{|\hat{S}_i|}\sum_{p\in\hat{S}_i} x_p - \frac{1}{|P\cap H_i|}\sum_{p\in P\cap H_i} x_p\Big| \le \epsilon(2^i r + 2L).$$

Lemma 1 is only for a fixed solution C. To guarantee the result for any $C \in \tilde{C} \pm L$, we discretize the range $\tilde{C} \pm L$. Imagine that we build a grid inside each $\mathbb{B}(\tilde{c}_j, L)$ for $1 \le j \le k$, where the grid side length is $\frac{\epsilon}{\sqrt{d}}L$. Denote by G_j the set of grid points inside each $\mathbb{B}(\tilde{c}_j, L)$, and then $\mathcal{G} = G_1 \times G_2 \times \cdots \times G_k$ contains $O\Big(\big(\frac{2\sqrt{d}}{\epsilon}\big)^{kd}\Big)$ k-tuple points of $\tilde{C} \pm L$ in total. We increase the sample size in Lemma 1 via replacing η by $\frac{\eta}{N\cdot|\mathcal{G}|}$ in the sample size "$O(\frac{1}{\epsilon^2}\log\frac{1}{\eta})$". As a consequence, through taking the union bound for the success probability, we have the following result.

Lemma 2. *S_i is the sample obtained from $P \cap H_i$ in Step 3 of Algorithm 1 for $0 \le i \le N$. Then with probability $1-\eta$,*

$$\Big|\frac{1}{|S_i|}\sum_{p\in S_i} x_p - \frac{1}{|P\cap H_i|}\sum_{p\in P\cap H_i} x_p\Big| \le \epsilon(2^i r + 2L).$$

for each $i = \{0, 1, \cdots, N\}$ and any $C \in \mathcal{G}$.

Next, we show that for any $C \in \tilde{C} \pm L$ (in particular the solutions in $\big(\tilde{C} \pm L\big) \setminus \mathcal{G}$), Lemma 2 is true. For any solution $C = \{c_1, \cdots, c_k\} \in \tilde{C} \pm L$, let $C' = \{c'_1, \cdots, c'_k\}$ be its nearest neighbor in $\mathcal{G}$, *i.e.*, c'_j is the grid point of the cell containing c_j in G_j, for $1 \le j \le k$. Also, denote by x'_p the distance $\min_{1\le j\le k}||p-c'_j||$. Then we consider to bound the error $\big|\frac{1}{|S_i|}\sum_{p\in S_i} x_p - \frac{1}{|P\cap H_i|}\sum_{p\in P\cap H_i} x_p\big|$ through C'. By using the triangle inequality, we have

$$\begin{aligned} & \Big|\frac{1}{|S_i|}\sum_{p\in S_i} x_p - \frac{1}{|P\cap H_i|}\sum_{p\in P\cap H_i} x_p\Big| \quad (23) \\ \le\ & \underbrace{\Big|\frac{1}{|S_i|}\sum_{p\in S_i} x_p - \frac{1}{|S_i|}\sum_{p\in S_i} x'_p\Big|}_{\text{(a)}} \\ +\ & \underbrace{\Big|\frac{1}{|S_i|}\sum_{p\in S_i} x'_p - \frac{1}{|P\cap H_i|}\sum_{p\in P\cap H_i} x'_p\Big|}_{\text{(b)}} \\ +\ & \underbrace{\Big|\frac{1}{|P\cap H_i|}\sum_{p\in P\cap H_i} x'_p - \frac{1}{|P\cap H_i|}\sum_{p\in P\cap H_i} x_p\Big|}_{\text{(c)}}. \end{aligned}$$

In (23), the term (b) is bounded by Lemma 2 since $C' \in \mathcal{G}$. To bound the terms (a) and (c), we study the difference $|x_p - x'_p|$ for each point p. Suppose the nearest neighbor of

p in C (*resp.*, C') is c_{j_1} (*resp.*, c'_{j_2}). Then,

$$\begin{aligned} x_p &= ||p - c_{j_1}|| \le ||p - c_{j_2}|| \\ &\le ||p - c'_{j_2}|| + ||c'_{j_2} - c_{j_2}|| \\ &\le ||p - c'_{j_2}|| + \epsilon L = x'_p + \epsilon L, \end{aligned} \tag{24}$$

where the last inequality comes from the fact that c'_{j_2} and c_{j_2} are in the same grid cell with side length $\frac{\epsilon}{\sqrt{d}}L$. Similarly, we have $x'_p \le x_p + \epsilon L$. Overall, $|x_p - x'_p| \le \epsilon L$. As a consequence, the terms (a) and (c) in (23) are both bounded by ϵL. Overall, (23) becomes

$$\begin{aligned} &\Big|\frac{1}{|S_i|}\sum_{p\in S_i} x_p - \frac{1}{|P\cap H_i|}\sum_{p\in P\cap H_i} x_p\Big| \\ \le\ & O(\epsilon)(2^i r + L). \end{aligned} \tag{25}$$

For convenience, we use P_H to denote the set $\cup_{i=0}^N (P\cap H_i)$.

Lemma 3. *Let $S_0, S_1, \cdots, S_N$ be the samples obtained in Algorithm 1. Then, with probability $1-\eta$,*

$$\begin{aligned} &\frac{1}{n-z}\Big|\sum_{i=0}^N \frac{|P\cap H_i|}{|S_i|}\sum_{p\in S_i} x_p - \sum_{p\in P_H} x_p\Big| \\ \le\ & O(\epsilon)\Big(\mathcal{K}_1^{-z}(P,\tilde{C}) + L\Big) \end{aligned} \tag{26}$$

for any $C \in \tilde{C} \pm L$.

Proof. For convenience, let $Err_i = \big|\frac{1}{|S_i|}\sum_{p\in S_i} x_p - \frac{1}{|P\cap H_i|}\sum_{p\in P\cap H_i} x_p\big|$ for $0 \le i \le N$. We directly have $Err_i \le O(\epsilon)(2^i r + L)$ from (25). Moreover, the left hand-side of (26) $= \frac{1}{n-z}\sum_{i=0}^N |P\cap H_i| \cdot Err_i$

$$\begin{aligned} &\le \frac{O(\epsilon)}{n-z}\sum_{i=0}^N |P\cap H_i|\cdot(2^i r + L) \\ &= O(\epsilon)\cdot\sum_{i=0}^N \frac{|P\cap H_i|}{n-z}2^i r + O(\epsilon)L. \end{aligned} \tag{27}$$

It is easy to know that the term $\sum_{i=0}^N \frac{|P\cap H_i|}{n-z}2^i r$ of (27) is at most $\frac{1}{n-z}(|P\cap H_0|r + 2\sum_{p\in P_H\setminus H_0} x_p) \le r + 2\mathcal{K}_1^{-z}(P,\tilde{C})$. Note we set $N = \lceil \log\frac{1}{\gamma}\rceil$ in Algorithm 1. Together with (13), we know $r \le \epsilon\mathcal{K}_1^{-z}(P,\tilde{C})$ and thus $\sum_{i=0}^N \frac{|P\cap H_i|}{n-z}2^i r \le O(1)\mathcal{K}_1^{-z}(P,\tilde{C})$. So (26) is true. □

Below, we always assume that (26) is true and consider to prove (20) of Theorem 1. The set P is partitioned into two parts: P_{in}^C and P_{out}^C by C, where P_{out}^C is the z farthest points to C (*i.e.*, the outliers) and $P_{in}^C = P \setminus P_{out}^C$. Similarly, the coreset S is also partitioned into two parts S_{in}^C and S_{out}^C by C, where S_{out}^C is the set of outliers with total weights z. In other words, we need to prove

$$\sum_{p\in S_{in}^C} w(p)x_p \approx \sum_{p\in P_{in}^C} x_p. \tag{28}$$

Consider two cases: (i) $P_H \setminus P_{in}^C = \emptyset$ and (ii) $P_H \setminus P_{in}^C \neq \emptyset$. Intuitively, the case (i) indicates that the set P_{in}^C occupies the whole region $\cup_{i=0}^N H_i$; the case (ii) indicates that the region $\cup_{i=0}^N H_i$ contains some outliers from P_{out}^C. In the following subsections, we prove that (20) holds for both cases. For ease of presentation, we use $w(U)$ to denote the total weight of a weighted point set U (please be not confused with $|U|$, which is the number of points in U).

3.1. Case (i): $P_H \setminus P_{in}^C = \emptyset$

We prove the following key lemma first.

Lemma 4. *If $P_H \setminus P_{in}^C = \emptyset$, $S_{in}^C = S_H \cup (P_{in}^C \setminus P_H)$ and $S_{out}^C = P_{out}^C$ (recall $S_H = \cup_{i=0}^N S_i$ from Algorithm 1).*

Proof. First, the assumption $P_H \setminus P_{in}^C = \emptyset$ implies

$$P_H \subset P_{in}^C; \tag{29}$$

$$|P_{in}^C \setminus P_H| = |P_{in}^C| - |P_H|. \tag{30}$$

In addition, since $S_H \subset P_H$, we have $S_H \subset P_{in}^C$ from (29). Consequently, the set $S_H \cup (P_{in}^C \setminus P_H) \subset P_{in}^C$. Therefore, for any $p \in S_H \cup (P_{in}^C \setminus P_H)$ and any $q \in P_{out}^C$, $x_p \le x_q$. Moreover, the set $S \setminus \Big(S_H \cup (P_{in}^C \setminus P_H)\Big)$

$$\begin{aligned} &= \Big(S_H \cup (P \setminus P_H)\Big) \setminus \Big(S_H \cup (P_{in}^C \setminus P_H)\Big) \\ &= (P \setminus P_H) \setminus (P_{in}^C \setminus P_H) \\ &\underbrace{=}_{\text{by (29)}} P \setminus P_{in}^C = P_{out}^C. \end{aligned} \tag{31}$$

Note $|P_{out}^C| = z$. As a consequence, S_{in}^C should be exactly the set $S_H \cup (P_{in}^C \setminus P_H)$, and $S_{out}^C = P_{out}^C$. □

Lemma 5. *If $P_H \setminus P_{in}^C = \emptyset$, (20) is true.*

Proof. Because the set S_{in}^C is equal to $S_H \cup (P_{in}^C \setminus P_H)$ from Lemma 4, the objective value $\mathcal{K}_1^{-z}(S,C) = \frac{1}{n-z}\Big(\sum_{p\in S_H} w(p)x_p + \sum_{p\in P_{in}^C\setminus P_H} x_p\Big) =$

$$\frac{1}{n-z}\Big(\sum_{i=0}^N \frac{|P\cap H_i|}{|S_i|}\sum_{p\in S_i} x_p + \sum_{p\in P_{in}^C\setminus P_H} x_p\Big). \tag{32}$$

From Lemma 3, the value of (32) is no larger than

$$\begin{aligned} &\le \frac{1}{n-z}\Big(\sum_{p\in P_H} x_p + O(\epsilon)(n-z)(\mathcal{K}_1^{-z}(P,\tilde{C}) + L) \\ &\quad + \sum_{p\in P_{in}^C\setminus P_H} x_p\Big). \end{aligned} \tag{33}$$

Note that $P_H \setminus P_{in}^C = \emptyset$, and thus the sum of the two terms $\sum_{p\in P_H} x_p$ and $\sum_{p\in P_{in}^C\setminus P_H} x_p$ in (33) is $\sum_{p\in P_{in}^C} x_p$. Therefore, $\mathcal{K}_1^{-z}(S,C) \le$

$$\begin{aligned} & \frac{1}{n-z}\sum_{p\in P_{in}^C} x_p + O(\epsilon)(\mathcal{K}_1^{-z}(P,\tilde{C})+L) \\ = \;& \mathcal{K}_1^{-z}(P,C) + O(\epsilon)(\mathcal{K}_1^{-z}(P,\tilde{C})+L). \end{aligned} \tag{34}$$

Similarly, we have $\mathcal{K}_1^{-z}(S,C) \ge \mathcal{K}_1^{-z}(P,C) - O(\epsilon)(\mathcal{K}_1^{-z}(P,\tilde{C})+L)$. Thus, (20) is true. □

3.2. Case (ii): $P_H \setminus P_{in}^C \neq \emptyset$

Since $S\setminus P_H = P\setminus P_H$ are the outermost $(1+1/\epsilon)z$ points to $\tilde{C}$, we have the following claim first.

Claim 1. *Either $S_{in}^C \setminus P_H \subseteq P_{in}^C \setminus P_H$ or $P_{in}^C \setminus P_H \subseteq S_{in}^C \setminus P_H$ is true.*

Lemma 6. *If $P_H \setminus P_{in}^C \neq \emptyset$, we have $x_p \le 2^N r + L$ for any $p \in S_{in}^C \cup P_{in}^C \cup P_H$.*

Proof. We consider the points in the three parts P_H, P_{in}^C, and S_{in}^C separately.

(1) Due to (21), we have $x_p \le 2^N r + L$ for any $p \in P_H$.

(2) Arbitrarily select one point p_0 from $P_H \setminus P_{in}^C$. By (21) again, we have $x_{p_0} \le 2^N r + L$. Also, because $P_H \setminus P_{in}^C \subset P_{out}^C$, we directly have $x_p \le x_{p_0}$ for any $p \in P_{in}^C$. Namely, $x_p \le 2^N r + L$ for any $p \in P_{in}^C$.

(3) Below, we consider the points in S_{in}^C. If $w(S_{in}^C \cap P_H) > \left|P_{in}^C \cap P_H\right|$, *i.e.*, P_H contains more inliers of S than that of P, then the outer region H_{out} should contain less inliers of S than that of P. Thus, from Claim 1, we have $S_{in}^C \setminus P_H \subseteq P_{in}^C \setminus P_H$. Hence, $S_{in}^C = (S_{in}^C \setminus P_H) \cup (S_{in}^C \cap P_H) \subseteq (P_{in}^C \setminus P_H) \cup P_H = P_{in}^C \cup P_H$. From **(1)** and **(2)**, we know $x_p \le 2^N r + L$ for any $p \in S_{in}^C$.

Else, $w(S_{in}^C \cap P_H) \le \left|P_{in}^C \cap P_H\right|$. Then $w(S_{in}^C \cap S_H) \le \left|P_{in}^C \cap P_H\right|$ since $S_{in}^C \cap P_H = S_{in}^C \cap S_H$. Because $w(S_H) = |P_H|$, we have

$$w(S_H \setminus S_{in}^C) \ge |P_H \setminus P_{in}^C|. \tag{35}$$

Also, the assumption $P_H \setminus P_{in}^C \neq \emptyset$ implies $w(S_H \setminus S_{in}^C) \ge |P_H \setminus P_{in}^C| > 0$, *i.e.,*

$$S_H \setminus S_{in}^C \neq \emptyset. \tag{36}$$

Arbitrarily select one point p_0 from $S_H \setminus S_{in}^C$. We know $x_{p_0} \le 2^N r + L$ since $p_0 \in S_H \setminus S_{in}^C \subset P_H$. Also, for any point $p \in S_{in}^C$, we have $x_p \le x_{p_0}$ because $p_0 \in S_H \setminus S_{in}^C \subset S_{out}^C$. Therefore $x_p \le 2^N r + L$. □

Lemma 7. *If $P_H \setminus P_{in}^C \neq \emptyset$, (20) is true.*

Proof. We prove the upper bound of $\mathcal{K}_1^{-z}(S,C)$ first. We analyze the clustering costs of the two parts $S_{in}^C \cap S_H$ and $S_{in}^C \setminus S_H$ separately.

$$\mathcal{K}_1^{-z}(S,C) = \frac{1}{n-z}\Big(\underbrace{\sum_{p\in S_{in}^C\cap S_H} w(p)x_p}_{\textbf{(a)}} + \underbrace{\sum_{p\in S_{in}^C\setminus S_H} x_p}_{\textbf{(b)}}\Big).$$

Note the points of $S_{in}^C \setminus S_H$ have unit-weight (since $S_{in}^C \setminus S_H \subseteq P \setminus P_H$ are the points from the outermost $(1+\frac{1}{\epsilon})z$ points of P). Obviously, the part **(a)** is no larger than

$$\begin{aligned} & \sum_{p\in S_H} w(p)x_p = \sum_{i=0}^{N} \frac{|P\cap H_i|}{|S_i|}\sum_{p\in S_i} x_p \\ \le \;& \sum_{p\in P_H} x_p + O(\epsilon)(n-z)(\mathcal{K}_1^{-z}(P,\tilde{C})+L) \end{aligned} \tag{37}$$

from Lemma 3. The set P_H consists of two parts $P_H \cap P_{in}^C$ and $P_H \setminus P_{in}^C$. From Lemma 6 and the fact $|P_H \setminus P_{in}^C| \le |P_{out}^C| = z$, we know $\sum_{p\in P_H\setminus P_{in}^C} x_p \le z(2^N r + L)$. Thus, the upper bound of the part **(a)** becomes

$$\begin{aligned} & \textstyle\sum_{p\in P_H\cap P_{in}^C} x_p + z(2^N r + L) + \\ & O(\epsilon)(n-z)(\mathcal{K}_1^{-z}(P,\tilde{C})+L). \end{aligned} \tag{38}$$

To bound the part **(b)**, we consider the size $|S_{in}^C \setminus S_H|$. Since the total weight of outliers is z, $w\big(S_H \cap S_{in}^C\big)$

$$\begin{aligned} &= w(S_H) - w\big(S_H \cap S_{out}^C\big) \\ &\ge w(S_H) - z \\ &= \big|P_H\big| - z \\ &\ge \big|P_H \cap P_{in}^C\big| - z. \end{aligned} \tag{39}$$

Together with the fact $w\big(S_H \cap S_{in}^C\big) + \big|S_{in}^C \setminus S_H\big| = \big|P_H \cap P_{in}^C\big| + \big|P_{in}^C \setminus P_H\big| = n - z$, we have

$$\big|S_{in}^C \setminus S_H\big| \le \big|P_{in}^C \setminus P_H\big| + z. \tag{40}$$

Therefore $\Big|\big(S_{in}^C \setminus S_H\big) \setminus \big(P_{in}^C \setminus P_H\big)\Big| \le z$ from Claim 1. Through Lemma 6 again, we know that the part **(b)** is no larger than $\sum_{p\in P_{in}^C\setminus P_H} x_p + \Big|\big(S_{in}^C \setminus S_H\big) \setminus \big(P_{in}^C \setminus P_H\big)\Big| \cdot (2^N r + L)$

$$\le \sum_{p\in P_{in}^C\setminus P_H} x_p + z(2^N r + L). \tag{41}$$

Putting (38) and (41) together, we have $\mathcal{K}_1^{-z}(S,C) \le$

$$\begin{aligned} & \mathcal{K}_1^{-z}(P,C) + O(\epsilon)(\mathcal{K}_1^{-z}(P,\tilde{C})+L) \\ & + \frac{2z}{n-z}(2^N r + L). \end{aligned} \tag{42}$$

Recall $\gamma = z/(n-z)$ in Algorithm 1. If $\gamma \geq \epsilon$, the size of our coreset S is at least $(1+1/\epsilon)z \geq n$; that is, S contains all the points of P. For the other case $\epsilon > \gamma$, together with (13), the term $\frac{2z}{n-z}(2^N r + L)$ in (42) is at most

$$O(\epsilon)(\mathcal{K}_1^{-z}(P,\tilde{C}) + L). \tag{43}$$

Overall, $\mathcal{K}_1^{-z}(S,C) \leq \mathcal{K}_1^{-z}(P,C) + O(\epsilon)(\mathcal{K}_1^{-z}(P,\tilde{C}) + L)$ via (42). So we complete the proof for the upper bound.

Now we consider the lower bound of $\mathcal{K}_1^{-z}(S,C)$. Denote by $X = S_H \cup (P_{in}^C \setminus P_H)$ and $Y = X \setminus S_{in}^C$. Obviously,

$$\mathcal{K}_1^{-z}(S,C) \geq \frac{1}{n-z}\Big(\underbrace{\sum_{p\in X} w(p)x_p}_{\textbf{(c)}} - \underbrace{\sum_{p\in Y} w(p)x_p}_{\textbf{(d)}}\Big).$$

From Lemma 3, the part **(c)** is at least

$$\sum_{p\in P_H} x_p - O(\epsilon)(\mathcal{K}_1^{-z}(P,\tilde{C}) + L) + \sum_{p\in P_{in}^C \setminus P_H} x_p \geq \sum_{p\in P_{in}^C} x_p - O(\epsilon)(\mathcal{K}_1^{-z}(P,\tilde{C}) + L). \tag{44}$$

Further, since $w(Y) \leq z$ and $Y \subseteq X \subseteq P_{in}^C \cup P_H$, the part **(d)** is no larger than $z(2^N r + L)$ from Lemma 6. Using the similar manner for proving the upper bound, we know that $\mathcal{K}_1^{-z}(S,C) \geq \mathcal{K}_1^{-z}(P,C) - O(\epsilon)(\mathcal{K}_1^{-z}(P,\tilde{C}) + L)$. □

4. Linear Regression with Outliers

In this section, we consider the problem of linear regression with outliers. Our algorithm and analysis are for the objective function $\mathcal{LR}_1^{-z}$, and the ideas can be extended to handle the objective function $\mathcal{LR}_2^{-z}$.

Theorem 2. *Algorithm 2 returns a point set S having the size $|S| = \tilde{O}(\frac{1}{\epsilon^2}d) + (1+\frac{1}{\epsilon})z$. Moreover, with probability at least $1-\eta$, for any $L > 0$ and any solution $h \in \tilde{h} \pm L$, we have*

$$\mathcal{LR}_1^{-z}(S,h) \in \mathcal{LR}_1^{-z}(P,h) \pm \epsilon\big(\mathcal{LR}_1^{-z}(P,\tilde{h}) + L\big). \tag{45}$$

Here, S is a weighted instance of linear regression with outliers, and the total weight of outliers is z (see Remark 1).

We still use P_H to denote the set $\cup_{i=0}^N (P \cap H_i)$. First, we need to prove that S_H is a good approximation of P_H. Given a hyperplane h, we define a random variable $x_p = |Res(p,h)|$ for each $p \in P$. If $p \in H_i$ for $0 \leq i \leq N$, similar to (21) and (22), we have the following bounds for x_p: $x_p \leq 2^i r + L$; $x_p \geq \max\{2^{i-1}r - L, 0\}$ if $i \geq 1$ and $x_p \geq 0$ if $i = 0$.

Then, we can apply the similar idea of Lemma 3 to obtain the following lemma, where the only difference is about the discretization on $\tilde{h} \pm L$. Recall that $\tilde{h}$ is defined by the coefficients $\tilde{h}_1, \cdots, \tilde{h}_d$ and the input set P is normalized within the region $\mathcal{R}_D$. We build a grid inside each vertical segment $\overline{l_j u_j}$ for $0 \leq j \leq d-1$, where $l_0 = (0, \cdots, 0, \tilde{h}_d - L)$, $u_0 = (0, \cdots, 0, \tilde{h}_d + L)$, and

$$l_j = (0, \cdots, 0, \underbrace{D}_{j-th}, 0, \cdots, 0, \tilde{h}_j D + \tilde{h}_d - L), \tag{46}$$

$$u_j = (0, \cdots, 0, \underbrace{D}_{j-th}, 0, \cdots, 0, \tilde{h}_j D + \tilde{h}_d + L) \tag{47}$$

for $j \neq 0$; the grid length is $\frac{\epsilon}{2d}L$. Denote by G_j the set of grid points inside the segment $\overline{l_j u_j}$. Obviously, $\mathcal{G} = G_0 \times G_1 \times \cdots \times G_{d-1}$ contains $(\frac{4d}{\epsilon})^d$ d-tuple points, and each tuple determines a $(d-1)$-dimensional hyperplane in $\tilde{h} \pm L$; moreover, we have the following claim.

Claim 2. *For each $h \in \tilde{h} \pm L$, there exist a hyperplane h' determined by a d-tuple points from $\mathcal{G}$, such that $|Res(p,h) - Res(p,h')| \leq \epsilon L$ for any $p \in P$.*

Lemma 8. *Let $S_0, S_1, \cdots, S_N$ be the samples obtained in Algorithm 2. Then, with probability $1-\eta$,*

$$\frac{1}{n-z}\Big|\sum_{i=0}^N \frac{|P\cap H_i|}{|S_i|}\sum_{p\in S_i} x_p - \sum_{p\in P_H} x_p\Big| \leq O(\epsilon)\Big(\mathcal{LR}_1^{-z}(P,\tilde{h}) + L\Big) \tag{48}$$

for any $h \in \tilde{h} \pm L$.

We fix a solution $h \in \tilde{h} \pm L$. Similar to the proof of Theorem 1 in Section 3, we also consider the two parts P_{in}^h and P_{out}^h of P partitioned by h, where P_{out}^h is the z farthest points to the hyperplane h (*i.e.,* the outliers) and

Algorithm 2 LAYERED SAMPLING FOR LIN1-OUTLIER

Input: An instance $P \subset \mathbb{R}^d$ of linear regression with z outliers, a solution $\tilde{h} = (\tilde{h}_1, \cdots, \tilde{h}_k)$, and two parameters $\epsilon, \eta \in (0,1)$.

1. Let $\gamma = z/(n-z)$ and $N = \lceil \log \frac{1}{\gamma} \rceil$. Compute the value r satisfying (18).
2. As described in (15), (16), and (17), the space is partitioned into $N+2$ layers $H_0, H_1, \cdots, H_N$ and H_{out}.
3. Randomly sample $\min\Big\{O(\frac{1}{\epsilon^2}d\log\frac{d}{\epsilon}\log\frac{N}{\eta}), |P\cap H_i|\Big\}$ points, denoted by S_i, from $P \cap H_i$ for $0 \leq i \leq N$.
4. For each point $p \in S_i$, set its weight to be $|P\cap H_i|/|S_i|$; let $S_H = \cup_{i=0}^N S_i$.

Output $S = S_H \cup (P \cap H_{out})$.

$P_{in}^h = P \setminus P_{out}^h$. Similarly, S is also partitioned into two parts S_{in}^h and S_{out}^h by h, where S_{out}^h is the set of outliers with total weights z. For case (i) $P_H \setminus P_{in}^h = \emptyset$ and case (ii) $P_H \setminus P_{in}^h \neq \emptyset$, we can apply almost the identical ideas in Section 3.1 and 3.2 respectively to prove (45).

5. Conclusion

To reduce the time complexities of existing algorithms for clustering and linear regression with outliers, we propose a new variant of coreset method which can guarantee the quality for any solution in a local range surrounding the given initial solution. Due to the space limit, we leave the complete experimental results to our supplement. In future, it is worth considering to apply our framework to a broader range of robust optimization problems, such as logistic regression with outliers and Gaussian mixture model with outliers.

A. Proof of Claim 1

Since S_{in}^C is the set of inliers to C, there must exist some value $r_S > 0$ such that

$$S_{in}^C = \{p \mid p \in S, \min_{1 \le j \le k} ||p - c_j|| \le r_S\}. \tag{49}$$

And therefore

$$S_{in}^C \setminus P_H = \{p \mid p \in S \setminus P_H, \min_{1 \le j \le k} ||p - c_j|| \le r_S\}. \tag{50}$$

Similarly, there exists some value $r_P > 0$ such that

$$P_{in}^C \setminus P_H = \{p \mid p \in P \setminus P_H, \min_{1 \le j \le k} ||p - c_j|| \le r_P\}. \tag{51}$$

Note $S \setminus P_H = P \setminus P_H$. So, if $r_S \le r_P$, we have $S_{in}^C \setminus P_H \subseteq P_{in}^C \setminus P_H$. Otherwise, $P_{in}^C \setminus P_H \subseteq S_{in}^C \setminus P_H$.

B. Proof of Claim 2

Let $h = (h_1, \cdots, h_d)$, and suppose $h' = (h'_1, \cdots, h'_d)$ is h's nearest neighbor in $\mathcal{G}$, *i.e.*, $|h'_d - h_d| \le \frac{\epsilon}{2d} L$ and $|Dh'_j + h'_d - Dh_j - h_d| \le \frac{\epsilon}{2d} L$ for $1 \le j \le d-1$. Then,

$$\begin{aligned} |h'_j - h_j| &\le \frac{1}{D}(\frac{\epsilon}{2d} L + |h'_d - h_d|) \\ &\le \frac{\epsilon}{Dd} L \end{aligned} \tag{52}$$

for $1 \le j \le d-1$. For any $p = (x_1, \cdots, x_d) \in \mathcal{R}_D$,

$$\begin{aligned} & |Res(p, h) - Res(p, h')| \\ \le & \sum_{j=1}^{d-1} |h'_j - h_j| \cdot |x_j| + |h'_d - h_d| \\ \le & \sum_{j=1}^{d-1} |h'_j - h_j| \cdot D + |h'_d - h_d| \\ \le & \epsilon L. \end{aligned} \tag{53}$$

References

Arthur, D. and Vassilvitskii, S. k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 1027–1035. Society for Industrial and Applied Mathematics, 2007.

Awasthi, P. and Balcan, M.-F. Center based clustering: A foundational perspective. 2014.

Bachem, O., Lucic, M., and Krause, A. Practical coreset constructions for machine learning. *arXiv preprint arXiv:1703.06476*, 2017.

Bhaskara, A., Vadgama, S., and Xu, H. Greedy sampling for approximate clustering in the presence of outliers. In *Advances in Neural Information Processing Systems*, pp. 11146–11155, 2019.

Bhatia, K., Jain, P., and Kar, P. Robust regression via hard thresholding. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pp. 721–729, 2015.

Blum, M., Floyd, R. W., Pratt, V., Rivest, R. L., and Tarjan, R. E. Time bounds for selection. *Journal of Computer and System Sciences*, 7(4):448–461, 1973.

Boutsidis, C., Drineas, P., and Magdon-Ismail, M. Near-optimal coresets for least-squares regression. *IEEE Trans. Information Theory*, 59(10):6880–6892, 2013. doi: 10.1109/TIT.2013.2272457.

Chandola, V., Banerjee, A., and Kumar, V. Anomaly detection: A survey. *ACM Computing Surveys (CSUR)*, 41(3): 15, 2009.

Chawla, S. and Gionis, A. k-means–: A unified approach to clustering and outlier detection. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, pp. 189–197. SIAM, 2013.

Chen, J., Azer, E. S., and Zhang, Q. A practical algorithm for distributed clustering and outlier detection. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*, pp. 2253–2262, 2018.

Chen, K. A constant factor approximation algorithm for k-median clustering with outliers. In *Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 826–835. Society for Industrial and Applied Mathematics, 2008.

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

Dasgupta, A., Drineas, P., Harb, B., Kumar, R., and Mahoney, M. W. Sampling algorithms and coresets for \ell_p regression. *SIAM Journal on Computing*, 38(5):2060–2078, 2009.

Ding, H. and Xu, J. Sub-linear time hybrid approximations for least trimmed squares estimator and related problems. In *30th Annual Symposium on Computational Geometry, SOCG'14, Kyoto, Japan, June 08 - 11, 2014*, pp. 110, 2014. doi: 10.1145/2582112.2582131.

Drineas, P., Mahoney, M. W., and Muthukrishnan, S. Sampling algorithms for l_2 regression and applications. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pp. 1127–1136. Society for Industrial and Applied Mathematics, 2006.

Erickson, J. and Seidel, R. Better lower bounds on detecting affine and spherical degeneracies. *Discrete & Computational Geometry*, 13:41–57, 1995. doi: 10.1007/BF02574027.

Feldman, D. Core-sets: An updated survey. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, 10(1), 2020.

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

Feldman, D., Schmidt, M., and Sohler, C. Turning big data into tiny data: Constant-size coresets for *k*-means, PCA and projective clustering. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2013, New Orleans, Louisiana, USA, January 6-8, 2013*, pp. 1434–1453, 2013.

Fichtenberger, H., Gillé, M., Schmidt, M., Schwiegelshohn, C., and Sohler, C. Bico: Birch meets coresets for k-means clustering. In *European Symposium on Algorithms*, pp. 481–492. Springer, 2013.

Friggstad, Z., Khodamoradi, K., Rezapour, M., and Salavatipour, M. R. Approximation schemes for clustering with outliers. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 398–414. SIAM, 2018.

Goodfellow, I. J., McDaniel, P. D., and Papernot, N. Making machine learning robust against adversarial inputs. *Commun. ACM*, 61(7):56–66, 2018.

Gupta, S. *Approximation algorithms for clustering and facility location problems*. PhD thesis, University of Illinois at Urbana-Champaign, 2018.

Gupta, S., Kumar, R., Lu, K., Moseley, B., and Vassilvitskii, S. Local search methods for k-means with outliers. *Proceedings of the VLDB Endowment*, 10(7):757–768, 2017.

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

Hawkins, D. M. The feasible solution algorithm for least trimmed squares regression. *Computational Statistics and Data Analysis*, 17, 1994. doi: 10.1016/0167-9473(92)00070-8.

Huang, L., Jiang, S., Li, J., and Wu, X. Epsilon-coresets for clustering (with outliers) in doubling metrics. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 814–825. IEEE, 2018.

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

Karnin, Z. S. and Liberty, E. Discrepancy, coresets, and sketches in machine learning. *CoRR*, abs/1906.04845, 2019.

Klivans, A. R., Kothari, P. K., and Meka, R. Efficient algorithms for outlier-robust regression. In *Conference On Learning Theory, COLT 2018, Stockholm, Sweden, 6-9 July 2018*, pp. 1420–1430, 2018.

Krishnaswamy, R., Li, S., and Sandeep, S. Constant approximation for k-median and k-means with outliers via iterative rounding. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 646–659. ACM, 2018.

Langberg, M. and Schulman, L. J. Universal ε-approximators for integrals. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*, pp. 598–607. SIAM, 2010.

Li, Y., Long, P. M., and Srinivasan, A. Improved bounds on the sample complexity of learning. *Journal of Computer and System Sciences*, 62(3):516–527, 2001.

Lucic, M., Faulkner, M., Krause, A., and Feldman, D. Training gaussian mixture models at scale via coresets. *The Journal of Machine Learning Research*, 18(1):5885–5909, 2017.

Mettu, R. R. and Plaxton, C. G. Optimal time bounds for approximate clustering. *Machine Learning*, 56(1-3):35–60, 2004.

Mount, D. M., Netanyahu, N. S., Piatko, C. D., Silverman, R., and Wu, A. Y. On the least trimmed squares estimator. *Algorithmica*, 69(1):148–183, 2014. doi: 10.1007/s00453-012-9721-8.

Mount, D. M., Netanyahu, N. S., Piatko, C. D., Wu, A. Y., and Silverman, R. A practical approximation algorithm for the LTS estimator. *Computational Statistics and Data Analysis*, 99:148–170, 2016. doi: 10.1016/j.csda.2016.01.016.

Munteanu, A., Schwiegelshohn, C., Sohler, C., and Woodruff, D. On coresets for logistic regression. In *Advances in Neural Information Processing Systems*, pp. 6561–6570, 2018.

Ott, L., Pang, L., Ramos, F. T., and Chawla, S. On integrated clustering and outlier detection. In *Advances in neural information processing systems*, pp. 1359–1367, 2014.

Phillips, J. M. Coresets and sketches. *Computing Research Repository*, 2016.

Rousseeuw, P. and van Driessen, K. Computing LTS regression for large data sets. *Data Min. Knowl. Discov.*, 12(1):29–45, 2006. doi: 10.1007/s10618-005-0024-4.

Rousseeuw, P. J. Least median of squares regression. *Journal of the American Statistical Association*, 79, 12 1984. doi: 10.1080/01621459.1984.10477105.

Shen, Y. and Sanghavi, S. Iterative least trimmed squares for mixed linear regression. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pp. 6076–6086, 2019.