Individually Fair Diversity Maximization

Ruien Li
School of Data Science and Engineering
East China Normal University
Shanghai, China
reli@stu.ecnu.edu.cn

Yanhao Wang[*]
School of Data Science and Engineering
East China Normal University
Shanghai, China
yhwang@dase.ecnu.edu.cn

Abstract

We consider the problem of diversity maximization from the perspective of individual fairness: given a set P of n points in a metric space, we aim to extract a subset S of size k from P so that (1) the diversity of S is maximized and (2) S is *individually fair* in the sense that every point in P has at least one of its $\frac{n}{k}$-nearest neighbors as its "representative" in S. We propose $(O(1), 3)$-bicriteria approximation algorithms for the individually fair variants of the three most common diversity maximization problems, namely, max-min diversification, max-sum diversification, and sum-min diversification. Specifically, the proposed algorithms provide a set of points where every point in the dataset finds a point within a distance at most 3 times its distance to its $\frac{n}{k}$-nearest neighbor while achieving a diversity value at most $O(1)$ times lower than the optimal solution. Numerical experiments on real-world and synthetic datasets demonstrate that the proposed algorithms generate solutions that are individually fairer than those produced by unconstrained algorithms and incur only modest losses in diversity.

1 Introduction

As machine learning (ML) algorithms are widely used in automated decision-making processes such as banking, hiring, and education, concerns have been raised about their negative social consequences [30, 32], e.g., discriminatory treatment against specific individuals. Recently, there has been extensive literature on *algorithmic fairness*, aiming to define the notion of fairness in ML problems and design effective and efficient fairness-aware algorithms [10, 31]. Such considerations have been taken into account in many data-driven ML tasks, including classification [12], clustering [44], ranking [34, 48], matching [41], and data summarization [8, 23].

This paper focuses on the diversity maximization problem and addresses its individually fair variants. *Diversity maximization* is a fundamental combinatorial optimization problem with broad applications in feature selection [47], search [15, 42], and recommendation [4, 7]. Generally, its objective is to extract a subset S of size k from a set P of n points so that the diversity of S (measured by the point-wise dissimilarity) is maximized. The existing studies on diversity maximization primarily consider three objectives, namely *max-min diversification*, which aims to maximize the minimum distance between any pair of selected points, *max-sum diversification*, which aims to maximize the sum of pairwise distances between selected points, and *sum-min diversification*, which aims to maximize the sum of the minimum distances from each selected point to its nearest neighbor among the other selected points. All three problems have been extensively investigated in the literature [6, 9, 20, 27], where they are also, respectively, referred to as *remote-edge*, *remote-clique*, and *remote-pseudoforest*. The unconstrained max-min diversification problem is NP-complete in metric spaces, and a greedy algorithm [16] offers a 2-approximation, which has also been shown to be tight [39]. For unconstrained max-sum and sum-min diversification problems, it is impossible to obtain an

[*] Corresponding Author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

approximation factor better than 2 in polynomial time if we assume that the planted clique conjecture holds [6, 22], while existing studies have proposed the best possible 2-approximation algorithm for max-sum diversification through greedy selection [19], and the best-known 8-approximation algorithm for sum-min diversification by randomization and solving linear programs (LPs) [6]. Given that each of these objectives highlights different aspects of diversity, we consider all of them in this paper to obtain a comprehensive understanding of the interplay between diversity and individual fairness in various application scenarios.

Currently, efficient algorithms for diversity maximization under *group fairness* or *partition matroid* constraints have been proposed in [2, 3, 33, 45, 46]. They consider that each point in P denotes an individual associated with a particular demographic attribute, e.g., gender or race. As such, P is divided into different demographic groups, and an algorithm is required to select a subset S from P not only to maximize the diversity of S but also to ensure the selection of a pre-determined number of points from every group in S for equitable representation. Despite these studies, little attention has been paid to diversity maximization under *individual fairness* constraints. Inspired by an individual notion of fairness for the facility location problem [21], the selection of a given point set P is fair if every point in P has a center among its $(|P|/k)$-closest neighbors. Compared with group fairness, individual fairness differs in two crucial aspects. First, it does not require access to group attributes, while group fairness relies on predefined features such as gender, age, or race. Second, the two notions consider different perspectives of fairness. While group fairness emphasizes equitable representation across groups, it may overlook disparities among individuals within each group. Individual fairness, by contrast, operates at the point level to ensure that each individual is adequately represented in the selected subset.

The need for individual fairness, in particular, is natural and reasonable in many scenarios [9, 13, 24, 38, 40]. For instance, consider a fast-food franchise that selects k supply outlets within a city. From the perspective of dispersion and diversity, we need to choose outlet locations that are spread out to avoid over-concentration in one region, thereby increasing overall market coverage and potential revenue. However, such dispersion strategies may still leave many individuals far from any chosen outlet, thus excluding them from convenient services. Incorporating individual fairness can address this issue by ensuring that each resident lies within a reasonable distance of at least one facility so that no individual is neglected. This balance not only enhances potential revenue but also improves the service experience for all customers, underscoring the need to balance diversity objectives with individual fairness considerations.

Our Contributions. With these motivations, we study the problem of α-fair k-selection under individual fairness constraints for the three common diversity maximization problems. Here, the parameter $\alpha \geq 1$ can be viewed as a fairness tolerance level. Larger α values allow a looser notion of fairness and thus potentially more diverse solutions, while $\alpha = 1$ corresponds to the strictest form of individual fairness. Moreover, our algorithms are analyzed under the notion of (β, γ)-*bicriteria approximation*, which means that they may relax the fairness constraints with a factor of γ (e.g., allowing each point to be covered within a small multiplicative slack) while still providing provable guarantees on the achieved diversity within a factor of β. Our main results include the following.

- **Max-Min Diversification:** For any $\alpha \geq 1$ and $k \in \mathbb{Z}_+$, we give a $(\beta, 3)$-bicriteria approximation algorithm for max-min diversification under individual fairness constraints if there exists a β-approximation algorithm for max-min diversification under partition matroid constraints. Using the approximation algorithm of [46] as a subroutine, the approximation factor can be instantiated as $(5 + \varepsilon, 3)$ for any $\varepsilon > 0$.
- **Max-Sum Diversification:** For any $\alpha \geq 1$ and $k \in \mathbb{Z}_+$, we give a $(\beta(4 + \varepsilon), 3)$-bicriteria approximation algorithm for max-sum diversification under individual fairness constraints if there exists a β-approximation algorithm for max-sum diversification under partition matroid constraints. Using the 2-approximation local search algorithm in [2] as a subroutine, the approximation factor can be instantiated as $(8 + \varepsilon, 3)$ for any $\varepsilon > 0$.
- **Sum-Min Diversification:** For any $\alpha \geq 1$ and $k < n/3$, we give a $(\beta(4 + \varepsilon), 3)$-bicriteria approximation algorithm for sum-min diversification under individual fairness constraints if there exists a β-approximation algorithm for sum-min diversification under partition matroid constraints. Using the LP-based 8-approximation algorithm in [6] as a subroutine, the approximation factor can be instantiated as $(32 + \varepsilon, 3)$ for any $\varepsilon > 0$.

To the best of our knowledge, these are the first algorithms with theoretical guarantees for individually fair max-min, max-sum, and sum-min diversification problems. Finally, we evaluate the empirical performance of our algorithms on real-world and synthetic data sets. The results demonstrate that our algorithms generate solutions that are individually fairer than those produced by unconstrained algorithms while incurring modest diversity losses. In addition, our algorithms also exhibit high time efficiency and scalability.

2 Related Work

(Unconstrained) Diversity Maximization. Diversity maximization has been studied from a graph-theoretic perspective since the 1990s. Ravi et al. [39] proposed a 2-approximation algorithm for max-min diversification and proved that the bound cannot be tighter unless P = NP. Hassin et al. [19] proposed a 2-approximation algorithm for max-sum diversification, and Bhaskara et al. [6] further proved that the approximation factor cannot be improved under the planted clique assumption [22]. Chandra and Halldórsson [9] proposed a $O(\log n)$-approximation algorithm for sum-min diversification. Bhaskara et al. [6] further proposed an 8-approximation algorithm for sum-min diversification and proved that the factor cannot be better than 2 under the same planted clique assumption. These algorithms cannot be used for diversity maximization variants with group or individual fairness constraints. Nevertheless, they often serve as building blocks for the design of fair diversity maximization algorithms.

Diversity Maximization under Group Fairness Constraints. Moumoulidou et al. [33] first proposed approximation algorithms for the group-fair variant of max-min diversification. Addanki et al. [3] improved the approximation ratios of the algorithms in [33]. Wang et al. [45] proposed streaming algorithms for the group-fair variant of max-min diversification. Wang et al. [46] developed an exact algorithm and a $(5+\varepsilon)$-approximation algorithm for max-min diversification with bounded-size group fairness constraints. Abbassi et al. [2] proposed 2-approximation local-search algorithms for max-sum diversification under matroid constraints. Bhaskara et al. [6] proposed an 8-approximation algorithm for sum-min diversification under matroid constraints based on randomization and linear programming. Mahabadi and Trajanovski [28] proposed two deterministic algorithms for sum-min diversification under group fairness constraints. One is a $O(m \cdot \log k)$-approximation algorithm with an exponential time complexity w.r.t. k, and the other is a $O(m^2 \cdot \log k)$-approximation algorithm that runs in polynomial time. Mahabadi and Trajanovski [28] also proposed coreset-based algorithms for max-sum and max-min diversification under group fairness constraints. However, the above algorithms only consider group fairness (or general matroid) constraints and cannot be directly used in the individually fair variants of diversity maximization.

Individual Fairness in Clustering Problems. Clustering has been studied extensively from a fairness perspective over the past few years. Most previous results consider clustering problems under the notion of *group fairness* [1, 10, 14, 23]. Jung et al. [21] first proposed the notion of *individual fairness* we use in this paper for facility location problems. Following this seminal work, individually fair clustering problems have attracted some attention in recent years. Mahabadi and Vakilian [29] studied center-based clustering problems, such as k-median, k-means, and k-center, with individual fairness constraints. Negahbani and Chakrabarty [35] explored individually fair k-clustering with general ℓ_p-norm objectives using a linear programming approach. Vakilian and Yalçiner [44] studied the problem of α-fair k-clustering with l_p-norm objectives, achieving improved approximation factors in both fairness and cost. Han et al. [17] considered individually fair k-center with outliers and proposed a 4-approximation algorithm. Bateni et al. [5] designed a fast local-search algorithm for individually fair k-means clustering with improved time complexity. These approaches provide inspiration for the algorithms presented in this paper. However, they cannot be applied directly to diversity maximization, as they are originally designed for clustering problems.

3 Problem Definition

Let (P, d) denote a metric space, where P is a set of n points and $d : P \times P \rightarrow \mathbb{R}_{\geq 0}$ is a distance function that measures the dissimilarity between any pair of points in P and satisfies the axioms of (i) *identity of indiscernibles*, (ii) *symmetry*, and (iii) *triangle inequality*. We use $S \subseteq P$ to denote a subset of points. The number of points to select, unless otherwise specified, is denoted by $k \in \mathbb{Z}_{+}$.

Then, we define the objective functions of the max-min, max-sum, and sum-min diversification as follows. They are all distance-based objectives; that is, they are formulated in terms of the pairwise distances between the chosen points. Let $d(u, v)$ be the distance between u and v, and for a set of points T, let $d(u, T) = \min_{v \in T} d(u, v)$. The diversity functions for the three objectives are as follows:

1. **Max-Min Diversification:** $\mathsf{div}_{\text{mm}}(S, P) = \min_{u \in S} d(u, S \backslash \{u\})$;
2. **Max-Sum Diversification:** $\mathsf{div}_{\text{ms}}(S, P) = \sum_{u \in S} \sum_{v \in S \backslash \{u\}} d(u, v)$;
3. **Sum-Min Diversification:** $\mathsf{div}_{\text{sm}}(S, P) = \sum_{u \in S} d(u, S \backslash \{u\})$.

Max-min diversification aims to maximize the minimum distance between any pair of distinct points in the selected set S. Intuitively, it encourages the selected points to be as far apart as possible, ensuring separation among all points. Max-sum diversification aims to maximize the sum of all pairwise distances among the points in the set S. Thus, it favors a set in which the points are collectively dispersed across the space. Sum-min diversification aims to maximize the sum of the distances from all points $u \in S$ to their nearest neighbor within $S \setminus \{u\}$. It strikes a balance between local and global diversity, promoting selections where each point is relatively well-separated from at least one close neighbor, without maximizing all pairwise distances.

Next, we introduce some notation to formally define the notion of individual fairness we consider in this paper. For every point $v \in P$, we use $B(v, r) := \{u \in P : d(v, u) \leq r\}$ to denote the subset of all points in P that are at distance at most r from v and call it the ball of radius r centered at v.

Definition 1 (Fair Radius). *Let $l \in [n]$ be a fairness parameter. For every point $v \in P$, we define the fair radius $r_l(v)$ as the minimum distance r such that $|B(v, r)| \geq \frac{n}{l}$. When $l = k$, we drop the subscript for simplicity and use $r(\cdot)$ to denote $r_k(\cdot)$.*

Then, we formally define the notion of α-fair k-selection [44], a variant of the individual fairness notion from [21, 29].

Definition 2 (α-Fair k-Selection). *A set of k points $S \subseteq P$ is α-fair if for every point $x \in P$, $d(x, S) \leq \alpha r_k(x)$.*

According to [21], there always exists a feasible solution when $\alpha \geq 2$. Ideally, we would like to find a solution with $\alpha = 1$, which would fully satisfy the original definition of individual fairness [21, 29]. However, since deciding whether a given set of points P admits a fair selection with $\alpha = 1$ is NP-hard [21], it is unlikely that such solutions can be found in polynomial time unless P = NP. Consequently, we aim to provide a bicriteria approximation guarantee instead.

Definition 3 (Bicriteria Approximation). *An algorithm is a (β, γ)-bicriteria approximation for α-fair k-selection w.r.t. a given diversity function if for any set of points P the solution SOL returned by the algorithm on P satisfies the following properties:*

1. $\mathrm{div}(\mathrm{OPT}, P) \leq \beta \cdot \mathrm{div}(\mathrm{SOL}, P)$*, where* OPT *denotes the optimal set of k points for α-fair k-selection of P w.r.t. the given diversity function. In particular,* $\mathrm{div}(\mathrm{OPT}, P) = 0$ *if an α-fair k-selection does not exist for P.*
2. SOL *is a $(\gamma \cdot \alpha)$-fair k-selection of P.*

Hardness of Approximation. Ravi et al. [39] showed that, unless P = NP, there does not exist any polynomial-time β-approximation algorithm for the unconstrained max-min diversification problem when $\beta < 2$. Bhaskara et al. [6] further proved that, under the planted clique conjecture, there exists no polynomial-time β-approximation algorithm for the unconstrained max-sum and sum-min diversification problems when $\beta < 2$. Since the unconstrained setting can be viewed as the special case of our individually fair setting with $\alpha = \infty$, the hardness results in the unconstrained case naturally generalize to our problems, as stated below.

Theorem 1. *There exists no polynomial-time β-approximation algorithm for individually fair max-min diversification with $\beta < 2$, unless P = NP.*

Theorem 2. *There exists no polynomial-time β-approximation algorithm for individually fair max-sum and sum-min diversification with $\beta < 2$ under the planted clique conjecture.*

```
Algorithm 1: IFRGenerate
Input: Fairness parameter $\alpha$
Output: A set of individual fairness regions for given parameters $\alpha, k$
Initialize the set of covered points $Z \leftarrow \emptyset$ and the set of centers of selected balls $\mathcal{C} \leftarrow \emptyset$
while $Z \neq P$ do
    $c \leftarrow \arg\min_{x \in P \setminus Z} r(x)$
    $\mathcal{C} \leftarrow \mathcal{C} \cup \{c\}$
    $Z \leftarrow Z \cup \{x \in P \setminus Z | d(x,c) \leq 2\alpha \cdot r(x)\}$
end while
return $\{B(c, \alpha \cdot r(c)) : c \in \mathcal{C}\}$
```

In previous studies on individually fair clustering [29, 44], connections between individual fairness and partition matroid constraints have been established. We now introduce the concept of matroid constraints so that our subsequent individually fair algorithms can be framed as solutions rooted in them. Note that the group fairness constraint has also been shown to be a type of matroid constraint.

Definition 4 (Matroid Constraint)**.** *A matroid $\mathcal{M}$ is defined as a family of subsets of the ground set of points $\mathcal{E}(\mathcal{M}) = P$, called independent sets. The set of independent sets S of a matroid $\mathcal{M}$ is denoted by $\mathcal{I}(\mathcal{M})$. For a given matroid $\mathcal{M}$, the associated matroid constraint is $S \in \mathcal{I}(\mathcal{M})$. As is standard, $\mathcal{M}$ is a uniform matroid of rank r if $\mathcal{I}(\mathcal{M}) := \{X \subseteq \mathcal{E}(\mathcal{M}) : |X| \leq r\}$. A partition matroid is the direct sum of uniform matroids. Note that uniform matroid constraints are equivalent to cardinality constraints, i.e., $|S| \leq k$. This definition follows [43].*

To satisfy individual fairness constraints, we introduce a special structure called *individual fairness regions*, which partitions the points based on their distances in the metric space. This notion is useful in our algorithms for connecting individual fairness to matroid constraints.

Definition 5 (Individual Fairness Region)**.** *A set of balls $\mathcal{B} = \{B(c_1, \alpha \cdot r(c_1)), \ldots, B(c_m, \alpha \cdot r(c_m))\}$, where $m \leq k$, is called a set of individual fairness regions if it satisfies the following properties:*

1. *For every $x \in P : d(x, \{c_1, \ldots, c_m\}) \leq 2\alpha \cdot r(x)$;*
2. *For any pair of centers c_i, c_j, $d(c_i, c_j) > 2\alpha \cdot \max\{r(c_i), r(c_j)\}$; in other words, individual fairness regions are disjoint from each other.*

4 Our Algorithms

In this section, we reduce our individually fair diversity maximization problem to data selection under partition matroid constraints. We first present an algorithm, IFRGenerate, that, given a set of points P and a fairness parameter α, returns a set of individual fairness regions.

Overview of Algorithm 1. IFRGenerate is designed to address the problem of maximizing diversity under individual fairness constraints by reducing it to the problem under partition matroid constraints. Similar algorithms have been used for individually fair clustering, and they are crucial for satisfying individual fairness constraints in clustering problems [29, 44]. The algorithm takes as input a fairness parameter α, the size of the input data set n, and the desired size of the output set k. The procedure begins by initializing two empty sets: a set of covered points $Z \leftarrow \emptyset$ and a set of selected ball centers $\mathcal{C} \leftarrow \emptyset$. It then enters a loop that continues as long as the set of uncovered points $P \setminus Z$ is non-empty. At each iteration, the algorithm selects a center c that has the minimal fair radius in the uncovered points, adds c to the set of centers $\mathcal{C}$, and updates the set of covered points Z by including all points $x \in P \setminus Z$ whose distance $d(x,c)$ is at most $2\alpha \cdot r(x)$. This process repeats until all points are covered. The output of the algorithm is a set of balls $\{B(c, \alpha r(c)) : c \in \mathcal{C}\}$, where each ball should consist of a selected center c and its radius $\alpha r(c)$. This set of balls can be treated as different groups, connecting our problem with that under the partition matroid constraints.

We now prove that IFRGenerate generates individual fairness regions as defined in Definition 5.

Lemma 1. *Let k be a positive integer and α be a parameter that indicates the desired fairness guarantee. Algorithm 1 returns a set of at most k individual fairness regions in $O(nk)$ time.*

Algorithm 2: OMSGENERATE

```
Input: Set of points $P$, desired number of selected points $k$, fairness parameter $\alpha$
Output: Individual fairness regions and $k_i$ in the original metric space
Compute a set of individual fairness regions $\mathcal{B} = \{B_1, \cdots, B_m\}$ on $(P, k, \alpha)$ using Algorithm 1
Let $P_0 = \{v_0 | v \in P \backslash (B_1 \cup \cdots \cup B_m)\}$ be the points not in individual fairness regions
$k_i = k - m + 1$ for all $i \in [m]$   ▷ {denote that we pick at most $k - m + 1$ centers from each
  individual fairness ball}
$k_0 = k$   ▷ {denote that we pick at most $k$ centers from $P_0$}
return $\{(P_0, k_0), (B_1, k_1), \cdots, (B_m, k_m)\}$
```

See the proof in Appendix B.1. Similar to individually fair clustering [29, 44], the benefit of a set of individual fairness regions is that it reduces the problem of finding an α-fair selection to a data selection problem with lower bound requirements, i.e., at least one point must be selected from each individual fairness region. Lemma 2 indicates that a set of points S is *feasible* w.r.t. a set of individual fairness regions $\mathcal{B}$, if for every ball $B \in \mathcal{B}, |S \cap B| > 0$.

Lemma 2. *Let $\mathcal{B} = \{B(c_1, \alpha \cdot r(c_1)), \ldots, B(c_m, \alpha \cdot r(c_m))\}$ be a set of individual fairness regions obtained from Algorithm 1 for a set of points P with parameters k and α. Then, any set of points S that is feasible w.r.t. $\mathcal{B}$ is (3α)-fair.*

See the proof in Appendix B.2. Based on the lemmas above, we now introduce our algorithm for maximizing diversity under individual fairness constraints. We first construct individual fairness regions on the original data P using Algorithm 2.

Overview of Algorithm 2. OMSGENERATE is designed to address the problem of selecting a diverse subset of points while satisfying the fairness constraints in the given dataset P, specifically by generating a set of individual fairness regions in P. The algorithm takes as input a set of points P, a desired number of selected points k, a fairness parameter α, and the original metric space. It begins by computing a set of individual fairness regions $\mathcal{B} = \{B_1, \ldots, B_m\}$ using Algorithm 1 (IFRGENERATE) on the input (P, k, α). Next, it identifies the set of points P_0 that are not covered by these fairness regions, i.e., points outside the union of all fairness regions in $\mathcal{B}$. For each individual fairness region B_i, the algorithm sets the maximum number of points that can be selected from B_i at $k_i = k - m + 1$. In addition, it sets $k_0 = k$, which is the maximum number of centers that can be selected from the uncovered points P_0. Note that $k_0, k_1, \ldots, k_m$ do not denote the number of points that should be selected but the maximum number that can be selected in a region. In practical implementations, the number of points to be selected may differ from $k_0, k_1, \ldots, k_m$, as they are only the upper bounds on the number of points to select. Our ultimate goal is always to select k points as the final result. Finally, the algorithm returns a set of individual fairness regions $\{(P_0, k_0), (B_1, k_1), \ldots, (B_m, k_m)\}$, representing the individual fairness regions in P and the maximum number of points that can be selected from each region. Since the computation can be mainly attributed to Algorithm 1, the time complexity of Algorithm 2 is $O(nk)$ as well.

Now, we demonstrate that with a β-approximation algorithm for different diversity maximization problems under partition matroid constraints, a bicriteria approximation for α-fair k-selection exists. We first give the approximation for max-min diversification under individual fairness constraints.

Theorem 3. *Suppose that there exists a β-approximation algorithm for max-min diversification under partition matroid constraints. Then, there exists a $(\beta, 3)$-bicriteria approximation for α-fair k-selection in max-min diversification.*

Proof Sketch. We show that the solution from a β-approximation algorithm (MAXMINALG) for max-min diversification under partition matroid constraints is a $(\beta, 3)$-bicriteria approximation for α-fair k-selection on metric space P. The solution SOL_G from MAXMINALG satisfies the partition matroid constraints, ensuring at least one point per fairness region B_i. By Lemma 2, this implies that SOL_G is (3α)-fair. We show that the diversity of the optimal solution OPT_G on P is equal to that of OPT'_G on a constructed instance P'. By mapping the solutions between P and P', we show $\mathsf{div}_{\text{mm}}(\text{OPT}_G, P) = \mathsf{div}_{\text{mm}}(\text{OPT}'_G, P')$.

For the α-fair optimal solution OPT_I on P, we can construct a feasible solution OPT'_C on P' with equal diversity while satisfying the partition matroid constraints. Since OPT'_G is optimal on P', $\mathsf{div}_{\text{mm}}(\text{OPT}'_G, P') \geq \mathsf{div}_{\text{mm}}(\text{OPT}'_C, P') = \mathsf{div}_{\text{mm}}(\text{OPT}_I, P)$. Given that MaxMinAlg is β-approximate, $\mathsf{div}_{\text{mm}}(\text{OPT}_G, P) \leq \beta \cdot \mathsf{div}_{\text{mm}}(\text{SOL}_G, P)$, thus $\mathsf{div}_{\text{mm}}(\text{OPT}_I, P) \leq \beta \cdot \mathsf{div}_{\text{mm}}(\text{SOL}_G, P)$. Therefore, SOL_G is a $(\beta, 3)$-bicriteria approximation. □

See the full proof in Appendix B.4.

After proving the existence of a bicriteria approximation algorithm for α-fair k-selection with a max-min objective, we then prove that the algorithm for the max-sum objective also exists.

Theorem 4. *Suppose that there exists a β-approximation algorithm for max-sum diversification under partition matroid constraints. Then, there exists a $(\beta(4+\varepsilon), 3)$-bicriteria approximation for α-fair k-selection in max-sum diversification.*

Proof Sketch. We show that the solution from a β-approximation algorithm (MaxSumAlg) for max-sum diversification under partition matroid constraints is a $(\beta(4+\varepsilon), 3)$-bicriteria approximation for α-fair k-selection on metric space P. We establish that $\mathsf{div}_{\text{ms}}(\text{OPT}_G, P) \geq \frac{1}{4+\varepsilon}\mathsf{div}_{\text{ms}}(\text{OPT}'_G, P')$. When $\mathsf{div}_{\text{ms}}(\text{OPT}_G, P) < \mathsf{div}_{\text{ms}}(\text{OPT}'_G, P')$, OPT'_G may include multiple copies of points in P. In the extreme case, k points in P' map to $\lfloor \frac{k}{2} \rfloor$ points in P, and we analyze basic cases (e.g., four points in P' from two in P) to show $\mathsf{div}_{\text{ms}}(\text{OPT}'_G, P') \leq (4+\varepsilon) \cdot \mathsf{div}_{\text{ms}}(\text{ORI}, P)$, where ORI are distinct points in P. Since $\mathsf{div}_{\text{ms}}(\text{OPT}_G, P) \geq \mathsf{div}_{\text{ms}}(\text{ORI}, P)$, we get $\mathsf{div}_{\text{ms}}(\text{OPT}_G, P) \geq \frac{1}{4+\varepsilon}\mathsf{div}_{\text{ms}}(\text{OPT}'_G, P')$.

For the α-fair optimal solution OPT_I on P, we construct OPT'_C on P' with equal diversity while satisfying the partition matroid constraints. Since OPT'_G is optimal on P', $\mathsf{div}_{\text{ms}}(\text{OPT}'_G, P') \geq \mathsf{div}_{\text{ms}}(\text{OPT}'_C, P') = \mathsf{div}_{\text{ms}}(\text{OPT}_I, P)$. In combination with the β-approximation of MaxSumAlg, $\mathsf{div}_{\text{ms}}(\text{OPT}_I, P) \leq (4+\varepsilon) \cdot \mathsf{div}_{\text{ms}}(\text{OPT}_G, P) \leq \beta(4+\varepsilon) \cdot \mathsf{div}_{\text{ms}}(\text{SOL}_G, P)$. Therefore, SOL_G is a $(\beta(4+\varepsilon), 3)$-bicriteria approximation. □

See the full proof in Appendix B.5.

We then consider the same circumstance for sum-min diversification.

Theorem 5. *Suppose that there exists a β-approximation algorithm for sum-min diversification under partition matroid constraints with $1 < k < n/3$. Then, there exists a $(\beta(4+\varepsilon), 3)$-bicriteria approximation for α-fair k-selection in sum-min diversification.*

Proof Sketch. We show that the solution from a β-approximation algorithm (SumMinAlg) for k-selection under partition matroid constraints for sum-min diversification is a $(\beta(4+\varepsilon), 3)$-bicriteria approximation for α-fair k-selection on metric space P. We establish that $\mathsf{div}_{\text{sm}}(\text{OPT}_G, P) \geq \frac{1}{4+\varepsilon}\mathsf{div}_{\text{sm}}(\text{OPT}'_G, P')$. When $\mathsf{div}_{\text{sm}}(\text{OPT}_G, P) < \mathsf{div}_{\text{sm}}(\text{OPT}'_G, P')$, OPT'_G includes multiple copies of the points in P. With $z \leq k$ distinct points in P corresponding to OPT'_G, pairs of copies contribute $2(k-z)\varepsilon\delta$ to $\mathsf{div}_{\text{sm}}(\text{OPT}'_G, P')$. Defining $D = \mathsf{div}_{\text{sm}}(\text{OPT}'_G, P') - 2(k-z)\varepsilon\delta$, we show $D \leq \mathsf{div}_{\text{sm}}(\text{OPT}^z_G, P)$, where OPT^z_G is the optimal z-point solution in P. By Lemma 4 (see Appendix B.6), $\mathsf{div}_{\text{sm}}(\text{OPT}^z_G, P) \leq 4 \cdot \mathsf{div}_{\text{sm}}(\text{OPT}_G, P)$ for $1 < k < n/3$. Since $2(k-z)\varepsilon\delta \leq \varepsilon \cdot \mathsf{div}_{\text{sm}}(\text{OPT}_G, P)$, we get $\mathsf{div}_{\text{sm}}(\text{OPT}'_G, P') \leq (4+\varepsilon) \cdot \mathsf{div}_{\text{sm}}(\text{OPT}_G, P)$.

For the α-fair optimal solution OPT_I on P, we construct OPT'_C on P' with equal diversity satisfying partition matroid constraints. Since OPT'_G is optimal on P', $\mathsf{div}_{\text{sm}}(\text{OPT}'_G, P') \geq \mathsf{div}_{\text{sm}}(\text{OPT}'_C, P') = \mathsf{div}_{\text{sm}}(\text{OPT}_I, P)$. Combining with the β-approximation of SumMinAlg, $\mathsf{div}_{\text{sm}}(\text{OPT}_I, P) \leq (4+\varepsilon) \cdot \mathsf{div}_{\text{sm}}(\text{OPT}_G, P) \leq \beta(4+\varepsilon) \cdot \mathsf{div}_{\text{sm}}(\text{SOL}_G, P)$. Therefore, SOL_G is a $(\beta(4+\varepsilon), 3)$-bicriteria approximation. □

See the full proof in Appendix B.6.

According to Theorems 3–5, we have the following theorems with specific approximation factors. Note that the computation of ε differs between these theorems; detailed explanations are provided in Appendix C.

Theorem 6. *For any $\alpha \geq 1$, there exists a $O(mkn + m^k \log \frac{1}{\varepsilon})$ time algorithm that computes a $(5+\varepsilon, 3)$-bicriteria approximate solution to the α-fair k-selection of max-min diversification under individual fairness constraints.*

The proof follows from Theorem 3 and the $(5+\varepsilon)$-approximation algorithm from [46] for max-min diversification under partition matroid constraints. We note that the time complexity of this algorithm is not linear w.r.t. k. More detailed discussions are provided in Appendix D.

Theorem 7. *For any $\alpha \geq 1$, there exists a polynomial time algorithm that computes a $(8+\varepsilon, 3)$-bicriteria approximate solution to the α-fair k-selection of max-sum diversification under individual fairness constraints.*

The proof follows from Theorem 4 and the 2-approximation algorithm in [2] for max-sum diversification under partition matroid constraints.

Theorem 8. *For any $\alpha \geq 1$, there exists a nearly linear-time algorithm that computes a $(32+\varepsilon, 3)$-bicriteria approximate solution to the α-fair k-selection of sum-min diversification under individual fairness constraints.*

The proof follows from Theorem 5 and the 8-approximation algorithm in [6] for sum-min diversification under partition matroid constraints.

5 Experiments

In this section, we empirically compare our algorithms with unconstrained diversity maximization algorithms. The experiments focus on presenting the trade-off between fairness and diversity loss of our algorithms.

Implementation. All experiments were carried out on a server with an Intel(R) Xeon(R) Gold 6134 CPU @3.20GHz (2 processors) and 128GB RAM running Windows Server 2019 Datacenter. The algorithms were implemented in Python 3. Our code and data are publicly available at `https://github.com/HonokaKousaka/IFDM`.

Data Sets. In the experiments, we used three public real-world data sets and one synthetic data set. The basic information for each data set is shown in Table 1. For MovieLens, the user vectors are obtained through matrix factorization using LIBMF [11]. We randomly sampled 1,000 points from each data set for evaluation.

Table 1: Statistics of data sets in the experiments, where n is the number of data points and dim is the dimensionality.

Dataset	**Description**	n	dim
CelebA [26]	Features for celebrity images extracted by VGG16	202,599	25,088
GloVe [37]	Global vectors for word representation	400,000	100
MovieLens [18]	User vectors obtained from the rating matrix	162,541	50
Gaussian	Gaussian blobs by `make_blobs` in scikit-learn [36]	1,000,000	20

Experimental Setup. For max-min diversification under individual fairness constraints, we implemented FMMD-S [46], a $(5+\varepsilon)$-approximation algorithm for max-min diversification under partition matroid constraints in $O(mkn + m^k \log \frac{1}{\varepsilon})$ time, and fixed $\varepsilon = 0.05$ for FMMD-S. We compared our algorithm with GMM [16], which provides a 2-approximation for unconstrained max-min diversification. For max-sum diversification under individual fairness constraints, we implemented the local search algorithm in [2], which is a 2-approximation algorithm for max-sum diversification under general matroid constraints in $O(\frac{n}{\varepsilon} \log(k))$ time, and set $\varepsilon = 0.05$ in the algorithm. We compared our algorithm with the greedy algorithm in [19], which guarantees a 2-approximation for unconstrained max-sum diversification. For sum-min diversification under individual fairness constraints, we implemented the coreset-based algorithm in [28], which provides an $O(m^2 \cdot \log k)$-approximation for sum-min diversification under group fairness constraints in polynomial time. To obtain an unconstrained solution for sum-min diversification, we placed all points in an individual fairness region and ran our algorithm accordingly. We fixed $\alpha = 1$ in all experiments so that each algorithm makes the best effort to ensure individual fairness. We implemented all algorithms in Python 3 and used the Gurobi optimizer to solve ILPs in FMMD-S. To compare with unconstrained optimal solutions for quantifying exact utility losses, we also ran the Gurobi optimizer to solve the ILPs for max-min, max-sum, and sum-min diversification within a 30-minute time limit.

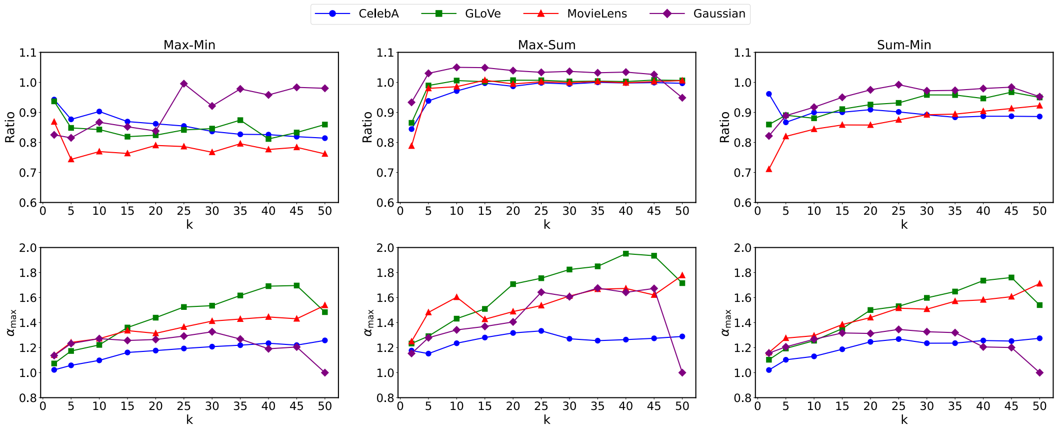

Figure 1: Overall experimental results. The first row illustrates the ratios of the diversity values of the solutions returned by our algorithms over the diversity values of the best solutions returned by unconstrained algorithms when $k = 2, 5, 10, \ldots, 50$ and $\alpha = 1$; the second row presents the $\alpha_{\max}$ values of the solutions returned by our algorithms.

Evaluation Measures. In terms of individual fairness, we define $\alpha_{\max} = \max_{x \in P} d(x, \mathrm{SOL})/r(x)$ as the performance metric. Specifically, $\alpha_{\max}$ indicates how close a solution is to satisfying the individual fairness constraint on P: The smaller $\alpha_{\max}$, the fairer the solution SOL. If $\alpha_{\max} \leq 1$, the solution SOL strictly satisfies the individual fairness constraint. In terms of utility, we compute the ratio of the diversity value of a fair solution to that of an unconstrained solution returned by an approximation or an exact ILP-based algorithm, using this ratio as the performance metric. The larger the ratio, the higher the utility of the solution. When using approximation algorithms to obtain unconstrained solutions, we ran them 10 times and selected the solution with the largest diversity value. We also ran our proposed algorithms ten times and used the average of both measures for evaluation.

Experimental Results. In Figure 1, we present the experimental results for three diversity objectives across four data sets. We observe that our algorithms consistently provide approximate solutions with diversity losses of no more than 30% compared to unconstrained (approximate) solutions, while always guaranteeing that the value of $\alpha_{\max}$ is below 2. In Table 2, we also present the ratios of the diversity values of our solutions to the optimal diversity values computed from the ILPs solved by Gurobi. As shown, our algorithms limit utility losses relative to unconstrained (optimal) solutions to at most 35%, further validating their effectiveness in ensuring diversity while satisfying individual fairness constraints. These results demonstrate that our algorithms strike an effective balance between individual fairness and diversity.

Table 2: Ratios of the diversity values of the solutions of our proposed algorithms over the diversity values of the optimal unconstrained solutions by ILPs when $k = 5, 10, 20$ and $\alpha = 1$.

Dataset	**Max-Min**			**Max-Sum**			**Sum-Min**		
	$k=5$	$k=10$	$k=20$	$k=5$	$k=10$	$k=20$	$k=5$	$k=10$	$k=20$
CelebA	0.827	0.807	0.794	0.923	0.967	0.985	0.703	0.769	0.842
GloVe	0.780	0.793	0.824	0.947	0.980	0.991	0.754	0.835	0.910
MovieLens	0.701	0.724	0.764	0.925	0.961	0.977	0.676	0.757	0.830
Gaussian	0.819	0.845	0.819	0.968	0.988	0.996	0.831	0.893	0.955

Figure 2 shows the running times of our algorithms on each data set. In terms of time efficiency, taking CelebA as an example, the process of computing an individually fair solution (including Algorithms 1 & 2, and the diversification algorithms with matroid constraints) takes no more than 4 seconds when $k \leq 50$ for max-min and sum-min diversification. For max-sum diversification, because ε is very small, the local search requires many iterations to meet the stop condition; thus, it takes about 300 seconds when $k = 50$.

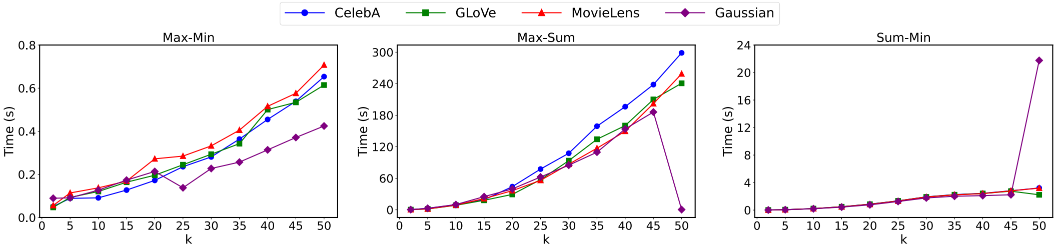

Figure 2: Running time (in seconds) of our algorithms when $k = 2, 5, 10, \dots, 50$ and $\alpha = 1$.

We observe that when $k = 50$, the synthetic Gaussian data set exhibits anomalous results in the ratios, the $\alpha_{\max}$ values, and the running time for max-sum and sum-min diversification. This phenomenon can be attributed to the data generation process, since the Gaussian data set is constructed by sampling points around 50 Gaussian centers. As a result, when $k = 50$, the data set naturally forms exactly 50 individual fairness regions, each containing 20 points. Consequently, every data point falls into one of these regions, and selecting one point from each region is sufficient to achieve high diversity while guaranteeing individual fairness. This leads to $\alpha_{\max} \le 1$ when $k = 50$; that is, the proposed algorithms provide solutions that strictly satisfy the individual fairness constraints. Moreover, for max-sum diversification, the local search procedure requires only a few iterations before meeting the stop condition because the initial solution already contains one point from each region. In contrast, for sum-min diversification, our proposed algorithm uses GMM to fit within each of the 50 fairness regions and determine potential candidates. Consequently, performing 50 independent GMM instances significantly increases the runtime when $k = 50$.

6 Conclusion

In this paper, we study the diversity maximization problem under individual fairness constraints, namely α-fair k-selection. By generating individual fairness regions to partition data points and utilizing existing approximation algorithms for diversity maximization under matroid constraints, we propose a $(5+\varepsilon, 3)$-bicriteria approximation algorithm for max-min diversification, $(8+\varepsilon, 3)$-bicriteria approximation algorithm for max-sum diversification, and $(32+\varepsilon, 3)$-bicriteria approximation algorithm for sum-min diversification for any $\alpha \ge 1$ in any metric space. Experimental results demonstrate that our proposed algorithms efficiently provide individually fair, highly diverse subsets on real-world and synthetic datasets.

There are still some open problems for future work. Given that the approximation factors of our algorithms are determined by those for diversity maximization under matroid constraints, a possible direction is to improve the approximation ratio for individually fair diversity maximization by utilizing better matroid-constrained diversity maximization algorithms. In addition, we acknowledge that we have not proved whether individually fair diversity maximization problems are strictly harder to approximate than their unconstrained counterparts. Establishing such hardness results remains open and would shed light on further improvements in approximation factors. Another promising avenue for exploration is to extend the problems and proposed algorithms to streaming, distributed, and deletion-robust settings.

Acknowledgments and Disclosure of Funding

This work was supported by the National Natural Science Foundation of China (Grant No. 62202169).

References

[1] Mohsen Abbasi, Aditya Bhaskara, and Suresh Venkatasubramanian. Fair clustering via equitable group representations. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 504–514, 2021.

[2] Zeinab Abbassi, Vahab S Mirrokni, and Mayur Thakur. Diversity maximization under matroid constraints. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 32–40, 2013.

[3] Raghavendra Addanki, Andrew McGregor, Alexandra Meliou, and Zafeiria Moumoulidou. Improved approximation and scalability for fair max-min diversification. In *25th International Conference on Database Theory*, pages 7:1–7:21, 2022.

[4] Azin Ashkan, Branislav Kveton, Shlomo Berkovsky, and Zheng Wen. Optimal greedy diversity for recommendation. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence*, pages 1742–1748, 2015.

[5] MohammadHossein Bateni, Vincent Cohen-Addad, Alessandro Epasto, and Silvio Lattanzi. A scalable algorithm for individually fair k-means clustering. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, pages 3151–3159, 2024.

[6] Aditya Bhaskara, Mehrdad Ghadiri, Vahab S. Mirrokni, and Ola Svensson. Linear relaxations for finding diverse elements in metric spaces. *Advances in Neural Information Processing Systems*, 29:4098–4106, 2016.

[7] Pablo Castells, Neil Hurley, and Saúl Vargas. Novelty and diversity in recommender systems. In Francesco Ricci, Lior Rokach, and Bracha Shapira, editors, *Recommender Systems Handbook*, pages 603–646. Springer, New York, NY, USA, 2022.

[8] Elisa Celis, Vijay Keswani, Damian Straszak, Amit Deshpande, Tarun Kathuria, and Nisheeth Vishnoi. Fair and diverse DPP-based data summarization. In *Proceedings of the 35th International Conference on Machine Learning*, pages 716–725, 2018.

[9] Barun Chandra and Magnús M Halldórsson. Approximation algorithms for dispersion problems. *J. Algorithms*, 38(2):438–465, 2001.

[10] Flavio Chierichetti, Ravi Kumar, Silvio Lattanzi, and Sergei Vassilvitskii. Fair clustering through fairlets. *Advances in Neural Information Processing Systems*, 30:5029–5037, 2017.

[11] Wei-Sheng Chin, Bo-Wen Yuan, Meng-Yuan Yang, Yong Zhuang, Yu-Chin Juan, and Chih-Jen Lin. LIBMF: a library for parallel matrix factorization in shared-memory systems. *J. Mach. Learn. Res.*, 17(1):2971–2975, 2016.

[12] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, pages 214–226, 2012.

[13] Erhan Erkut. The discrete p-dispersion problem. *Eur. J. Oper. Res.*, 46(1):48–60, 1990.

[14] Mehrdad Ghadiri, Samira Samadi, and Santosh Vempala. Socially fair k-means clustering. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 438–448, 2021.

[15] Sreenivas Gollapudi and Aneesh Sharma. An axiomatic approach for result diversification. In *Proceedings of the 18th International Conference on World Wide Web*, pages 381–390, 2009.

[16] Teofilo F Gonzalez. Clustering to minimize the maximum intercluster distance. *Theor. Comput. Sci.*, 38:293–306, 1985.

[17] Lu Han, Dachuan Xu, Yicheng Xu, and Ping Yang. Approximation algorithms for the individually fair k-center with outliers. *J. Glob. Optim.*, 87(2):603–618, 2023.

[18] F. Maxwell Harper and Joseph A. Konstan. The movielens datasets: History and context. *ACM Trans. Interact. Intell. Syst.*, 5(4):19:1–19:19, 2016.

[19] Refael Hassin, Shlomi Rubinstein, and Arie Tamir. Approximation algorithms for maximum dispersion. *Oper. Res. Lett.*, 21(3):133–137, 1997.

[20] Piotr Indyk, Sepideh Mahabadi, Mohammad Mahdian, and Vahab S Mirrokni. Composable core-sets for diversity and coverage maximization. In *Proceedings of the 33rd ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pages 100–108, 2014.

[21] Christopher Jung, Sampath Kannan, and Neil Lutz. Service in your neighborhood: Fairness in center location. In *1st Symposium on Foundations of Responsible Computing*, pages 5:1–5:15, 2020.

[22] Richard M. Karp. The probabilistic analysis of some combinatorial search algorithms. Technical Report UCB/ERL M581, University of California, Berkeley, Apr 1976. URL `http://www2.eecs.berkeley.edu/Pubs/TechRpts/1976/28848.html`.

[23] Matthäus Kleindessner, Pranjal Awasthi, and Jamie Morgenstern. Fair k-center clustering for data summarization. In *Proceedings of the 36th International Conference on Machine Learning*, pages 3448–3457, 2019.

[24] Michael J Kuby. Programming models for facility dispersion: The p-dispersion and maxisum dispersion problems. *Geogr. Anal.*, 19(4):315–329, 1987.

[25] Yash Kurkure, Miles Shamo, Joseph Wiseman, Sainyam Galhotra, and Stavros Sintos. Faster algorithms for fair max-min diversification in R^d. *Proc. ACM Manag. Data*, 2(3):137:1–137:26, 2024.

[26] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *2015 IEEE International Conference on Computer Vision*, pages 3730–3738, 2015.

[27] Sepideh Mahabadi and Shyam Narayanan. Improved diversity maximization algorithms for matching and pseudoforest. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 25:1–25:22, 2023.

[28] Sepideh Mahabadi and Stojan Trajanovski. Core-sets for fair and diverse data summarization. *Advances in Neural Information Processing Systems*, 36:78987–79011, 2023.

[29] Sepideh Mahabadi and Ali Vakilian. Individual fairness for k-clustering. In *Proceedings of the 37th International Conference on Machine Learning*, pages 6586–6596, 2020.

[30] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Comput. Surv.*, 54(6):115:1–115:35, 2022.

[31] Shira Mitchell, Eric Potash, Solon Barocas, Alexander D'Amour, and Kristian Lum. Algorithmic fairness: Choices, assumptions, and definitions. *Annu. Rev. Stat. Appl.*, 8:141–163, 2021.

[32] Brent Daniel Mittelstadt, Patrick Allo, Mariarosaria Taddeo, Sandra Wachter, and Luciano Floridi. The ethics of algorithms: Mapping the debate. *Big Data Soc.*, 3(2):1–21, 2016.

[33] Zafeiria Moumoulidou, Andrew McGregor, and Alexandra Meliou. Diverse data selection under fairness constraints. In *24th International Conference on Database Theory*, pages 13:1–13:25, 2021.

[34] Harikrishna Narasimhan, Andrew Cotter, Maya Gupta, and Serena Wang. Pairwise fairness for ranking and regression. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04): 5248–5255, 2020.

[35] Maryam Negahbani and Deeparnab Chakrabarty. Better algorithms for individually fair *k*-clustering. *Advances in Neural Information Processing Systems*, 34:13340–13351, 2021.

[36] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.*, 12:2825–2830, 2011.

[37] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543, 2014.

[38] Nikolaos Ploskas, Kostas Stergiou, and Dimosthenis C Tsouros. The p-dispersion problem with distance constraints. In *29th International Conference on Principles and Practice of Constraint Programming*, pages 30:1–30:18, 2023.

[39] Sekharipuram S Ravi, Daniel J Rosenkrantz, and Giri Kumar Tayi. Heuristic and special case algorithms for dispersion problems. *Oper. Res.*, 42(2):299–310, 1994.

[40] SS Ravi, Daniel J Rosenkrantz, and Giri K Tayi. Approximation algorithms for facility dispersion. In *Handbook of Approximation Algorithms and Metaheuristics*, pages 347–364. Chapman and Hall/CRC, 2018.

[41] Govind S Sankar, Anand Louis, Meghana Nasre, and Prajakta Nimbhorkar. Matchings with group fairness constraints: Online and offline algorithms. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pages 377–383, 2021.

[42] Rodrygo L. T. Santos, Craig Macdonald, and Iadh Ounis. Exploiting query reformulations for web search result diversification. In *Proceedings of the 19th International Conference on World Wide Web*, pages 881–890, 2010.

[43] Alexander Schrijver. *Combinatorial Optimization: Polyhedra and Efficiency*. Springer, Berlin, Heidelberg, 2003.

[44] Ali Vakilian and Mustafa Yalçiner. Improved approximation algorithms for individually fair clustering. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, pages 8758–8779, 2022.

[45] Yanhao Wang, Francesco Fabbri, and Michael Mathioudakis. Streaming algorithms for diversity maximization with fairness constraints. In *2022 IEEE 38th International Conference on Data Engineering*, pages 41–53, 2022.

[46] Yanhao Wang, Michael Mathioudakis, Jia Li, and Francesco Fabbri. Max-min diversification with fairness constraints: Exact and approximation algorithms. In *Proceedings of the 2023 SIAM International Conference on Data Mining*, pages 91–99, 2023.

[47] Sepehr Abbasi Zadeh, Mehrdad Ghadiri, Vahab S. Mirrokni, and Morteza Zadimoghaddam. Scalable feature selection via distributed diversity maximization. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1):2876–2883, 2017.

[48] Meike Zehlike, Francesco Bonchi, Carlos Castillo, Sara Hajian, Mohamed Megahed, and Ricardo Baeza-Yates. FA*IR: A fair top-k ranking algorithm. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 1569–1578, 2017.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: See the Abstract and Introduction sections.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitations are discussed in the Conclusion section.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: See the Lemmas and Theorems as well as their proofs in the paper.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We have described the experimental setup in Section 5.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We have released our code and data in a GitHub repository.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: See details in Section 5.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: The experimental results are stable, since the experiments are run across four datasets in various parameter settings. We also ran each algorithm that involves approximation and randomization ten times and reported the best (for baselines) or the average (for our proposed algorithms) of each evaluation metric.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See details in Section 5.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our work does not involve crowdsourcing, contract work. Our work does not involve interactions between researchers and human participants. The datasets being used are open-source or generated by scikit-learn, and we have correctly made citations. We do not identify any deviation from the Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: In this work, we aim to maximize commonly used diversity objective functions while ensuring a certain level of individual fairness. Our algorithm does not involve human intervention in data distribution or classification. Our work contributes to raising awareness of fairness and diversity in ML-based decision-making processes. Generally, we do not foresee any immediate and direct harmful impacts from this work.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The datasets being used in our paper are open-source or generated by scikit-learn. Therefore, our work does not pose such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We list the licenses for the assets we use in our work. These assets have been correctly and properly cited in our paper.

    Code: Our implementation of the algorithm for sum-min diversification under partition matroid constraints is partly based on the code from [29], which has an MIT license.

    Datasets: CelebA is available for non-commercial research purposes, as claimed by the dataset creators on their official website. MovieLens can be used for any research purposes under their customized Usage License. GloVe is available under the Public Domain Dedication and License, as claimed by the creators.

    Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We have released our code in a well-documented format.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The algorithm design and proof in this paper do not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Algorithm 3 (PSGENERATE) (Used for Proof Only)

Consider an instance of α-fair k-selection on a point set P. Let $\mathcal{B}$ be the set of individual fairness regions of P with parameters k and α constructed using Algorithm 2. Then, given an instance of α-fair k-selection, Algorithm 3 builds an instance of k-selection under partition matroid constraints.

Overview of Algorithm 3. Existing studies on individually fair clustering [44] have implemented a similar instance to generate an individually fair solution for clustering problems. Our implementation of Algorithm 3 is a little different, since the instance generated is not directly involved in our solution generation process. However, it serves a critical role in our theoretical analysis. Unlike prior work, where the constructed instance is typically used as a direct component in producing the final clustering solution, our approach leverages the instance primarily to facilitate the proof of our theorem guarantees, ensuring that the diversity properties hold under the individual fairness constraints. PSGENERATE takes as input a point set P, the desired number of selected points k, a fairness parameter α, and an accuracy parameter $\varepsilon < 1/2$. It begins by computing a set of individual fairness regions $\mathcal{B} = B_1, \ldots, B_m$ using Algorithm 2 on (P, k, α). Next, it creates copies of the original point set P and each individual fairness region B_i, denoted as $\overline{P}_0$ and $\overline{B}_i$ respectively, and constructs an extended set P' by including $\overline{P}_0$ and $\overline{B}_i$. Note that P' has two distinct copies of the points that belong to an individual fairness ball of $\mathcal{B}$. A modified distance function d' is then defined, where $d'(u, u) = 0$ for all $u \in P'$, $d'(v_x, u_y) = d(v, u)$ for distinct $v, u \in P'$ with $v \neq u$, and $d'(v_x, v_y) = \varepsilon\delta$ for $v_x, v_y \in P'$, where $\delta \leftarrow \min_{x,y \in P} d(x, y)$. The algorithm finally returns the tuple $(P', \{(\overline{P}_0, \overline{k}_0), (\overline{B}_1, \overline{k}_1), \ldots, (\overline{B}_m, \overline{k}_m)\}, d')$, providing an instance corresponding to the given α-fair k-selection problem for the proof purpose. Given that the individual fairness regions in P have been computed, the time complexity of Algorithm 3 is $O(n)$, since most of the steps only require iterating over all points in P. Note that we do not need to implement Algorithm 3 in practice.

Algorithm 3: PSGENERATE

```
Input: set of points $P$, desired number of selected points $k$, fairness parameter $\alpha$, accuracy
  parameter $\varepsilon < 1/2$
Output: an instance of $k$-selection under partition matroid constraints w.r.t. the given instance of
  $\alpha$-fair $k$-selection for the proof
Compute the set $\{(P_0, k_0), (B_1, k_1), \cdots, (B_m, k_m)\}$ on $(P, k, \alpha)$ using Algorithm 2
Let $\overline{P}_0 = \{v_0 | v \in P\}$ be a copy of $P$
Let $\overline{B}_i = \{v_i | v \in B_i\}$ be a copy of $B_i$ for all $B_i \in \mathcal{B}$
$P' \leftarrow \overline{P}_0 \cup (\bigcup_{B_i \in \mathcal{B}} \overline{B}_i)$    ▷{$P'$ has two distinct copies of the points that belong to an
  individual fairness ball of $\mathcal{B}$.}
▷Construct a distance function $d' : P' \times P' \to \mathbb{R}^+$
$\overline{k}_0 = k_0$    ▷{denotes that we pick at most $k_0$ points from $\overline{P}_0$}
$\overline{k}_i = k_i$ for all $i \in [m]$    ▷{denotes that we pick at most $k_i$ points from $\overline{B}_i$}
Let $\delta \leftarrow \min_{x,y \in P} d(x, y)$
Let $d'(u, u) = 0$ for all $u \in P'$
Let $d'(v_x, u_y) = d(v, u)$ for all $v_x, u_y \in P'$ where $v \neq u$
Let $d'(v_x, v_y) = \varepsilon\delta$ for all $v_x, v_y \in P'$
return $(P', \{(\overline{P}_0, \overline{k}_0), (\overline{B}_1, \overline{k}_1), \cdots, (\overline{B}_m, \overline{k}_m)\}, d')$
```

We show that the distance function d' in Algorithm 3 is a metric distance.

Lemma 3. *The distance function $d' : P' \times P' \to \mathbb{R}^+$ constructed in Algorithm 3 constitutes a metric space. (See the proof in Appendix B.3.)*

B Missing Proofs

B.1 Proof of Lemma 1

Proof. First, we show that the set of centers returned by Algorithm 1 satisfies property (1) of the individual fairness regions. For every point $x \in P$ let c_x denote the first center added to $\mathcal{C}$ such that $d(x, c_x) \leq 2\alpha \cdot r(x)$. Hence, $d(x, \mathcal{C}) \leq d(x, c_x) \leq 2\alpha \cdot r(x)$ where the last inequality follows from the fact that c_x marks x as covered. Next, consider the iteration of the algorithm in which a center c is added to $\mathcal{C}$. Since c is an uncovered point, its distance to any other center c' that is already in $\mathcal{C}$ is more than $2\alpha \cdot r(c) = 2\alpha \cdot \max\{r(c), r(c')\}$ where the equality follows from the fact that centers are picked in a non-decreasing order of their fair radius. Hence, for any pair of centers in $\mathcal{C}$, property (2) holds. Finally, by property (2), the balls of radius $r(\cdot)$ around the centers present in $\mathcal{C}$ are disjoint. Moreover, according to the definition of fair radius, each of the balls $\{B(c, r(c))\}_{c\in\mathcal{C}}$ contains at least n/k points. Hence, the number of individual fairness regions is at most k.

Now we consider the time complexity of Algorithm 1. When generating individual fairness regions, the first step (Line 5) is to iterate over all radii to find the minimal fair radius, whose worst time complexity is $O(n)$. The time complexity of Line 6 is obviously $O(1)$. The final step (Line 7) requires iterating over points in $P\backslash Z$, whose time complexity is $O(n)$. Considering the outer `while` loop runs at most k times, the total time complexity of Lines 5, 6, and 7 is $O(nk)$, $O(k)$, and $O(nk)$, respectively. Thus, the total time complexity for generating individual fairness regions is $O(nk)$. □

B.2 Proof of Lemma 2

Proof. Let S be a set of cluster centers that is feasible w.r.t. $\mathcal{B}$. For every point $x \in P$, let c_x denote the first center chosen by Algorithm 1 such that $d(x, c_x) \leq 2\alpha \cdot r(x)$. Moreover, let s_x denote the center in S such that $s_x \in B(c_x, \alpha \cdot r(c_x))$. Then, for any point $x \in P$, we have

$$\begin{aligned} d(x, s_x) &\leq d(x, c_x) + d(c_x, s_x) \\ &\leq 2\alpha \cdot r(x) + d(c_x, s_x) \\ &\leq 2\alpha \cdot r(x) + \alpha \cdot r(c_x) \\ &\leq 3\alpha \cdot r(x), \end{aligned}$$

where the first inequality follows from the triangle inequality, the second inequality follows from the property (1) of individual fairness regions, the third inequality follows since $s_x \in B(c_x, \alpha \cdot r(c_x))$, and the last inequality follows since centers are added in a non-decreasing order of their fair radius in line 5 of Algorithm 1, leading to $r(c_x) \leq r(x)$. □

B.3 Proof of Lemma 3

Proof. Let $u, v, w \in (\mathcal{F} \cup \mathcal{M})$ be three arbitrary points and let u_P, v_P, w_P be their corresponding points from P. First, we prove that $d'(u, v) = 0 \iff u = v$. If $u = v$, the distance $d'(u, v)$ is set to zero by line 10. To show the other direction, if $d'(u, v) = 0$ then the constraint $u = v$ for the assignment in line 10 is satisfied since $d(u_P, v_P) > 0$ for all $u_p \neq v_p$ (line 11) and $d'(u, v) = \varepsilon\delta > 0$ when $u_p = v_p$ and $u \neq v$ (line 12). Secondly, we prove the symmetric property $d'(u, v) = d'(v, u)$. If $d'(u, v) = 0$, then by the first part $u = v$ and therefore $d'(v, u) = 0 = d'(u, v)$. Assume $d'(u, v) > 0$, which implies $u \neq v$. If $u_P \neq v_P$, then by line 11 and the metric properties of d, $d'(u, v) = d(u_P, v_P) = d(v_P, u_P) = d'(v, u)$ holds. Lastly, we show that the triangle inequality $d'(u, w) \leq d'(u, v) + d'(v, w)$ holds. If $u = w$ then by the first property, $d'(u, w) = 0$ so the equality holds. Assume $u \neq w$ and consider their corresponding points u_P, w_P.

1. If $u_P = w_P$, then $d'(u, w) = \varepsilon\delta$. If $v_P = u_P$, then $d'(u, v) = d'(u, w) = \varepsilon\delta$ and therefore $d'(u, w) \leq d'(u, v) + d'(v, w)$ already holds. If $v_P \neq u_P$, then $d'(u, v) = d(u_P, v_P) \geq \min_{x,y\in P} d(x, y) \geq \varepsilon\delta$. Thus, $d'(u, w) \leq d'(u, v) + d'(v, w)$ holds.
2. If $u_P \neq w_P$, then $d'(u, w) = d(u_P, w_P) \geq \varepsilon\delta$. Note that ($u_P = v_P$ and $v_P = w_P$) can not hold, so consider the remaining three cases:
   (a) $v_P = w_P$ and $u_P \neq v_P$. Then $d'(u, w) = d'(u, v)$ and $d'(u, w) \leq d'(u, v) + d'(v, w)$;
   (b) $u_P = v_P$ and $v_P \neq w_P$. Then $d'(u, w) = d'(v, w)$ and $d'(u, w) \leq d'(u, v) + d'(v, w)$;
   (c) $u_P \neq v_P$ and $v_P \neq w_P$. Then $d'(u, w) = d(u_P, w_P)$, $d'(v, w) = d'(v_P, w_P)$ and since $d(\cdot)$ satisfies the triangle inequality, $d'(u, w) \leq d'(u, v) + d'(v, w)$ holds.

By combining the above results, we conclude that $d'(\cdot, \cdot)$ is a metric distance. □

B.4 Proof of Theorem 3

Proof. Let MaxMinAlg be a β-approximation algorithm for k-selection under partition matroid constraints for max-min diversification. Consider $\{(P_0, k_0), (B_1, k_1), \ldots, (B_m, k_m)\}$ as the original metric space with individual fairness regions on P constructed by Algorithm 2. Consider an instance of α-fair k-selection on P and let c be the instance of k-selection under partition matroid constraints constructed by Algorithm 3 with input parameters P, k, and α. We show that the solution returned by MaxMinAlg$(\{(P_0, k_0), (B_1, k_1), \ldots, (B_m, k_m)\})$ is a $(\beta, 3)$-bicriteria approximate solution of the given instance of α-fair k-selection on P.

Let SOL_G be the solution returned by MaxMinAlg$(\{(P_0, k_0), (B_1, k_1), \ldots, (B_m, k_m)\})$. Let OPT_G be the optimal solution on $(\{(P_0, k_0), (B_1, k_1), \ldots, (B_m, k_m)\})$ maximizing the diversity for max-min diversification on P under partition matroid constraints. We let OPT'_G be the optimal solution on $(P', \{(\overline{P}_0, \overline{k}_0), (\overline{B}_1, \overline{k}_1), \ldots, (\overline{B}_m, \overline{k}_m)\}, d')$ from Algorithm 3 maximizing the diversity for max-min diversification on P' under partition matroid constraints. We also let OPT_I be the optimal solution on P which maximizes the diversity with individual fairness constraints satisfied.

Fairness Approximation: Given that SOL_G is constructed under partition matroid constraints, for each $i \in [m]$, $|B_i \cap \text{SOL}_G| \geq 1$. Hence, by Lemma 2, SOL_G is a (3α)-fair k-selection of P.

Diversity Approximation: We first prove that $\mathsf{div}_{\text{mm}}(\text{OPT}_G, P) = \mathsf{div}_{\text{mm}}(\text{OPT}'_G, P')$. No matter what OPT_G is, we can always construct COR'_G in P' whose diversity is equivalent to OPT_G in P. We start with an initially empty set of centers COR'_G. In the first step, for each $B \in \mathcal{B}$, let c_i denote an arbitrary center in $\text{OPT}_G \cap B_i$, and then we add the point $c \in \overline{B}_i$ corresponding to c_i to COR'_G. Next, in the second step, for each o_0 in the rest points of OPT_G, we add the point $o \in \overline{P}_0$ corresponding to o_0 to COR'_G. It is obvious that COR'_G has exactly k distinct centers and the pairwise distances between COR'_G are the same as those between OPT_G. Therefore, $\mathsf{div}_{\text{mm}}(\text{OPT}_G, P) = \mathsf{div}_{\text{mm}}(\text{COR}'_G, P')$. Considering that $\mathsf{div}_{\text{mm}}(\text{OPT}'_G, P') \geq \mathsf{div}_{\text{mm}}(\text{COR}'_G, P')$, we have $\mathsf{div}_{\text{mm}}(\text{OPT}_G, P) \leq \mathsf{div}_{\text{mm}}(\text{OPT}'_G, P')$.

Now we assume that $\mathsf{div}_{\text{mm}}(\text{OPT}_G, P) < \mathsf{div}_{\text{mm}}(\text{OPT}'_G, P')$. If there do not exist two points u', v' in OPT'_G that are the copies of the same point in P, then we can always find the original k points in P (which are corresponding to OPT'_G) whose diversity is greater than current $\mathsf{div}_{\text{mm}}(\text{OPT}_G, P)$ and equals $\mathsf{div}_{\text{mm}}(\text{OPT}'_G, P')$, which is a contradiction. If there exist two points u', v' in OPT'_G that are the copies of the same point in P, then the diversity is $\varepsilon\delta$ given the definition of max-min diversification, which is much smaller than $\mathsf{div}_{\text{mm}}(\text{OPT}_G, P)$, leading to a contradiction. Therefore, $\mathsf{div}_{\text{mm}}(\text{OPT}_G, P) = \mathsf{div}_{\text{mm}}(\text{OPT}'_G, P')$.

Next, we connect the partition matroid constraints on P' with individual fairness constraints on P. By the definition of α-fairness, each point $v \in P$ must have a center in OPT_I within distance at most $\alpha \cdot r(v)$. Hence, for each individual fairness region $B \in \mathcal{B}$, $|\text{OPT}_I \cap B| \geq 1$. For each $i \in [m]$, let c_i be the copy of an arbitrary center $c \in \text{OPT}_I \cap B_i$ in the set $\overline{B}_i$. For the remaining points in OPT_I, we pick their corresponding copies in the set $\overline{P}_0$. Let OPT'_C denote the constructed solution for the instance P'. Since OPT'_C picks exactly one point from each set $\overline{B}_i$, for $i \in [m]$, and exactly $k - m$ points from $\overline{P}_0$, OPT'_C is a feasible solution for max-min diversification under partition matroid constraints on instance $(P', \{(\overline{P}_0, \overline{k}_0), (\overline{B}_1, \overline{k}_1), \ldots, (\overline{B}_m, \overline{k}_m)\}, d')$. Since the pairwise distances between OPT_I are the same as those between OPT'_C, we can have $\mathsf{div}_{\text{mm}}(\text{OPT}_I, P) = \mathsf{div}_{\text{mm}}(\text{OPT}'_C, P')$. Considering that OPT'_G is the optimal solution on instance $(P', \{(\overline{P}_0, \overline{k}_0), (\overline{B}_1, \overline{k}_1), \ldots, (\overline{B}_m, \overline{k}_m)\}, d')$, we can have $\mathsf{div}_{\text{mm}}(\text{OPT}'_G, P') \geq \mathsf{div}_{\text{mm}}(\text{OPT}'_C, P')$. Hence, we have

$$\mathsf{div}_{\text{mm}}(\text{OPT}_I, P) \leq \mathsf{div}_{\text{mm}}(\text{OPT}'_G, P') = \mathsf{div}_{\text{mm}}(\text{OPT}_G, P') \leq \beta \cdot \mathsf{div}_{\text{mm}}(\text{SOL}_G, P).$$

Thus, the diversity of max-min diversification under individual fairness constraints of P using SOL_G is within a β factor of the diversity of any optimal α-fair k-selection of P. □

B.5 Proof of Theorem 4

Proof. We use the same notations as those in Proof of Theorem 3. The only difference is that MaxMinAlg is changed into MaxSumAlg, referring to a β-approximation algorithm for k-selection under partition matroid constraints for max-sum diversification. We show that the solution returned by MaxSumAlg$(\{(P_0, k_0), (B_1, k_1), \ldots, (B_m, k_m)\})$ is a $(\beta(4 + \varepsilon), 3)$-bicriteria approximate solution of the given instance of α-fair k-selection on P.

Diversity Approximation: We first prove that $(4+\varepsilon)\cdot \mathsf{div}_{\text{ms}}(\text{OPT}_G, P) \geq \mathsf{div}_{\text{ms}}(\text{OPT}'_G, P')$. We can get $\mathsf{div}_{\text{ms}}(\text{OPT}_G, P) \leq \mathsf{div}_{\text{ms}}(\text{OPT}'_G, P')$ through the same process shown in the same part in Proof of Theorem 3. Let us focus on the case when $\mathsf{div}_{\text{ms}}(\text{OPT}_G, P) < \mathsf{div}_{\text{ms}}(\text{OPT}'_G, P')$ occurs. In this case, there are two or more points in OPT'_G being copies of the same point in OPT_G. The extreme situation is that each point p'_1 in OPT'_G can always find another point p'_2 in OPT'_G, and their original point in P is the same point. Therefore, in general situations, there are at most $\lfloor \frac{k}{2} \rfloor$ points in P whose two copies in P' are both selected.

In order to obtain the diversity of the extreme situation, we now consider the basic case where there are only four points in P' that are selected, and there are only two points in P, which are the original points of the four points in P'. W.l.o.g., let EXP'_4 be the four points in P' containing a'_1, a'_2, a'_3, a'_4 and EXP_2 be the two distinct original points in P containing a_1, a_3, where a'_1, a'_2 are the copies of a_1, and a'_3, a'_4 are the copies of a_3. From the definition, we can have $d'(a'_1, a'_2) = d'(a'_3, a'_4) = \varepsilon\delta$, $d'(a'_1, a'_3) = d'(a'_1, a'_4) = d'(a'_2, a'_3) = d'(a'_2, a'_4) = d(a_1, a_3)$. It is obvious that $\mathsf{div}_{\text{ms}}(\text{EXP}_2, P) = d(a_1, a_3)$, while we have $\mathsf{div}_{\text{ms}}(\text{EXP}'_4, P') = d'(a'_1, a'_2) + d'(a'_3, a'_4) + d'(a'_1, a'_3) + d'(a'_1, a'_4) + d'(a'_2, a'_3) + d'(a'_2, a'_4) = 4 \cdot \mathsf{div}_{\text{ms}}(\text{EXP}_2, P) + 2\varepsilon\delta$.

Now consider the situation for OPT'_G is not extreme, i.e., there exists at least one point p'_1 in OPT'_G that is unable to find another point p'_2 in OPT'_G satisfying $d'(p'_1, p'_2) = \varepsilon\delta$. There are two types of basic situations. W.l.o.g., in the first situation, we let EXP'_3 be the 3 points in P' containing a'_1, a'_2, a'_3 and EXP_2 be the 2 distinct original points in P containing a_1, a_2, where a'_2, a'_3 are the copies of a_2, and a'_1 is the only copy of a_1. From the definition, we can have $d'(a'_2, a'_3) = \varepsilon\delta$, $d'(a'_1, a'_2) = d'(a'_1, a'_3) = d(a_1, a_2)$. We can have $\mathsf{div}_{\text{ms}}(\text{EXP}'_3, P') = d'(a'_1, a'_2) + d'(a'_1, a'_3) + d'(a'_2, a'_3) = 2 \cdot \mathsf{div}_{\text{ms}}(\text{EXP}_2, P) + \varepsilon\delta \leq 4 \cdot \mathsf{div}_{\text{ms}}(\text{EXP}_2, P) + 2\varepsilon\delta$. In the second situation, we let EXP'_2 be the 2 points in P' containing a'_1, a'_2 and EXP_2 be the 2 original points in P containing a_1, a_2, where a'_1 is the only copy of a_1, and a'_2 is the only copy of a_2 as well. We have $d'(a'_1, a'_2) = d(a_1, a_2)$ and $\mathsf{div}_{\text{ms}}(\text{EXP}'_2, P') = \mathsf{div}_{\text{ms}}(\text{EXP}_2, P) \leq 4 \cdot \mathsf{div}_{\text{ms}}(\text{EXP}_2, P) + 2\varepsilon\delta$.

For a general OPT'_G in P', $\mathsf{div}_{\text{ms}}(\text{OPT}'_G)$ is a combination of the three basic cases mentioned above. We let ORI be the distinct original points in P which are corresponding to OPT'_G in P'. We can have $4 \cdot \mathsf{div}_{\text{ms}}(\text{ORI}, P) + \lfloor \frac{k}{2} \rfloor \varepsilon\delta \geq \mathsf{div}_{\text{ms}}(\text{OPT}'_G, P')$. Given that there at least $\lfloor \frac{k}{2} \rfloor$ points in ORI, we have $\mathsf{div}_{\text{ms}}(\text{ORI}, P) \geq \lfloor \frac{k}{2} \rfloor \delta$, and we can further have $(4+\varepsilon) \cdot \mathsf{div}_{\text{ms}}(\text{ORI}, P) \geq \mathsf{div}_{\text{ms}}(\text{OPT}'_G, P')$. Next, we assume $|\text{ORI}| = k_s \geq \lfloor \frac{k}{2} \rfloor$, and we let $\text{OPT}^{k_s}_G$ be the optimal solution for k_s-point max-sum diversification under partition matroid constraints in P. We can have $\mathsf{div}_{\text{ms}}(\text{OPT}_G, P) \geq \mathsf{div}_{\text{ms}}(\text{OPT}^{k_s}_G, P)$, since the addition of any other point to $\text{OPT}^{k_s}_G$ would increase the diversity. Given that $\mathsf{div}_{\text{ms}}(\text{OPT}^{k_s}_G, P) \geq \mathsf{div}_{\text{ms}}(\text{ORI}, P)$, we can have $\mathsf{div}_{\text{ms}}(\text{OPT}_G, P) \geq \mathsf{div}_{\text{ms}}(\text{ORI}, P)$. Therefore, we can come to the conclusion that $(4+\varepsilon) \cdot \mathsf{div}_{\text{ms}}(\text{OPT}_G, P) \geq \mathsf{div}_{\text{ms}}(\text{OPT}'_G, P')$.

Through the same process as that in proof of Theorem 3, we have

$$\mathsf{div}_{\text{ms}}(\text{OPT}_I, P) \leq \mathsf{div}_{\text{ms}}(\text{OPT}'_G, P') \leq (4+\varepsilon) \cdot \mathsf{div}_{\text{ms}}(\text{OPT}_G, P) \leq \beta(4+\varepsilon) \cdot \mathsf{div}_{\text{ms}}(\text{SOL}_G, P).$$

Thus, the diversity of max-sum diversification under individual fairness constraints of P using SOL_G is within a $\beta(4+\varepsilon)$ factor of the diversity of any optimal α-fair k-selection of P. □

B.6 Proof of Theorem 5

Proof. We use the same notations as those in the proof of Theorem 3. The only difference is that MaxMinAlg is changed into SumMinAlg, referring to a β-approximation algorithm for k-selection under partition matroid constraints for sum-min diversification. We show that the solution returned by SumMinAlg$(\{(P_0, k_0), (B_1, k_1), \ldots, (B_m, k_m)\})$ is a $(\beta(4+\varepsilon), 3)$-bicriteria approximate solution of the given instance of α-fair k-selection on P.

Diversity Approximation: We first prove that $(4+\varepsilon) \cdot \mathsf{div}_{\text{sm}}(\text{OPT}_G, P) \geq \mathsf{div}_{\text{sm}}(\text{OPT}'_G, P')$. We can get $\mathsf{div}_{\text{sm}}(\text{OPT}_G, P) \leq \mathsf{div}_{\text{sm}}(\text{OPT}'_G, P')$ through the same process shown in the same part in Proof of Theorem 3.

Let us focus on the case when $\mathsf{div}_{\text{sm}}(\text{OPT}_G, P) < \mathsf{div}_{\text{sm}}(\text{OPT}'_G, P')$ occurs. In this case, there are two or more points in OPT'_G being copies of the same point in OPT_G. We assume that there are z distinct original points in P corresponding to OPT'_G in P', $z \leq k$. Now we focus on two points a'_1, a'_2 in OPT'_G satisfying $d'(a'_1, a'_2) = \varepsilon\delta$ if they exist. It can be inferred that these kinds of points contribute

$2(k-z)\varepsilon\delta$ to the overall diversity, since $d'(a'_1, \text{OPT}'_G \backslash a'_1) = d'(a'_2, \text{OPT}'_G \backslash a'_2) = d'(a'_1, a'_2) = \varepsilon\delta$ and there are $2(k-z)$ points from OPT'_G involved in this calculation.

Let us define $D = \mathsf{div}_{\text{sm}}(\text{OPT}'_G, P') - 2(k-z)\varepsilon\delta$. We notice that D does not equal the diversity of the original z points in P. This is because some points in the original z points that are copied to OPT'_G have contributed $\varepsilon\delta$ to $\mathsf{div}_{\text{sm}}(\text{OPT}'_G, P')$, so they cannot make any other contribution to D. Let OPT^z_G be the optimal solution for z-point sum-min diversification under partition matroid constraints in P. From what we have discussed above, we can get $D \leq \mathsf{div}_{\text{sm}}(\text{OPT}^z_G, P)$.

Now we are going to introduce a lemma [6] so as to compare $\mathsf{div}_{\text{sm}}(\text{OPT}^z_G, P)$ and $\mathsf{div}_{\text{sm}}(\text{OPT}_G, P)$.

Lemma 4. *Let (P, d) be a metric space, and $n = |P|$. Suppose $1 < k < n/3$ is the target number of elements. Let S' be any subset of V of size $\leq k$. Then we can efficiently find an $S \subseteq V$ of size $= k$, such that $\mathsf{div}_{sm}(S, P) \geq \frac{1}{4}\mathsf{div}_{sm}(S', P)$. (See the proof in Appendix B.7.)*

Based on Lemma 4, for OPT^z_G, we can always find a TEMP $\subseteq P$ of size $= k$, such that $4 \cdot \mathsf{div}_{\text{sm}}(\text{TEMP}, P) \geq \mathsf{div}_{\text{sm}}(\text{OPT}^z_G, P)$. Obviously, $\mathsf{div}_{\text{sm}}(\text{TEMP}, P) \leq \mathsf{div}_{\text{sm}}(\text{OPT}_G, P)$, since OPT_G is the optimal solution. We can now have

$$\mathsf{div}_{\text{sm}}(\text{OPT}^z_G, P) \leq 4 \cdot \mathsf{div}_{\text{sm}}(\text{OPT}_G, P)$$

if $1 < k < n/3$. Now we have

$$\begin{aligned}\mathsf{div}_{\text{sm}}(\text{OPT}'_G, P') &= D + 2(k-z)\varepsilon\delta \\ &\leq \mathsf{div}_{\text{sm}}(\text{OPT}^z_G, P) + 2(k-z)\varepsilon\delta \\ &\leq 4 \cdot \mathsf{div}_{\text{sm}}(\text{OPT}_G, P) + 2(k-z)\varepsilon\delta.\end{aligned}$$

As mentioned in the proof of Theorem 4, $k - z \geq \lfloor \frac{k}{2} \rfloor$. Therefore, we have $2(k-z)\varepsilon\delta \leq 2\lfloor \frac{k}{2} \rfloor \varepsilon\delta \leq k\varepsilon\delta \leq \varepsilon \cdot \mathsf{div}_{\text{sm}}(\text{OPT}_G, P)$. Therefore, we have $\mathsf{div}_{\text{sm}}(\text{OPT}'_G, P') \leq (4+\varepsilon) \cdot \mathsf{div}_{\text{sm}}(\text{OPT}_G, P)$.

Hence, we finally come to the conclusion that

$$\begin{aligned}\mathsf{div}_{\text{sm}}(\text{OPT}_I, P) &\leq \mathsf{div}_{\text{sm}}(\text{OPT}'_G, P') \\ &\leq (4+\varepsilon) \cdot \mathsf{div}_{\text{sm}}(\text{OPT}_G, P) \\ &\leq \beta(4+\varepsilon) \cdot \mathsf{div}_{\text{sm}}(\text{SOL}_G, P).\end{aligned}$$

Thus, the diversity of sum-min diversification under individual fairness constraints of P using SOL_G is within a $\beta(4+\varepsilon)$ factor of the diversity of any optimal α-fair k-selection of P. □

B.7 Proof of Lemma 4

Proof. Let us suppose $1 < k < n/3$, and let S' be a set of size $r < k$. We may assume that $r \geq 2$, and so we can suppose that $S' = \{u_1, u_2, \ldots, u_r\}$, for $1 < r < k$.

Let us partition P into $P_1, P_2, \ldots, P_r$, where P_i is the set of vertices whose closest neighbor in S' is u_i. Without loss of generality, assume that $|P_1| \geq |P_2| \geq \cdots \geq |P_r|$. Also, let d_i denote the minimum distance from u_i to the rest of S'. We consider two cases:

Case 1 ($|P_1| \geq k$): First, consider the set of points $W_1 := \{u_2, u_3, \ldots, u_r\} \cup T$, where T is an arbitrary set of $k-r+1$ points in P_1. We claim that the diversity of W_1 is at least $(d_2+d_3+\cdots+d_r)/2$. This is because for any $i > 1$, every point in P_1 is at a distance at least $d_i/2$ from u_i (to see this, consider some $v \in P_i$; we know that $d(v, u_1) \leq d(v, u_i)$, by the definition of P_i; thus, if $d(v, u_i) < d_i/2$, we must have $d(v, u_1) < d_i$, a contradiction). Second, let v be the u_i that is furthest from u_1. Clearly, we have $d(u_1, v) \geq d_1$. Now consider the set of points $W_2 := \{v\} \cup T$, where T is any set of $k-1$ vertices in P_1. From the same argument as above, the diversity of W_2 is at least $d_1/2$. Now, one of the sets above must have diversity $\geq (d_1 + d_2 + \cdots + d_r)/4 \geq \mathsf{div}_{\text{sm}}(S', P)/4$. This completes the argument in this case.

Case 2 ($|P_1| < k < n/3$): In other words, all the sets V_i have size $< k$. Now, let s be the smallest index for which $|P_1 \cup \cdots \cup P_s| \geq k$. Since all the $|P_i|$ are smaller than $k < n/3$, we certainly have $s < r$. Furthermore, we must have $|P_{s+1} \cup \cdots \cup P_r| \geq k$. Now, define $W_1 := \{u_1, u_2, \ldots, u_s\} \cup T$, where T is an arbitrary set of $k-s$ elements from $P_{s+1} \cup \cdots \cup P_r$ and $W_2 := \{u_{s+1}, \ldots, u_r\} \cup T'$, where T' is an arbitrary set of $k-s$ elements from $P_1 \cup \cdots \cup P_s$. By the above argument, the diversity of W_1 is at least $(d_1 + d_2 + \cdots + d_s)/2$, and that of W_2 is at least $(d_{s+1} + \cdots + d_r)/2$. As before, one of these quantities is at least $\mathsf{div}_{\text{sm}}(S', P)/4$. □

C Discussion on Error Parameter in Theorems 6–8

C.1 Error Parameter in Theorem 6

The algorithm used in Theorem 6 is FMMD-S from [46], which is a $(5+\varepsilon)$-approximation algorithm for max-min diversification under group fairness constraints running in $O(mkn + m^k \log \frac{1}{\varepsilon})$ time. Therefore, ε in Theorem 6 directly inherits the error parameter in FMMD-S.

C.2 Error Parameter in Theorem 7

In the proof of Theorem 4, ε is an accuracy parameter in Algorithm 3. Therefore, ε in Theorem 7 is the accuracy parameter in Algorithm 3. In Section 5, ε is a small positive constant used to control the threshold for accepting local improvements. Thus, a smaller ε can potentially lead to better results while incurring higher computational overhead.

C.3 Error Parameter in Theorem 8

In the proof of Theorem 5, ε is an accuracy parameter in Algorithm 3. The algorithm used in Theorem 8 is from [6] with no other parameter involved in its approximation factor. Therefore, ε in Theorem 8 is just the accuracy parameter in Algorithm 3.

D Extended Theoretical Analysis for Max-Min Diversification

In Theorem 6, we show that the time complexity of our algorithm is exponential w.r.t. k, as it uses FMMD-S [46] as a subroutine. This arises because max-min diversification with individual fairness is at least as hard as its partition matroid-constrained version (Theorem 3). To date, however, no algorithm can simultaneously satisfy the following four requirements: (i) the solution exactly meets the partition matroid constraint; (ii) the time complexity is polynomial; (iii) the solution provides a constant-factor approximation guarantee for max-min diversification; and (iv) the algorithm works in general metric spaces.

To provide different trade-offs among these requirements, we show that our algorithm can work with different algorithms for max-min diversification with group fairness constraints. The main results are summarized as follows:

1. **FMMD-S [46]:** According to the concept of group fairness defined in [46], after executing Algorithm 2, one can directly apply FMMD-S by restricting the number of points selected in each individual fairness region to lie between 1 and $k - m + 1$. The solution achieves a 5-approximation while strictly satisfying group fairness constraints, as indicated in Theorem 6. However, FMMD-S relies on solving an ILP for solution computation and thus has an exponential time complexity w.r.t. k.
2. **Fair-Greedy-Flow [3]:** According to the concept of group fairness defined in [3], after executing Algorithm 2, one can directly apply Fair-Greedy-Flow by pre-specifying for each individual fairness region a constant $k_i \geq 1$ such that the sum of all k_i's is equal to k. This algorithm runs in $O(nkm^3)$ time but only provides an $O(m)$-approximation, where m is the number of individual fairness regions.
3. **MFD [25]:** According to the definition of group fairness in [25], after executing Algorithm 2, one can directly apply MFD by pre-specifying for each individual fairness region a constant $k_i \geq 1$ such that the sum of all k_i's is equal to k. This algorithm has a near-linear time complexity of $O(nk)$ and achieves a constant approximation. However, the algorithm is randomized and its solutions may not always satisfy the fairness constraints. In addition, MFD is specific to Euclidean space.

Each of these methods does not meet at least one of the four requirements. Considering the empirical performance of our algorithm, we chose the FMMD-S algorithm [46] in our implementation.