On Fair Epsilon Net and Geometric Hitting Set

Mohsen Dehghankar
University of Illinois Chicago
Chicago, IL, USA
mdehgh2@uic.edu

Stavros Sintos
University of Illinois Chicago
Chicago, IL, USA
stavros@uic.edu

Abolfazl Asudeh
University of Illinois Chicago
Chicago, IL, USA
asudeh@uic.edu

ABSTRACT

Fairness has emerged as a formidable challenge in data-driven decisions. Many of the data problems, such as creating compact data summaries for approximate query processing, can be effectively tackled using concepts from computational geometry, such as ε-nets. However, these powerful tools have yet to be examined from the perspective of fairness.

To fill this research gap, we add fairness to classical geometric approximation problems of ε-net, ε-sample, and geometric hitting set. We introduce and address two notions of group fairness: demographic parity, which requires preserving group proportions from the input distribution, and custom-ratios fairness, which demands satisfying arbitrary target ratios.

We develop two algorithms to enforce fairness—one based on sampling and another on discrepancy theory. The sampling-based algorithm is faster and computes a fair ε-net of size which is only larger by a $\log(k)$ factor compared to the standard (unfair) ε-net, where k is the number of demographic groups. The discrepancy-based algorithm is slightly slower (for bounded VC dimension), but it computes a smaller fair ε-net. Notably, we reduce the fair geometric hitting set problem to finding fair ε-nets. This results in a $O(\log \mathrm{OPT} \times \log k)$ approximation of a fair geometric hitting set.

Additionally, we show that under certain input distributions, constructing fair ε-samples can be infeasible, highlighting limitations in fair sampling. Beyond the theoretical guarantees, our experimental results validate the practical effectiveness of the proposed algorithms. In particular, we achieve zero unfairness with only a modest increase in output size compared to the unfair setting.

PVLDB Reference Format:
Mohsen Dehghankar, Stavros Sintos, and Abolfazl Asudeh. On Fair Epsilon Net and Geometric Hitting Set. PVLDB, 19(5): 1032 - 1045, 2026.
doi:10.14778/3796195.3796213

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/UIC-InDeXLab/FairNet.

Proceedings of the VLDB Endowment, Vol. 19, No. 5 ISSN 2150-8097.
doi:10.14778/3796195.3796213

This material is based upon work supported in part by the National Science Foundation under Award No. 2348919.

1 INTRODUCTION

1.1 Motivation

As algorithmic decisions continue to influence critical aspects of modern human life, from resource allocation and recommendation to hiring and even predictive policing, the need to ensure fairness in data-driven systems has become increasingly urgent.

Geometric approximation algorithms and notions such as ε-nets facilitate *Approximate Query Processing* (AQP) by providing compact data summaries, aka data representations, that preserve key properties of large datasets [47, 65]. These geometric tools are especially valuable due to their provable guarantees of approximation quality and computational efficiency. ε-nets, for example, are well-known as small subsets of data that guarantee to contain at least one sample from each range larger than ε, and hence can be used for fast query answering. To further clarify this, let us consider the following example for approximate database range-query processing.

Example 1. (Part 1) *Consider a map dataset T with two filtering attributes x: `long` and y: `lat`, and $n = 18$ tuples shown as points in Figure 1. The range predicates in this relation are in the form of axis-parallel rectangles. For example, r_1 in Figure 1 corresponds with the following SQL query:*

```
SELECT * FROM T
    WHERE -87.73<=long<=-85.5 AND 41.86<=lat<=41.94
```

In a very large setting, where the goal is to quickly identify a tuple matching the query, one can model the problem as a range space $(X, \mathcal{R})$ where X is the set of points in T and $\mathcal{R}$ is the universe of all axis-parallel rectangles.
Let $\varepsilon = \frac{5}{18}$. An ε-net on $(X, \mathcal{R})$ is a subset of T that guarantees to contain at least one point from any possible range with cardinality at least 5. The points highlighted in green form such an ε-net. For example, the rectangle r_1, which encompasses five points, has one point from the ε-net. Using only the set of points in the ε-net, one can quickly find a point satisfying any (sufficiently large) range query. For example, the highlighted point in the bottom-right of r_1 is the point in the ε-net satisfying the above SQL query. □

In § 7, we will illustrate some of the other applications of ε-nets, including k-Nearest Neighbors and top-k queries.

The traditional formulations of these problems, however, operate under the assumption of homogeneity in the data, neglecting the underlying (demographic) group structure that may be critical in fairness-sensitive applications. To better motivate this, let us consider Example 1 once more:

Example 1. (Part 2) *Suppose the tuples in Figure 1 belong to two demographic groups specified by the color of the points (i.e., `{blue,red}`). One can notice the selected points for ε-net mainly belong to the `red` group. As a result, answering range queries using this set will favor the red group by mostly returning a tuple from this group.* □

Example 1 highlights a concrete use case of query answering on map data for finding points of interest [2, 68, 92], where businesses such as restaurants are the tuples on the map. In particular, consider an interactive application, such as Yelp or Google Maps, where users

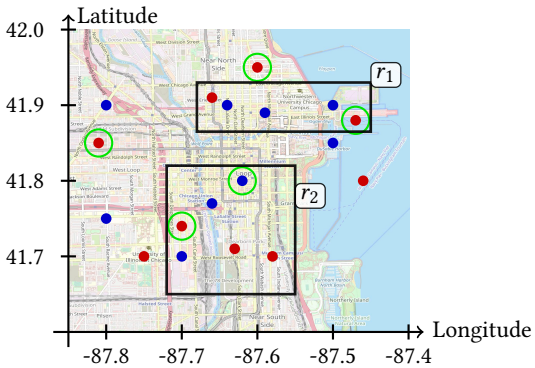

Figure 1: Illustration of a toy example on a Chicago map with an unfair ε-net, highlighted with green circles.

explore the map by panning and zooming to certain (rectangular) regions on their screen. Biases in the outputs of these applications against (businesses owned by) certain demographic groups is a well-known concern [86].

In order to make the UI interactive, such an application needs to quickly find some points that match the specified region [2, 28]. Using an ε-net as the dataset representative, the system can quickly find and display some points that match the query, without scanning the entire dataset. Now, if some groups are underrepresented in the ε-net (e.g., only a few black-owned restaurants are selected), those groups will have a lower exposure (e.g., the black-owned restaurants are highlighted disproportionately less on the maps), causing potentially less income for those groups.

A *fair* ε-net ensures that the selected points are representative of all groups, and using it for approximate query answering is not discriminatory against some groups.

Using standard geometric sampling in this example can introduce bias. As shown in Figure 1, the standard ϵ-net result (green annotated points) may capture mostly restaurants from the red group, particularly when that group is the majority of the dataset. This imbalance skews the recommendations toward the majority group, violating fairness principles in recommender systems [94]. Our fairness-aware algorithms address this issue by ensuring balanced group representation with **minimal overhead**. The randomized variant guarantees fairness with only an $O(\log k)$ increase in the sample size, where k is the number of groups, while the deterministic variant achieves the same goal with merely a constant-size increase. These results highlight an effective trade-off between fairness and efficiency; our approach forces equitable recommendations and full neighborhood coverage without significantly increasing the overall sample size.

1.2 Technical Contributions

In this paper, we introduce fairness on three foundational geometric approximation problems: the construction of ε-nets, ε-samples, and the geometric hitting set problem. Our goal is to enforce fairness constraints while preserving strong approximation guarantees.

We formalize our problems based on two group-fairness notions: (a) demographic parity (DP), which requires preserving the group ratios from the input data distribution, and (b) custom-ratios fairness (CR), which generalizes DP by allowing arbitrary target ratios for different groups.

Problem	Fairness Notion	Proposed Algorithms
Fair ε-net	Demographic Parity	Fair Monte-Carlo (§ 4.1)
		Fair Discrepancy-based (§ 4.2)
		Fair Sketch-and-merge (§ 4.2.2)
	Custom-ratios	Reduction to FGHS (§ 4.3)
Fair ε-sample	Demographic Parity	Fair Monte-Carlo (§ 4.4)
		Fair Sketch-and-merge (§ 4.4)
	Custom-ratios	Infeasible (§ 4.4)
Fair Geometric Hitting Set (FGHS)	Custom-ratios	Reduction to DP Fair ε-net (§ 5)

Table 1: Summary of the proposed fairness-aware algorithms under different fairness measures.

To address demographic parity in the construction of ε-nets, we propose two algorithms: a Monte Carlo Randomized algorithm based on sampling and a deterministic discrepancy-based algorithm. We further generalize both approaches to the weighted setting, where each point carries a weight and fairness is measured based on the total weight of each color group.

To address the custom-ratio fairness,[1] we reduce the fair ε-net problem to the Fair Geometric Hitting Set problem. To solve this, we develop a fair LP-based algorithm that constructs geometric hitting sets (equivalently, geometric set covers), satisfying both demographic parity and custom-ratio constraints. A summary of the proposed algorithms is provided in Table 1.

In addition to the theoretical analysis, we conduct comprehensive experimental evaluations on real-world and synthetic datasets on several applications of ε-nets and hitting sets. Our experiments verified our theoretical findings since our algorithms' outputs satisfied the fairness requirements while minimally increasing the output size compared to the regular (unfair) outputs. Our experiments further demonstrate the efficiency of our algorithms across diverse settings, including different dataset sizes, dimensionalities, numbers of demographic groups, and their distribution patterns.

Paper Organization. The remainder of the paper is structured as follows. § 7 discusses the applications of geometric sampling in database systems.§ 2 introduces the necessary preliminaries and formal definitions. § 3 provides an overview of classical results on ε-nets, ε-samples, and the Geometric Hitting Set problem. In § 4, we present our algorithms for constructing fair ε-nets, followed by a discussion on fair ε-samples in Sub§ 4.4. § 5 details our approach to the Fair Geometric Hitting Set problem. Related work is discussed in § 6. Finally, § 8 presents the experimental results of the algorithms.

2 PRELIMINARIES

In this section, we begin by introducing the necessary notation, followed by formal definitions of the preliminary concepts. We then provide a formulation of the problem studied in this paper.

2.1 Notations

Range spaces: A range space $(X, \mathcal{R})$ consists of a finite set of points X, where $|X| = n$, and each point p_i $(1 \leq i \leq n)$ lies in a d-dimensional space $\mathbb{R}^d$. Associated with X is a family of subsets $\mathcal{R}$, referred to as ranges, where each $R_j \in \mathcal{R}$ is a subset of X. While the

[1] It is important to note that any algorithm satisfying the more general custom-ratio (CR) fairness also satisfies demographic parity (DP); however, it may not do so efficiently.

set of ranges $\mathcal{R}$ may be infinite in general, we denote by $m = |\mathcal{R}|$ the number of ranges when $\mathcal{R}$ is finite.

The *dual* of a range space $(X, \mathcal{R})$ is another range space, denoted by $(\mathcal{R}, \mathcal{X})$, in which each range in $\mathcal{R}$ is treated as a point, and the new family of ranges $\mathcal{X}$ is defined as below:

$$\mathcal{X} = \{\mathcal{R}(p) \mid p \in X\},$$

where $\mathcal{R}(p) = \{R \in \mathcal{R} \mid p \in R\}$. That is, each point p in the original space defines a range in the dual space, consisting of all ranges in $\mathcal{R}$ that contain p.

Demographic groups: We assume a fixed set of demographic groups, represented as a finite set of colors $C = \{c_1, c_2, \ldots, c_k\}$. Each point $p_i \in X$ is associated with a color $c(p_i) \in C$. This induces a partition of the point set X into demographic groups: for each $c \in C$, let $X_c = \{p \in X \mid c(p) = c\}$ denote the subset of points of color c. Generally, for every subset $Y \subseteq X$, we set $Y_c = \{p \in Y \mid c(p) = c\}$ for each $c \in C$. We use the terms 'demographic group' and 'color' interchangeably.

Weights: In the weighted setting, each point $p_i \in X$ is associated with a weight denoted by w_i. For any subset $S \subseteq X$, the total weight of the points in S is given by $w(S) = \sum_{p_i \in S} w_i$.

2.2 Definitions

For the range space $(X, \mathcal{R})$, a subset $S \subseteq X$ is said to be *shattered* by $\mathcal{R}$ if, for every subset $T \subseteq S$, there exists a range $R \in \mathcal{R}$ such that $R \cap S = T$. Define $\mathcal{R}_{|T}$ as the set of ranges induced on T, in other words, $\mathcal{R}_{|T} = \{R \cap T \mid R \in \mathcal{R}\}$.

The *VC dimension* of the range space $(X, \mathcal{R})$, denoted by VC $(X, \mathcal{R})$, is the maximum size of a subset $S \subseteq X$ that can be shattered by $\mathcal{R}$. If arbitrarily large shattered subsets exist, the VC dimension is said to be infinite. We will use d as the VC dimension where the range space is clear from the context.

In the following, we formally define ε-net, ε-sample, and the Geometric Hitting Set problem [20, 47].

Definition 1 (ε-net). *Fix a value $\varepsilon \leq 1$. A subset $\mathcal{N} \subseteq X$ is called an ε-net of $(X, \mathcal{R})$, if for any range $R \in \mathcal{R}$ we have:*

$$\frac{|R|}{|X|} \geq \varepsilon \Longrightarrow |\mathcal{N} \cap R| \geq 0$$

Definition 2 (ε-sample). *Fix a value $\varepsilon \leq 1$. A subset $\mathcal{A} \subseteq X$ is called an ε-sample of $(X, \mathcal{R})$, if for any range $R \in \mathcal{R}$ we have:*

$$\left| \frac{|\mathcal{A} \cap R|}{|\mathcal{A}|} - \frac{|X \cap R|}{|X|} \right| \leq \varepsilon$$

Definition 3 (Geometric Hitting Set). *Given a range space $(X, \mathcal{R})$ with a bounded VC-dimension. The Geometric Hitting Set problem asks to find the smallest subset of points $X^* \subseteq X$ to hit all the ranges in $\mathcal{R}$, i.e.:* $\forall R \in \mathcal{R},\ \exists p \in X^*$ *such that* $p \in R$

X^* is called the smallest hitting set of $(X, \mathcal{R})$.

The dual of this problem, formulated over the dual range space, is referred to as the Geometric Set Cover problem (where the dual VC-dimension is bounded). Consequently, most of the results presented in this paper for Geometric Hitting Set also extend to the Set Covering setting.

2.3 Problems

We now introduce fair variants of the problems by incorporating fairness constraints into their definitions.

Definition 4 (Fair ε-net). *Given a range space $(X, \mathcal{R})$, a finite set of colors $C = \{c_1, \ldots, c_k\}$ representing k groups, a parameter $\varepsilon \in (0, 1)$, and a vector $\mathcal{T} = (\tau_1, \tau_2, \cdots, \tau_k)$ of k ratios for the groups, such that for every $l \in [k]$, $\tau_l \leq 1$ and $\sum_{l \in [k]} \tau_l = 1$, the goal is to compute a set $\mathcal{N} \subseteq X$ of minimum size such that $\mathcal{N}$ is an ε-net of $(X, \mathcal{R})$ and for every $l \in [k]$,* $\dfrac{|\mathcal{N} \cap X_{c_l}|}{|\mathcal{N}|} = \tau_l.$

The Fair ε-sample problem can be defined similarly.

Definition 5 (Fair ε-sample). *Given the same input as in the Fair ε-net problem, the goal is to compute a set $\mathcal{A} \subseteq X$ of minimum size such that $\mathcal{A}$ is an ε-sample of $(X, \mathcal{R})$ and for every $l \in [k]$,*

$$\frac{|\mathcal{A} \cap X_{c_l}|}{|\mathcal{A}|} = \tau_l.$$

Definition 6 (Fair Geometric Hitting Set, FGHS). *Given a range space $(X, \mathcal{R})$ with a bounded VC-dimension and the color ratios $\mathcal{T} = (\tau_1, \tau_2, \cdots, \tau_k)$ defined similarly to the Fair ε-net problem. The Fair Geometric Hitting Set (FGHS) problem[2] asks for the smallest subset $X^* \subseteq X$ satisfying the following conditions:*

(1) X^* is **fair** *with respect to $\mathcal{T}$, i.e., $\frac{|X^* \cap X_{c_l}|}{|X^*|} = \tau_l, \forall l \in [k]$.*
(2) X^* hits all the ranges in $\mathcal{R}$.

2.4 Fairness

In all the above problems, we may either address the *demographic parity* or *custom-ratio* fairness constraints defined as below.

We say that a subset of points $S \subseteq X$ satisfies **demographic parity (DP)** if, for each color, it maintains the same ratio as the ground set X. In other words, the ratios $\mathcal{T}$ are defined as

$$\tau_l = \frac{|X_{c_l}|}{|X|}, \quad \forall l \in [k].$$

The same definition is valid for a subset of ranges (instead of points) if the demographic groups are defined on ranges.

In a more general setting, we consider any arbitrary ratios $\mathcal{T}$ and try to satisfy the fairness accordingly; we call it **custom-ratio (CR)** constraint. However, it is sometimes impossible to satisfy *any ratio* in some cases.

Example 2. *Consider a range space consisting of 10 points, where 9 are blue and only 1 is red. Suppose that any valid hitting set must contain more than two points. In this case, it is impossible to satisfy a 50%-50% color ratio in any hitting set, as there are insufficient red points to meet the fairness constraint.*

Demographic parity, also known as statistical parity, is a natural and widely adopted notion of fairness in data management and machine learning literature [15, 21, 35, 84, 85, 104]. It ensures to maintain the same group proportions as in the original dataset.

Custom-ratio fairness generalizes the fairness constraints beyond maintaining the dataset ratios. This enables the enforcement

[2]Note that the Geometric Set Cover and Geometric Hitting Set problems are equivalent by duality. The solutions we propose for Fair Hitting Set naturally extend to *Fair Geometric Set Cover (FGSC)*, where the groups are defined over the sets.

of fairness based on social norms, to, for example, reverse historical discriminations reflected in data [39]. For instance, consider the businesses such as restaurants as the data points in Example 1. Suppose that due to various biases, group distributions among the business owners is different from their underlying distribution in the society. This is a well-known issue for black-owned businesses in the US [17]. In such cases, the custom-ratio fairness allows to move the group distributions closer to the ones in the society. Overall, custom-ratio fairness offers greater flexibility in defining fairness. This broader formulation has also been explored as a general fairness framework in recent works [31, 42, 43].

3 BACKGROUND

In this section, we briefly review classical results related to the problems defined in Section 2.2. References to this background will be made in subsequent sections as needed.

ε-nets and ε-samples: A classical approach to constructing ε-nets is based on random sampling:

Theorem 1 (ε-net [47]). *Given a range space $(X, \mathcal{R})$ with bounded VC dimension d. A random sample (with replacement) of size $\lambda \geq \max\left(\frac{4}{\varepsilon}\log\frac{4}{\varphi}, \frac{8d}{\varepsilon}\log\frac{16}{\varepsilon}\right)$ is an ε-net with probability at least $1-\varphi$.*

A similar result characterizes the number of random samples required to construct an ε-sample:

Theorem 2 (ε-sample [91]). *Given a range space $(X, \mathcal{R})$ with bounded VC-dim d. A random sample (with replacement) of size $\gamma \geq \frac{c_0}{\varepsilon^2}\left(d\log\frac{d}{\varepsilon} + \log\frac{1}{\varphi}\right)$ is an ε-sample with probability at least $1-\varphi$, for a large enough constant c_0.*

Deterministic constructions of ε-nets and ε-samples can also be achieved via discrepancy methods [25, 26], which we discuss in the relevant section, in the context of building deterministic fair ε-nets.

Geometric Hitting Set: The Hitting Set and Set Cover problems are dual and algorithmically equivalent, which is known to be NP-hard. In the general (non-geometric) setting, a standard greedy algorithm—which iteratively selects the point that hits the most (remaining) ranges—yields a $\log n$ approximation. However, in geometric settings with bounded VC dimension d, improved algorithms achieve a $\log d$ approximation factor.

To achieve this improvement, one approach uses the multiplicative weight update method [5, 44], while another uses LP relaxation, reducing the Hitting Set problem to an instance of the *weighted* ε-net problem [22, 58].

Assumptions. Similarly to [25], we assume that $n = |X| = 2^{\xi}$ for a positive integer ξ, i.e., the number of points in X is a power of 2. Furthermore, we make the mild assumption that for every $c \in C$, $|X_c| = 2^{\xi_c}$, where ξ_c is a positive integer. Our goal in all cases is to construct fair ε-nets of small size – ideally comparable to the size of unfair ε-nets. In order to achieve this goal, we assume that $\tau_\ell \cdot \lambda \geq 1$, for every $\ell \in [k]$. If these inequalities do not hold, then the size of a fair ε-net might be much larger (even k times larger) than the size of the (unfair) ε-net. Even if this assumption does not hold, then all our algorithms are correct, however the approximation factor and the running time of some of our algorithms might increase by a factor $\frac{1}{\lambda \cdot \min_{\ell\in[k]}\tau_\ell}$.

Aspect	Fair Monte-Carlo	Fair Discrepancy-based
Size Increase Factor	$O(\log k)$	Constant $O(1)$
Running time	$O(n)$	$O(m \cdot n \log n)$ for simple discrepancy
		$O\left(n \cdot \frac{d^{3d}}{\varepsilon^{2d}} \log^d\left(\frac{d}{\varepsilon}\right)\right)$ for Sketch-and-Merge
Weighted Case	Similar guarantees to unweighted	Very slow (depends on $\sum_i w_i$)
Determinism	Monte Carlo Randomized (always fair if successful)	Deterministic if $\mathcal{R}$ is materialized (always fair)
Implementation	Easier to implement	More complex

Table 2: High-level comparison of fair Monte-Carlo sampling-based and fair discrepancy-based algorithms for addressing DP in ε-nets.

4 FAIR ε-NETS

In this section, we discuss our algorithms for the Fair ε-net and ε-sample problems. First, in Sections 4.1 and 4.2, we address fair ε-net under the demographic parity, followed by the custom-ratios fairness in § 4.3. We will then study fair ε-sample in § 4.4.

We propose two categories of algorithms for constructing fair ε-nets that satisfy demographic parity: a *Monte-Carlo sampling-based* and a *discrepancy-based* algorithm. The sampling-based approach returns a larger fair ε-net efficiently, while the discrepancy-based method algorithms return a smaller fair ε-net at the expense of increased runtime. We design two discrepancy-based algorithms. While they both return a fair ε-net of (asymptotically) the same size, the second one is more efficient when d is small and m is large. A comparative summary of all our algorithms is presented in Table 2.

4.1 A Monte-Carlo Randomized Algorithm for Satisfying Demographic Parity

Our first algorithm is a sampling-based Monte Carlo randomized algorithm that finds a fair ε-net satisfying demographic parity with a high probability. At a high level, the algorithm is developed based on the observation that a random sample from X should be "near-fair" for each color c_i, in the sense that the ratio of samples from c_i in the sample set should be close to the required ratio τ_i. Hence, with a high probability, an $O(\log k)$-factor increase in the size of the set is enough to satisfy the demographic parity ratios.

Algorithm. Draw a set $\mathcal{N}$ of $\lambda \geq \max\{\frac{4}{\varepsilon}\log\frac{4}{\varphi/2}, \frac{8d}{\varepsilon}\log\frac{16}{\varepsilon}\}$ uniform random samples (with replacement) from X. Let $v = 2\ln(k \cdot \frac{1}{\varphi/2})$. For every color $c \in C$ add $\max\{(1+v)\cdot\frac{|X_c|}{|X|}\cdot\lambda - |\mathcal{N}_c|, 0\}$ arbitrary points of color c to $\mathcal{N}$. Let $\mathcal{S}$ be the new set after we traversed all colors $c \in C$. Return $\mathcal{S}$. A pseudo-code of this algorithm is presented in Algorithm 1.

Correctness and runtime analysis.

Lemma 3. *For every color $c \in C$, $|\mathcal{N}_c| \leq (1+v)\cdot\frac{|X_c|}{|X|}\cdot\lambda$, with probability at least $1-\varphi/2$.*

Proof. This is a result of applying Chernoff's inequality on the number of points sampled from each color. See Technical Report [32] for complete proof. □

Lemma 4. *The set $\mathcal{S}$ is a fair ε-net of $(X, \mathcal{R})$ and $|\mathcal{S}| = O((1+v)\lambda)$, with probability at least $1-\varphi$.*

Proof. This is a result of applying the well known Theorem 1 and Lemma 3. See Technical Report [32] for detailed proof. □

Lemma 5. *The algorithm's time complexity is $O(|X|)$.*

Algorithm 1 Fair Monte-Carlo (FMC) algorithm for building an ε-net satisfying DP fairness.

Require: Range space $(X, \mathcal{R})$, Epsilon ε.
Ensure: The DP fair ε-net $\mathcal{S}$, found by sampling.

function FMC($X, \mathcal{R}, \varepsilon$)
  $v \leftarrow 2\ln(k \cdot \frac{1}{\varphi/2})$
  $\mathcal{N} \leftarrow$ a random subset of size λ from X
  $\mathcal{S} \leftarrow copy(\mathcal{N})$ ▷ Initialize a new set for fair ε-net.
  for All colors $c \in C$ **do**
    $Tmp \leftarrow$ arbitrary $(1+v)\frac{|X_c|}{|X|}\lambda - |\mathcal{N}_c|$ points from X_c
    $\mathcal{S} \leftarrow \mathcal{S} \cup Tmp$
  return $\mathcal{S}$ ▷ This is a fair ε-net with probability $\geq 1-\varphi$.

Proof. We place all points from X in a table. We sample λ indexes and we add the corresponding points in $\mathcal{N}$. Then we go through each element in the table and we add it in $\mathcal{N}$ if needed in $O(1)$ time. □

Theorem 6. *Given a range space $(X, \mathcal{R})$, with $|X| = n$, VC dimension d, DP constraints $\mathcal{T}$, and parameters $\varepsilon, \varphi \in (0,1)$, there exists a randomized algorithm that constructs a fair ε-net with respect to $\mathcal{T}$ of size $O(\frac{1}{\varepsilon}\max\{\log\frac{1}{\varphi}, d\log\frac{1}{\varepsilon}\}\cdot\log(\frac{k}{\varphi}))$ with probability at least $1-\varphi$, in $O(n)$ time, where k is the number of different colors in set X.*

4.1.1 Weighted fair ε-net. Our randomized algorithm can be extended to the fair ε-net problem over a weighted set of points. Assume that each point $p_i \in X$ is associated with a weight w_i. Let $W_c = \sum_{p_i \in X_c} w_i$ be the sum of weights of points with color $c \in C$. For simplicity and without loss of generality, we assume that $\sum_{c\in C} W_c = 1$. A weighted ε-net is a subset $\mathcal{S} \subseteq X$ such that for every $R \in \mathcal{R}$ with $\sum_{p_i \in R} w_i \geq \varepsilon$, it holds that $R \cap \mathcal{S} \neq \emptyset$. In terms of fairness, the DP ratios $\mathcal{T}$ are defined as $\tau_\ell = W_{c_\ell}$, for each $\ell \in [k]$. The goal is to compute a set $\mathcal{S} \subseteq X$ such that, for every $R \in \mathcal{R}$ with $\sum_{p_i \in R} w_i \geq \varepsilon$, it holds that $R \cap \mathcal{S} \neq \emptyset$ (weighted ε-net) and for every $c_\ell \in C$, $\frac{|\mathcal{S}_{c_\ell}|}{|\mathcal{S}|} = \tau_\ell$. As we had in the unweighted case, we assume for simplicity that the ratios in $\mathcal{T}$ are chosen in that way such that $W_{c_\ell} \cdot \lambda = \tau_\ell \cdot \lambda \geq 1$.

It is known [47], that a set of λ random samples from X with respect to the weights of the points returns a weighted (unfair) ε-net. Hence, intuitively, we can extend the algorithm from the unweighted case to the weighted case, as follows.

Algorithm. Draw a set $\mathcal{N}$ of λ random samples with respect to the weights of the points from X. Let $v = 2\ln(k \cdot \frac{1}{\varphi/2})$. For every color $c \in C$, add $\max\{(1+v)\cdot\frac{|X_c|}{|X|}\cdot\lambda - |\mathcal{N}_c|, 0\}$ arbitrary points of color c to $\mathcal{N}$. Let $\mathcal{S}$ be the new set after we traversed all colors $c \in C$. Return $\mathcal{S}$.

Using the same arguments as in Lemma 3, we can show that $\mathbb{P}\left(\exists c \in C,\ |\mathcal{N}_c| \geq (1+v)\frac{|X_c|}{|X|}\cdot\lambda\right) \leq \frac{\varphi}{2}$. Skipping the details, we conclude with the following theorem.

Theorem 7. *Given a range space $(X, \mathcal{R})$ over a weighted set of points X, with $|X| = n$, VC dimension d, DP constraints $\mathcal{T}$, and parameters $\varepsilon, \varphi \in (0,1)$, there exists a randomized algorithm that constructs a weighted fair ε-net with respect to $\mathcal{T}$ of $O\left(\frac{1}{\varepsilon}\max\{\log\frac{1}{\varphi}, d\log\frac{1}{\varepsilon}\}\cdot\log\frac{k}{\varphi}\right)$ size with probability at least $1-\varphi$, where k is the number of different colors in set X. The time complexity of the algorithm is $O(n)$.*

4.2 Discrepancy-based Algorithms for Satisfying Demographic Parity

The Monte-Carlo sampling-based algorithm proposed in the previous section, while being efficient, returns a fair ε-net, which is larger than standard ε-nets by roughly a $\log(k)$ factor. Besides, similar to other randomized Monte-Carlo algorithms, it does not always find a valid solution. In this section, we propose two fair discrepancy-based algorithms to address these issues; we propose deterministic methods that construct smaller fair ε-nets. Such results come at the cost of increased running time. While both discrepancy-based algorithms return a fair ε-net of (asymptotically) the same size, the first one is more efficient when m is small and d is large, while the second one is more efficient when d is small and m is large.

4.2.1 Discrepancy-based algorithm for fair ε-net. We first describe the notion of discrepancy to construct an ε-net. After that, we adjust this method to construct a **fair** ε-net. More details on the definitions and the proofs relating to the standard ε-net construction using discrepancy can be found in [25, 44].

Before we start the description of our algorithm, we give some useful definitions. Let $(X, \mathcal{R})$ be a range space with $|X| = n$ and $|\mathcal{R}| = m$. Define a coloring function $\kappa : X \to \{-1, +1\}$ that assigns a color $+1$ or -1 to each point in X. The coloring κ differs from the demographic groups and should not be mistaken with colors/groups $C = \{c_1, c_2, \cdots, c_k\}$. The *discrepancy* of κ over a range $R \in \mathcal{R}$ is defined as: $|\kappa(R)| = \left|\sum_{p\in R}\kappa(p)\right|$. The *discrepancy* of the function κ is defined as: $\text{disc}(\kappa) = \max_{R\in\mathcal{R}}|\kappa(R)|$. Namely, it is the color difference in the most unbalanced range based on coloring $\kappa(\cdot)$. The discrepancy of the range space $(X, \mathcal{R})$ is defined as the best achievable discrepancy by a coloring function, $\text{disc}(X) = \min_{\kappa: X\to\{-1,+1\}}\text{disc}(\kappa)$.

A *matching* Π of X is a set of pairs $\{(p_i, p_j) \mid p_i, p_j \in X\}$ that partitions the entire set X into $n/2$ pairs. A coloring κ is said to be *compatible* with a matching Π, if for any pair $(p_i, p_j) \in \Pi$, we have: $\kappa(p_i) + \kappa(p_j) = 0$.

The following procedure defines a major step in building ε-nets using this method:

Random Halving. Let Π be an *arbitrary* matching on X. For each pair $(p_i, p_j) \in \Pi$, randomly either color p_i as $+1$ and p_j as -1, or the other way around, by tossing a fair coin. Then, without loss of generality, only keep the points with color $+1$. Let $X^{(1)}$ be the result set with $|X^{(1)}| = \frac{n}{2}$.

The next lemma that uses the Random Halving procedure is proven in [44].

Lemma 8. *We are given a range space $(X, \mathcal{R})$ with $|X| = n$ and $|\mathcal{R}| = m$. Let Π be an arbitrary matching of X and let κ be the coloring constructed by applying the procedure Random Halving. With a probability more than $\frac{1}{2}$, we have:*

$$\forall R \in \mathcal{R}, \quad |\kappa(R)| \leq \Delta = \sqrt{2n\ln(4m)}.$$

The construction of Lemma 8 can be made deterministic by the method of *conditional expectations* [44], assuming that we have access to all the ranges in $\mathcal{R}$:

Lemma 9. *Given a range space $(X, \mathcal{R})$ with $|X| = n$ and $|\mathcal{R}| = m$. Let Π be an arbitrary matching on X. There exists a deterministic*

method Halving$(X, \mathcal{R}, \Pi)$ *that constructs a coloring function* $\kappa : X \to \{-1, 1\}$ *which is compatible with matching* Π *in* $O(n \cdot m)$ *time, such that*

$$\forall R \in \mathcal{R},\ |\kappa(R)| \le \Delta.$$

Using the notion and properties of discrepancy, we describe our algorithm for constructing a fair ε-net.

Algorithm. Our algorithm works in iterations $i = 1, \ldots, U$, where $U = \log \frac{n}{c_0 \frac{d}{\varepsilon} \log \frac{d}{\varepsilon}}$ and c_0 is a sufficiently large constant. Initially, $X^{(0)} = X$. In the i-th iteration, we construct a *fair matching* $\Pi_f^{(i-1)}$ on $X^{(i-1)}$, as follows. Let $Y = X^{(i-1)}$. For each pair $p_{j_1}, p_{j_2} \in Y$ with $c(p_{j_1}) = c(p_{j_2})$, we add the pair (p_{j_1}, p_{j_2}) in $\Pi_f^{(i-1)}$ and remove p_{j_1}, p_{j_2} from Y. We repeat the same pairing procedure until $Y = \emptyset$. Then, we run the halving procedure from Lemma 9, executing Halving $\left(X^{(i-1)}, \mathcal{R}_{|X^{(i-1)}}, \Pi_f^{(i-1)}\right)$. Let $X^{(i)}$ be the points with color +1 returned by the halving procedure. In the end, we return $X^{(U)}$.

Correctness and runtime analysis. We first show that the set returned by our algorithm is an ε-net of small size. Then we show that this is also a fair ε-net.

Lemma 10. *Our algorithm returns an ε-net of size $O(\frac{d}{\varepsilon} \log \frac{d}{\varepsilon})$.*

Proof. This is inspired by the known result from [25] and applying Lemma 9. See Technical Report [32] for complete proof. □

Lemma 11. *Our algorithm returns a fair ε-net.*

Proof. This comes from the fact that the halving steps, maintain the ratios from each color. See Technical Report [32] for complete proof. □

Lemma 12. *The running time of our algorithm is $O(n \cdot m \cdot \log n)$.*

Proof. In each iteration i of our algorithm, we construct a fair matching in $O(|X^{(i-1)}|) = O(n)$ time by making a pass over all points in $X^{(i-1)}$. The Halving procedure from Lemma 9 runs in $O(|X^{(i-1)}| \cdot |\mathcal{R}_{|X^{(i-1)|}}|) = O(n \cdot m)$ time. We execute $U = O(\log n)$ iterations of the algorithm so the total running time is $O(n \cdot m \cdot \log n)$. □

Putting everything together, we conclude with Theorem 13.

Theorem 13. *Given a range space $(X, \mathcal{R})$, with $|X| = n$, $|\mathcal{R}| = m$, VC dimension d, DP constraints $\mathcal{T}$, and a parameter $\varepsilon \in (0, 1)$, there exists a deterministic algorithm that constructs a fair ε-net with respect to $\mathcal{T}$, of size $O(\frac{d}{\varepsilon} \log \frac{d}{\varepsilon})$ in $O(n \cdot m \cdot \log n)$ time.*

A comparison of the two algorithms, discrepancy-based and sampling-based, is provided in Table 2.

Weighted case: If the points X have weights, a simple approach would be to replicate each point according to its weight w_i (assuming all the weights are integers; otherwise, we multiply them by a large value). However, the running time of this method would depend on $\sum_i w_i$ (integer weights), which can be significantly larger than n. The same argument is valid for space usage.

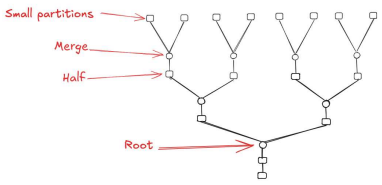

Figure 2: Sketch-and-Merge approach on a partition tree.

4.2.2 Discrepancy with Sketch-and-Merge. In this subsection, we discuss an alternative deterministic algorithm that is more efficient than the algorithm from Theorem 13 if the VC dimension is small. At a high level, the algorithm works as follows. We first construct a fair ε-sample of small size satisfying DP constraints using a hierarchical bottom-up approach (resulting in a tree structure visualized in Figure 2). Then we apply the algorithm from Theorem 13 on the fair ε-sample set to construct a fair ε-net. The algorithm is faster because, this time, the expensive algorithm from Theorem 13 is executed on a small set, i.e., a fair ε-sample. The correctness follows from the fact that an ε_1-net of a ε_2-sample of $(X, \mathcal{R})$, is an $(\varepsilon_1 + \varepsilon_2)$-net of $(X, \mathcal{R})$ [25].

Algorithm 2 Fair Sketch-and-Merge algorithm (FSM) for Finding an ε-net satisfying DP fairness.

Require: Range space $(X, \mathcal{R})$, Epsilon ε, the value 2^p for the size of small partitions.
Ensure: The DP fair ε-net $\mathcal{S}$, found by sketch-and-merge.

function FSM$(X, \mathcal{R}, \varepsilon, 2^p)$
  $\mathcal{P} \leftarrow$ partition X to $\frac{n}{2^p}$ equally-sized **fair** subsets.
  $\mathcal{L} \leftarrow \mathcal{P}$ ▷ All nodes in current level.
  while $|\mathcal{L}| > 1$ **do** ▷ Until reaching root
    $\mathcal{L}' \leftarrow [\ ]$ ▷ placeholder of the next level.
    for *siblings* (v_1, v_2) *in* $\mathcal{L}$ with parent u **do**
      $X'_u \leftarrow X_{v_1} \cup X_{v_2}$ ▷ Merge each pair of siblings
      $\Pi_f^{(u)} \leftarrow$ a fair matching constructed on X'_u
      $X_u \leftarrow$ Halving$(X'_u, \mathcal{R}_{|X'_u}, \Pi_f^{(u)})$
      $\mathcal{L}'.append(u)$
    $\mathcal{L} \leftarrow \mathcal{L}'$
  root $\leftarrow \mathcal{L}[0]$
  $\widehat{\mathcal{X}} \leftarrow X_{\text{root}}$
  while $|\widehat{\mathcal{X}}| > c_0 \frac{4d}{\varepsilon^2} \log \frac{2d}{\varepsilon}$ **do** ▷ Until finding $\frac{\varepsilon}{2}$-sample
    Construct fair matching Π_f on $\widehat{\mathcal{X}}$
    $\widehat{\mathcal{X}} \leftarrow$ Halving$(\widehat{\mathcal{X}}, \mathcal{R}_{|\widehat{\mathcal{X}}}, \Pi_f)$
  $\mathcal{X} \leftarrow \widehat{\mathcal{X}}$ ▷ The $\frac{\varepsilon}{2}$-sample.
  $\mathcal{S} \leftarrow$ Execute algorithm from Theorem 13 on $(\mathcal{X}, \mathcal{R}_{|\mathcal{X}})$
  return $\mathcal{S}$

Algorithm. Let p be a small parameter value that is set later. Let $\mathcal{P}$ be the partitioning of X into $\frac{n}{2^p}$ equally-sized fair subsets. Each partition $P \in \mathcal{P}$ contains 2^p points of the same color $c \in C$. We construct an empty binary partition tree with $\frac{n}{2^p}$ leaf nodes. For a node v of the partition tree, let $X_v \subseteq X$ be the points stored in node v. Initially, $X_v = \emptyset$ for every node v in the partition tree. Each partition $P \in \mathcal{P}$ is assigned to a leaf node v_P in the partition tree and $X_{v_P} = P$. Let $\mathcal{L}_j$ be the set of nodes in the j-th level of the tree. Initially, $\mathcal{L}_0 = \{v_P \mid P \in \mathcal{P}\}$ be the set of leaf nodes of the partition tree. We repeat the following until we reach the root of the tree. For every pair of sibling nodes v_1, v_2 in $\mathcal{L}_j$ with

parent $u \in \mathcal{L}_{j+1}$ we define $X'_u = X_{v_1} \cup X_{v_2}$ (Merging step). We construct a fair matching $\Pi_f^{(u)}$ on X'_u as we had in the algorithm of § 4.2.1, and we execute Halving$(X'_u, \mathcal{R}_{|X'_u}, \Pi_f^{(u)})$. Let X_u be the set of points from X'_u with color +1 returned by the halving procedure (Halving step). Let root be the root of the partition tree and $X^{(\text{root})}$ the points stored in it following the procedure above. We set $\widehat{\mathcal{X}} = X_{\text{root}}$. While $|\widehat{\mathcal{X}}| > c_0 \frac{4d}{\varepsilon^2} \log(\frac{2d}{\varepsilon})$, we repeat the following recursive approach. We construct a fair matching Π_f on $\widehat{\mathcal{X}}$ and we execute Halving$(\widehat{\mathcal{X}}, \mathcal{R}_{|\widehat{\mathcal{X}}}, \Pi_f)$. Let $\widehat{\mathcal{X}}$ be the set of points with color +1 returned by the halving procedure. The first time that we find $|\widehat{\mathcal{X}}| < c_0 \frac{4d}{\log} \frac{2d}{\varepsilon}$, we stop the repetitions and we set $\mathcal{X} = \widehat{\mathcal{X}}$. In the end, we run the algorithm from Theorem 13 on the range space $(\mathcal{X}, \mathcal{R}_{|\mathcal{X}})$ with parameter $\varepsilon/2$, and let $\mathcal{S}$ be the returned set of points. We return $\mathcal{S}$. We show the pseudocode of our algorithm in Algorithm 2. The overall structure of our algorithm using a partition is shown in Figure 2.

Correctness and runtime analysis.

Lemma 14. *The set $\mathcal{X}$ is a fair $\frac{\varepsilon}{2}$-sample of $(X, \mathcal{R})$.*

Proof. We show that $\mathcal{X}$ is both an $\frac{\varepsilon}{2}$-sample and a fair sample. For the proof, please refer to the Technical Report [32]. □

Lemma 15. *The returned set $\mathcal{S}$ is a fair ε-net of $(X, \mathcal{R})$.*

Proof. It is known from [25] that an ε_1-net of a ε_2-sample of $(X, \mathcal{R})$, is an $(\varepsilon_1 + \varepsilon_2)$-net of $(X, \mathcal{R})$. Similarly, it is straightforward to see that a fair ε_1-net of a fair ε_2-sample of $(X, \mathcal{R})$, is a fair $(\varepsilon_1 + \varepsilon_2)$-net of $(X, \mathcal{R})$. From Lemma 14, we know that $\mathcal{X}$ is a fair $\frac{\varepsilon}{2}$-sample of $(X, \mathcal{R})$. The algorithm from Theorem 13 is executed on $(\mathcal{X}, \mathcal{R}_{|\mathcal{X}})$ with parameter $\frac{\varepsilon}{2}$, and it returns the set $\mathcal{S}$ which is a fair $\frac{\varepsilon}{2}$-net of $(\mathcal{X}, \mathcal{R}_{|\mathcal{X}})$. Hence, $\mathcal{S}$ is also a fair ε-net of $(X, \mathcal{R})$. □

Lemma 16. *The running time of the algorithm is*

$$O\left(n \cdot \frac{d^{3d}}{\varepsilon^{2d}} \log^d\left(\frac{d}{\varepsilon}\right)\right)$$

.

Proof. This comes from the tree partitioning of the points. We refer the reader to the Technical Report [32] for more details. □

Putting everything together, we conclude with Theorem 17.

Theorem 17. *Given a range space $(X, \mathcal{R})$, with $|X| = n$, VC dimension d, DP constraints $\mathcal{T}$, and a parameter $\varepsilon \in (0, 1)$, there exists a deterministic algorithm that constructs a fair ε-net with respect to $\mathcal{T}$, of size $O\left(\frac{d}{\varepsilon} \log \frac{d}{\varepsilon}\right)$ in $O\left(n \cdot \frac{d^{3d}}{\varepsilon^{2d}} \log^d\left(\frac{d}{\varepsilon}\right)\right)$ time.*

4.3 Satisfying Custom-Ratio Fairness

In this section, we discuss the second, more general, fairness definition where the user provides the ratios $\mathcal{T} = \{\tau_1, \tau_2, \cdots, \tau_k\}$ and requires the returned ε-net $\mathcal{S}$ to satisfy $\frac{|\mathcal{S}_{c_i}|}{|\mathcal{S}|} = \tau_i$ for every $c_i \in C$.

Through the whole subsection, we assume that we have access to the ranges $\mathcal{R}$; in other words, the ranges are explicitly given to us as input. This assumption is valid in most cases; for example, the discrepancy approach to deterministically find an ε-net is based on this assumption [25, 44]. Assuming that we have access to the ranges $\mathcal{R}$ and $|\mathcal{R}| = m$, finding an ε-net of the range space $(X, \mathcal{R})$ can be reduced to the *hitting set* problem. In other words, based on the Definition 1, $\mathcal{N}$ is an ε-net of $(X, \mathcal{R})$, if and only if it hits all the ranges in $\mathcal{R}_\varepsilon = \{R \mid R \in \mathcal{R}, \ \frac{|R|}{|X|} \geq \varepsilon\}$.

Given the range space $(X, \mathcal{R})$, one can first filter out the light ranges and keep only the *heavy* ranges, aka $\mathcal{R}_\varepsilon$. Then, compute a hitting set for the new range space $(X, \mathcal{R}_\varepsilon)$. Any algorithm solving the hitting set problem would result in solving the ε-net problem.

$$\varepsilon\text{-net} \xrightarrow[\text{over } \mathcal{R}_\varepsilon]{\text{reduction}} \text{Hitting Set}$$

In the fairness context, any fair hitting set given the constraints $\mathcal{T}$ is also a fair ε-net. Solving the Fair Hitting Set problem is discussed in [31] in the general setting. They provide a randomized $O(\log n)$-approximation algorithm for the fair set cover that runs in $O(\text{poly}(n, m))$ time. The approximation factor holds in expectation. The algorithm in [31] is a greedy algorithm that picks the most promising points at each step. As a result, by applying the same algorithm, we can compute a **fair** ε-net for the range space $(X, \mathcal{R})$ with expected size $O(\text{OPT} \cdot \log n)$, where OPT is the fair ε-net with the minimum size. We note that this algorithm does not find a fair ε-net with an absolute size as we had in Theorems 6, 13, and 17. Instead, our algorithm for custom-ratios returns a fair ε-net with size larger by an (expected) $O(\log n)$ factor from the smallest possible fair ε-net with respect to the CR constraints $\mathcal{T}$.

Theorem 18. *Given a range space $(X, \mathcal{R})$, with $|X| = n$, $|\mathcal{R}| = m$, CR constraints $\mathcal{T}$, and a parameter $\varepsilon \in (0, 1)$, there exists a randomized algorithm that constructs a fair ε-net with respect to $\mathcal{T}$, of expected size $O(\text{OPT} \cdot \log(n))$ in $O(\text{poly}(n, m))$ time, where* OPT *is the size of the smallest fair ε-net with respect to $\mathcal{T}$.*

The algorithm from Theorem 18 works in general range spaces; however, assuming a range space $(X, \mathcal{R})$ with a bounded VC-dim would result in better approximation algorithms [22, 27, 40, 56]. The Fair Geometric Hitting Set (FGHS) problem is discussed in detail in § 5. Here, we only use the final result we get for the FGHS problem and use it to derive an approximation algorithm for constructing a fair ε-net satisfying CR constraints $\mathcal{T}$. In Theorem 22, we show an $O\left(\max\{\log \frac{1}{\varphi}, d \log(\text{OPT}_{FGHS})\} \cdot \log(\frac{k}{\varphi})\right)$-approximation algorithm (with probability at least $1 - \varphi$) for the Fair Geometric Hitting Set problem (on CR constraints), where OPT_{FGHS} is the optimum solution. By leveraging the algorithm from Theorem 22 on $(X, \mathcal{R}_\varepsilon)$, we return a fair ε-net $\mathcal{S}$ with respect to CR constraints $\mathcal{T}$ of size $O\left(\max\{\log \frac{1}{\varphi}, d \log(\text{OPT})\} \cdot \log(\frac{k}{\varphi}) \cdot \text{OPT}\right)$ with probability at least $1 - \varphi$. The algorithm's time complexity is $O(n \cdot m + \mathcal{M}(n, m + k))$, where $\mathcal{M}(n, m + k)$ is the running time to solve a linear program with $O(n)$ variables and $O(m + k)$ constraints (Theorem 22).

Theorem 19. *Given a range space $(X, \mathcal{R})$, with $|X| = n$, $|\mathcal{R}| = m$, CR constraints $\mathcal{T}$, and a parameter $\varepsilon \in (0, 1)$, there exists a randomized algorithm that constructs a fair ε-net with respect to $\mathcal{T}$, of size $O\left(\max\{\log \frac{1}{\varphi}, d \log(\text{OPT})\} \cdot \log(\frac{k}{\varphi}) \cdot \text{OPT}\right)$ with probability at least $1 - \varphi$, in $O(n \cdot m + \mathcal{M}(n, m + k))$ time, where* OPT *is the size of the smallest fair ε-net with respect to $\mathcal{T}$, k is the number of colors in X, and $\mathcal{M}(n, m + k)$ is the running time to solve a linear program with $O(n)$ variables and $O(m + k)$ constraints.*

It is worth mentioning that in § 5, the Fair Geometric Hitting Set is solved by a reduction to the weighted fair ε-net with DP constraints. As a result, the whole algorithm reduces an instance of the fair ε-net with CR constraint to an instance of the weighted fair ε-net problem with DP constraints.

$$\text{Fair } \varepsilon\text{-net (CR)} \xrightarrow{\text{red.}} \text{FGHS (CR)} \xrightarrow{\text{red.}} \text{Weighted Fair } \varepsilon\text{-net (DP)}$$

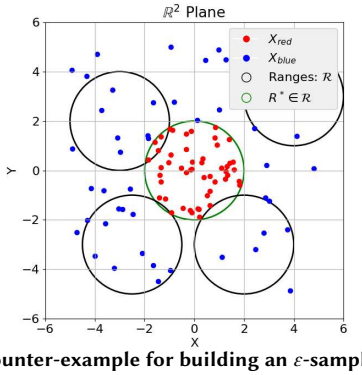

Figure 3: A counter-example for building an ε-sample satisfying any CR constraints. In this example, any ε-sample must contain $\frac{1}{2} \pm \varepsilon$ fraction of red points. The reason is that there exists a range $R^* \in \mathcal{R}$ that separates all reds from all blue points, and we have $\frac{|X_{red}|}{|X|} = \frac{1}{2}$.

Algorithm 3 Fair Geometric Hitting Set through LP relaxation

Require: Range space $(X, \mathcal{R})$, set of custom ratios $\mathcal{T}$.
Ensure: The Fair Geometric Hitting Set $\mathcal{S}$.

```
function FGLP(X, R, T)
    ε, w⃗ ← LPSolver(X, R, T)
    S ← Execute algorithm from Theorem 7 on (X, R) with parameter ε and weights w⃗
    return S
function LPSolver(X, R, T)
    Solves the LP in Equation 3.
    return ε and the weights w⃗ = {w_1, ··· , w_n}.
```

4.4 Fair ε-sample

The fair ε-sample problem under both demographic parity (DP) and custom-ratio (CR) constraints is addressed by extending the techniques introduced for ε-nets. For DP, fair ε-samples can be constructed following similar principles as in Section 4. However, satisfying arbitrary CR constraints is not always feasible. For example, as shown in Figure 3, if a range in the space separates all members of a demographic group, then any fair ε-sample must preserve the group's proportion within an ε margin, limiting the range of achievable target ratios. We present a formal argument and provide a counterexample illustrating this limitation in the technical report [32].

5 FAIR GEOMETRIC HITTING SET

In this section, we give an algorithm for satisfying the CR constraints in the Geometric Hitting Set. The goal of FGHS is to find the smallest subset $\mathcal{S}^* \subseteq X$, given the ratios $\mathcal{T} = (\tau_1, \tau_2, \cdots, \tau_k)$, such that:

(1) $\forall c_\ell \in C,\ \frac{|\mathcal{S}^*_{c_\ell}|}{|\mathcal{S}^*|} = \tau_l$,
(2) $\forall R \in \mathcal{R},\ R \cap \mathcal{S}^* \neq \emptyset$.

Equivalently, one can consider Fair Geometric Set Cover defined in the dual range space where sets are associated with colors. There are two popular algorithms for solving the (unfair) Geometric Hitting Set problem. The first approach relies on the Multiplicative Weight Update method [5, 40, 56]. This is a greedy algorithm that starts by choosing a subset of points. If this subset is not a hitting set for $\mathcal{R}$, they re-weight all the points that belong to at least a range that is not hit by the current solution. Iteratively repeating this process results in finding a $\log \mathrm{OPT}_{\mathrm{GHS}}$ approximation solution, where $\mathrm{OPT}_{\mathrm{GHS}}$ is the optimum geometric hitting set [44].

The second approach formulates the problem as an Integer Program, and solves the relaxed Linear Program [22, 27, 44, 58]. Our algorithm is a variant of this approach that adds the fairness constraints inside the LP formulation. Here, we assume that we have access to all ranges $\mathcal{R}$ with $|\mathcal{R}| = m$. The Integer Program (IP) for solving the FGHS problem is as follows:

$$\begin{aligned} \min \quad & \sum_{p_i \in X} z_i && && (1)\\ \text{s.t.} \quad & \sum_{p_i \in R} z_i \geq 1, && \forall R \in \mathcal{R} \\ & \sum_{p_i \in X_{c_\ell}} z_i = \tau_\ell \cdot \sum_{p_i \in X} z_i, && \forall c_\ell \in C \\ & z_i \in \{0,1\}, && \forall i \in [n] \end{aligned}$$

In this formulation, z_i is a variable that indicates whether the point $p_i \in X$ is selected in the final cover. The first type of constraints ensures that all ranges are hit. The second type of constraints ensures the CR fairness constraints. The equivalent Linear Program (LP) formulation considers the real domain $[0, 1]$ for variables z_i. By using a variable $f = \sum_i z_i$, we can rewrite it as:

$$\begin{aligned} \min \quad & f && && (2)\\ \text{s.t.} \quad & \sum_{p_i \in X} z_i = f \\ & \sum_{p_i \in R} z_i \geq 1, && \forall R \in \mathcal{R} \\ & \sum_{p_i \in X_{c_\ell}} z_i = \tau_\ell \cdot f, && \forall c_\ell \in C \\ & z_i \in [0,1], && \forall i \in [n], \quad f \geq 0 \end{aligned}$$

Now, define $\bar{w}_i = \frac{z_i}{f}$ and $\bar{\varepsilon} = \frac{1}{f}$. We can rewrite the LP as:

$$\begin{aligned} \max \quad & \bar{\varepsilon} && && (3)\\ \text{s.t.} \quad & \sum_{p_i \in X} \bar{w}_i = 1 \\ & \sum_{p_i \in R} \bar{w}_i \geq \bar{\varepsilon}, && \forall R \in \mathcal{R} \\ & \sum_{p_i \in X_{c_\ell}} \bar{w}_i = \tau_\ell, && \forall c_\ell \in C \\ & \bar{w}_i \in [0,1], && \forall i \in [n], \quad \bar{\varepsilon} \geq 0 \end{aligned}$$

Interestingly, the LP in (3) is interpreted as follows. The LP sets some weights to the points such that every range in $\mathcal{R}$ has a total weight of at least $\bar{\varepsilon}$, and the total weight of the points with color c_ℓ is τ_ℓ. Intuitively, a weighted fair $\bar{\varepsilon}$-net as defined in 4.1.1. Using these observations, we are ready to describe our algorithm for the FGHS problem.

Algorithm. Construct the instance of the LP (3). Use an LP solver to solve the LP (3). Let ε be the value of variable $\bar{\varepsilon}$ in the optimum LP solution and w_i be the value of the variable $\bar{w}_i$ for every $p_i \in X$ in the optimum LP solution. Let $\vec{w} = \{w_1, \ldots, w_n\}$ be the vector of all points' weights. We run the algorithm from Theorem 7 on $(X, \mathcal{R})$ with DP constraints $\mathcal{T}$, parameter ε, assuming that each point $p_i \in X$ has weight w_i. Let $\mathcal{S}$ be the weighted fair ε-net returned by the algorithm from Theorem 7. We return $\mathcal{S}$. The pseudocode is shown in Algorithm 3.

Correctness and runtime analysis. Let OPT_{FGHS} be the size of the optimum solution for the FGHS problem on $(X, \mathcal{R})$.

Lemma 20. *The algorithm returns a hitting set of size*

$$O\left(\max\{\log\frac{1}{\varphi}, d\log(\mathsf{OPT}_{FGHS})\}\cdot\log(\frac{k}{\varphi})\cdot\mathsf{OPT}_{FGHS}\right)$$

with probability at least $1-\varphi$.

Proof. See the Technical Report [32] for the proof. □

Let $\mathcal{M}(x, y)$ be the running time to solve an LP with $O(x)$ variables and $O(y)$ constraints.

Lemma 21. *Our algorithm has a time complexity of* $O(n\cdot m + \mathcal{M}(n, m+k))$.

Proof. See the Technical Report [32] for the proof. □

Putting everything together, we conclude with the next theorem.

Theorem 22. *Given a range space* $(X, \mathcal{R})$ *with* $|X| = n$, $|\mathcal{R}| = m$, *CR constraints* $\mathcal{T}$, *and a parameter* $\varphi \in (0, 1)$, *there exists a randomized algorithm that constructs a fair geometric hitting set of* $(X, \mathcal{R})$, *of* $O\left(\max\{\log\frac{1}{\varphi}, d\log(\mathsf{OPT}_{FGHS})\}\cdot\log(\frac{k}{\varphi})\cdot\mathsf{OPT}_{FGHS}\right)$ *size with probability at least* $1-\varphi$, *in* $O(n\cdot m + \mathcal{M}(n, m+k))$ *time, where* OPT_{FGHS} *is the size of the optimum FGHS solution,* k *is the number of different colors in set* X, *and* $\mathcal{M}(n, m+k)$ *is the running time to solve an LP with* $O(n)$ *variables and* $O(m+k)$ *constrains.*

6 RELATED WORK

It is now almost a decade that fairness has become a central topic in data-driven systems, and has been studied across various domains, including AI and Machine Learning [16, 63], data management [6, 87], and in algorithms [51].

Fairness-aware Data Management. Fairness in data-driven systems has been extensively studied in responsible data management [87], and various data-centric fairness-aware approaches have been proposed for different components of the responsible data-analytics pipeline [67]. The research in this domain include causal data repair methods aimed at ensuring fairness [74, 75], techniques for identifying and tuning problematic data slices [89], fairness-aware query adjustment [1, 53, 73, 84, 102], coverage-based methods for addressing representation bias [10, 78], label correction approaches promoting individual fairness [100], methods for mining the underrepresented and underperforming minority groups [30], fair and privacy-preserving data generation techniques [71], as well as fairness-aware data imputation [76, 103], entity matching [64, 77, 81], data cleaning [88], data integration [24, 67], and data augmentation [36, 82], among others. Beyond these, fairness has been investigated in approximate data processing (AQP) data structures, including data-informed hashmaps [79], locality-sensitive hashing [12], and count-min [80].

Algorithmic Fairness. This area includes satisfying fairness constraints while optimizing sub-modular functions [93], and satisfying fairness in Coverage Maximization problems [7, 14], Set Cover [31], Hitting Sets [49], and Facility Locations [50]. Fairness considerations have also been extended to a variety of other algorithmic domains, in related problems like Matching [37, 41], Resource Allocation [61], Ranking [9, 98], and Clustering [59, 90]. To the best of our knowledge none of the prior work study ε-nets, ε-samples, and geometric hitting set through the fairness lens.

7 APPLICATION TO DATABASE SYSTEMS

In Example 1, we highlighted the application of ε-nets as **dataset summaries for range queries**, where the goal is to compress potentially very large datasets to a small subset that contains at least one tuple from any possible (axis-parallel) range query. The ε-net can then facilitate AQP by quickly identifying a tuple in the set that satisfies a query range. In the following, we briefly outline some other applications of ε-nets, ε-samples, and geometric hitting set in database systems and beyond.

ε-samples are foundational to **Approximate Query Processing (AQP)** because they provide small yet statistically representative subsets of the data. By ensuring that the selectivity of every query range deviates by at most ε from that on the full data, they enable consistent accuracy guarantees for aggregate functions such as COUNT, SUM, and AVG. Modern systems like VerdictDB [70] implement hierarchical sampling schemes that mirror the layered construction of ε-samples, using them to propagate uncertainty estimates through query plans. Similarly, PASS[55] and JanusAQP [54] build multi-level uniform or stratified samples to estimate aggregates even in sparse data regions, effectively implementing hierarchical ε-samples to capture both dense and under-represented areas. These structures form the bridge between classical VC-dim-based theory and practical data synopses, making ε-samples the statistical backbone of scalable AQP engines. ε-nets serve a complementary role: rather than approximating frequencies, they guarantee coverage. In database terms, an ε-net ensures that every "heavy" query range is hit by at least one representative in the synopsis. This property is directly applicable to range emptiness and range reporting queries in spatial or multi-dimensional databases, where the goal is to quickly determine whether any qualifying tuple exists without scanning the full dataset.

Approximate Nearest Neighbor (ANN) is popular in various settings, including *geospatial data management*, where one would like to retrieve the k nearest tuples with minimum Euclidean distance to a given query point q [33, 72, 97]. In such cases, the queries can be viewed as d-balls (d-dimensional hyperspheres) centered at q. In spatial indexing or geospatial search [62, 65], an ε-net provides a provable bound that every sufficiently dense region will be represented by at least one tuple, which can drastically reduce false negatives in ANN searches [29]. Given a large dataset of tuples, an ε-net provides a small subset that guarantees to hit the top-k (for the corresponding ε) of any given d-ball. Hence, the NN of the ε-net to q belongs to the kNN of the complete dataset.

Another application of ε-nets is in **Regret–minimizing representatives for top-k queries** [3, 8, 11, 52, 66]: compact sets that minimize a notion of regret in approximate top-k query processing. By defining the range space as the universe of half-spaces, and setting $\varepsilon = \frac{k}{n}$, the ε-net guarantees at least one tuple in the top-k results of any linear ranking function [8]. Furthermore, in [3, 52], the k-regret minimizing set is defined as geometric hitting set.

Fair ε-nets extend the traditional "coverage" guarantee to incorporate fairness—ensuring that the selected subset of items includes representatives from all groups in proportion to their desired ratios. In this context, the goal is to construct a small set of items that can "cover" the preferences of all potential users, while also satisfying group fairness among the items themselves. For example,

in a movie recommendation system, we may want to select a set of movies such that every user finds at least one movie they like, but at the same time ensure that the selection contains a balanced representation of genres (e.g., comedies, dramas, documentaries). Thus, fair ε-nets provide a principled framework for ensuring both coverage and diversity in the selected set. For example, Zheng et al. [101] studied fair representative sets in database systems through a variation of the k-regret minimizing set problem. Building on techniques from [3, 52], this task can be formulated as an instance of fair geometric hitting set, allowing our proposed algorithms to yield efficient solutions with provable theoretical guarantees.

Beyond database systems, ε-nets, geometric hitting Sets, and set covers have applications across other domains, including machine learning, such as sample-efficient learning, active learning, clustering, and interoperability. For example, ε-nets offer a principled way to select small informative subsets for active learning under geometric and distributional assumptions, with applications in learning linear separators [13], convex bodies [45], and online settings [18]. In clustering, coresets based on ε-nets and hitting sets enable efficient approximations for k-means and k-median objectives [38, 46]. Hitting set formulations support interpretable and sparse models like the Set Covering Machines [48, 60] and predictive checklists [99], where minimal feature sets capture key behaviors. They also find use in motion planning [83] and sensor network coverage [95].

Integration Challenges.This work sets the foundation for integrating fair ε-nets into the database systems. However, several challenges remain. Real-world database workloads involve multi-relational joins, non-Euclidean predicates, and dynamic updates that fall outside the static geometric assumptions of ε-theory. Aligning ε-samples or ε-nets with relational query optimizers, and balancing fairness constraints with performance are non-trivial tasks. Furthermore, incorporating fairness-aware variants introduces multi-objective optimization trade-offs among accuracy, representativeness, and runtime efficiency. Addressing these challenges will require tighter coordination between database systems research and geometric approximation theory to build practical, fair, and theoretically sound summarization tools for next-generation data systems.

8 EXPERIMENTS

We evaluate the proposed algorithms on (a) real-world datasets from related applications and (b) synthetic data. Our code and artifacts are publicly available.[3] Due to the space constraints, a detailed discussion of the experiments and additional results is provided in the technical report [32].

8.1 Experimental Setups

We implemented the following four tasks to compare the proposed algorithms with baselines:

- **Database Summarization for Range Queries:** As illustrated in Example 1, the goal in this setting is to find a small, representative sample of the database table. Given a collection of range queries $Q = q_1, q_2, \cdots, q_m$, the objective is to select a small *fair* representative subset of tuples such that each heavy query is hit by at least one tuple from the sample.
- **Neighborhood Hitting on Geographic Data:** Given a set of geographic coordinates representing individual locations, we aim to select a small *fair* subset of individuals (or facility points) such that every neighborhood has at least one selected point within the sample — effectively hitting all heavy spatial neighborhoods. As further discussed in § 7, such a data summary facilitates approximate nearest neighbor search, where the goal is to return one of the top−ε closest tuples in the vicinity of a query point.
- **Rank Regret Representative:** The goal of this task is to find a small subset of tuples that hit the top-l of any ranking function [11]. A ranking function is defined as a linear combination of the dataset attributes. The top-l tuples are determined by computing the dot product between each tuple and the weight vector f, followed by sorting based on the resulting scores. Our objective is to find a *fair* rank-regret representative of the dataset.
- **Synthetic Data:** In this setup, we construct synthetic datasets with tunable parameters such as range size m, dataset size n, VC-dimension d, and demographic distributions. The goal is to evaluate algorithms under varying geometric complexity.

Algorithms: We evaluate the following set of algorithms, as introduced in the previous sections and summarized in Table 1:

- **ε-net:** As a set of baselines, for finding a standard (unfair) ε-net, we consider *Sampling*, *Discrepancy*, and *Sketch-and-Merge (SM)* – See § 3.[4] In all the reports, "**Sampling**" refers to the baseline unfair ε-net construction.
- **Fair ε-net:** For fair ε-nets, we use *Fair Monte-Carlo (FMC)* algorithm and *Fair Sketch-and-Merge (FSM)* as introduced in § 4.
- **Hitting Set:** For standard (unfair) hitting sets, we employ the *Geometric LP (GLP)* algorithm as a baseline, where LP relaxation is used to solve the geometric hitting set problem.
- **Fair Hitting Set:** For the fair hitting set, we use the fair version, *Fair Geometric LP (FGLP)* as introduced in § 5.

Measuring (un)fairness: Fairness constraints are enforced based on either demographic parity (DP) or custom-ratios (CR) fairness. We evaluate fairness in the final results by considering two measures: ℓ_2-norm and ℓ_∞-norm. Assume $\mathcal{N} \subseteq X$ is the output of one of the algorithms and $C = \{c_1, c_2, \cdots, c_k\}$ are the colors (demographic groups). The ℓ_2-norm fairness calculates the ℓ_2 distance of colors in $\mathcal{N}$ versus the desired constraint vector $\mathcal{T}$:

$$\mathcal{F}_2(\mathcal{N}, \mathcal{T}) = \left\| \left(\frac{|\mathcal{N}_{c_1}|}{|\mathcal{N}|}, \ldots, \frac{|\mathcal{N}_{c_k}|}{|\mathcal{N}|} \right) - \mathcal{T} \right\|_2 = \sqrt{\sum_{i \le k} \left(\frac{|\mathcal{N}_{c_i}|}{|\mathcal{N}|} - \tau_i \right)^2} \quad (4)$$

We also define ℓ_∞-norm fairness: $\mathcal{F}_\infty(\mathcal{N}, \mathcal{T}) = \max_{i \le k} \left| \frac{|\mathcal{N}_{c_i}|}{|\mathcal{N}|} - \tau_i \right|$.

Similarly, we define $\mathcal{F}_0$ as an ℓ_0-based unfairness measure that **counts** the number of groups not satisfying their target ratio[5]:

$$\mathcal{F}_0(\mathcal{N}, \mathcal{T}) = \frac{1}{k} \sum_{i=1}^{k} \mathbb{I}\left(\frac{|\mathcal{N}_{c_i}|}{|\mathcal{N}|} \neq \tau_i \right).$$

Using ℓ-norm distances to measure unfairness is a standard approach in the fairness and resource allocation literature [19, 57, 96]. As we will see in the experiments, our methods achieve almost zero-unfairness in most settings.[6] By *zero-unfairness*, we mean that the value of $\mathcal{F}_2$, $\mathcal{F}_\infty$ and $\mathcal{F}_0$ is exactly 0. One can verify that this corresponds to the best achievable fairness level under most commonly used unfairness metrics.

[3] Github repository

[4] Whenever we don't mention *"Fair"* in the name of an algorithm, we mean the baseline unfair version. We sometimes refer to unfair algorithms as "Standard".

[5] In our experiments, the indicator function $\mathbb{I}$ checks equality up to three digits after the decimal point and the final unfairness value is normalized between 0 and 1.

[6] When finding a fair sample is feasible.

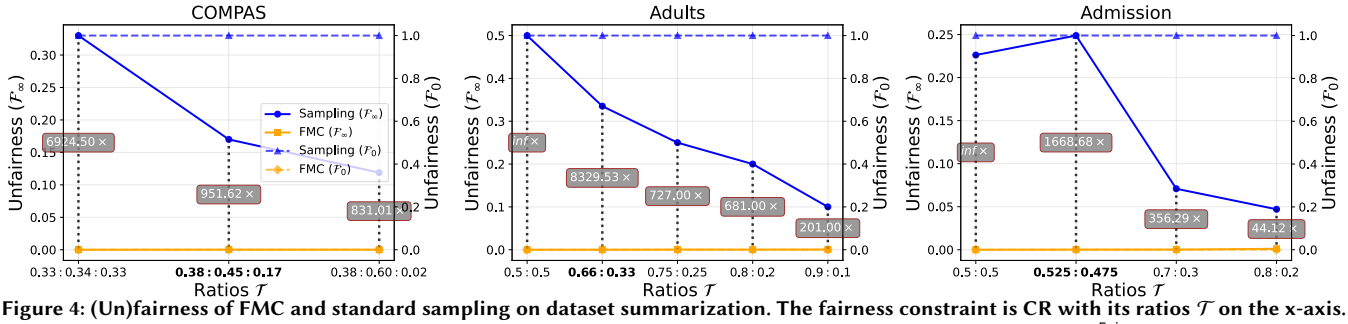

Figure 4: (Un)fairness of FMC and standard sampling on dataset summarization. The fairness constraint is CR with its ratios $\mathcal{T}$ on the x-axis.

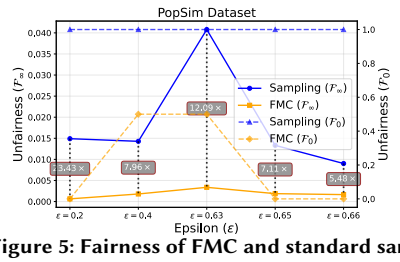

Figure 5: Fairness of FMC and standard sampling on the PopSim dataset vs. the value of ε.

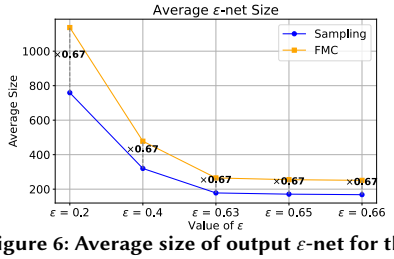

Figure 6: Average size of output ε-net for the PopSim dataset according to DP constraint.

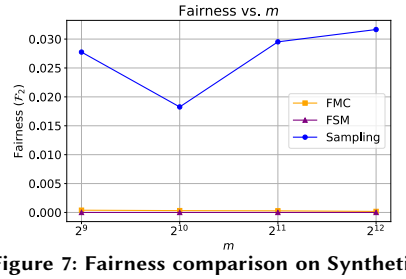

Figure 7: Fairness comparison on Synthetic Sampling task. m is the number of 2D ranges.

Table 3: Average running time of Fair Monte-Carlo and standard sampling algorithms for the dataset summarization task.

Dataset	Sampling Time (s)	FMC Time (s)
Adults	2.1244	2.2044 (**1.03**×)
COMPAS	2.4343	2.5470 (**1.05**×)
Admissions	0.0032	0.0038 (**1.19**×)

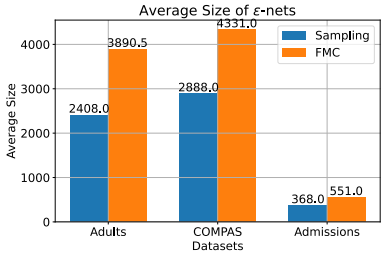

Figure 8: Comparing the average size of ε-net produced by standard and fair algorithms for dataset summarization task.

Datasets: We use four real-world datasets in our experiments. PopSim [69], Adult [34], COMPAS [4], and College Admissions [23]. We also use a synthetic dataset with axis-aligned rectangles in 2D and half-spaces in higher dimensions. A detailed description of these datasets is provided in the technical report [32]. PopSim [69] includes 2 million individual records annotated with racial information. The Adult [34] dataset contains approximately 50K records, each described by multiple demographic and employment attributes. The COMPAS [4] dataset comprises around 7,000 defendant records, with race used as the sensitive attribute.

8.2 Experiment Results

Database Summarization for Range Queries. For this task, we use three datasets: Adult, COMPAS, and College Admission. For each dataset, we generate a collection of range queries that must be hit by the output sample. These ranges are constructed as hyper-rectangles in the feature space. For each attribute, we select a threshold to partition the data into two groups, resulting in up to 2^d hyper-rectangles (d is the number of features). We then compute an ε-net over the data points to hit these ranges for varying values of ε. Figure 4 illustrates the fairness ($\mathcal{F}_\infty$ and $\mathcal{F}_0$) of the FMC algorithm compared to the baseline (unfair) sampling approach under CR constraints. In the CR setting, the goal is to match a target ratio $\mathcal{T}$ of demographic groups in the output. FMC successfully enforces the desired ratio with zero unfairness, while the baseline approach fails to do so. Notably, as the target ratio for the *minority group* increases (towards the left side of x-axis), the standard algorithm struggles even more to satisfy the fairness constraint. In these cases, the standard sampling might not even pick any point from the minor group, which results in higher average values of $\mathcal{F}_\infty$. The average output size of each method is also reported in Figure 8, where we can observe that the fair variants of the algorithms achieve zero unfairness while introducing a relatively small increase in the output size. Table 3 reports the average running time of both algorithms across the three datasets. The results show that the FMC algorithm shows only a minimal overhead compared to the standard sampling approach. Additional experimental results are presented in the technical report [32], where we compare the fairness metrics of these methods across varying values of ε (Figure 17). Consistent trends in the improvement of fairness achieved by FMC can be observed. Similar results were also observed on $\mathcal{F}_2$ fairness.

Neighborhood Hitting. For this task, we use the PopSim dataset, which contains the geographic locations of individuals. Each location is represented as a 2D point, and ranges are defined as balls of fixed radius centered around a randomly selected subset of points. The objective is to construct a fair ε-net—a subset of individuals that hits all densely populated regions. We apply both the FMC algorithm and the standard sampling approach to this setting. We also construct the Fair Hitting Set for this range space. To compute the hitting set, we use the GLP algorithm and its fairness-aware variant, FGLP. Figure 5 shows that FMC achieves zero unfairness under the DP constraint, whereas the standard sampling method results in a significantly unfair sample. Figure 6 further illustrates how the average size of the output sample varies with different values of

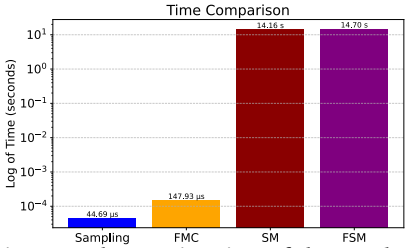

Figure 9: The running time of the standard methods and their fair variants on synthetic rectangle range space in 2D.

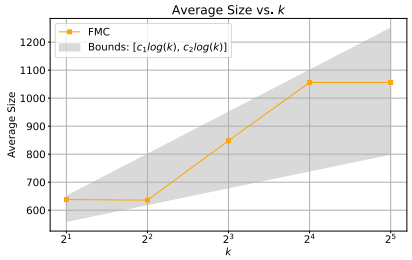

Figure 10: Average size of the ε-net vs. k on synthetic dataset with rectangle ranges and DP constraint. The gray area is the log trend.

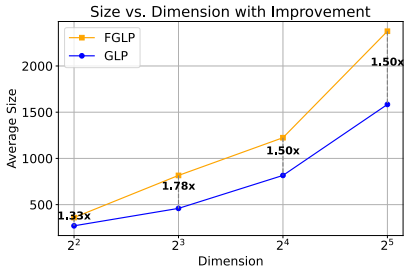

Figure 11: Average size of the hitting set vs. d. In this setting, we consider half-spaces in $\mathbb{R}^d$ space with DP constraint and $k = 2$.

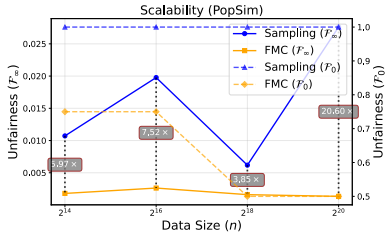

Figure 12: Unfairness on larger dataset size.

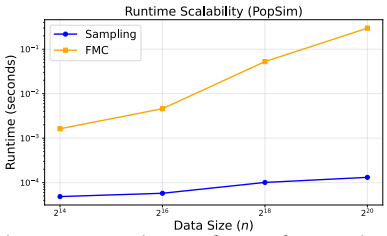

Figure 13: Runtime on larger dataset sizes.

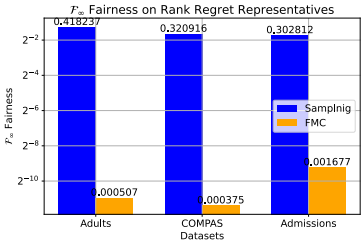

Figure 14: Fairness in Rank Regret Task.

Table 4: GLP v.s. FGLP on PopSim dataset for Neighborhood Hitting.

Algorithm	Output Size	Fairness ($\mathcal{F}_\infty$)	Runtime (μs)
GLP	336	0.093	108
FGLP	503	**0.002**	403

Table 5: Comparing hitting set algorithms on synthetic data.

Algorithm	Half-space ($d > 2$)	Rectangles ($d = 2$)
FGLP	$\mathcal{F}_\infty$: **0.00** Time: 29.77s	$\mathcal{F}_\infty$: **0.00** Time: 1.31s
GLP	$\mathcal{F}_\infty$: 0.06 Time: 20.22s	$\mathcal{F}_\infty$: 0.28 Time: 0.76s

ε. As ε decreases, more points are required to construct an ε-net that hits all heavy ranges. In addition, the fair method introduces only a minimal increase in sample size compared to the standard approach (logarithmic to k). A comparison of the running time of these two algorithms based on values ε is provided in the technical report [32]. Figures 12 and 13 show a comparison of the FMC and the sampling methods on larger datasets (larger instances of PopSim 5). As reflected in Figure 13, FMC ran in only 0.3 seconds for $n = 1$M, confirming its scalability. We also applied the GLP and FGLP algorithms to compute a hitting set for this range space. A comparison of their outputs is presented in Table 4. The results show that FGLP reduces unfairness 45 times, while maintaining a comparable sample size, especially considering the full dataset consists of 2M points.

Synthetic Sampling. For this task, we generated a collection of synthetic datasets consisting of randomly sampled points and geometric ranges in the plane (rectangles) and higher dimensions (half-spaces). Details of the construction process are provided in the technical report [32]. We varied the dimensionality, the number of colors, and the number of ranges in these datasets.

Results on ε-nets: Figure 7 compares the fairness ($\mathcal{F}_2$) of the standard Sampling method with two fair variants: Fair Monte Carlo (FMC) and Fair Sketch-and-Merge (FSM). Both FMC and FSM achieve near-zero unfairness, whereas the standard sampling method exhibits significantly higher unfairness. Figure 9 presents the running time comparison between standard Sampling and FMC, as well as between standard Sketch-and-Merge (SM) and its fair counterpart, FSM. The results show that the additional computational cost introduced by the fair methods is minimal while they significantly improve fairness. Additional results comparing the output size and runtime of standard sampling and Discrepancy-based methods are provided in the technical report [32]. Figure 10 shows the output size of the FMC algorithm as the **number of colors** in the point set increases. The results show that an exponential increase in the number of colors leads to a linear growth in output size, aligning with the theoretical result ($\log k$ factor).

Results on Geometric Set Cover: Table 5 presents a comparison of runtime and fairness of the two algorithms for hitting set problem. FGLP achieves zero unfairness with only a minimal increase in runtime. Figure 11 compares the output size of these algorithms across **different** d. Overall, the fair setting results in a slightly larger output size compared to the standard version, while achieving near-zero unfairness. As d increases, the growth in size remains nearly constant since it depends on $\log k$ rather than the VC-dim.

Rank Regret Representatives. To generate Rank Regret Representatives, we use three real-world datasets: Adult, COMPAS, and Admissions. The goal is to construct an ε-net that intersects the heavy top-l regions of any ranking function defined over the dataset features. We evaluate this across different values of $\varepsilon = \frac{l}{n}$ [11]. The results are consistent with previous settings. Figure 14 presents the fairness comparison across the datasets averaged over multiple ε values. More results are provided in the technical report [32].

9 CONCLUSION

We studied the geometric approximation problems of ε-net, ε-sample, and geometric hitting set through the lens of fairness. We formulated the problems using two notions of group fairness and proposed efficient randomized and deterministic algorithms with small approximation factors to address them. In addition to the theoretical guarantees, our experimental evaluations further demonstrated the effectiveness of our algorithms across various tasks and datasets.

REFERENCES

[1] Chiara Accinelli, Simone Minisi, and Barbara Catania. 2020. Coverage-based Rewriting for Data Preparation.. In *EDBT/ICDT Workshops*.

[2] Vladimir Agafonkin. 2016. *Clustering millions of points on a map with Supercluster*. Mapbox. https://blog.mapbox.com/clustering-millions-of-points-on-a-map-with-supercluster-272046ec5c97 Accessed: 2025-07-06.

[3] Pankaj K Agarwal, Nirman Kumar, Stavros Sintos, and Subhash Suri. 2017. Efficient algorithms for k-regret minimizing sets. *arXiv preprint arXiv:1702.01446* (2017).

[4] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. 2016. Machine Bias – ProPublica. https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing. Accessed: 2025-04-16.

[5] Sanjeev Arora, Elad Hazan, and Satyen Kale. 2012. The multiplicative weights update method: a meta-algorithm and applications. *Theory of computing* 8, 1 (2012), 121–164.

[6] Abolfazl Asudeh. 2021. Enabling responsible data science in practice. *ACM SIGMOD Blog* (2021).

[7] Abolfazl Asudeh, Tanya Berger-Wolf, Bhaskar DasGupta, and Anastasios Sidiropoulos. 2023. Maximizing coverage while ensuring fairness: A tale of conflicting objectives. *Algorithmica* 85, 5 (2023), 1287–1331.

[8] Abolfazl Asudeh, Gautam Das, HV Jagadish, Shangqi Lu, Azade Nazi, Yufei Tao, Nan Zhang, and Jianwen Zhao. 2022. On finding rank regret representatives. *ACM Transactions on Database Systems (TODS)* 47, 3 (2022), 1–37.

[9] Abolfazl Asudeh, HV Jagadish, Julia Stoyanovich, and Gautam Das. 2019. Designing fair ranking schemes. In *Proceedings of the 2019 international conference on management of data*. 1259–1276.

[10] Abolfazl Asudeh, Zhongjun Jin, and HV Jagadish. 2019. Assessing and remedying coverage for a given dataset. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. IEEE, 554–565.

[11] Abolfazl Asudeh, Azade Nazi, Nan Zhang, Gautam Das, and HV Jagadish. 2019. RRR: Rank-regret representative. In *Proceedings of the 2019 International Conference on Management of Data*. 263–280.

[12] Martin Aumuller, Sariel Har-Peled, Sepideh Mahabadi, Rasmus Pagh, and Francesco Silvestri. 2021. Fair near neighbor search via sampling. *ACM SIGMOD Record* 50, 1 (2021), 42–49.

[13] Maria-Florina Balcan and Phil Long. 2013. Active and passive learning of linear separators under log-concave distributions. In *Conference on Learning Theory*. PMLR, 288–316.

[14] Sayan Bandyapadhyay, Aritra Banik, and Sujoy Bhore. 2021. On fair covering and hitting problems. In *Graph-Theoretic Concepts in Computer Science: 47th International Workshop, WG 2021, Warsaw, Poland, June 23–25, 2021, Revised Selected Papers 47*. Springer, 39–51.

[15] Solon Barocas, Moritz Hardt, and Arvind Narayanan. 2017. Fairness in machine learning. *Nips tutorial* 1 (2017), 2017.

[16] Solon Barocas, Moritz Hardt, and Arvind Narayanan. 2023. *Fairness and machine learning: Limitations and opportunities*. MIT Press.

[17] Timothy Bates. 2006. The urban development potential of black-owned businesses. *Journal of the American Planning Association* 72, 2 (2006), 227–237.

[18] Sujoy Bhore, Devdan Dey, and Satyam Singh. 2024. Online Epsilon Net and Piercing Set for Geometric Concepts. *arXiv preprint arXiv:2410.07059* (2024).

[19] Asia J Biega, Krishna P Gummadi, and Gerhard Weikum. 2018. Equity of attention: Amortizing individual fairness in rankings. In *The 41st international acm sigir conference on research & development in information retrieval*. 405–414.

[20] Hervé Brönnimann and Michael T Goodrich. 1994. Almost optimal set covers in finite VC-dimension: (preliminary version). In *Proceedings of the tenth annual symposium on Computational geometry*. 293–302.

[21] Toon Calders, Faisal Kamiran, and Mykola Pechenizkiy. 2009. Building classifiers with independency constraints. In *2009 IEEE international conference on data mining workshops*. IEEE, 13–18.

[22] Timothy M Chan and Sariel Har-Peled. 2009. Approximation algorithms for maximum independent set of pseudo-disks. In *Proceedings of the twenty-fifth annual symposium on Computational geometry*. 333–340.

[23] Eswar Chand. 2020. Admission Dataset. https://www.kaggle.com/datasets/eswarchandt/admission. Accessed: 2025-04-16.

[24] Jiwon Chang, Bohan Cui, Fatemeh Nargesian, Abolfazl Asudeh, and HV Jagadish. 2024. Data distribution tailoring revisited: cost-efficient integration of representative data. *The VLDB Journal* 33, 5 (2024), 1283–1306.

[25] B. Chazelle. 2000. *The Discrepancy Method: Randomness and Complexity*. Cambridge University Press. https://books.google.com/books?id=dmOPmEh6LdYC

[26] Bernard Chazelle and Jiři Matoušek. 1996. On linear-time deterministic algorithms for optimization problems in fixed dimension. *Journal of Algorithms* 21, 3 (1996), 579–597.

[27] Chandra Chekuri, Kenneth L Clarkson, and Sariel Har-Peled. 2012. On the set multicover problem in geometric settings. *ACM Transactions on Algorithms (TALG)* 9, 1 (2012), 1–17.

[28] Valery Chupurnoff. 2024. Displaying a large number of objects on a map. Online blog post. https://mappable.world/blog/displaying-a-large-number-of-objects-on-a-map Accessed July 6, 2025.

[29] Mohsen Dehghankar and Abolfazl Asudeh. 2025. HENN: A Hierarchical Epsilon Net Navigation Graph for Approximate Nearest Neighbor Search. *arXiv preprint arXiv:2505.17368* (2025).

[30] Mohsen Dehghankar and Abolfazl Asudeh. 2025. Mining the Minoria: Unknown, Under-represented, and Under-performing Minority Groups. *Proceedings of the VLDB Endowment* (2025).

[31] Mohsen Dehghankar, Rahul Raychaudhury, Stavros Sintos, and Abolfazl Asudeh. 2025. Fair Set Cover. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1* (Toronto ON, Canada) *(KDD '25)*. Association for Computing Machinery, New York, NY, USA, 189–200. doi:10.1145/3690624.3709184

[32] Mohsen Dehghankar, Stavros Sintos, and Abolfazl Asudeh. 2025. On Fair Epsilon Net and Geometric Hitting Set (Technical Report). *CoRR, abs/2507.08758* (2025). https://arxiv.org/abs/2507.08758

[33] Subramaniam Dhanabal and SJIJCA Chandramathi. 2011. A review of various k-nearest neighbor query processing techniques. *International Journal of Computer Applications* 31, 7 (2011), 14–22.

[34] Dheeru Dua and Casey Graff. 2019. UCI Machine Learning Repository: Adult Data Set. https://archive.ics.uci.edu/ml/datasets/adult. Accessed: 2025-04-16.

[35] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*. 214–226.

[36] Mahdi Erfanian, HV Jagadish, and Abolfazl Asudeh. 2024. Chameleon: Foundation Models for Fairness-Aware Multi-Modal Data Augmentation to Enhance Coverage of Minorities. *Proceedings of the VLDB Endowment* 17, 11 (2024), 3470–3483.

[37] Seyed Esmaeili, Sharmila Duppala, Davidson Cheng, Vedant Nanda, Aravind Srinivasan, and John P Dickerson. 2023. Rawlsian fairness in online bipartite matching: Two-sided, group, and individual. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 5624–5632.

[38] Dan Feldman and Michael Langberg. 2011. A unified framework for approximating and clustering data. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*. 569–578.

[39] Sandra Fredman. 2017. Reversing discrimination. In *Global Minority Rights*. Routledge, 307–332.

[40] Yoav Freund and Robert E Schapire. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences* 55, 1 (1997), 119–139.

[41] David García-Soriano and Francesco Bonchi. 2020. Fair-by-design matching. *Data Mining and Knowledge Discovery* 34 (2020), 1291–1335.

[42] Shivam Gupta, Ganesh Ghalme, Narayanan C Krishnan, and Shweta Jain. 2023. Efficient algorithms for fair clustering with a new notion of fairness. *Data Mining and Knowledge Discovery* 37, 5 (2023), 1959–1997.

[43] Rashida Hakim, Ana-Andreea Stoica, Christos H Papadimitriou, and Mihalis Yannakakis. 2024. The Fairness-Quality Tradeoff in Clustering. *Advances in Neural Information Processing Systems* 37 (2024), 117509–117542.

[44] Sariel Har-Peled. 2011. *Geometric approximation algorithms*. Number 173. American Mathematical Soc.

[45] Sariel Har-Peled, Mitchell Jones, and Saladi Rahul. 2021. Active-learning a convex body in low dimensions. *Algorithmica* 83 (2021), 1885–1917.

[46] Sariel Har-Peled and Soham Mazumdar. 2004. On coresets for k-means and k-median clustering. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*. 291–300.

[47] David Haussler and Emo Welzl. 1986. Epsilon-nets and simplex range queries. In *Proceedings of the second annual symposium on Computational geometry*. 61–71.

[48] Zakria Hussain, Sandor Szedmak, and John Shawe-Taylor. 2004. The linear programming set covering machine. *Pattern Analysis, Statistical Modelling and Computational Learning* (2004).

[49] Tanmay Inamdar, Lawqueen Kanesh, Madhumita Kundu, Nidhi Purohit, and Saket Saurabh. 2023. Fixed-Parameter Algorithms for Fair Hitting Set Problems. In *48th International Symposium on Mathematical Foundations of Computer Science*.

[50] Christopher Jung, Sampath Kannan, and Neil Lutz. 2019. A center in your neighborhood: Fairness in facility location. *arXiv preprint arXiv:1908.09041* (2019).

[51] Jon Kleinberg, Jens Ludwig, Sendhil Mullainathan, and Ashesh Rambachan. 2018. Algorithmic fairness. In *Aea papers and proceedings*, Vol. 108. American Economic Association 2014 Broadway, Suite 305, Nashville, TN 37203, 22–27.

[52] Nirman Kumar and Stavros Sintos. 2018. Faster approximation algorithm for the k-regret minimizing set and related problems. In *2018 Proceedings of the Twentieth Workshop on Algorithm Engineering and Experiments (ALENEX)*. SIAM, 62–74.

[53] Jinyang Li, Yuval Moskovitch, Julia Stoyanovich, and HV Jagadish. 2023. Query refinement for diversity constraint satisfaction. *Proceedings of the VLDB Endowment* 17, 2 (2023), 106–118.

[54] Xi Liang, Stavros Sintos, and Sanjay Krishnan. 2023. Janusaqp: Efficient partition tree maintenance for dynamic approximate query processing. In *2023 IEEE 39th International Conference on Data Engineering (ICDE)*. IEEE, 572–584.

[55] Xi Liang, Stavros Sintos, Zechao Shang, and Sanjay Krishnan. 2021. Combining aggregation and sampling (nearly) optimally for approximate query processing. In *Proceedings of the 2021 International Conference on Management of Data*. 1129–1141.

[56] Nick Littlestone. 1988. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine learning* 2 (1988), 285–318.

[57] Yiran Liu, Ke Yang, Zehan Qi, Xiao Liu, Yang Yu, and Chengxiang Zhai. 2024. Prejudice and caprice: A statistical framework for measuring social discrimination in large language models. *CoRR* (2024).

[58] Philip M Long. 2001. Using the pseudo-dimension to analyze approximation algorithms for integer programming. In *Workshop on Algorithms and Data Structures*. Springer, 26–37.

[59] Yury Makarychev and Ali Vakilian. 2021. Approximation algorithms for socially fair clustering. In *Conference on Learning Theory*. PMLR, 3246–3264.

[60] Mario Marchand, Mohak Shah, John Shawe-Taylor, and Marina Sokolova. 2003. The set covering machine with data-dependent half-spaces. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*. 520–527.

[61] Tasfia Mashiat, Xavier Gitiaux, Huzefa Rangwala, Patrick Fowler, and Sanmay Das. 2022. Trade-offs between group fairness metrics in societal resource allocation. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*. 1095–1105.

[62] Jiří Matoušek. 1994. Geometric range searching. *ACM Computing Surveys (CSUR)* 26, 4 (1994), 422–461.

[63] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. 2021. A survey on bias and fairness in machine learning. *ACM computing surveys (CSUR)* 54, 6 (2021), 1–35.

[64] Mohammad Hossein Moslemi and Mostafa Milani. 2024. Threshold-independent fair matching through score calibration. In *Proceedings of the Conference on Governance, Understanding and Integration of Data for Effective and Responsible AI*. 40–44.

[65] Nabil H Mustafa and Kasturi Varadarajan. 2017. Epsilon-approximations & epsilon-nets. In *Handbook of Discrete and Computational Geometry*. Chapman and Hall/CRC, 1241–1267.

[66] Danupon Nanongkai, Atish Das Sarma, Ashwin Lall, Richard J Lipton, and Jun Xu. 2010. Regret-minimizing representative databases. *Proceedings of the VLDB Endowment* 3, 1-2 (2010), 1114–1124.

[67] Fatemeh Nargesian, Abolfazl Asudeh, and HV Jagadish. 2022. Responsible data integration: Next-generation challenges. In *Proceedings of the 2022 international conference on management of data*. 2458–2464.

[68] Rostislav Netek, Jan Brus, and Ondrej Tomecka. 2019. Performance testing on marker clustering and heatmap visualization techniques: a comparative study on javascript mapping libraries. *ISPRS international journal of geo-information* 8, 8 (2019), 348.

[69] Khanh Duy Nguyen, Nima Shahbazi, and Abolfazl Asudeh. 2023. PopSim: An Individual-level Population Simulator for Equitable Allocation of City Resources. *arXiv preprint arXiv:2305.02204* (2023).

[70] Yongjoo Park, Barzan Mozafari, Joseph Sorenson, and Junhao Wang. 2018. Verdictdb: Universalizing approximate query processing. In *Proceedings of the 2018 International Conference on Management of Data*. 1461–1476.

[71] David Pujol, Amir Gilad, and Ashwin Machanavajjhala. 2023. PreFair: Privately Generating Justifiably Fair Synthetic Data. *Proceedings of the VLDB Endowment* 16, 6 (2023), 1573–1586.

[72] Nick Roussopoulos, Stephen Kelley, and Frederic Vincent. 1995. Nearest neighbor queries. In *Proceedings of the 1995 ACM SIGMOD international conference on Management of data*. 71–79.

[73] Senjuti Basu Roy, Baruch Schieber, and Nimrod Talmon. 2024. Fairness in Preference Queries: Social Choice Theories Meet Data Management. *Proceedings of the VLDB Endowment* 17, 12 (2024), 4225–4228.

[74] Babak Salimi, Bill Howe, and Dan Suciu. 2020. Database repair meets algorithmic fairness. *ACM SIGMOD Record* 49, 1 (2020), 34–41.

[75] Babak Salimi, Luke Rodriguez, Bill Howe, and Dan Suciu. 2019. Interventional fairness: Causal database repair for algorithmic fairness. In *Proceedings of the 2019 international conference on management of data*. 793–810.

[76] Sebastian Schelter, Tammo Rukat, and Felix Biessmann. 2021. JENGA: A framework to study the impact of data errors on the predictions of machine learning models. (2021).

[77] Nima Shahbazi, Nikola Danevski, Fatemeh Nargesian, Abolfazl Asudeh, and Divesh Srivastava. 2023. Through the Fairness Lens: Experimental Analysis and Evaluation of Entity Matching. *Proceedings of the VLDB Endowment* 16, 11 (2023), 3279–3292.

[78] Nima Shahbazi, Yin Lin, Abolfazl Asudeh, and HV Jagadish. 2023. Representation bias in data: A survey on identification and resolution techniques. *Comput. Surveys* 55, 13s (2023), 1–39.

[79] Nima Shahbazi, Stavros Sintos, and Abolfazl Asudeh. 2024. Fairhash: A fair and memory/time-efficient hashmap. *Proceedings of the ACM on Management of Data* 2, 3 (2024), 1–29.

[80] Nima Shahbazi, Stavros Sintos, and Abolfazl Asudeh. 2025. Fair-Count-Min: Frequency Estimation under Equal Group-wise Approximation Factor. *arXiv preprint arXiv:2505.18919* (2025).

[81] Nima Shahbazi, Jin Wang, Zhengjie Miao, and Nikita Bhutani. 2024. Fairness-aware data preparation for entity matching. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*. IEEE, 3476–3489.

[82] Shubham Sharma, Yunfeng Zhang, Jesús M Ríos Aliaga, Djallel Bouneffouf, Vinod Muthusamy, and Kush R Varshney. 2020. Data augmentation for discrimination prevention and bias disambiguation. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*. 358–364.

[83] Seiji Shaw, Aidan Curtis, Leslie Pack Kaelbling, Tomás Lozano-Pérez, and Nicholas Roy. 2024. Towards practical finite sample bounds for motion planning in TAMP. *arXiv preprint arXiv:2407.17394* (2024).

[84] Suraj Shetiya, Ian P Swift, Abolfazl Asudeh, and Gautam Das. 2022. Fairness-aware range queries for selecting unbiased data. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE, 1423–1436.

[85] Ashudeep Singh and Thorsten Joachims. 2018. Fairness of exposure in rankings. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 2219–2228.

[86] Mohit Singhal, Javier Pacheco, Seyyed Mohammad Sadegh Moosavi Khorzooghi, Tanusree Debi, Abolfazl Asudeh, Gautam Das, and Shirin Nilizadeh. 2025. Auditing Yelp's Business Ranking and Review Recommendation Through the Lens of Fairness. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 19. 1798–1816.

[87] Julia Stoyanovich, Bill Howe, and Hosagrahar Visvesvaraya Jagadish. 2020. Responsible data management. *Proceedings of the VLDB Endowment* 13, 12 (2020).

[88] Ki Hyun Tae, Yuji Roh, Young Hun Oh, Hyunsu Kim, and Steven Euijong Whang. 2019. Data cleaning for accurate, fair, and robust models: A big data-AI integration approach. In *Proceedings of the 3rd international workshop on data management for end-to-end machine learning*. 1–4.

[89] Ki Hyun Tae and Steven Euijong Whang. 2021. Slice tuner: A selective data acquisition framework for accurate and fair machine learning models. In *Proceedings of the 2021 International Conference on Management of Data*. 1771–1783.

[90] Suhas Thejaswi, Bruno Ordozgoiti, and Aristides Gionis. 2021. Diversity-aware k-median: Clustering with fair center representation. In *Machine Learning and Knowledge Discovery in Databases. Research Track: European Conference, ECML PKDD 2021, Bilbao, Spain, September 13–17, 2021, Proceedings, Part II 21*. Springer, 765–780.

[91] Vladimir N. Vapnik and Alexey Ya. Chervonenkis. 1971. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications* 16, 2 (1971), 264–280. doi:10.1137/1116025

[92] Volodymyr Agafonkin and contributors. 2025. Leaflet – a JavaScript library for interactive maps. https://leafletjs.com/. Accessed July 6, 2025.

[93] Yanhao Wang, Yuchen Li, Francesco Bonchi, and Ying Wang. 2022. Balancing Utility and Fairness in Submodular Maximization (Technical Report). *arXiv preprint arXiv:2211.00980* (2022).

[94] Zeyu Wang, Klint Qinami, Ioannis Christos Karakozis, Kyle Genova, Prem Nair, Kenji Hata, and Olga Russakovsky. 2020. Towards fairness in visual recognition: Effective strategies for bias mitigation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 8919–8928.

[95] Weili Wu, Zhao Zhang, Wonjun Lee, Dingzhu Du, et al. 2020. Optimal coverage in wireless sensor networks. (2020).

[96] Yixuan Xu, Steven Jecmen, Zimeng Song, and Fei Fang. 2023. A one-size-fits-all approach to improving randomness in paper assignment. *Advances in Neural Information Processing Systems* 36 (2023), 14445–14468.

[97] Bin Yao, Feifei Li, and Piyush Kumar. 2010. K nearest neighbor queries and knn-joins in large relational databases (almost) for free. In *2010 IEEE 26th International Conference on Data Engineering (ICDE 2010)*. IEEE, 4–15.

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

[99] Haoran Zhang, Quaid Morris, Berk Ustun, and Marzyeh Ghassemi. 2021. Learning optimal predictive checklists. *Advances in neural information processing systems* 34 (2021), 1215–1229.

[100] Hantian Zhang, Ki Hyun Tae, Jaeyoung Park, Xu Chu, and Steven Euijong Whang. 2023. iflipper: Label flipping for individual fairness. *Proceedings of the ACM on Management of Data* 1, 1 (2023), 1–26.

[101] Jiping Zheng, Yuan Ma, Wei Ma, Yanhao Wang, and Xiaoyang Wang. 2022. Happiness maximizing sets under group fairness constraints. *Proceedings of the VLDB Endowment* 16, 2 (2022), 291–303.

[102] Jiongli Zhu, Sainyam Galhotra, Nazanin Sabri, and Babak Salimi. 2023. Consistent Range Approximation for Fair Predictive Modeling. *Proc. VLDB Endow.* (2023).

[103] Jiongli Zhu and Babak Salimi. 2024. Overcoming Data Biases: Towards Enhanced Accuracy and Reliability in Machine Learning. *IEEE Data Eng. Bull.* 47, 1 (2024), 18–35.

[104] Indre Zliobaite. 2015. On the relation between accuracy and fairness in binary classification. *arXiv preprint arXiv:1505.05723* (2015).