arXiv:1906.08484v1 [cs.DS] 20 Jun 2019

Coresets for Clustering with Fairness Constraints

Lingxiao Huang* Shaofeng H.-C. Jiang† Nisheeth K. Vishnoi‡

June 21, 2019

Abstract

In a recent work, [19] studied the following "fair" variants of classical clustering problems such as k-means and k-median: given a set of n data points in $\mathbb{R}^d$ and a binary type associated to each data point, the goal is to cluster the points while ensuring that the proportion of each type in each cluster is roughly the same as its underlying proportion. Subsequent work has focused on either extending this setting to when each data point has multiple, non-disjoint sensitive types such as race and gender [6], or to address the problem that the clustering algorithms in the above work do not scale well [39, 7, 5]. The main contribution of this paper is an approach to clustering with fairness constraints that involve *multiple, non-disjoint* types, that is *also scalable*. Our approach is based on novel constructions of coresets: for the k-median objective, we construct an ε-coreset of size $O(\Gamma k^2 \varepsilon^{-d})$ where Γ is the number of distinct collections of groups that a point may belong to, and for the k-means objective, we show how to construct an ε-coreset of size $O(\Gamma k^3 \varepsilon^{-d-1})$. The former result is the first known coreset construction for the fair clustering problem with the k-median objective, and the latter result removes the dependence on the size of the full dataset as in [39] and generalizes it to multiple, non-disjoint types. Plugging our coresets into existing algorithms for fair clustering such as [5] results in the fastest algorithms for several cases. Empirically, we assess our approach over the **Adult** and **Bank** dataset, and show that the coreset sizes are much smaller than the full dataset; applying coresets indeed accelerates the running time of computing the fair clustering objective while ensuring that the resulting objective difference is small.

*EPFL, Switzerland. Email: huanglingxiao1990@126.com
†Weizmann Institute of Science, Israel. Email: shaofeng.jiang@weizmann.ac.il
‡Yale University, USA. Email: nisheeth.vishnoi@yale.edu

Contents

1 Introduction

Clustering algorithms are widely used in automated decision-making tasks, e.g., unsupervised learning [40], feature engineering [30, 25], and recommendation systems [9, 37, 20]. With the increasing applications of clustering algorithms in human-centric contexts, there is a growing concern that, if left unchecked, they can lead to discriminatory outcomes for protected groups. For instance, the proportion of a minority group assigned to some cluster can be far from its underlying proportion, even if clustering algorithms do not take the sensitive attribute into its decision making [19]. Such an outcome may, in turn, lead to unfair treatment of minority groups, e.g., women may receive proportionally fewer job recommendations with high salary [21, 36] due to their underrepresentation in the cluster of high salary recommendations.

To address this issue, Chierichetti et al. [19] recently proposed the fair clustering problem that requires the clustering assignment to be *balanced* with respect to a binary sensitive attribute. Given a set X of n data points in $\mathbb{R}^d$ and a binary type associated to each data point, the goal is to cluster the points such that the proportion of each type in each cluster is roughly the same as its underlying proportion, while ensuring that the clustering objective is minimized. Subsequent work has focused on either extending this setting to when each data point has multiple, non-disjoint sensitive types [6] (Definition 2.3), or to address the problem that the clustering algorithms do not scale well [19, 38, 39, 7, 5].

Due to the scale at which one is required to clustering, several existing fair clustering algorithms have to take samples instead of using the full dataset, since their running time is at least quadratic in the input size [19, 38, 7, 6]. Very recently, Backurs et al. [5] propose a nearly linear approximation algorithm for fair k-median, but it only works for a binary type. It is still unknown whether there exists a scalable approximation algorithm for multiple sensitive types [5]. To improve the running time of fair clustering algorithms, a powerful technique called ε-coreset was introduced. Roughly, a coreset for fair clustering is a small weighted point set, such that for any k-subset and any fairness constraint, the fair clustering objective computed over the coreset is approximately the same as that computed from the full dataset (Definition 2.1). Thus, a coreset can be used as a proxy for the full dataset – one can apply any fair clustering algorithm on the coreset, achieve a good approximate solution on the full dataset, and hope to speed up the algorithm. As mentioned in [5], using coresets can indeed accelerate the computation time and save the storage space for fair clustering problems. Another benefit is that one may want to compare the clustering performance under different fairness constraints, and hence it may be more efficient to repeatedly use coresets. Currently, the only known result for coresets for fair clustering is by Schmidt et al. [39], who constructed an ε-coreset for fair k-means clustering. However, their coreset size includes a $\log n$ factor and only restricts to a single sensitive type. Moreover, there is no known coreset construction for other commonly-used clusterings, e.g., fair k-median.[1]

Our contributions. The main contribution of this paper is the efficient construction of coresets for clustering with fairness constraints that involve multiple, non-disjoint types. Technically, we show the existence of ε-coresets of size independent on n for both fair k-

[1]For another clustering called k-center, Har-Peled's coreset result [26] can be easily extended to the fair k-center clustering problem by applying Theorem 4.2.

Table 1: Summary of coreset results. $T_1(n)$ and $T_2(n)$ denote the running time of an $O(1)$-approximate algorithm for k-median/means, respectively.

	k-Median size	k-Median construction time	k-Means size	k-Means construction time
[39]			$O(\Gamma k\varepsilon^{-d-2}\log n)$	$\tilde{O}(k\varepsilon^{-d-2}n\log n + T_2(n))$
This	$O(\Gamma k^2\varepsilon^{-d})$	$O(k\varepsilon^{-d+1}n + T_1(n))$	$O(\Gamma k^3\varepsilon^{-d-1})$	$O(k\varepsilon^{-d+1}n + T_2(n))$

median and fair k-means, summarized in Table 1. Let Γ denote the number of distinct collections of groups that a point may belong to.

- Our coreset for fair k-median is of size $O(\Gamma k^2\varepsilon^{-d})$ (Theorem 4.1), which is the first known coreset to the best of our knowledge.
- For fair k-means, our coreset is of size $O(\Gamma k^3\varepsilon^{-d-1})$ (Theorem 5.1), which improves the result of [39] by an $\Theta(\frac{\log n}{\varepsilon k^2})$ factor and generalizes it to multiple, non-disjoint types.
- As mentioned in [5], applying coresets can accelerate the running time of fair clustering algorithms, while suffering only an additional $(1+\varepsilon)$ factor in the approxiation ratio. Setting $\varepsilon = \Omega(1)$ and plugging our coresets into existing algorithms [39, 6, 5], we directly achieve scalable fair clustering algorithms, summarized in Table 2.

We present novel technical ideas to deal with fairness constraints for coresets.

- Our first technical contribution is a reduction to the case $\Gamma = 1$ (Theorem 4.2) which greatly simplifies the problem. Our reduction not only works for our specific construction, but also for all coreset constructions in general.
- Furthermore, to deal with the $\Gamma = 1$ case, we provide several interesting geometric observations for the arrangement of the optimal fair k-median/means clustering (Lemma 4.1), which may be of independent interest.

We implement our algorithm and conduct experiments on **Adult** and **Bank** datasets, that consist of ~40000 records each.

- A vanilla implementation results in a coreset with size depend on ε^{-d}. Our implementation is inspired by our theoretical results and produces coresets whose size is much smaller in practice. This improved implementation is still within the framework of our analysis, and the same worst case theoretical bound still holds.
- To validate the performance of our implementation, we experiment with varying ε for both fair k-median and k-means. As expected, the empirical error is well under the theoretical guarantee ε, and the size does not suffer from the ε^{-d} factor. Specifically, for fair k-median, we achieve 10% empirical error using only about 500 points for both data sets, and we achieve similar error using 2000 points (which is only 5% of the original data set) for the k-means case. In addition, our coreset for fair k-means is comparable to that of [39] in both the size and the empirical error.
- The small size of the coreset translates to more than 200x speedup (with error ~10%) in the running time of computing the fair clustering objective when the fair constraint F is given. We also apply our coreset on the scalable fair clustering algorithm [5], and drastically improve the running time of the algorithm by approximately 30 times.

Table 2: Summary of fair clustering algorithms. Δ denotes the maximum number of groups that a point may belong to, and "multi" means the algorithm can handle multiple non-disjoint types.

	k-Median			k-Means		
	multi	approx. ratio	time	multi	approx. ratio	time
[19]		O(1)	$\Omega(n^2)$			
[39]					$1+\varepsilon$	$n^{O(k/\varepsilon)}$
[5]		$\tilde{O}(d\log n)$	$O(dn\log n + T_1(n))$			
[7]		$(3.488, 1)$	$\Omega(n^2)$		$(4.675, 1)$	$\Omega(n^2)$
[6]	✓	$(O(1), 4\Delta+4)$	$\Omega(n^2)$	✓	$(O(1), 4\Delta+4)$	$\Omega(n^2)$
This		$\tilde{O}(d\log n)$	$O(dlk^2\log(lk) + T_1(lk^2))$		$O(1)$	$(lk)^{O(k/\varepsilon)}$
This	✓	$(O(1), 4\Delta+4)$	$\Omega(l^{2\Delta}k^4)$	✓	$(O(1), 4\Delta+4)$	$\Omega(l^{2\Delta}k^6)$

1.1 Other related works

There are increasingly more works on fair clustering algorithms. Chierichetti et al. [19] introduced the fair clustering problem for a binary type and obtained approximation algorithms for fair k-median/center. Backurs et al. [5] improved the running time to nearly linear for fair k-median, but the approximation ratio is $\tilde{O}(d\log n)$. Rösner and Schmidt [38] designed a 14-approximate algorithm for fair k-center, and the ratio is improved to 5 by [7]. For fair k-means, Schmidt et al. [39] introduced the notion of fair coresets, and presented an efficient streaming algorithm. More generally, Bercea et al. [7] proposed a bi-criteria approximation for fair k-median/means/center/supplier/facility location. Very recently, Bera et al. [6] presented a bi-criteria approximation algorithm for fair (k, z)-clustering problem (Definition 2.3) with arbitrary group structures (potentially overlapping), and Anagnostopoulos et al. [4] improved their results by proposing the first constant-factor approximation algorithm. It is still open to design a near linear time $O(1)$-approximate algorithm for the fair (k, z)-clustering problem.

There are other fair variants of clustering problems. Ahmadian et al. [3] studied a variant of the fair k-center problem in which the number of each type in each cluster has an upper bound, and proposed a bi-criteria approximation algorithm. Chen et al. [18] studied the fair clustering problem in which any n/k points are entitled to form their own cluster if there is another center closer in distance for all of them. Kleindessner et al. [32] investigate the fair k-center problem in which each center has a type, and the selection of the k-subset is restricted to include a fixed amount of centers belonging to each type. In another paper [33], they developed fair variants of spectral clusterings (a heuristic k-means clustering framework) by incorporating the proportional fairness constraints proposed by [19].

The notion of coreset was first proposed by Agarwal et al. [1]. There has been a large body of work for unconstrained clustering problems in Euclidean spaces [2, 26, 17, 27, 34, 22, 23, 8]). Apart from these, for the general (k, z)-clustering problem, Feldman and Langberg [22] presented an ε-coreset of size $\tilde{O}(dk\varepsilon^{-2z})$ in $\tilde{O}(nk)$ time. Huang et al. [28] showed an ε-coreset of size $\tilde{O}(\mathrm{ddim}(X)\cdot k^3\varepsilon^{-2z})$, where $\mathrm{ddim}(X)$ is doubling dimension that measures the intrinsic dimensionality of a space. For the special case of k-means, Braverman et al. [8] improved the size to $\tilde{O}(k\varepsilon^{-2}\cdot\min\{k/\varepsilon, d\})$ by a dimension reduction approach. Works such as [22] use importance sampling technique which avoid the size factor ε^{-d}, but it is unknown if such approaches can be used in fair clustering.

2 Problem definition

Consider a set $X \subseteq \mathbb{R}^d$ of n data points, an integer k (number of clusters), and l groups $P_1, \ldots, P_l \subseteq X$. An *assignment constraint*, which was proposed by Schmidt et al. [39], is a $k \times l$ integer matrix F. A clustering $\mathcal{C} = \{C_1, \ldots, C_k\}$, which is a k-partitioning of X, is said to satisfy assignment constraint F if

$$|C_i \cap P_j| = F_{ij}, \ \forall i \in [k], j \in [l].$$

For a k-subset $C = \{c_1, \ldots, c_k\} \subseteq X$ (the center set) and $z \in \mathbb{R}_{>0}$, we define $\mathcal{K}_z(X, F, C)$ as the minimum value of $\sum_{i\in[k]} \sum_{x\in C_i} d^z(x, c_i)$ among all clustering $\mathcal{C} = \{C_1, \ldots, C_k\}$ that satisfies F, which we call the optimal fair (k, z)-clustering value. If there is no clustering satisfying F, $\mathcal{K}_z(X, F, C)$ is set to be infinity. The following is our notion of coresets for fair (k, z)-clustering. This generalizes the notion introduced in [39] which only considers a partitioned group structure.

Definition 2.1 (**Coreset for fair clustering).** *Given a set $X \subseteq \mathbb{R}^d$ of n points and l groups $P_1, \ldots, P_l \subseteq X$, a weighted point set $S \subseteq \mathbb{R}^d$ with weight function $w : S \to \mathbb{R}_{>0}$ is an ε-coreset for the fair (k, z)-clustering problem, if for each k-subset $C \subseteq \mathbb{R}^d$ and each assignment constraint $F \in \mathbb{Z}_{\geq 0}^{k\times l}$, it holds that $\mathcal{K}_z(S, F, C) \in (1 \pm \varepsilon) \cdot \mathcal{K}_z(X, F, C)$.*

Since points in S might receive fractional weights, we change the definition of $\mathcal{K}_z$ a little, so that in evaluating $\mathcal{K}_z(S, F, C)$, a point $x \in S$ may be partially assigned to more than one cluster and the total amount of assignments of x equals $w(x)$.

The currently most general notion of fairness in clustering was proposed by [6], which enforces both upper bounds and lower bounds of any group's proportion in a cluster.

Definition 2.2 (**(α, β)-proportionally-fair).** *A clustering $\mathcal{C} = (C_1, \ldots, C_k)$ is (α, β)-proportionally-fair ($\alpha, \beta \in [0, 1]^l$), if for each cluster C_i and $j \in [l]$, it holds that $\alpha_j \leq \frac{|C_i\cap P_j|}{|C_i|} \leq \beta_j$.*

The above definition directly implies for each cluster C_i and any two groups $P_{j_1}, P_{j_2} \in [l]$, $\frac{\alpha_{j_1}}{\beta_{j_2}} \leq \frac{|C_i\cap P_{j_1}|}{|C_i\cap P_{j_2}|} \leq \frac{\beta_{j_1}}{\alpha_{j_2}}$. In other words, the fraction of points belonging to groups P_{j_1}, P_{j_2} in each cluster is bounded from both sides. Indeed, similar fairness constraints have been investigated by works on other fundamental algorithmic problems such as data summarization [13], ranking [15, 41], elections [11], personalization [16, 12], classification [10], and online advertising [14]. Naturally, Bera et al. [6] also defined the fair clustering problem with respect to (α, β)-proportionally-fairness as follows.

Definition 2.3 (**(α, β)-proportionally-fair (k, z)-clustering).** *Given a set $X \subseteq \mathbb{R}^d$ of n points, l groups $P_1, \ldots, P_l \subseteq X$, and two vectors $\alpha, \beta \in [0, 1]^l$, the objective of (α, β)-proportionally-fair (k, z)-clustering is to find a k-subset $C = \{c_1, \ldots, c_k\} \in \mathbb{R}^d$ and (α, β)-proportionally-fair clustering $\mathcal{C} = \{C_1, \ldots, C_k\}$, such that the objective function $\sum_{i\in[k]} \sum_{x\in C_i} d^z(x, c_i)$ is minimized.*

Our notion of coresets is very general, and we relate our notion of coresets to the (α, β)-proportionally-fair clustering problem, via the following observation, which is similar to Proposition 5 in [39].

Proposition 2.1. *Given a k-subset C, the assignment restriction required by (α, β)-proportionally-fairness can be modeled as a collection of assignment constraints.*

As a result, if a weighted set S is an ε-coreset satisfying Definition 2.1, then for any $\alpha, \beta \in [0,1]^l$, the (α, β)-proportionally-fair (k, z)-clustering value computed from S must be a $(1\pm\varepsilon)$-approximation of that computed from X.

Remark 2.1. *Definition 2.2 enforces fairness by looking at the proportion of a group in each cluster. We can also consider another type of constraints over the number of group points in each cluster, defined as follows.*

Definition 2.4 ((α, β)-fair)**.** *We call a clustering $\mathcal{C} = \{C_1, \ldots, C_k\}$ (α, β)-fair ($\alpha, \beta \in \mathbb{Z}_{\geq 0}^{k\times l}$), if for each cluster C_i and each $j \in [l]$, we have $\alpha_{ij} \leq |C_i \cap P_j| \leq \beta_{ij}$.*

We can similarly define the (α, β)-fair (k, z)-clustering problem with respect to the above definition as in Definition 2.3, and Proposition 2.1 still holds in this case. Hence, an ε-coreset for fair (k, z)-clustering also preserves the clustering objective of the (α, β)-fair (k, z)-clustering problem.

3 Technical overview

We introduce novel techniques to tackle the fairness constraints. Recall that Γ denotes the number of distinct collections of groups that a point may belong to. Our first technical contribution is a general reduction to the $\Gamma = 1$ case which works for any coreset construction algorithm (Theorem 4.2). The idea is to divide X into Γ parts with respect to the groups that a point belongs to, and construct a fair coreset with parameter $\Gamma = 1$ for each group. The observation is that the union of these coresets is a coreset for the original instance and Γ.

Our coreset construction for the case $\Gamma = 1$ is based on the framework of [27]. The main observation of [27] is that it suffices to deal with X that lies on a line. In their analysis, a crucially used property is that the clustering for any given center partitions X into k contiguous parts on the line. However, this property might not hold in fair clustering. Nonetheless, we manage to show a new structural lemma, that the optimal fair k-median/means clustering partitions X into $O(k)$ contiguous intervals. For fair k-median, the key geometric observation is that there always exists a center whose corresponding optimal fair k-median cluster forms a contiguous interval (Claim 4.1), and this combined with an induction implies the optimal fair clustering partitions X into $2k - 1$ intervals. For fair k-means, we show that each optimal fair cluster actually forms a single contiguous interval. Thanks to the new structural properties, plugging in a slightly different set of parameters in [27] yields fair coresets.

4 Coresets for fair k-median clustering

In this section, we construct coresets for fair k-median ($z = 1$). For each $x \in X$, denote $\mathcal{P}_x = \{i \in [l] : x \in P_i\}$ as the collection of groups that x belongs to. Let Γ denote the number of distinct $\mathcal{P}_x$'s. Let $T_z(n)$ denotes the running time of a constant approximation algorithm for the (k, z)-clustering problem. The main theorem is as follows.

Theorem 4.1 (Coreset for fair k-median). *There exists an algorithm that constructs an ε-coreset for the fair k-median problem of size $O(\Gamma k^2 \varepsilon^{-d})$, in $O(k\varepsilon^{-d+1} n + T_1(n))$ time.*

Note that Γ is usually small. For instance, if there is only a single sensitive attribute [39], then each $\mathcal{P}_x$ is a singleton and $\Gamma = l$. More generally, let Λ denote the maximum number of groups that any point belongs to, then $\Gamma \leq l^{\Lambda}$, but there is only $O(1)$ sensitive attributes for each point.

The main technical difficulty for the coreset construction is to deal with the assignment constraints. We make an important observation (Theorem 4.2), that one only needs to prove Theorem 4.1 for the case $l = 1$, and we thus focus on the case $l = 1$. This theorem is a generalization of Theorem 7 in [39], and the coreset of [39] actually extends to arbitrary group structure thanks to our theorem.

Theorem 4.2 (Reduction from l groups to a single group). *Suppose there exists an ε-coreset for the fair (k, z)-clustering problem of size t for the case $l = 1$, then there exists an ε-coreset for the fair (k, z)-clustering problem of size Γt.*

Proof. Consider the case that $\Gamma = 1$ in which all $\mathcal{P}_x$s are the same. Hence, this case can be degenerated to $l = 1$ and has an ε-coreset of size t by assumption. Divide the point set X into $X^{(1)}, \ldots, X^{(\Gamma)}$ by $\mathcal{P}_x$, i.e., for each $i \in [\Gamma]$, all collections $\mathcal{P}_x$ ($x \in X^{(i)}$) are the same, denoted by $\mathcal{P}_i$. For each $i \in [\Gamma]$, suppose $S^{(i)}$ is an ε-coreset for the fair (k, z)-clustering problem of $X^{(i)}$ where each point in $S^{(i)}$ belongs to all groups in $\mathcal{P}_i$. Let $S := \bigcup_{i \in [l]} S^{(i)}$. It is sufficient to prove S is an ε-coreset for the fair (k, z)-clustering problem of X.

Given a k-subset $C \subseteq \mathbb{R}^d$ and an assignment constraint F, let $C_1^\star, \ldots, C_k^\star$ be the optimal fair clustering of the instance (X, F, C). Then for each collection $X^{(i)}$ ($i \in [\Gamma]$), we construct an assignment constraint $F^{(i)} \in \mathcal{Z}^{k \times l}$ as follows: for each $j_1 \in [k]$ and $j_2 \in [l]$, let $F^{(i)}_{j_1, j_2} = 0$ if $j_2 \notin \mathcal{P}_i$ and $\left| C^\star_{j_1} \cap X^{(i)} \right|$ if $j_2 \in \mathcal{P}_i$, i.e., $F^{(i)}_{j_1, j_2}$ is the number of points within $X^{(i)}$ that belong to $C_{j_1} \cap P_{j_2}$. By definition, we have that for each $j_1 \in [k]$ and $j_2 \in [l]$,

$$F_{j_1, j_2} = \sum_{i \in [\Gamma]} F^{(i)}_{j_1, j_2}. \tag{1}$$

Then

$$\begin{aligned}
\mathcal{K}_z(X, F, C) =& \sum_{i \in [l]} \mathcal{K}_z(X^{(i)}, F^{(i)}, C) && \text{(Defns. of } \mathcal{K}_z \text{ and } F^{(i)}) \\
\geq& (1 - \varepsilon) \cdot \sum_{i \in [l]} \mathcal{K}_z(S^{(i)}, F^{(i)}, C) && \text{(Defn. of } S^{(i)}) \\
\geq& (1 - \varepsilon) \cdot \mathcal{K}_z(S, F, C) && \text{(Optimality and Eq. (1)).}
\end{aligned}$$

Similarly, we can prove that $\mathcal{K}_z(S, F, C) \geq (1 - \varepsilon)\mathcal{K}_z(X, F, C)$. It completes the proof. $\square$

Our coreset construction for both fair k-median and k-means are similar to that in [27], except we use a different set of parameters. In a high level, the algorithm reduces general instances to instances where data lie on a line, and it only remains to give a coreset for the line case.

4.1 The line case

Since $l = 1$, we describe F as an integer vector in $\mathbb{Z}_{\geq 0}^k$. For a weighted point set S with weight $w : S \to \mathbb{R}_{\geq 0}$, we define the *mean* of S by $\overline{S} := \frac{1}{|S|}\sum_{p\in S} w(p)\cdot p$ and the *error* of S by $\Delta(S) := \sum_{p\in S} w(p)\cdot d(p,\overline{S})$.

Construction. Our construction is the same to [27], except that we need a different set of parameters. Denote OPT as the optimal value of the unconstrained k-median clustering. We first compute an $O(1)$-approximation to OPT such that we can set a threshold ξ satisfying that $\xi = \Theta(\varepsilon\cdot\mathsf{OPT}) \leq \frac{\varepsilon\cdot\mathsf{OPT}}{30k}$. We consider the points from left to right (on the line) and group them into batches in a greedy way: each batch B is a maximal point set satisfying that $\Delta(B) \leq \xi$. Let $\mathcal{B}(X)$ denote the collection of all batches. The coreset S is defined by $S := \bigcup_{B\in\mathcal{B}(X)} \overline{B}$ where each point $\overline{B}$ has weight $|B|$.

Analysis. In [27], it was shown that S is an $\varepsilon/3$-coreset for the unconstrained k-median clustering problem. In their analysis, it is crucially used that the optimal clustering partitions X into k contiguous intervals. Unfortunately, the nice "contiguous" property does not hold in our case because of the assignment constraint $F \in \mathbb{R}^k$. To resolve this issue, we prove a new structural property (Lemma 4.1) that the optimal fair k-median clustering actually partitions X into only $O(k)$ contiguous intervals.

Lemma 4.1 (**Fair k-median clustering consists of $2k-1$ contiguous intervals).** *Suppose $X := \{x_1,\dots,x_n\} \subset \mathbb{R}^d$ lies on the real line where $x_1 \leq \ldots \leq x_n$. For any k-subset $C = (c_1,\dots,c_k) \in \mathbb{R}^d$ and any fairness constraints $F \in \mathbb{Z}_{\geq 0}^k$, there exists an optimal fair k-median clustering that partitions X into at most $2k-1$ contiguous intervals.*

Proof. We prove by induction on k. The induction hypothesis is that, for any $k \geq 1$, Lemma 4.1 holds for any data set X, any k-subset $C \subseteq \mathbb{R}^d$ and any assignment constraint $F \in \mathbb{Z}_{\geq 0}^k$. The base case $k = 1$ holds trivially since all points in X must be assigned to c_1.

Assume the lemma holds for $k-1$ ($k \geq 2$) and we will prove the inductive step k. Let $C_1^\star,\dots,C_k^\star$ be the optimal fair k-median clustering w.r.t. C and F, where $C_i^\star \subseteq X$ is the subset assigned to center c_i. We present the structural property in Claim 4.1, whose proof is given later.

Claim 4.1. *There exists $i \in [k]$ such that $C_i^\star$ consists of exactly one contiguous interval.*

We continue the proof of the inductive step by constructing a reduced instance (X', F', C') where a) $C' := C \setminus \{c_{i_0}\}$; b) $X' = X \setminus C_{i_0}^\star$; c) F' is formed by removing the i_0-th coordinate of F. Applying the hypothesis on (X', F', C'), we know the optimal fair $(k-1)$-median clustering consists of at most $2k-3$ contiguous intervals. Combining with $C_{i_0}^\star$ which has exactly one contiguous interval would increase the number of intervals by at most 2. Thus, we conclude that the optimal fair k-median clustering for (X, F, C) has at most $2k - 1$ contiguous intervals. This finishes the inductive step.

Finally, we complete the proof of Claim 4.1. We first prove the following fact for preparation.

Fact 4.1. *Suppose $p, q \in \mathbb{R}^d$. Define $f : \mathbb{R} \to \mathbb{R}$ as $f(x) := d(x, p) - d(x, q)$ (here we abuse the notation by treating x as a point in the x-axis of $\mathbb{R}^d$). Then f is either ID or DI.*[2]

Proof. Let h_p and h_q be the distance from p and q to the x-axis respectively, and let u_p and u_q be the corresponding x-coordinate of p and q. We have

$$f(x) = \sqrt{(x - u_p)^2 + h_p^2} - \sqrt{(x - u_q)^2 + h_q^2}.$$

Then we can regard p, q as two points in $\mathbb{R}^2$ by letting $p = (u_p, h_p)$ and $q = (u_q, h_q)$. Also we have

$$f'(x) = \frac{x - u_p}{\sqrt{(x - u_p)^2 + h_p^2}} - \frac{x - u_q}{\sqrt{(x - u_q)^2 + h_q^2}} = \frac{x - u_p}{d(x, p)} - \frac{x - u_q}{d(x, q)}.$$

W.l.o.g. assume that $u_p \leq u_q$. Next, we rewrite $f'(x)$ with respect to $\cos(\angle pxu_p)$ and $\cos(\angle qxu_q)$.

1. If $x \leq u_p$. Then $f'(x) = \frac{d(x,u_q)}{d(x,q)} - \frac{d(x,u_p)}{d(x,p)} = \cos(\angle qxu_q) - \cos(\angle pxu_p)$.
2. If $u_p < x \leq u_q$. Then $f'(x) = \frac{d(x,u_p)}{d(x,p)} + \frac{d(x,u_q)}{d(x,q)} = \cos(\angle pxu_p) + \cos(\angle qxu_q)$.
3. If $x > u_q$. Then $f'(x) = \frac{d(x,u_p)}{d(x,p)} - \frac{d(x,u_q)}{d(x,q)} = \cos(\angle pxu_p) - \cos(\angle qxu_q)$.

Denote the intersecting point of line pq and the x-axis to be y. Specificially, if $h_p = h_q$, we denote $y = -\infty$. Note that $f'(x) = 0$ if and only if $x = y$. Now we analyze $f'(x)$ in two cases (whether or not $h_p \leq h_q$).

- Case i): $h_p \leq h_q$ which implies that $y < u_p$. When x goes from $-\infty$ to u_p, first $f'(x) \leq 0$ and then $f'(x) \geq 0$. When $x > u_p$, $f'(x) \geq 0$.
- Case ii): $h_p > h_q$ which implies that $y > u_q$. When $x \leq u_q$, $f'(x) \geq 0$. When x goes from u_q to $+\infty$, first $f'(x) \geq 0$ and then $f'(x) \leq 0$.

Therefore, $f(x)$ is either DI or ID. □

Proof of Claim 4.1. Suppose for the contrary that for any $i \in [k]$, $C_i^\star$ consists of at least two contiguous intervals. Pick any i and suppose $S_L, S_R \subseteq C_i^\star$ are two contiguous intervals such that S_L lies on the left of S_R. Let y_L denote the rightmost point of S_L and y_R denote the leftmost point of S_R. Since S_L and S_R are two distinct contiguous intervals, there exists some point $y \in X$ between y_L and y_R such that $y \in C_j^\star$ for some $j \neq i$. Define $g : \mathbb{R} \to \mathbb{R}$ as $g(x) := d(x, c_j) - d(x, c_i)$. By Fact 4.1, we know that $g(x)$ is either ID or DI.

[2] ID means that the function f first (non-strictly) increases and then (non-strictly) decreases. DI means the other way round.

If g is ID, we swap the assignment of y and $y_{\min} := \arg\min_{x\in\{y_L,y_R\}} g(x)$ in the optimal fair k-median clustering. Since g is ID, for any interval P with endpoints p and q, $\min_{x\in P} g(x) = \min_{x\in\{p,q\}} g(x)$. This fact together with $y_L \le y \le y_R$ implies that $g(y_{\min}) - g(y) \le 0$. Hence, the change of the objective is

$$d(y,c_i) - d(y,c_j) - d(y_{\min},c_i) + d(y_{\min},c_j) = g(y_{\min}) - g(y) \le 0.$$

This contradicts with the optimality of $C^\star$ and hence g has to be DI.

Next, we show that there is no $y' \in C_j^\star$ such that $y' < y_L$ or $y' > y_R$. We prove by contradiction and only focus on the case of $y' < y_L$, since the case of $z > y_R$ can be proved similarly by symmetry. We swap the assignment of y_L and $y_{\max} := \arg\max_{x\in\{y,y'\}} g(x)$ in the optimal fair k-median clustering. The change of the objective is

$$\begin{aligned}&d(y_L,c_j) - d(y_L,c_i) - d(y_{\max},c_j) + d(y_{\max},c_i)\\ =&g(y_L) - g(y_{\max}) \le 0,\end{aligned}$$

where the last inequality is by the fact that g is DI. This contradicts the optimality of $C^\star$. Hence, we conclude such y' does not exist.

Therefore, $\forall x \in C_j^\star$, $y_L < x < y_R$. By assumption, $C_j^\star$ consists of at least two contiguous intervals within (y_L, y_R). However, we can actually do exactly the same argument for $C_j^\star$ as in the i case, and eventually we would find a j' such that $C_{j'}^\star$ lies inside a strict smaller interval (y'_L, y'_R) of X, where $y_L < y'_L < y'_R < y_R$. Since n is finite, we cannot do this procedure infinitely, which is a contradiction. This finishes the proof of Claim 4.1. ◻

Theorem 4.3shows the correctness of our coreset for the line case. The proof idea is similar to that of Lemma 2.8 in [27], in which we can show that the assignment difference of a contiguous interval between X and S is upper bounded by $O(\frac{\varepsilon\cdot\mathsf{OPT}}{k})$.

Theorem 4.3 (Coreset for fair k-median when X lies on a line). *Let X be a set of n points lying on a line in $\mathbb{R}^d$. Let $S = \bigcup_{B\in\mathcal{B}(X)} \overline{B}$ be the coreset in which each point $\overline{B}$ has weight $|B|$. Then S is an $\varepsilon/3$-coreset for fair k-median clustering of X.*

4.2 Proof of Theorem 4.3

Proof. The proof idea is similar to that of Lemma 2.8 in [27]. We first rotate space such that the line is on the x-axis and assume that $x_1 \le x_2 \le \ldots \le x_n$. Given an assignment constraint $F \in \mathbb{R}^k$ and a k-subset $C = \{c_1,\ldots,c_k\} \subseteq \mathbb{R}^d$, let c'_i denote the projection of point c_i to the real line and assume that $c'_1 \le c'_2 \le \ldots \le c'_k$. Our goal is to prove that

$$|\mathcal{K}_1(S,F,C) - \mathcal{K}_1(X,F,C)| \le \frac{\varepsilon}{3}\cdot\mathcal{K}_1(X,F,C).$$

By the construction of S, we build up a mapping $\pi : X \to S$ by letting $\pi(x) = \overline{B}$ for any $x \in B$. For each $i \in [k]$, let C_i denote the collection of points assigned to c_i in the optimal fair k-median clustering of X. By Lemma 4.1, $C_1,\ldots,C_k$ partition the line into at most $2k-1$ intervals $\mathcal{I}_1,\ldots,\mathcal{I}_t$ ($t \le 2k-1$), such that all points of any interval $\mathcal{I}_i$ are assigned to the same center. Denote an assignment function $f : X \to C$ by $f(x) = c_i$ if $x \in C_i$. Let $\widehat{\mathcal{B}}$ denote the set of all batches B, which intersects with more than one intervals $\mathcal{I}_i$, or alternatively, the interval $\mathcal{I}(B)$ contains the projection of a center point of C to the x-axis. Clearly, $|\widehat{\mathcal{B}}| \le 2k-2+k = 3k-2$. For each batch $B \in \widehat{\mathcal{B}}$, we have

$$\sum_{x \in B} d(\pi(x), f(x)) - d(x, f(x)) \overset{\text{triangle ineq.}}{\le} \sum_{x \in B} |d(x, \pi(x))| = \sum_{x \in B} |d(x, \overline{B})| \overset{\text{Defn. of } B}{\le} \frac{\varepsilon \mathsf{OPT}}{30k}. \tag{2}$$

Note that $X \setminus \bigcup_{B \in \widehat{\mathcal{B}}} B$ can be partitioned into at most $3k-1$ contiguous intervals. Denote these intervals by $\mathcal{I}'_1, \ldots, \mathcal{I}'_{t'}$ ($t' \le 3k-1$). By definition, all points of each interval $\mathcal{I}'_i$ are assigned to the same center whose projection is outside $\mathcal{I}'_i$. Then by the proof of Lemma 2.8 in [27], we have that for each $\mathcal{I}'_i$,

$$\sum_{x \in \mathcal{I}'_i} d(\pi(x), f(x)) - d(x, f(x)) \le 2\xi = \frac{\varepsilon \mathsf{OPT}}{15k}. \tag{3}$$

Combining Inequalities (2) and (3), we have

$$\begin{aligned}
&\mathcal{K}_1(S,F,C) - \mathcal{K}_1(X,F,C) \le \sum_{x \in X} d(\pi(x), f(x)) - d(x, f(x)) \quad (\text{Defn. of } \mathcal{K}_1(S,F,C)) \\
=& \sum_{B \in \widehat{\mathcal{B}}} \sum_{x \in B} d(\pi(x), f(x)) - d(x, f(x)) \\
&+ \sum_{i \in [t]} \sum_{x \in \mathcal{I}'_i} d(\pi(x), f(x)) - d(x, f(x)) \qquad (4) \\
\le& (3k-2) \cdot \frac{\varepsilon \mathsf{OPT}}{30k} + (3k-1) \cdot \frac{\varepsilon \mathsf{OPT}}{15k} \quad (\text{Ineqs. (2) and (3)}) \\
\le& \frac{\varepsilon \mathsf{OPT}}{3} \le \frac{\varepsilon}{3} \cdot \mathcal{K}_1(X,F,C).
\end{aligned}$$

To prove the other direction, we can regard S as a collection of n unweighted points and consider the optimal fair k-median clustering of S. Again, the optimal fair k-median clustering of S partitions the x-axis into at most $2k-1$ contiguous intervals, and can be described by an assignment function $f' : S \to C$. Then we can build up a mapping $\pi' : S \to X$ as the inverse function of π. For each batch B, let S_B denote the collection of $|B|$ unweighted points located at $\overline{B}$. We have the following inequality that is similar to Inequality (2)

$$\sum_{x \in S_B} d(\pi'(x), f'(x)) - d(x, f'(x)) \le \frac{\varepsilon \mathsf{OPT}}{30k}.$$

Suppose a contiguous interval $\mathcal{I}$ consists of several batches and satisfies that all points of $\mathcal{I} \cap S$ are assigned to the same center by f' whose projection is outside $\mathcal{I}$. Then by the proof of Lemma 2.8 in [27], we have that

$$\sum_{B \in \mathcal{I}} \sum_{x \in S_B} d(\pi'(x), f'(x)) - d(x, f'(x)) \le 0.$$

Then by a similar argument as for Inequality (4), we can prove the other direction

$$\mathcal{K}_1(X,F,C) - \mathcal{K}_1(S,F,C) \le \frac{\varepsilon}{3} \cdot \mathcal{K}_1(X,F,C),$$

which completes the proof. $\square$

4.3 Extending to higher dimension

The extension is the same as that of [27]. For completeness, we describe the detailed procedure for coresets for fair k-median.

1. We start with computing an approximate k-subset $C^\star = \{c_1, \dots, c_k\} \subseteq \mathbb{R}^d$ such that $\mathsf{OPT} \le \mathcal{K}_1(X, C^\star) \le c \cdot \mathsf{OPT}$ for some constant $c > 1$.[3]

2. Then we partition the point set X into sets $X_1, \dots, X_k$ satisfying that X_i is the collection of points closest to c_i.

3. For each center c_i, we take a unit sphere centered at c_i and construct an $\frac{\varepsilon}{3c}$-net N_{c_i}[4] on this sphere. By Lemma 2.6 in [27], $|N_{c_i}| = O(\varepsilon^{-d+1})$ and may be computed in $O(\varepsilon^{-d+1})$ time. Then for every $p \in N_{c_i}$, we emit a ray from c_i to p. Overall, there are at most $O(k\varepsilon^{-d+1})$ lines.

4. For each $i \in [k]$, we project all points of X_i onto the closest line around c_i. Let $\pi : X \to \mathbb{R}^d$ denote the projection function. By the definition of $\frac{\varepsilon}{3c}$-net, we have that $\sum_{x \in X} d(x, \pi(x)) \le \varepsilon \cdot \mathsf{OPT}/3$ which indicates that the projection cost is negligible. Then for each line, we compute an $\varepsilon/3$-coreset of size $O(k\varepsilon^{-1})$ for fair k-median by Theorem 4.3. Let S denote the combination of coresets generated from all lines.

Proof of Theorem 4.1. Since there are at most $O(k\varepsilon^{-d+1})$ lines and the coreset on each line is of size at most $O(k\varepsilon^{-1})$ by Theorem 4.3, the total size of S is $O(k^2\varepsilon^{-d})$. For the correctness, by the optimality of OPT (which is *unconstrained* optimal), for any given assignment constraint $F \in \mathbb{R}^k$ and any k-subset $C \subseteq \mathbb{R}^d$, $\mathsf{OPT} \le \mathcal{K}_1(X, F, C)$. Combining this fact with Theorem 4.3, we have that S is an ε-coreset for fair k-median clustering, by the same argument as in Theorem 2.9 of [27]. For the running time, we need $T_1(n)$ time to compute $C^\star$ and APX and the remaining construction time is upper bounded by $O(k\varepsilon^{-d+1}n)$ – the projection process to lines. This completes the proof. $\square$

Remark 4.1. *In fact, it suffices to emit a set of rays such that the total cost of projecting points to the rays is at most $\frac{\varepsilon \cdot \mathsf{OPT}}{3}$. This observation is crucially used in our implementations (Section 6) to reduce the size of the coreset, particularly to avoid the construction of the $O(\varepsilon)$-net which is of $O(\varepsilon^{-d})$ size.*

5 Coresets for fair k-means clustering

In this section, we show how the construction of coresets for fair k-means. Similar to the fair k-median case, we apply the approach in [27]. The main theorem is as follows.

Theorem 5.1 (Coreset for fair k-means). *There exists an algorithm that constructs ε-coreset for the fair k-means problem of size $O(\Gamma k^3 \varepsilon^{-d-1})$, in $O(k^2\varepsilon^{-d+1}n + T_2(n, d, k))$ time.*

[3] For example, we can set $c = 10$ by [31].

[4] An ε-net Q means that for any point p in the unit sphere, there exists a point $q \in Q$ satisfying that $d(p, q) \le \varepsilon$.

Note that the above result improves the coreset size of [39] by a $O(\frac{\log n}{\varepsilon k^2})$ factor. Similar to the fair k-median case, it suffices to prove for the case $l = 1$. Recall that an assignment constraint for $l = 1$ can be described by a vector $F \in \mathbb{R}^k$. Denote OPT to be the optimal k-means value without any assignment constraint.

5.1 The line case

Similar to [27], we first consider the case that X is a point set on the real line. For a weighted point set S with weight $w : S \to \mathbb{R}_{\geq 0}$, we denote the *mean* of S by $\overline{S} := \frac{1}{|S|}\sum_{p\in S} w(p)\cdot p$, and the *error* of S by $\Delta(S) := \sum_{p\in S} w(p)\cdot d^2(p, \overline{S})$.

Construction. Same to [27], we consider the points from left to right and group them into batches in a greedy way: each batch B is a maximal point set satisfying that $\Delta(B) \leq \xi$ where $\xi = \frac{\varepsilon^2\mathsf{OPT}}{200k^2}$. Let $\mathcal{I}(B)$ denote the smallest closed segment containing all the points of a batch B. Let $\mathcal{B}(X)$ denote the collection of all batches. For each batch B, we construct a collection $\mathcal{J}(B)$ of two weighted points satisfying Lemma 5.1. The coreset is defined by $S = \bigcup_{B\in\mathcal{B}(X)} \mathcal{J}(B)$.

Lemma 5.1 (Lemmas 3.2 and 3.4 in [27]). *The number of batches is $O(k^2/\varepsilon^2)$. For each batch B, there exist two weighted points $q_1, q_2 \in \mathcal{I}(B)$ together with weight w_1, w_2 satisfying that*

- $w_1 + w_2 = |B|$.
- *Let $\mathcal{J}(B)$ denote the collection of two weighted points q_1 and q_2. Then we have $\overline{\mathcal{J}(B)} = \overline{B}$ and $\Delta(B) = \Delta(\mathcal{J}(B))$.*
- *Given any point $q \in \mathbb{R}^d$, we have*

$$\mathcal{K}_2(B, q) = \Delta(B) + |B|\cdot d^2(q, \overline{B}) = \mathcal{K}_2(\mathcal{J}(B), q).$$

Analysis. We argue that S is indeed an $\varepsilon/3$-coreset for the fair k-means clustering problem. By Theorem 3.5 in [27], S is an $\varepsilon/3$-coreset for k-means clustering of X. However, we need to handle additional assignment constraints. To address this, we introduce the following lemma showing that every optimal cluster satisfying the given assignment constraint is within a contiguous interval.

Lemma 5.2 (Clusters are contiguous for fair k-means). *Suppose $X = \{x_1, \ldots, x_n\}$ where $x_1 \leq x_2 \leq \ldots \leq x_n$. Given an assignment constraint $F \in \mathbb{R}^k$ and a k-subset $C = \{c_1, \ldots, c_k\} \subseteq \mathbb{R}^d$. Then letting $C_i := \left\{x_{1+\sum_{j<i} F_j}, \ldots, x_{\sum_{j\leq i} F_j}\right\}$ ($i \in [k]$), we have*

$$\mathcal{K}_2(X, F, C) = \sum_{i\in[k]}\sum_{x\in C_i} d^2(x, c_i).$$

Proof. Let c_i' denote the projection of point c_i to the real line and assume that $c_1' \leq c_2' \leq \ldots \leq c_k'$. We slightly abuse the notation by regarding point c_i' as a real value. We prove the lemma by contradiction. Let $C_1, \ldots, C_k$ be the optimal fair clustering. By contradiction we assume that there exists $i_1 < i_2$ and $j_1 < j_2$ such that $x_{j_1} \in C_{i_2}$ and $x_{j_2} \in C_{i_1}$. By the definitions of c'_{i_1} and c'_{i_2}, we have that

$$d(c'_{i_1}, x_{j_1}) + d(c'_{i_2}, x_{j_2}) \le d(c'_{i_1}, x_{j_2}) + d(c'_{i_2}, x_{j_1}), \tag{5}$$

and

$$\max\left\{d(c'_{i_1}, x_{j_1}), d(c'_{i_2}, x_{j_2})\right\} \le \max\left\{d(c'_{i_1}, x_{j_2}), d(c'_{i_2}, x_{j_1})\right\}. \tag{6}$$

Combining Inequalities (5) and (6), we argue that

$$d^2(c'_{i_1}, x_{j_1}) + d^2(c'_{i_2}, x_{j_2}) \le d^2(c'_{i_1}, x_{j_2}) + d^2(c'_{i_2}, x_{j_1}) \tag{7}$$

by proving the following claim.

Claim 5.1. *Suppose $a, b, c, d \ge 0$, $a + b \le c + d$ and $a, b, c \le d$. Then $a^2 + b^2 \le c^2 + d^2$.*

Proof. If $a + b \le d$, then we have $a^2 + b^2 \le (a + b)^2 \le d^2 \le c^2 + d^2$. So we assume that $a + b > d$. Let $e = a + b - d > 0$. Since $a + b \le c + d$, we have $e^2 \le c^2$. Hence, it suffices to prove that $a^2 + b^2 \le e^2 + d^2$. Note that

$$e^2 + d^2 = (a + b - d)^2 + d^2 = a^2 + b^2 + (d - a)(d - b) \ge a^2 + b^2,$$

which completes the proof. □

Now we come back to prove Lemma 5.1. We have the following inequality.

$$\begin{aligned}
& d^2(x_{j_1}, c_{i_1}) + d^2(x_{j_2}, c_{i_2}) \\
=& d^2(x_{j_1}, c'_{i_1}) + d^2(c'_{i_1}, c_{i_1}) + d^2(x_{j_2}, c'_{i_2}) + d^2(c'_{i_2}, c_{i_2}) && \text{(The Pythagorean theorem)} \\
\le& d^2(x_{j_1}, c'_{i_2}) + d^2(c'_{i_1}, c_{i_1}) + d^2(x_{j_2}, c'_{i_1}) + d^2(c'_{i_2}, c_{i_2}) && \text{(Ineq. (7))} \\
=& d^2(x_{j_1}, c_{i_2}) + d^2(x_{j_2}, c_{i_1}). && \text{(The Pythagorean theorem)}
\end{aligned}$$

It contradicts with the assumption that $x_{j_1} \in C_{i_2}$ and $x_{j_2} \in C_{i_1}$. Hence, we complete the proof. □

Now we are ready to give the following theorem.

Theorem 5.2 (Coreset for fair k-means when X lies on a line). *Let X be a set of n points lying on a line in $\mathbb{R}^d$. Let $S = \bigcup_{B \in \mathcal{B}(X)} \mathcal{J}(B)$ be the coreset constructed as in Lemma 5.1. Then S is an $\varepsilon/3$-coreset for fair k-means clustering of X.*

Proof. The proof is similar to that of Theorem 3.5 in [27]. The running time analysis is exactly the same. Hence, we only focus on the correctness analysis in the following. We first rotate space such that the line is on the x-axis and assume that $x_1 \le x_2 \le \ldots \le x_n$. Given an assignment constraint $F \in \mathbb{R}^k$ and a k-subset $C = \{c_1, \ldots, c_k\} \subseteq \mathbb{R}^d$, let c'_i denote the projection of point c_i to the real line and assume that $c'_1 \le c'_2 \le \ldots \le c'_k$. Our goal is to prove that

$$|\mathcal{K}_2(S, F, C) - \mathcal{K}_2(X, F, C)| \le \frac{\varepsilon}{3} \cdot \mathcal{K}_2(X, F, C).$$

By Lemma 5.2, we have that the optimal fair clustering of X should be $C_i := \left\{x_{1+\sum_{j<i} F_j}, \ldots, x_{\sum_{j\le i} F_j}\right\}$ for each $i \in [k]$. Hence, $\mathcal{I}(C_1), \ldots, \mathcal{I}(C_k)$ are disjoint intervals. Similarly, the optimal fair clustering of X should be to scan weighted points in S from left to right and cluster points of total weight F_i to c_i.[5] If a batch $B \in \mathcal{B}(X)$ lies completely within some interval $\mathcal{I}(C_i)$, then it does not contribute to the overall difference $|\mathcal{K}_2(S, F, C) - \mathcal{K}_2(X, F, C)|$ by Lemma 5.1.

Thus, the only problematic batches are those that contain an endpoint of $\mathcal{I}(C_1), \ldots, \mathcal{I}(C_k)$. There are at most $k-1$ such batches. Let B be one such batch and $\mathcal{J}(B) = \{q_1, q_2\}$ be constructed as in Lemma 5.1. For $i \in [k]$, let $V_i := \mathcal{I}(C_i) \cap B$. Let T denote the collection of the w_1 left side points within B and $T' = B \setminus T$. Note that w_1 may be fractional and hence T may include a fractional point. Denote

$$\eta := \sum_{i\in[k]} \sum_{x\in V_i\cap T} d^2(x, q_1) + \sum_{i\in[k]} \sum_{x\in V_i\cap T'} d^2(x, q_2).$$

We have that

$$\begin{aligned}
\eta =& \sum_{i\in[k]} \sum_{x\in V_i\cap T} \left(d(x, \overline{B}) - d(q_1, \overline{B})\right)^2 + \sum_{i\in[k]} \sum_{x\in V_i\cap T'} \left(d(x, \overline{B}) - d(q_2, \overline{B})\right)^2 \\
\le& \sum_{i\in[k]} \sum_{x\in V_i\cap T} \left(d^2(x, \overline{B}) + d^2(q_1, \overline{B})\right) + \sum_{i\in[k]} \sum_{x\in V_i\cap T'} \left(d^2(x, \overline{B}) + d^2(q_2, \overline{B})\right) \\
=& \Delta(B) + \Delta(\mathcal{J}(B)) = 2\Delta(B) \qquad \text{(Lemma 5.1)} \\
\le& \frac{\varepsilon^2 \mathsf{OPT}}{100k} \qquad \text{(Construction of } B\text{)}.
\end{aligned} \tag{8}$$

Then we can upper bound the contribution of B to the overall difference $|\mathcal{K}_2(S, F, C) - \mathcal{K}_2(X, F, C)|$

[5] Recall that a weighted point can be partially assigned to more than one cluster.

by

$$
\begin{aligned}
&\left|\sum_{i\in[k]}\sum_{x\in V_i\cap T}\left(d^2(x,c_i)-d^2(q_1,c_i)\right)+\sum_{i\in[k]}\sum_{x\in V_i\cap T'}\left(d^2(x,c_i)-d^2(q_2,c_i)\right)\right| \\
\le&\sum_{i\in[k]}\sum_{x\in V_i\cap T}\left|d^2(x,c_i)-d^2(q_1,c_i)\right|+\sum_{i\in[k]}\sum_{x\in V_i\cap T'}\left|d^2(x,c_i)-d^2(q_2,c_i)\right| \\
=&\sum_{i\in[k]}\sum_{x\in V_i\cap T} d(x,q_1)\left(d(x,c_i)+d(q_1,c_i)\right)+\sum_{i\in[k]}\sum_{x\in V_i\cap T'} d(x,q_2)\left(d(x,c_i)+d(q_2,c_i)\right) \\
\le&\sum_{i\in[k]}\sum_{x\in V_i\cap T} d(x,q_1)\left(2d(x,c_i)+d(x,q_1)\right)+\sum_{i\in[k]}\sum_{x\in V_i\cap T'} d(x,q_2)\left(2d(x,c_i)+d(x,q_2)\right) \\
=&\sum_{i\in[k]}\sum_{x\in V_i\cap T} d^2(x,q_1)+\sum_{i\in[k]}\sum_{x\in V_i\cap T'} d^2(x,q_2) \\
&+2\sum_{i\in[k]}\sum_{x\in V_i\cap T} d(x,q_1)d(x,c_i)+2\sum_{i\in[k]}\sum_{x\in V_i\cap T'} d(x,q_2)d(x,c_i) \qquad (9)\\
\le&\eta+2\sqrt{\eta}\sqrt{\sum_{i\in[k]}\sum_{x\in V_i} d^2(x,c_i)} \qquad \text{(Defn. of } \eta \text{ and Cauchy-Schwarz)} \\
\le&\frac{\varepsilon^2\mathsf{OPT}}{50k}+\frac{2\varepsilon}{7k}\sqrt{\mathsf{OPT}\cdot\mathcal{K}_2(X,F,C)} \qquad \text{(Ineq. (8))} \\
\le&\frac{\varepsilon^2\mathsf{OPT}}{100k}+\frac{2\varepsilon}{10k}\cdot\frac{\mathsf{OPT}+\sum_{i\in[k]}\sum_{x\in V_i} d^2(x,c_i)}{2} \\
\le&\frac{\varepsilon\mathsf{OPT}}{5k}+\frac{\varepsilon\sum_{i\in[k]}\sum_{x\in V_i} d^2(x,c_i)}{10k}.
\end{aligned}
$$

Since there are at most $k-1$ such batches, we conclude that the their total contribution to the error $|\mathcal{K}_2(S,F,C)-\mathcal{K}_2(X,F,C)|$ can be upper bounded by

$$
\frac{\varepsilon\mathsf{OPT}}{5}+\frac{\varepsilon\mathcal{K}_2(X,F,C)}{10k}\le\frac{\varepsilon}{3}\cdot\mathcal{K}_2(X,F,C).
$$

It completes the proof. □

5.2 Extending to higher dimension

The extension is almost the same to fair k-median, except that we apply Theorem 5.2 to construct the coreset on each line. Let S denote the combination of coresets generated from all lines.

Proof of Theorem 5.1. By the above construction, the coreset size is $O(k^3\varepsilon^{-d-1})$. For the correctness, Theorem 3.6 in [27] applies an important fact that for any k-subset $C\subseteq\mathbb{R}^d$,

$$
\mathcal{K}_2(X,C^\star)\le c\cdot\mathcal{K}_2(X,C).
$$

In our setting, we have a similar property. Note that for any given `assignment constraint` $F\in\mathbb{R}^k$ and any k-subset $C\subseteq\mathbb{R}^d$, we have

$$
\mathcal{K}_2(X,C^\star)\le c\cdot\mathcal{K}_2(X,F,C).
$$

Then combining this fact with Theorem 5.2, we have that S is an ε-coreset for the fair k-means clustering problem, by the same argument as that of Theorem 3.6 in [27]. $\square$

6 Empirical results

We implement our algorithm and evaluate its performance on real datasets. The implementation mostly follows our description of algorithms, but a vanilla implementation would bring in an ε^{-d} factor in the coreset size. To avoid this, as observed in Remark 4.1, we may actually emit any set of rays as long as the total projection cost is bounded, instead of ε^{-d} rays. We implement this idea by finding the smallest integer m and m lines, such that the minimum cost of projecting data onto m lines is within the error threshold. In our implementation for fair k-means, we adopt the widely used Lloyd's heuristic [35] to find the m lines, where the only change to Lloyd's heuristic is that, for each cluster, we need to find a *line* that minimizes the projection cost instead of a point, and we use SVD to efficiently find this line optimally. Unfortunately, the above approach does not work for the fair k-median, as the SVD does not give the optimal line. As a result, we still need to construct the ε-net, but we alternatively employ some heuristics to find the net adaptively w.r.t. the dataset.

Our evaluation is conducted on the **Adult** and **Bank** [19, 39] datasets, each consists of ~40000 records. For both datasets, we choose numerical features to form a vector in $\mathbb{R}^d$ for each record, where $d = 6$ for **Adult** and $d = 10$ for **Bank**. We use ℓ_2 to measure the distance of these vectors, and normalize each dimension to be within $[0, 1]$ so that features with large numerical range could not dominate the distance measure. We choose two sensitive types for each dataset: sex and marital for **Adult**; marital and default for **Bank**, which results in 9 groups in **Adult** and 6 in **Bank**. We pick $k = 3$ (i.e. number of clusters) throughout our experiment. We define the *empirical error* as $|\frac{\mathcal{K}_z(S,F,C)}{\mathcal{K}_z(X,F,C)} - 1|$ (which is the same measure as ε) for some F and C. To evaluate the empirical error, we draw 500 independent random samples of (F, C) and report the maximum empirical error among these samples. For each (F, C), the fair clustering objectives $\mathcal{K}_z(\cdot, F, C)$ may be formulated as integer linear programs (ILP). We use **CPLEX** [29] to solve the ILP's, and report the average running time[6] T_X and T_S for evaluating the objective on dataset X and coreset S respectively. We also showcase the speed-up to a recently published, practically efficient, $O(\log n)$-approximate algorithm for fair k-median [5] that works for a binary type. We refer to this algorithm as **FairTree**. We slightly modify the implementation of **FairTree** to make it efficiently work on top of our coreset.

A recent work [39] presented a coreset construction for the fair k-means, whose implementation is based on the BICO library which is a high-performance coreset-based library for computing k-means clustering [24]. We evaluate the performance of our coreset for fair k-means against the BICO implementation. As a remark for the BICO implementation, BICO does not support the parameter ε, but a hinted size of the resulted coreset. Hence, we start with evaluating our coreset, and set the hinted size for BICO as the size of our coreset. For the k-median case, we are not aware of any other previous coreset implementation and hence no baseline is compared with.

[6]The experiments are conducted on a 4-Core desktop CPU with 64 GB RAM.

Table 3: performance of ε-coresets for fair k-median w.r.t. varying ε.

ε	Adult				Bank			
	emp. err.	size	T_S (ms)	T_X (ms)	emp. err.	size	T_S (ms)	T_X (ms)
10%	0.84%	15867	1484	7284	1.82%	1367	63	5404
15%	1.80%	8585	622	-	5.44%	515	24	-
20%	3.17%	4773	293	-	9.66%	231	13	-
25%	4.55%	2791	160	-	11.84%	145	10	-
30%	6.11%	1636	87	-	15.96%	107	9	-
35%	7.48%	1075	56	-	19.89%	88	8	-
40%	9.06%	696	36	-	35.07%	48	7	-

Table 4: performance of ε-coresets for fair k-means w.r.t. varying ε, also compared with **BICO**, on **Adult**.

ε	emp. err.		coreset size		T_S (ms)		T_X (ms)
	Ours	**BICO**	Ours	**BICO**	Ours	**BICO**	
10%	2.55%	1.23%	17665	18144	1688	1534	7548
15%	5.55%	4.97%	7164	6868	438	389	-
20%	8.56%	9.97%	3421	3474	185	187	-
25%	10.67%	14.19%	1906	1909	99	98	-
30%	13.68%	20.98%	1363	1412	71	71	-
35%	18.44%	24.65%	773	719	39	35	-
40%	20.94%	25.53%	559	550	29	29	-

6.1 Results

Table 3 and 4 summarize the accuracy-size trade-off of our coreset for fair k-median and k-means respectively, under different error guarantee ε. A key finding is that the size of the coreset does not suffer from the ε^{-d} factor thanks to our optimized implementation. As for the fair k-median, the empirical error of our coreset is well under control. In particular, to achieve 10% empirical error (even 5% for **Bank**), only about 500 points, which is about 1 percent of data, are necessary for both datasets, and this results in a ~200x acceleration in evaluating the objective.

Our coreset also works well for fair k-means, and it offers significant acceleration of evaluating the objective. Compared with **BICO**, our coreset offers better accuracy-size trade-off when ε is relatively large, and ours has a much lower empirical error when the size is below ~3500. However, it performs worse than **BICO** when ε is small. One explanation is that the way our algorithm works might not capture the pattern of our datasets. Recall that our algorithm emits rays and project points such that the projection cost is bounded, but we find this part becomes a bottleneck when ε is small. Intuitively, if the dataset is well clustered around a few lines, our algorithm should offer superior performance; however, this might not be the case in our datasets. On the other hand, when ε is relatively large, projecting to a few rays incurs acceptable error, and our coreset offers better overall performance.

Table 5 demonstrates the speed-up to **FairTree** with the help of our coreset. Our coreset

Table 5: speed-up of **FairTree** [5] via a 0.5-coreset

	Adult			Bank		
	objective	# nodes	time (ms)	objective	# nodes	time (ms)
FairTree	10634	91389	12220	8518	74543	11001
Ours	9840	1007	453	5993	158	344

offers about 30x speed-up while maintaining similar objective [7]. We observe that a crucial step of **FairTree** is an HST decomposition, and the overall performance of the algorithm depends on the decomposition. As shown in Table 5, the number of nodes generated by the HST decomposition is drastically reduced by about 100 times using our coreset, and this explains the speed-up.

7 Conclusion and future work

This paper constructs ε-coresets for the fair k-median/means clustering problem of size independent on the full dataset, and when the data may have multiple, non-disjoint types. Our coreset for fair k-median is the first known coreset construction to the best of our knowledge. For fair k-means, we improve the coreset size of the prior result [39], and extend it to multiple non-disjoint types. Our correctness analysis depends on several new geometric observations that may have independent interest. The empirical results show that our coresets are indeed much smaller than the full dataset and result in significant reductions in the running time of computing the fair clustering objective.

Our work leaves several interesting futural directions. For unconstrained clustering, there exist several works using the sampling approach such that the coreset size does not depend exponentially on the Euclidean dimension d. It is interesting to investigate whether sampling approaches can be applied for constructing fair coresets and achieve similar size bound as the unconstrained setting. Another interesting direction is to construct coresets for general fair (k, z)-clustering beyond k-median/means/center.

References

[1] Pankaj K Agarwal, Sariel Har-Peled, and Kasturi R Varadarajan. Approximating extent measures of points. *Journal of the ACM (JACM)*, 51(4):606–635, 2004.

[2] Pankaj K Agarwal and Cecilia Magdalena Procopiuc. Exact and approximation algorithms for clustering. *Algorithmica*, 33(2):201–226, 2002.

[3] Sara Ahmadian, Alessandro Epasto, Ravi Kumar, and Mohammad Mahdian. Clustering without over-representation. In *The 36th International Conference on Machine Learning (ICML)*, 2019.

[7] We note that **FairTree** is $O(\log n)$-approximate so a difference of factor $O(1)$ is entirely acceptable.

[4] Aris Anagnostopoulos, Luca Becchetti, Matteo Böhm, Adriano Fazzone, Stefano Leonardi, Cristina Menghini, and Chris Schwiegelshohn. Principal fairness: Removing bias via projections. In *The 36th International Conference on Machine Learning (ICML)*, 2019.

[5] Arturs Backurs, Piotr Indyk, Krzysztof Onak, Baruch Schieber, Ali Vakilian, and Tal Wagner. Scalable fair clustering. In *The 36th International Conference on Machine Learning (ICML)*, 2019.

[6] Suman K. Bera, Deeparnab Chakrabarty, and Maryam Negahbani. Fair algorithms for clustering. *CoRR*, abs/1901.02393, 2019.

[7] Ioana O Bercea, Martin Groß, Samir Khuller, Aounon Kumar, Clemens Rösner, Daniel R Schmidt, and Melanie Schmidt. On the cost of essentially fair clusterings. *arXiv preprint arXiv:1811.10319*, 2018.

[8] Vladimir Braverman, Dan Feldman, and Harry Lang. New frameworks for offline and streaming coreset constructions. *CoRR*, abs/1612.00889, 2016.

[9] Robin Burke, Alexander Felfernig, and Mehmet H Göker. Recommender systems: An overview. *AI Magazine*, 32(3):13–18, 2011.

[10] L. Elisa Celis, Lingxiao Huang, Vijay Keswani, and Nisheeth K. Vishnoi. Classification with fairness constraints: A meta-algorithm with provable guarantees. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 319–328. ACM, 2019.

[11] L. Elisa Celis, Lingxiao Huang, and Nisheeth K. Vishnoi. Multiwinner voting with fairness constraints. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 144–151. AAAI Press, 2018.

[12] L. Elisa Celis, Sayash Kapoor, Farnood Salehi, and Nisheeth K. Vishnoi. Controlling polarization in personalization: An algorithmic framework. In *Fairness, Accountability, and Transparency in Machine Learning*, 2019.

[13] L. Elisa Celis, Vijay Keswani, Damian Straszak, Amit Deshpande, Tarun Kathuria, and Nisheeth K. Vishnoi. Fair and diverse dpp-based data summarization. In *International Conference on Machine Learning*, pages 715–724, 2018.

[14] L. Elisa Celis, Anay Mehrotra, and Nisheeth K. Vishnoi. Towards controlling discrimination in online ad auctions. In *International Conference on Machine Learning*, 2019.

[15] L. Elisa Celis, Damian Straszak, and Nisheeth K. Vishnoi. Ranking with fairness constraints. In *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018)*, volume 107, page 28. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2018.

[16] L. Elisa Celis and Nisheeth K. Vishnoi. Fair personalization. In *Fairness, Accountability, and Transparency in Machine Learning*, 2017.

[17] Ke Chen. On k-median clustering in high dimensions. In *SODA*, pages 1177–1185. Society for Industrial and Applied Mathematics, 2006.

[18] Xingyu Chen, Brandon Fain, Charles Lyu, and Kamesh Munagala. Proportionally fair clustering. In *The 36th International Conference on Machine Learning (ICML)*, 2019.

[19] Flavio Chierichetti, Ravi Kumar, Silvio Lattanzi, and Sergei Vassilvitskii. Fair clustering through fairlets. In *Advances in Neural Information Processing Systems*, pages 5029–5037, 2017.

[20] Joydeep Das, Partha Mukherjee, Subhashis Majumder, and Prosenjit Gupta. Clustering-based recommender system using principles of voting theory. In *2014 International Conference on Contemporary Computing and Informatics (IC3I)*, pages 230–235. IEEE, 2014.

[21] Amit Datta, Michael Carl Tschantz, and Anupam Datta. Automated experiments on ad privacy settings: A tale of opacity, choice, and discrimination. *Proceedings on Privacy Enhancing Technologies*, 2015(1):92–112, 2015.

[22] D. Feldman and M. Langberg. A unified framework for approximating and clustering data. In *STOC*, pages 569–578, 2011.

[23] Dan Feldman, Melanie Schmidt, and Christian Sohler. Turning big data into tiny data: Constant-size coresets for k-means, PCA and projective clustering. In *SODA*, pages 1434–1453, 2013.

[24] Hendrik Fichtenberger, Marc Gillé, Melanie Schmidt, Chris Schwiegelshohn, and Christian Sohler. BICO: BIRCH meets coresets for k-means clustering. In *ESA*, 2013.

[25] Elena L Glassman, Rishabh Singh, and Robert C Miller. Feature engineering for clustering student solutions. In *Proceedings of the first ACM conference on Learning@ scale conference*, pages 171–172. ACM, 2014.

[26] Sariel Har-Peled. Clustering motion. *Discrete & Computational Geometry*, 31(4):545–565, 2004.

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

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

[29] IBM. IBM ILOG CPLEX optimization studio CPLEX user's manual, version 12 release 6, 2015.

[30] Sheng-Yi Jiang, Qi Zheng, and Qian-Sheng Zhang. Clustering-based feature selection. *Acta Electronica Sinica*, 36(12):157–160, 2008.

[31] Tapas Kanungo, David M Mount, Nathan S Netanyahu, Christine D Piatko, Ruth Silverman, and Angela Y Wu. A local search approximation algorithm for k-means clustering. *Computational Geometry*, 28(2-3):89–112, 2004.

[32] Matthäus Kleindessner, Pranjal Awasthi, and Jamie Morgenstern. Fair k-center clustering for data summarization. In *The 36th International Conference on Machine Learning (ICML)*, 2019.

[33] Matthäus Kleindessner, Samira Samadi, Pranjal Awasthi, and Jamie Morgenstern. Guarantees for spectral clustering with fairness constraints. In *The 36th International Conference on Machine Learning (ICML)*, 2019.

[34] Michael Langberg and Leonard J. Schulman. Universal ε-approximators for integrals. In *SODA*, pages 598–607, 2010.

[35] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982.

[36] Claire Cain Miller. Can an algorithm hire better than a human? *The New York Times*, 25, 2015.

[37] Manh Cuong Pham, Yiwei Cao, Ralf Klamma, and Matthias Jarke. A clustering approach for collaborative filtering recommendation using social network analysis. *J. UCS*, 17(4):583–604, 2011.

[38] Clemens Rösner and Melanie Schmidt. Privacy preserving clustering with constraints. In *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

[39] Melanie Schmidt, Chris Schwiegelshohn, and Christian Sohler. Fair coresets and streaming algorithms for fair k-means clustering. *arXiv preprint arXiv:1812.10854*, 2018.

[40] Pang-Ning Tan, Michael Steinbach, Vipin Kumar, et al. Cluster analysis: basic concepts and algorithms. *Introduction to data mining*, 8:487–568, 2006.

[41] Ke Yang and Julia Stoyanovich. Measuring fairness in ranked outputs. In *Proceedings of the 29th International Conference on Scientific and Statistical Database Management*, page 22. ACM, 2017.