Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence (IJCAI-23)
Special Track on AI for Good

Fast and Differentially Private Fair Clustering

Junyoung Byun[1,2] , **Jaewook Lee**[1]
[1]Department of Industrial Engineering, Seoul National University
[2]Institute of Engineering Research, Seoul National University
{quswns95, jaewook}@snu.ac.kr

Abstract

This study presents the first differentially private and fair clustering method, built on the recently proposed density-based fair clustering approach. The method addresses the limitations of fair clustering algorithms that necessitate the use of sensitive personal information during training or inference phases. Two novel solutions, the Gaussian mixture density function and Voronoi cell, are proposed to enhance the method's performance in terms of privacy, fairness, and utility compared to previous methods. The experimental results on both synthetic and real-world data confirm the compatibility of the proposed method with differential privacy, achieving a better fairness-utility trade-off than existing methods when privacy is not considered. Moreover, the proposed method requires significantly less computation time, being at least 3.7 times faster than the state-of-the-art.

1 Introduction

Recently, there has been a significant increase in attention devoted to the issue of fairness in machine learning due to the revelation that the outputs generated by multiple machine learning algorithms exhibit a bias towards certain demographic groups. The utilization of such biased algorithms can lead to adverse effects for individuals who belong to socially marginalized communities. An empirical study reported that the COMPAS software utilized by courts in the United States tends to exhibit a higher prediction of recidivism rate for Black Americans than is accurate [Mehrabi *et al.*, 2021]. Furthermore, several instances of discrimination in face recognition technology have been brought to light, including the categorization of white faces as more attractive and Asian faces being labeled as having closed eyes [Howard and Borenstein, 2018]. As a result, it is imperative to address the unequal treatment exhibited by machine learning algorithms, which is closely related to the United Nations' Sustainable Development Goals [Vinuesa *et al.*, 2020].

Studies on fairness in machine learning have primarily centered on supervised learning techniques, such as classification [Zafar *et al.*, 2017; Agarwal *et al.*, 2018] and regression [Berk *et al.*, 2017]. However, it is crucial to also consider the fairness of unsupervised learning, as biased unsupervised models can result in adverse outcomes in real-world scenarios. For instance, when the result of customer segmentation is closely tied to a sensitive variable, minority groups may be unfairly excluded from opportunities such as targeted marketing benefits. Since the pioneering work of [Chierichetti *et al.*, 2017], several studies on fair clustering have been conducted. The concept of "balance" is the most widely used notion of fairness in fair clustering and is a clustering-specific variant of disparate impact. The majority of these studies are based on fairlet decomposition, which balances the coupling of data points based on the sensitivity variables and distances between points.

Fairness in machine learning has a strong correlation with privacy, as demographic information is considered personal and private. The revelation of sensitive variables can result in disparate treatment [Krieger and Fiske, 2006], a legal principle that prohibits decision-making based on such information. As a result, many fair machine learning models aim to minimize the use of sensitive variables [Lahoti *et al.*, 2020], particularly during the inference phase. However, such models may not be effective in scenarios where the collection of sensitive information is prohibited. To address privacy concerns regarding demographic information, the concept of differential privacy (DP) [Dwork, 2006] was introduced in fair classification to limit the impact of sensitive variables on model outputs [Jagielski *et al.*, 2019]. However, differential privacy has yet to be applied to fair clustering, as the fairlet decomposition requires repeated access to sensitive variables in the construction of fairlets, making it incompatible with DP.

In this study, we introduce a pioneering approach for fair clustering that ensures differential privacy. Specifically, we present a novel differentially private fair labeling function for the density-based fair clustering method [Lee *et al.*, 2021]. To address the limitations of the existing method in handling high-dimensional data, we propose two solutions. Firstly, instead of utilizing a kernel-based support function, our approach employs Gaussian mixtures (GMs). Secondly, we determine adjacency of centers based on the intersection of their Voronoi cells instead of relying on the existence of transition points (TPs). By utilizing the GM density function and Voronoi cells, our proposed method eliminates the requirement for gradient system integrations, leading to substantial improvements in both efficacy and efficiency.

2 Related Works

The fair clustering method has undergone numerous advancements since its introduction in [Chierichetti *et al.*, 2017]. To overcome the super-quadratic time requirement of the original fairlet decomposition method [Chierichetti *et al.*, 2017], approximate fairlet decomposition methods were proposed in [Bera *et al.*, 2019] and [Backurs *et al.*, 2019] that only require nearly linear time. Moreover, [Huang *et al.*, 2019] proposed an algorithm to determine a coreset before applying any efficient fair decomposition method.

In addition to the k-center methods, fairlet decomposition was applied to other clustering methods such as correlation clustering [Ahmadian *et al.*, 2020b] and hierarchical clustering [Ahmadian *et al.*, 2020a]. [Li *et al.*, 2020] introduced a deep fair clustering method by incorporating a loss function to maintain a similar cluster assignment distribution among demographic groups.

Several studies have also proposed classification models that achieve fairness and DP simultaneously. [Cummings *et al.*, 2019] proposed an efficient algorithm that is approximately fair in cases where ensuring DP prohibits fairness. [Mozannar *et al.*, 2020] introduced a fair learning algorithm with local DP to protect sensitive information. [Tran *et al.*, 2021] extended the method in [Jagielski *et al.*, 2019] to deep learning by incorporating Lagrangian duality into the training of neural networks.

3 Preliminaries

3.1 Balance for Fair Clustering

Consider a set of n data points, denoted as $\{\mathbf{x}_i\}$, that belong to data space $\mathcal{X} := \mathbb{R}^p$. Each data point $\mathbf{x}_i$ is associated with a binary sensitive variable z_i, which can take on either the value of 0 or 1. In accordance with the definition presented in [Chierichetti *et al.*, 2017], the balance of a subset D is calculated as follows:

$$\mathrm{Bal}(D) := \min\left(\frac{|D_0|}{|D_1|}, \frac{|D_1|}{|D_0|}\right), \tag{1}$$

where $|D_0|$ and $|D_1|$ denote the number of instances in D with $z = 0$ and $z = 1$, respectively.

Additionally, the balance of a set of clusters $\mathbf{C} = C_1, \ldots, C_m$ can be determined as follows:

$$\mathrm{Bal}(\mathbf{C}) := \min_{C \in \mathbf{C}} \mathrm{Bal}(C). \tag{2}$$

3.2 Density-based Fair Clustering

With a smooth real-valued density function $f : \mathcal{X} \to \mathbb{R}$, the level set of the density function $L_f(\gamma)$ for $\gamma > 0$ can be decomposed into disjoint clusters, as follows

$$L_f(\gamma) := \{\mathbf{x} \in \mathcal{X} : f(\mathbf{x}) \leq \gamma\} = C_1 \cup \ldots \cup C_m. \tag{3}$$

Density-based clustering methods typically begin by estimating a density function, which is assumed to have a specific form. For instance, when using support vector domain description (SVDD) [Tax and Duin, 1999], the density function is estimated as follows:

$$f(\mathbf{x}) = K(\mathbf{x}, \mathbf{x}) - 2\sum_i \alpha_i K(\mathbf{x}_i, \mathbf{x}) + \sum_{i,j} \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j), \tag{4}$$

where $K(\cdot, \cdot)$ is a kernel function and α_i are the estimated parameters. The gradient system is then constructed from the estimated density function as follows:

$$\frac{d\mathbf{x}}{dt} = -G(\mathbf{x})^{-1}\nabla f(\mathbf{x}), \tag{5}$$

where $G(\mathbf{x})$ is a positive definite matrix for all $\mathbf{x}$.

A stable equilibrium point (SEP) is an equilibrium vector ($\nabla f(\mathbf{s}) = 0$), whose corresponding Jacobian matrix $J(\mathbf{s})$ has only positive eigenvalues. The basin cell of an SEP is defined as follows:

$$\overline{B(\mathbf{s})} := \mathrm{cl}\{\mathbf{x}(0) \in \mathcal{R} : \lim_{t\to\infty}\mathbf{x}(t) = \mathbf{s}\}. \tag{6}$$

The entire data space, $\mathcal{X}$, can be partitioned into separate basin cells under mild conditions, which allows for the labeling of any unseen data. A data point in the basin cell of an SEP is labeled as belonging to the same cluster as the SEP.

An index-k equilibrium vector is an equilibrium vector whose corresponding Jacobian matrix has k negative eigenvalues and $p - k$ positive eigenvalues. If an index-one equilibrium vector, or TP, $\mathbf{t}_{ij} \in \overline{B(\mathbf{s}_i)} \cap \overline{B(\mathbf{s}_j)}$ exists between two SEPs $\mathbf{s}_i$ and $\mathbf{s}_j$, the two SEPs are said to be adjacent and the distance between them is estimated as $f(\mathbf{t}_{ij})$. The SEPs are then hierarchically merged based on their proximity to form the desired number of clusters.

In [Lee *et al.*, 2021], a fair clustering method was proposed by modifying the hierarchical merging procedure of SEPs. The method adds a fairness constraint to the distance measure by incorporating a trade-off between fairness and utility as follows:

$$\begin{aligned}\hat{f}(\mathbf{t}_i) &:= f(\mathbf{t}_i) \\ &- \lambda \cdot (\mathrm{Bal}(C_{i1} \cup C_{i2}) - \min(\mathrm{Bal}(C_{i1}), \mathrm{Bal}(C_{i2}))),\end{aligned} \tag{7}$$

where $i1$ and $i2$ are indices of the two clusters to which each of the two SEPs corresponding to the TP $\mathbf{t}_i$ belongs, and λ is a hyper-parameter to adjust the trade-off between fairness and utility.

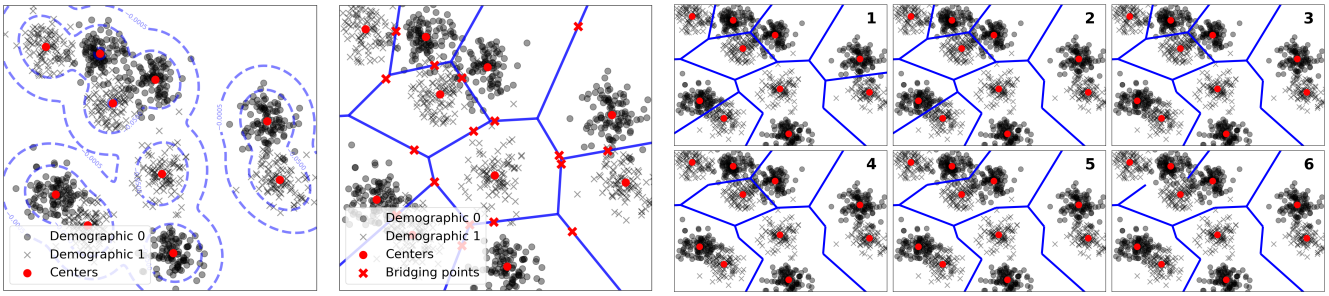
Though method in [Lee *et al.*, 2021] has an advantage in capturing complex, nonconvex clusters, it faces challenges in terms of efficiency and performance, particularly for high-dimensional data. The process of determining SEPs and TPs through (5) requires a large number of iterations, and the effectiveness of the algorithm in determining TPs becomes uncertain as the dimensionality of the data grows.

3.3 Differential Privacy

DP is a concept in privacy-preserving data analysis that bounds the impact of any single individual's data on the outcome of a query. This is achieved by adding random noise to the query results, which limits the extent to which the output can be altered by any change in the input.

Definition 1 (Differential privacy)**.** *A randomized algorithm $\mathcal{Q}$ is (ϵ, δ)-DP if for all $A \subseteq \mathrm{Range}(\mathcal{Q})$ and for all neighboring datasets $D, D' \in \mathcal{D}$ such that $||D - D'||_1 = 1$ (differ in only one data point),*

$$\Pr[\mathcal{Q}(D) \in A] \leq \exp(\epsilon)\Pr[\mathcal{Q}(D) \in A] + \delta. \tag{8}$$



(a) (b) (c)

Figure 1: The overall framework of the proposed method, demostrated with a toy example. (a) 1. Estimation of the GM density function, which can be modified to ensure DP for all variables. (b) 2. Determination of the Voronoi cells of the centers and the corresponding BPs. (c) 3. Fair and DP hierarchical merging of the centers until $K = 4$, with the number in each figure indicating the order of merging.

If $\delta = 0$, $\mathcal{Q}$ is considered ϵ-DP, where ϵ is referred to as the privacy budget, and smaller ϵ values indicate stronger privacy. δ is the probability of failure to satisfy ϵ-DP.

We present two widely-used mechanisms to modify algorithms to satisfy DP. The mechanisms are based on the concept of sensitivity, which is defined as follows:

Definition 2 (Sensitivity)**.** *The l_o-sensitivity of a function $g : \mathcal{D} \to \mathbb{R}^k$ is given by:*

$$\Delta_o g := \max_{D,D' \in \mathcal{D}, ||D-D'||_1 = 1} ||g(D) - g(D')||_o. \quad (9)$$

The Laplace and Gaussian mechanisms are defined as follows:

Definition 3 (Laplace mechanism)**.** *For any function $g : \mathcal{D} \to \mathbb{R}^k$, the Laplace mechanism is defined as:*

$$\mathcal{Q}_L(D, f, \epsilon) := f(D) + \mathbf{y} \quad (10)$$

where $\mathbf{y} = (y_1, \ldots, y_k)$ is a vector of k i.i.d random variables $y_i \sim \mathrm{Lap}(\Delta_1 g/\epsilon)$, and Lap denotes the Laplace distribution.

Definition 4 (Gaussian mechanism)**.** *For any function $g : \mathcal{D} \to \mathbb{R}^k$, the Gaussian mechanism is defined as:*

$$\mathcal{Q}_G(D, f, \epsilon, \delta) := f(D) + \mathbf{y} \quad (11)$$

where $\mathbf{y} = (y_1, \ldots, y_k)$ is a vector of k i.i.d random variables $y_i \sim \mathcal{N}(0, \frac{2 \cdot \ln(1.25/\delta) \cdot (\Delta_2 g)^2}{\epsilon^2})$.

The Laplace and Gaussian mechanisms are widely recognized for ensuring ϵ-DP and (ϵ, δ)-DP, respectively. Two important properties of DP are the post-processing property and the composition property.

Remark 1 (Post-processing)**.** *If a randomized algorithm $\mathcal{Q}$ satisfies (ϵ, δ)-DP, then for any randomized mapping $\mathcal{Q}'$, $\mathcal{Q}' \circ \mathcal{Q}$ also satisfies (ϵ, δ)-DP.*

Remark 2 (Composition)**.** *If two randomized algorithms $\mathcal{Q}_1$ and $\mathcal{Q}_2$ satisfy ϵ_1-DP and ϵ_2-DP, respectively, then the mapping $\mathcal{Q}_{1,2}(\cdot) = (\mathcal{Q}_1(\cdot), \mathcal{Q}_2(\cdot))$ satisfies $(\epsilon_1 + \epsilon_2)$-DP.*

The two properties imply that accessing data multiple times will consume the privacy budget, while post-processing mappings that do not access the data will not consume the privacy budget.

The parallel composition property states that releasing the results of multiple randomized algorithms, each operating on disjoint datasets, will not increase privacy loss.

Remark 3 (Parallel composition)**.** *Let for all $i = 1, \ldots, d$, dataset D_i disjoint to others and $\mathcal{Q}_i$ whose input is D_i satisfy ϵ_i-DP, then releasing all $\mathcal{Q}_i$s guarantees $\max_i \epsilon_i$-DP.*

4 Proposed Method

4.1 Overall Framework

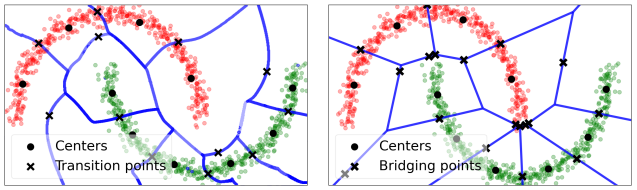
The proposed method involves three stages, as illustrated in Figure 1. The initial stage involves the estimation of the GM density function for the sample data. The GM training process does not compromise sensitive information and therefore does not consume privacy budget. However, in certain cases, it may be necessary to protect other variables, and thus a differentially private variant of the GM training has been introduced to ensure the privacy of all variables. Once the GM has been trained, the centers of the Gaussian distributions are treated as approximations of the density function's modes, taking over the role of the SEPs. The second step involves determining the adjacency and distance between the centers with their Voronoi cells. Specifically, centers with intersecting Voronoi cells are considered adjacent and the distance between the centers is estimated using (7), where the TPs are substituted with bridging points (BPs). Finally, the centers are hierarchically merged in a differentially private manner.

4.2 Density Estimation with Gaussian Mixtures

The GM density function is expressed as :

$$f(\mathbf{x}) = \sum_{k=1}^{\kappa} w_k (2\pi)^{-p/2} |\mathbf{\Sigma}_k|^{-1/2} e^{-(\mathbf{x}-\mathbf{m}_k)^T \mathbf{\Sigma}_k^{-1} (\mathbf{x}-\mathbf{m}_k)/2}, \quad (12)$$

where for $k = 1, \ldots, \kappa$, $w_k, \mathbf{m}_k$, and $\mathbf{\Sigma}_k$ are the weight, mean, and covariance of each Gaussian component, respectively. The computational cost of the sequential minimal optimization algorithm for SVDD is between linear and quadratic in n [Platt, 1998], while the computational cost of



(a) Basin cells and TPs (b) Voronoi cells and BPs

Figure 2: Comparison of the basin cell and the Voronoi cell.

the expectation-maximization (EM) algorithm for GM is linear in n. Furthermore, the use of GM as the density function reduces the computational cost of determining the SEPs, which is $O(n \cdot nsv)$ for SVDD (nsv denotes the number of support vectors) and $O(n \cdot \kappa)$ for GM, by using the centers $\mathbf{m}_k$ as approximate SEPs.

Another advantage of the GM density function is its ability to provide DP for all variables, not just sensitive ones. Evaluating kernel-based density functions requires access to support vectors, a subset of the training data. This leads to the need for adding excessive noise when determining SEPs through (5), rendering the algorithm ineffective. The trained GM density function, on the other hand, does not require any information from the training data for evaluation, which provides the post-processing property for the following steps.

4.3 Finding Bridging Points with Voronoi Cells

While using the density of the TPs as the dissimilarity measure between adjacent SEPs is common in density-based clustering, determining the TPs is computationally intensive and can become a bottleneck in the labeling process. Additionally, as the dimension of the data increases, the density allocated to areas where no data sample exists becomes minuscule, making it difficult to effectively determine TPs between different clusters using publicly available algorithms. For further information on determining TPs, refer to Algorithm 1 in [Lee and Lee, 2006].

In order to address the issues associated with the TPs, we utilize the Voronoi cells of the centers of the trained GM density function and its associated BPs. The Voronoi cell $V(\mathbf{m}_k)$ of a center $\mathbf{m}_k$ is defined as the closure of the set of all data points whose closest center is $\mathbf{m}_k$. The entire data space can be partitioned into individual Voronoi cells, similar to the basin cells. The dissimilarity between two adjacent centers $\mathbf{m}_i$ and $\mathbf{m}_j$ is determined by the density of the corresponding BP $\mathbf{b}_{ij}$, which is given by:

$$\mathbf{b}_{ij} := \operatorname{argmin}_{\mathbf{x}}\{f(\mathbf{x}) : ||\mathbf{x}-\mathbf{m}_i|| = ||\mathbf{x}-\mathbf{m}_j||\},$$
$$s.t.\ ||\mathbf{x}-\mathbf{m}_i|| = ||\mathbf{x}-\mathbf{m}_j|| < ||\mathbf{x}-\mathbf{m}_k||, \forall k \neq i, j. \quad (13)$$

This definition extends the one presented in [Kim *et al.*, 2014] by adding a constraint that the BP between the two centers should not be closer to any other center. This constraint is crucial for positioning the BP at the interface between the two Voronoi cells. Figure 2 illustrates the difference between basin cells and Voronoi cells for the same centers. It can be observed that some adjacent basin cells did not have any TPs, while BPs were found for all adjacent Voronoi cells (one BP, located in the upper right, is outside the range of the figure).

Algorithm 1 Fair labeling of sub-clusters with DP

procedure FLDP(density function f, sensitive variable $\mathbf{z} \subset \{0,1\}^n$, desired number of clusters K, set of initial clusters $\{C_i\}_{i=1}^a$, set of adjacency points $\{\mathbf{t}_j\}_{j=1}^b$, privacy budget ϵ)
  for i in $1, \ldots, a$ **do**
    $v(C_i) = \{k | \mathbf{x}_k \in C_i, k = 1, \ldots, n\}$
    $\mathrm{Sum}_z(C_i) = \sum_{k \in v(C_i)} z_k + \mathrm{Lap}(1/\epsilon)$
  Calculate $d(C_{j1}, C_{j2}) = f(\mathbf{t}_j) - \lambda(\mathrm{Bal}(\mathrm{C_{j1}} \cup \mathrm{C_{j2}}) - \min(\mathrm{Bal}(C_{j1}), \mathrm{Bal}(C_{j2})))$ for $j = 1, \ldots, b$, where $\mathrm{Bal}(C) = \min(\frac{\mathrm{Sum}_z(C)}{|C| - \mathrm{Sum}_z(C)}, \frac{|C| - \mathrm{Sum}_z(C)}{\mathrm{Sum}_z(C)})$ and $j1$, $j2$ are the indices of two adjacent modes w.r.t $\mathbf{t}_j$.
  Hierarchically merge the initial clusters, following steps 4 and 5 in [Lee *et al.*, 2021].
  return $\{C_i\}$

4.4 Differentially Private Fair Labeling

The proposed method requires the values of sensitive variables only when calculating the balances in (7). This design choice ensures that sensitive variables of test samples are not required during inference, a common practice in fair machine learning studies. The calculation of balance requires the size of each cluster and the sum of the sensitive variables belonging to each cluster, which can be calculated assuming that the sensitive variable can have a value of zero or one. To guarantee data privacy, randomness can be added to the sum of sensitive variables of each cluster.

Algorithm 1 outlines the procedure for the proposed fair labeling method that ensures ϵ-DP, denoted as FLDP. The Laplace mechanism is utilized to provide DP protection. A key aspect of the algorithm is that Laplace noise is only added to the initial clusters, and not to subsequent iterations, due to the post-processing property of DP. Additionally, the noise amount is not impacted by the number of clusters, as the initial clusters are disjoint. In scenarios where the ownership of data and computational resources are separated, such as in a cloud service environment, the algorithm also preserves privacy against the modeler. This is achieved by requiring only the sum of sensitive variables for each initial cluster, rather than information about individual samples, thereby enhancing privacy. The data owner can calculate $\mathrm{Sum}_z(C_i)$ and provide it to the modeler for privacy preservation.

Proposition 1. *Algorithm 1 preserves ϵ-DP.*

Proof. We defer the proof to the Appendix. □

4.5 Fast and Differentially Private Fair Clustering

The proposed method can be consolidated in Algorithm 2, which involves two sub-procedures in addition to FLDP, namely GMDP and FindBPs. GMDP is a procedure for training a differentially private GM model, using a modified version of the DP-EM algorithm from [Park *et al.*, 2017; Byun *et al.*, 2023] that optimizes parallel composition. Note

Algorithm 2 Fast and differentially private fair clustering

procedure FairClusterDP(input data $\{\mathbf{x}_i\}_{i=1}^n \in [-1,1]^{n\times p}$, sensitive variable $\mathbf{z} \subset \{0,1\}^n$, desired number of clusters K, initial GM parameters $\{\mathbf{m}_k^0\}_{k=1}^\kappa, \{\mathbf{\Sigma}_k^0\}_{k=1}^\kappa$, total privacy budget (ϵ, δ), privacy budget for labeling ϵ', number of iterations τ, number of batches B, step size T)
  $\rho = (\sqrt{\log(1/\delta)+\epsilon} - \sqrt{\log(1/\delta)})^2 - \epsilon'^2/2$
  $\epsilon'' = \rho + 2\sqrt{\rho \log(1/\delta)}$
  $\{w_k\}_{k=1}^\kappa, \{\mathbf{m}_k\}_{k=1}^\kappa, \{\mathbf{\Sigma}_k\}_{k=1}^\kappa = \mathsf{GMDP}(\{\mathbf{x}_i\}_{i=1}^n, \{\mathbf{m}_k^0\}_{k=1}^\kappa, \{\mathbf{\Sigma}_k^0\}_{k=1}^\kappa, (\epsilon'', \delta), \tau, B)$
  Define f as (12).
  $\{\mathbf{b}_j\}_{j=1}^b = \mathsf{FindBPs}(\{\mathbf{m}_k\}_{k=1}^\kappa, f, T)$
  for k in $1, \ldots, \kappa$ **do**
    $C_k = \{\mathbf{x}_i | \mathbf{x}_i \in \overline{V(\mathbf{m}_k)}, i = 1, \ldots, n\}$
  $\{C_i\} = \mathsf{FLDP}(f, \mathbf{z}, K, \{C_k\}_{k=1}^\kappa, \{\mathbf{b}_j\}_{j=1}^b, \epsilon')$

that GMDP is used only when DP is obligatory for all variables, including non-sensitive ones, and may be substituted with a standard GM in most cases. FindBPs is a procedure for determining the BPs, as detailed in Section 4.3. The two sub-procedures are further detailed in the Appendix.

The proposed method also possesses an advantage in labeling unseen data compared to existing methods. Unlike fairlet-based approaches, which necessitate retraining for new samples due to the absence of fairlets for new samples, the proposed density-based fair clustering does not require retraining, leveraging the benefit of traditional density-based clustering where the Voronoi cells encompass the full data space. Therefore, the inference process in the proposed method is simpler than that of existing methods. Additionally, the method described in [Lee *et al.*, 2021] requires iterating the dynamical system to determine the corresponding SEPs (SEPs) for new samples, while the proposed method only requires determining the nearest centers. Another option to further incorporate density is to assign the samples to the sub-clusters with the highest density, i.e. $\mathrm{argmax}_k f_k(\mathbf{x})$ if $f(\mathbf{x}) = \sum_k f_k(\mathbf{x})$.

Proposition 2. *Algorithm 2 preserves (ϵ, δ)-DP.*

Proof. We defer the proof to the Appendix. □

5 Experiments

In this section, the proposed method is evaluated on a diverse set of synthetic and real-world datasets, and its performance is compared with that of existing methods.

5.1 Datasets

In contrast, **Gaussian** consists of 18 convex Gaussain blobs, which are well suited for simple clustering methods. The total number of samples for the **Gaussian** was 500. Following [Lee *et al.*, 2021], we assigned a sensitive variable to the two datasets to create a trade-off between clustering performance and fairness.

In the evaluation, two synthetic datasets, namely **Twomoon** and **18-Gaussian**, were generated to represent non-convex and convex clusters, respectively. The **Twomoon** dataset consisted of 1000 samples arranged in two curved clusters, while the **18-Gaussian** dataset comprised 18 convex Gaussian blobs, with a total of 500 samples, making it suitable for simple clustering methods. In accordance with [Lee *et al.*, 2021], a sensitive variable was assigned to each of the two datasets to create a balance between clustering performance and fairness.

For real-world datasets, three commonly used datasets in fair machine learning research, from the UCI machine learning repository, were utilized: **Bank** [Moro *et al.*, 2014], **Adult** [Kohavi and others, 1996], and **Diabetes** [Strack *et al.*, 2014]. In this section, the experimental results are mainly focused on the **Twomoon** and **Adult** datasets. Further results on the other datasets can be found in the Appendix.

5.2 Settings

The proposed method was compared against four other fair clustering methods, namely **FairSVC**[1] [Lee *et al.*, 2021], **Fairlet**[2] [Chierichetti *et al.*, 2017], **FairScale**[3] [Backurs *et al.*, 2019], and **FairAlg**[4] [Bera *et al.*, 2019]. In all experiments, the trade-off between fairness and clustering performance was observed for different numbers of clusters, $K = 2, \ldots, 9$. The balance was used as the fairness criterion, and Silhouette score [Rousseeuw, 1987] and Davies-Bouldin index [Davies and Bouldin, 1979] were used as the clustering performance metrics. The closer the balance was to one, the better the result, as the same number of samples was considered from each demographic group. A higher Silhouette score represents improved clustering results, while a lower Davies-Bouldin score suggests superior clustering performance. The computation time was also measured to evaluate the efficiency of the proposed method.

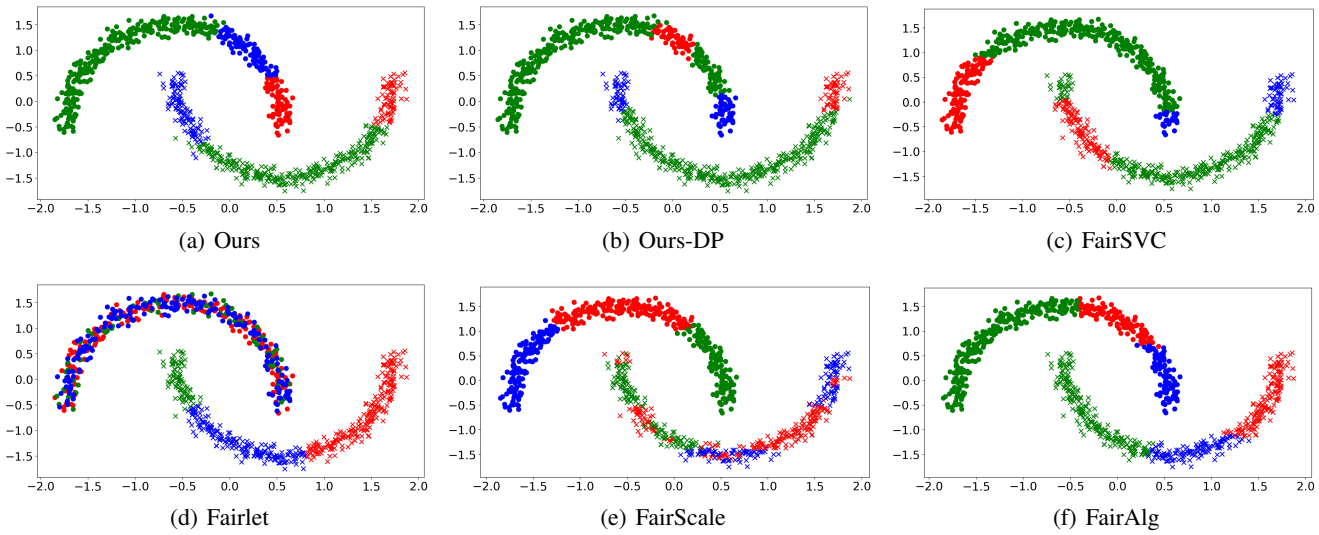
FairSVC was implemented using MATLAB, while the other methods were implemented using Python. However, the implementation of **FairScale** and **FairAlg** included calls to MATLAB and CPLEX APIs, respectively, within Python. All experiments were repeated ten times and the average of each metric was recorded.

5.3 Results

Results on Synthetic Datasets

As demonstrated in Figure 3, the outcomes of each method displayed a considerable disparity. The clustering results generated by **FairAlg** were comparable to those produced by unfair methods. In an effort to enhance balance, the fairlet decomposition methods **Fairlet** and **FairScale** resulted in distorted cluster shapes, where different clusters were intermingled. Conversely, **FairSVC** and the proposed method generated relatively natural and fair clustering results as the points belonging to the same original cluster (basin cell or Voronoi cell) were designated as the same cluster. It should be noted that, while the proposed method achieved a balance of 1, **FairSVC** achieved a balance of approximately 0.4. Figure 3(b) depicts the labeling outcome of the proposed method

[1] https://github.com/wj926/Fair_SVC
[2] https://github.com/guptakhil/fair-clustering-fairlets
[3] https://github.com/talwagner/fair_clustering
[4] https://github.com/nicolasjulioflores/fair_algorithms_for_clustering



(a) Ours (b) Ours-DP (c) FairSVC

(d) Fairlet (e) FairScale (f) FairAlg

Figure 3: Visualization of clustering results on Twomoon dataset with $K = 3$.

Dataset	Time (s)				
	Ours	FairSVC	Fairlet	FairScale	FairAlg
Twomoon	0.151	3.130 (x20.728)	499.861 (x3310.338)	4.230 (x28.013)	246.266 (x1630.901)
18-Gaussian	0.415	2.611 (x6.292)	55.238 (x133.104)	4.342 (x10.463)	38.75 (x93.373)

Table 1: Computation time on synthetic datasets. The numbers in parentheses indicate the ratio of the computation time of the compared methods to that of the proposed method.

when Laplace noise was incorporated to ensure DP ($\epsilon = 1$). This noise only impacts the dissimilarity between the initial clusters, thereby preserving the aforementioned property and resulting in relatively undisturbed cluster shapes.

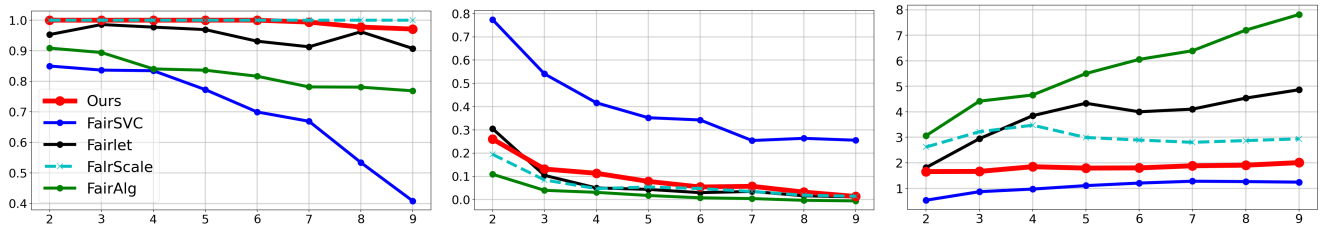
Table 1 compares the computation time of the proposed method with those of existing methods. It was found that the proposed method was significantly more efficient than the existing methods, particularly for non-convex datasets where solving optimization problems is more challenging. On the **Twomoon** dataset, the proposed method was approximately 20-30 times faster than **FairSVC** and **FairScale**, 1600 times faster than **FairAlg**, and 3000 times faster than **Fairlet**. On the **18-Gaussian** dataset, the proposed method was approximately 6.3 times faster than **FairSVC**, approximately 10.4 times faster than **FairScale**, approximately 90 times faster than **FairAlg**, and approximately 133 times faster than **Fairlet**. The results highlight the improved efficiency of the proposed method compared to the existing methods.

By considering the results obtained on the synthetic datasets, the proposed method demonstrated a better fairness-utility trade-off compared to **FairSVC** and natural clustering results compared to fairlet decomposition methods. Additionally, the proposed method showed remarkable computational efficiency, outperforming all other methods in terms of computation time. Nevertheless, the proposed method has a disadvantage in terms of cluster size uniformity. As illustrated in Figure 3(a), the clusters are not of equal size, with some clusters being significantly larger than others. This property may have a detrimental effect on the clustering metrics, as a large cluster may result in high intra-cluster measures and increase the numerator of the Davies-Bouldin index, leading to an overall increase in the index.

Results on Real-world Datasets

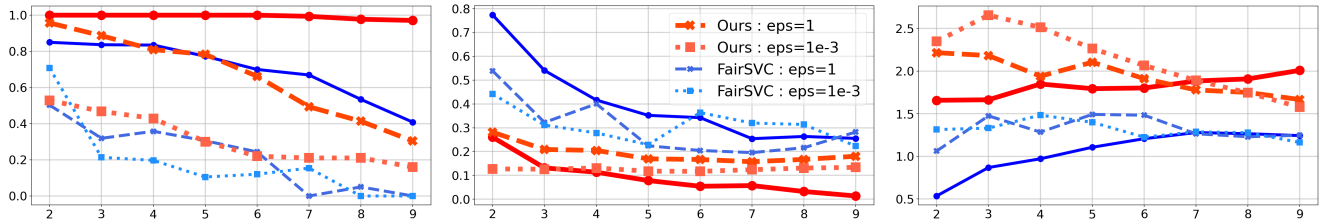
We first evaluated the performance of the proposed method without incorporating DP, as the other compared methods besides **FairSVC** are not capable of incorporating DP. The results in terms of balance score and clustering metrics on the **Adult** dataset are shown in Figure 4. Only **FairScale** maintained a perfect balance score for all values of K, and the proposed method demonstrated a slight decrease in balance score at high values of K. Meanwhile, **FairAlg** and **FairSVC** showed relatively low balance scores, with the balance score of **FairSVC** dropping to about 0.4 at $K = 9$. The proposed method achieved the best trade-off between fairness and utility, as evidenced by its generally higher Silhouette score and lower Davies-Bouldin index compared to the other methods, with the exception of **FairSVC**, which showed a significantly low balance score. The difference in clustering measures between the proposed method and **FairScale** is notable, further emphasizing the superior fairness-utility trade-off achieved by the proposed method.

Privacy-utility Trade-off We examined the impact of adding noise to the algorithm to preserve DP on the performance of the proposed method. We considered the scenario where only the privacy of sensitive variables was of concern and compared the proposed method with the **FairSVC** algorithm. The experiments for the scenario where the privacy of all variables are protected can be found in the Appendix. The privacy budget ϵ was set to $1, 0.001$, where a value of $\epsilon = 1$ represents moderate privacy and $\epsilon = 0.001$ represents very strong privacy. The results of the clustering on the **Adult** dataset under DP are depicted in Figure 5. The red lines in the figure represent the results of the proposed method and



(a) Balance (b) Silhouette (c) Davies-Bouldin

Figure 4: Clustering results on Adult dataset. The x-axis and y-axis indicate the number of clusters and the metric value, respectively.



(a) Balance (b) Silhouette (c) Davies-Bouldin

Figure 5: Clustering results with DP on Adult dataset. The x-axis and y-axis indicate the number of clusters and the metric value, respectively.

Dataset	Time (s)				
	Ours	FairSVC	Fairlet	FairScale	FairAlg
Adult	1.688	35.337 (x20.940)	504.064 (x298.690)	6.406 (x3.796)	229.944 (x135.664)
Bank	1.441	98.853 (x68.620)	614.567 (x426.608)	8.413 (x5.840)	242.060 (x168.028)
Diabetes	1.173	107.607 (x91.716)	582.217 (x496.235)	8.840 (x7.535)	158.047 (x134.706)

Table 2: Computation time on real-world datasets. The numbers in parentheses indicate the ratio of the computation time of the compared methods to that of the proposed method.

the blue lines represent the results of **FairSVC**. The balance of the proposed method was consistently higher than that of **FairSVC** for any value of K, regardless of the value of ϵ. Both methods showed a significant decrease in balance as K increased. However, while the balance of the proposed method with $\epsilon = 1$ remained above 0.3, and the balance with $\epsilon = 0.001$ was around 0.2, the balance of **FairSVC** with any ϵ dropped to close to zero for $K = 9$. Furthermore, the balance score of the proposed method showed a gradual decrease with the increase of privacy level, whereas the balance of **FairSVC** was not significantly affected by the value of ϵ. This indicates that the performance of **FairSVC** was severely impacted even with a small amount of added DP noise, while the proposed method was more resilient to such noise. In terms of clustering measures, **FairSVC** showed superior results at the cost of fairness. The proposed method displayed better clustering performance with $\epsilon = 1$ compared to $\epsilon = 0.001$, thus highlighting the clear decline in the fairness-utility trade-off as privacy level increased. However, no such pattern was observed in the case of **FairSVC**.

Computation Time The computation time for the three datasets is summarized in Table 2. The proposed method was found to be faster than the existing methods in all datasets, by at least several times. As the number of initial clusters used was higher for the real datasets than the synthetic datasets, the computation time of the proposed method increased in comparison. The computation times of **Fairlet** and **FairAlg** were mainly impacted by the number of samples and not significantly affected by the dimension of the dataset. Among the compared methods, **FairScale** was the fastest and was at least 3.7 times slower than the proposed method. In comparison with the proposed method, the computation time of the other methods was found to be high, ranging from tens of times to several hundred times, depending on the dataset.

In summary, the proposed method demonstrates superior results in balancing fairness and utility, and has been shown to be significantly faster than existing methods. Additionally, it demonstrates greater compatibility with DP in comparison to FairSVC, which is currently the only available method that supports DP implementation.

6 Discussion

We have proposed a new density-based fair clustering approach that meets DP standards. By incorporating the GM density function and Voronoi cell, the proposed method improves upon existing methods in every aspect, while maintaining DP with limited randomness. The experimental results demonstrate the superior performance of the proposed method in terms of privacy, fairness, utility, and efficiency. To further enhance input privacy, the proposed method can be combined with cryptographic techniques such as homomorphic encryption and secure multi-party computation.

Ethical Statement

Numerous instances have been documented wherein machine learning algorithms exhibit discriminatory behavior against various demographic groups, with adverse outcomes for vulnerable populations. The focus of our study is to directly confront the problem of social inequality by presenting a fair machine learning model that aims to provide equitable opportunities to multiple demographic groups. As such, our proposed model is significant in the context of advancing two of the Sustainable Development Goals, specifically Gender Equality and Reducing Inequality.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (No. 2019R1A2C2002358, 2022R1A5A6000840) and by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT) (No. 2021-0-02068, 2022-0-00984).

References

[Agarwal *et al.*, 2018] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. A reductions approach to fair classification. In *International Conference on Machine Learning*, pages 60–69. PMLR, 2018.

[Ahmadian *et al.*, 2020a] Sara Ahmadian, Alessandro Epasto, Marina Knittel, Ravi Kumar, Mohammad Mahdian, Benjamin Moseley, Philip Pham, Sergei Vassilvitskii, and Yuyan Wang. Fair hierarchical clustering. *Advances in Neural Information Processing Systems*, 33:21050–21060, 2020.

[Ahmadian *et al.*, 2020b] Sara Ahmadian, Alessandro Epasto, Ravi Kumar, and Mohammad Mahdian. Fair correlation clustering. In *International Conference on Artificial Intelligence and Statistics*, pages 4195–4205. PMLR, 2020.

[Backurs *et al.*, 2019] Arturs Backurs, Piotr Indyk, Krzysztof Onak, Baruch Schieber, Ali Vakilian, and Tal Wagner. Scalable fair clustering. In *International Conference on Machine Learning*, pages 405–413. PMLR, 2019.

[Bera *et al.*, 2019] Suman Bera, Deeparnab Chakrabarty, Nicolas Flores, and Maryam Negahbani. Fair algorithms for clustering. *Advances in Neural Information Processing Systems*, 32, 2019.

[Berk *et al.*, 2017] Richard Berk, Hoda Heidari, Shahin Jabbari, Matthew Joseph, Michael Kearns, Jamie Morgenstern, Seth Neel, and Aaron Roth. A convex framework for fair regression. *arXiv preprint arXiv:1706.02409*, 2017.

[Byun *et al.*, 2023] Junyoung Byun, Yujin Choi, and Jaewook Lee. Improving the utility of differentially private clustering through dynamical processing. *arXiv preprint arXiv:2304.13886*, 2023.

[Chierichetti *et al.*, 2017] Flavio Chierichetti, Ravi Kumar, Silvio Lattanzi, and Sergei Vassilvitskii. Fair clustering through fairlets. *Advances in Neural Information Processing Systems*, 30, 2017.

[Cummings *et al.*, 2019] Rachel Cummings, Varun Gupta, Dhamma Kimpara, and Jamie Morgenstern. On the compatibility of privacy and fairness. In *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*, pages 309–315, 2019.

[Davies and Bouldin, 1979] David L Davies and Donald W Bouldin. A cluster separation measure. *IEEE transactions on pattern analysis and machine intelligence*, (2):224–227, 1979.

[Dwork, 2006] Cynthia Dwork. Differential privacy. In *Automata, Languages and Programming: 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II 33*, pages 1–12. Springer, 2006.

[Howard and Borenstein, 2018] Ayanna Howard and Jason Borenstein. The ugly truth about ourselves and our robot creations: the problem of bias and social inequity. *Science and engineering ethics*, 24(5):1521–1536, 2018.

[Huang *et al.*, 2019] Lingxiao Huang, Shaofeng Jiang, and Nisheeth Vishnoi. Coresets for clustering with fairness constraints. *Advances in Neural Information Processing Systems*, 32, 2019.

[Jagielski *et al.*, 2019] Matthew Jagielski, Michael Kearns, Jieming Mao, Alina Oprea, Aaron Roth, Saeed Sharifi-Malvajerdi, and Jonathan Ullman. Differentially private fair learning. In *International Conference on Machine Learning*, pages 3000–3008. PMLR, 2019.

[Kim *et al.*, 2014] Kyoungok Kim, Youngdoo Son, and Jaewook Lee. Voronoi cell-based clustering using a kernel support. *IEEE Transactions on Knowledge and Data Engineering*, 27(4):1146–1156, 2014.

[Kohavi and others, 1996] Ron Kohavi et al. Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. In *Kdd*, volume 96, pages 202–207, 1996.

[Krieger and Fiske, 2006] Linda Hamilton Krieger and Susan T Fiske. Behavioral realism in employment discrimination law: Implicit bias and disparate treatment. *California Law Review*, 94(4):997–1062, 2006.

[Lahoti *et al.*, 2020] Preethi Lahoti, Alex Beutel, Jilin Chen, Kang Lee, Flavien Prost, Nithum Thain, Xuezhi Wang, and Ed Chi. Fairness without demographics through adversarially reweighted learning. *Advances in neural information processing systems*, 33:728–740, 2020.

[Lee and Lee, 2006] Jaewook Lee and Daewon Lee. Dynamic characterization of cluster structures for robust and inductive support vector clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1869–1874, 2006.

[Lee *et al.*, 2021] Woojin Lee, Hyungjin Ko, Junyoung Byun, Taeho Yoon, and Jaewook Lee. Fair clustering with fair correspondence distribution. *Information Sciences*, 581:155–178, 2021.

[Li *et al.*, 2020] Peizhao Li, Han Zhao, and Hongfu Liu. Deep fair clustering for visual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9070–9079, 2020.

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

[Moro *et al.*, 2014] Sérgio Moro, Paulo Cortez, and Paulo Rita. A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, 62:22–31, 2014.

[Mozannar *et al.*, 2020] Hussein Mozannar, Mesrob Ohannessian, and Nathan Srebro. Fair learning with private demographic data. In *International Conference on Machine Learning*, pages 7066–7075. PMLR, 2020.

[Park *et al.*, 2017] Mijung Park, James Foulds, Kamalika Choudhary, and Max Welling. Dp-em: Differentially private expectation maximization. In *Artificial Intelligence and Statistics*, pages 896–904. PMLR, 2017.

[Platt, 1998] John Platt. Fast training of support vector machines using sequential minimal optimization. In *Advances in Kernel Methods - Support Vector Learning*. MIT Press, January 1998.

[Rousseeuw, 1987] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

[Strack *et al.*, 2014] Beata Strack, Jonathan P DeShazo, Chris Gennings, Juan L Olmo, Sebastian Ventura, Krzysztof J Cios, and John N Clore. Impact of hba1c measurement on hospital readmission rates: analysis of 70,000 clinical database patient records. *BioMed research international*, 2014, 2014.

[Tax and Duin, 1999] David MJ Tax and Robert PW Duin. Support vector domain description. *Pattern recognition letters*, 20(11-13):1191–1199, 1999.

[Tran *et al.*, 2021] Cuong Tran, Ferdinando Fioretto, and Pascal Van Hentenryck. Differentially private and fair deep learning: A lagrangian dual approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 9932–9939, 2021.

[Vinuesa *et al.*, 2020] Ricardo Vinuesa, Hossein Azizpour, Iolanda Leite, Madeline Balaam, Virginia Dignum, Sami Domisch, Anna Felländer, Simone Daniela Langhans, Max Tegmark, and Francesco Fuso Nerini. The role of artificial intelligence in achieving the sustainable development goals. *Nature communications*, 11(1):233, 2020.

[Zafar *et al.*, 2017] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P Gummadi. Fairness constraints: Mechanisms for fair classification. In *Artificial intelligence and statistics*, pages 962–970. PMLR, 2017.