SynthFair: Ensuring Subgroup Fairness in Classification via Synthetic Data Generation

Begüm Hattatoğlu[1], Abdulhakim A. Qahtan[1⋆], Heysem Kaya[1], and Yannis Velegrakis[1,2]

[1] Department of Information and Computing Sciences
Utrecht University, Utrecht 3584CC, The Netherlands
[2] Department of Information Engineering and Computer Sciences
University of Trento, Trento, Italy

Abstract. Machine Learning (ML) models are used in a wide range of applications, which affects societies either directly or indirectly in daily life. Ensuring fairness in the decisions of these applications is a challenging task that has attracted the attention of researchers from different fields. However, considering a single sensitive attribute when measuring the fairness of a dataset or the outcomes of an ML model could be misleading when the data contains multiple sensitive attributes. In this paper, we study the problem of unfair decisions of the ML models for data with multiple sensitive attributes. The sensitive attributes are used to define the different demographic subgroups. Our study shows that the imbalanced representation of the different demographic subgroups in the population is one of the most important reasons behind the biased predictions of the ML models. To handle this problem, we propose a framework called 'SynthFair' to ensure fairness among the subgroups without changing the original class labels or removing the sensitive attributes from the data. SynthFair uses synthetic data generation for ensuring fair classification such that classifiers are trained on balanced datasets with similar number of records per subgroup. Experimental results over widely used benchmarks show that our framework yields consistent improvements compared to a set of bias mitigation methods.

Keywords: Machine learning, fairness measures, bias mitigation algorithms, clustering, synthetic data, classification

1 Introduction

The prevalence of Machine Learning (ML) in a wide range of applications has significantly affected the daily life. Machine learning models can handle big volumes of data for complex computational tasks. They are used for decision-making in business and government systems [24], in recommender systems, advertisements, hiring systems, and others. Besides, people usually have subjective opinions and points of view that might lead to bias in their decisions, which can be avoided

⋆ Corresponding author

using ML models. Unfortunately, ML models are not always objective. A large number of models have been identified to show bias against specific groups of the society [1, 20, 26], including Amazon's free same-day delivery and the COMPAS recidivism estimation tool. These tools show significant discrimination against specific neighborhood and community subgroups.

The bias in the outcomes of the ML models can be a result of the bias in the training data, the representativeness of the classes or the absence of informative features. In order to identify and quantify the bias in the datasets or the outcomes of the machine learning models, different fairness measures were proposed [6, 8, 11, 12, 17, 27, 30]. Based on these measures, different bias mitigation algorithms [3, 11, 12, 14–18, 22, 25, 29, 31–33] have been developed to reduce the bias in the outcomes of the ML models. Most of these algorithms focus on reducing the bias by optimizing the algorithms according to a given fairness measure. However, there is no consensus on the best bias mitigation algorithm to be used and which fairness measure should be considered [13].

In this paper, we propose SynthFair, a pre-processing framework that mitigates the bias in the outcomes of the ML models. SynthFair generates synthetic data to balance the representation of the different subgroups in the training dataset. Obviously, the generated synthetic examples (records) belong to the under-represented subgroups in the dataset. We use the Synthetic Minority Over-sampling Technique (SMOTE) [4] and the conditional Generative Adversarial Networks (cGANs) [28] to generate the synthetic data. However, it is important to produce synthetic examples that mimic the patterns in the real data. SMOTE interpolates the examples from a given class, which may generate examples that are deep within the other class. Therefore, we cluster the data and generate examples within the clusters to improve the quality of the generated examples. This is based on the fact that original data examples in each cluster have higher similarity to each other. After generating the synthetic data using SMOTE, we introduce three strategies to train the classifiers: i) combine the data in all clusters and train a single classifier; ii) train a classifier per cluster, assign each test sample to the nearest cluster, and use the classifier of that cluster for predicting the label of the test samples; iii) use a weighing mechanism to determine the contribution of each classifier in deciding the labels of the new samples.

Moreover, we also use GANs for generating the synthetic examples as they have been proven to be effective in generating examples that are similar to the original examples and in capturing the statistical patterns in the data. We use a special type of GANs that is called conditional GANs (cGANs), where we condition the GANs to generate examples from the under-represented groups only. For cGANs, we do not cluster the data before generating the synthetic examples so we train a single classifier using all the training data.

To summarize, our contributions are as follows.

– We provide a theoretical analysis for mitigating the bias in datasets and ML models' outcomes based on the disparate impact ratio.

– We develop a bias-mitigation framework that consistently improves the fairness by generating high quality examples from the minority class.

– We introduce multiple strategies for training fair classifiers and evaluate our framework on three real world datasets that are widely used as benchmarks for evaluating the fairness of the ML models.

2 Background and Related Work

In this paper, we propose a pre-processing framework that mitigates bias in a given dataset by generating synthetic examples to balance the dataset. We decompose the dataset D into three subsets of attributes $D = \{X, S, Y\}$. Here, X represents the set of attributes that do not contain sensitive information regarding individuals, S is the set of sensitive attributes containing sensitive information, and $Y \in \{-ve, +ve\}$ is the class label. Let the $+ve$ label represent the favorable class label. We use G_p/G_u to represent the examples of privileged/unprivileged groups, respectively. We denote the set of predicted labels by $\widehat{Y}$. It is worth noting that X, S, Y form a column-wise partitioning of the attributes of D, while the demographic groups G_p/G_u partition the data in a row-wise fashion. The used notations are described in Table 1.

Notation	Description
$D(X,S,Y)$	training dataset
$T(X,S,Y)$	testing dataset
X	the set of attributes with non-sensitive information about individuals
S	the set of attributes with sensitive information
$Y/\widehat{Y}$	the original/predicted class labels of the instances in a given dataset, respectively
D_{G_p}	$D_{G_p} = \{\mathbf{x} \in D \mid S(\mathbf{x}) = G_p\}$ the set of records that belong to the privileged group
D_{G_u}	$D_{G_u} = \{\mathbf{x} \in D \mid S(\mathbf{x}) = G_u\}$ the set of records that belong to the unprivileged group
$D^+_{G_p}$	$D^+_{G_p} = \{\boldsymbol{x} \in D_G \mid Y(\boldsymbol{x}) = 1\}$
$D_{G_u}+$	$D_{G_u}+ = \{\boldsymbol{x} \in D_{G_u} \mid Y(\boldsymbol{x}) = 1\}$
$N^+_{G_p}, N^+_{G_u}$	$N^+_{G_p} = \lvert D^+_{G_p}\rvert$, $N^+_{G_u} = \lvert D^+_{G_u}\rvert$
N^+	$N^+ = N^+_{G_p} + N^+_{G_u}$
F_{m_i}	fairness metric
A_{m_i}	performance metric

Table 1: Notation.

After profiling a set of benchmark datasets that are used for evaluating the fairness of the ML models, we found that the datasets are imbalanced, either in the representativeness or in the ratio of getting the desirable class label for the different demographic groups. Hence, we focus on solving the data imbalance problem by proposing a framework for generating more examples from the underrepresented group/class.

2.1 Fairness Measures

The problem of algorithmic fairness has been extensively studied during the last decade. A set of studies focused on formulating fairness measures that can de-

termine the bias in the datasets and in the outcomes of the ML models. Other studies focused on improving fairness according to one or more of the fairness measures. Our work is mainly focused on proposing a pre-processing bias mitigation framework.

Various algorithmic fairness measures have been formulated to quantify the bias in a dataset or the outcomes of an ML model. They can be used to measure bias in different stages of the machine learning pipeline. A set of these measures ensures the fairness between the different demographic groups such as *Demographic Parity* [8, 17], *Predictive Parity* [27], *Equalized Odds* [30], *Equal Opportunity* [12], *Overall Accuracy Equality* [2] and *Treatment Equality* [2]. These measures ensure that the rate of getting the favorable class label is almost the same for the different demographic groups. Other fairness measures consider the outcomes on the individual level. Examples of this type of fairness measure include *individual fairness* [8] and *Consistency* [32]. These measures consider an algorithm to be fair if it provides the same output for two individuals who have different values only in the sensitive attributes. In what follows, we provide formal definitions of five common fairness measures that are also used in our study.

Demographic Parity (DP): this measure states that the instances in both unprivileged and privileged groups should have equal probability of getting the favorable class label. That is: $P\left[\widehat{Y} = 1 \mid S = G_u\right] = P\left[\widehat{Y} = 1 \mid S = G_p\right]$. The same definition can be applied to measure the bias in the original dataset by substituting $\widehat{Y}$ by the original labels Y.

Disparate Impact Ratio (DIR): is defined as the ratio between the probability of privileged and unprivileged groups getting the positive outcomes. According to the American Civil Rights Act [9], a dataset or a classifier is considered fair if its DIR is at least 0.8. DIR can be formulated as:

$$\mathrm{DIR(D)} = \frac{P\left[\widehat{Y} = 1 | S = G_u\right]}{P\left[\widehat{Y} = 1 | S = G_p\right]}, \tag{1}$$

to measure the fairness of the ML model. Our target is increasing DIR such that it becomes greater than 0.8, ideally reaching 1.

Equalized Odds (EO): this measure states that instances from privileged and unprivileged groups should have equal True Positive Rate (TPR) and False Positive Rate (FPR). That is:

$$P\left[\widehat{Y} = 1 \mid S = G_u, Y = 1\right] = P\left[\widehat{Y} = 1 \mid S = G_p, Y| = 1\right] \wedge$$
$$P\left[\widehat{Y} = 1 \mid S = G_u, Y = 0\right] = P\left[\widehat{Y} = 1 \mid S = G_p, Y = 0\right].$$

Predictive Parity: to deem a classifier as fair in terms of predictive parity, both protected and unprotected groups should have the same positive predictive value. It is formalized as: $P\left[Y = 1 \mid \widehat{Y} = 1, S = G_p\right] = P\left[Y = 1 \mid \widehat{Y} = 1, S = G_u\right]$.

Consistency: this individual fairness measure determines how similar the labels are for similar instances in a dataset based on the k-neighbors of the instance. Thus, instances should have the same labels if they are similar in terms of features. This measure is formulated as:

$$\text{Consistency} = 1 - \frac{1}{|D|}\sum_{i=1}^{|D|}\left|\widehat{y}_i - \frac{1}{k}\sum_{\mathbf{x}_j \in kNN(\mathbf{x}_i)}\widehat{y}_j\right|,$$

where $kNN(\mathbf{x})$ represents the set of the closest k neighbors of instance $\mathbf{x}$.

It should be noted that statistical measures cannot guarantee fairness for individuals or more fine-grained sub-groups of the unprivileged groups [6]. Moreover, treating the individuals similarly does not necessarily imply fair treatment. Furthermore, there is a disagreement among different fairness measures since their goals and the considered criteria are different, which is formalized and proven with the *impossibility theorem* [5, 19, 21]. According to this theorem, it is impossible to satisfy both *equalized odds* and *predictive parity*.

2.2 Bias Mitigation Algorithms

There are several bias mitigation techniques (algorithms) that improve fairness while taking the performance of ML models in consideration. These algorithms fall in three categories:

Pre-processing algorithms solve the problem by mitigating the bias in the training data so classifiers do not learn the bias. Examples include *preferential sampling* [14] and oversampling techniques such as *massaging* [15]. Other techniques remove causal relationship between the sensitive attributes and the decision variable such as *interventional fairness* [25], or change the weight of the records (e.g., *reweighing* [3]). In *learning fair representations* (LFR) [32], the goal is to find an appropriate intermediate representation of a given dataset that encodes the data as accurately as possible while concealing any information about the sensitive attributes. *Fair class balancing* [29] on the other hand, balances the classes without considering the subgroups.

In-Processing Algorithms train the classifiers to improve the fairness according to a specific measure instead of focusing only on the performance measures. They are mostly limited to the chosen classifier. Zafar et al. [31] used *regularized logistic regression* and regularized support vector machines. Kamishima et al. [17] proposed regularized prejudice remover, which can be applied on any probabilistic classifier. Adversarial learning is used as an in-processing technique

Datasets	German			Adult			COMPAS		
Ratios	SG (%)	+ve (%)	-ve (%)	SG (%)	+ve (%)	-ve (%)	SG (%)	+ve (%)	-ve (%)
a1: 0, a2: 0	10.5	5.8	4.7	4.8	1	3.8	49.8	22.1	27.7
a1: 1, a2: 0	20.5	14.3	6.2	21.7	6.5	15.2	30.7	18.3	12.4
a1: 0, a2: 1	8.5	5.2	3.3	7.5	3.5	4	10.4	6.6	3.8
a1: 1, a2: 1	60.5	44.7	15.8	66	39	27	9.1	5.9	3.2

Table 2: The ratios of the subgroups that exist in the German, Adult and COMPAS datasets. These datasets are widely used for testing bias mitigation algorithms. The attributes "a1" and "a2" correspond to the sensitive attributes ([Age, Gender] in German, [Gender, Race] in Adult, and [Gender, Race] in COMPAS) and (+ve = positive, and -ve = negative)

to ensure fairness in [33]. Ristanoski [22] proposed an empirical loss-based tuning on support vector machines.

Post-Processing Algorithms change the predicted outcomes of classifiers based on certain rules or constraints to ensure fairness. This type of algorithms include *reject option classification* [16], *ensure equalized odds* [12], *avoid proxy discrimination* and *avoid unresolved discrimination* [18]. Unfortunately, the post-processing might incur ethical considerations when adjusting labels of examples that might deserve to receive the favorable class label.

3 Theoretical Analysis

In this section, we perform a theoretical analysis to improve algorithmic fairness. In our analysis, we target to minimize the impact of improving fairness on the performance of the classifiers.

Even though the fairness definitions based on the measures given in Section 2.1 are clear, there is no agreement on what should be considered fair. For example, for an ML model to be deemed fair according to the the equalized odds (EO), the difference between the terms on the different side of the equal sign should be close to 0. However, there is no agreement on the cutoff value to consider the algorithm as fair or not. The only exception is the Disparate Impact Ratio (DIR) which is based on the 80% rule defined in US law. For this reason, in our analysis, we focus on improving the DIR.

How to increase the value of the DIR? Let $|D|$ be the number of instances in the dataset D, N^+ be the total number positive examples in the dataset, $N^+_{G_p}/N^+_{G_u}$ be the number positive examples from the privileged/unprivileged groups, respectively. Let ξ be the percentage value of DIR that is computed from the original dataset (DIR(D) $= \xi/100$). Our goal is to increase the value of DIR by $\delta/100$, with $0 < \delta < 125 - \xi$, to make DIR(C) close to or greater than 80%, where C is a given classifier. To do so, we should increase (or decrease) the number of positive predictions from the unprivileged (privileged) groups. However, increasing the positive predictions of the unprivileged groups is favored because of the ethical considerations.

Let $p(Y = 1 \mid S = G_p) = \frac{N^+_{G_p}}{N_{G_p}}$, and $p(Y = 1 \mid S = G_u) = \frac{N^+_{G_u}}{N_{G_u}}$. Since, DIR(D) = ξ% then:

$$\frac{N^+_{G_u}/N_{G_u}}{N^+_{G_p}/N_{G_p}} = \frac{\xi}{100} \text{ and } N^+_{G_u} = \frac{\xi N_{G_u} N^+_{G_p}}{100 N_{G_p}}. \tag{2}$$

To increase the value of DIR(C) to $(\xi + \delta)$%, we need:

$$\frac{\left(N^+_{G_u} + \epsilon\right)/N_{G_u}}{\left(N_{G_p} - \gamma\right)/N_{G_p}} = \frac{\xi + \delta}{100}, \tag{3}$$

where ϵ is the number of instances (records) from the unprivileged group that should be predicted positive while their original label is negative. Conceptually, ϵ can take any integer value between 0 and $(N_{G_u} - N^+_{G_u})$. Conversely, $0 < \gamma < N^+_{G_p}$ is the number of instances from the privileged group that should be predicted negative while their original label is positive. Solving for ϵ and γ, we first get:

$$\frac{\left(N^+_{G_u} + \epsilon\right) N_{G_p}}{\left(N^+_{G_p} - \gamma\right) N_{G_u}} = \frac{\xi + \delta}{100}. \tag{4}$$

Substituting $N^+_{G_u}$ from Eq. (2) in Eq. (4), we get:

$$(\xi + \delta)\left(N^+_{G_p} - \gamma\right) N_{G_u} = 100 N_{G_p}\left(\frac{\xi N_{G_u} N^+_{G_p}}{100 N_{G_p}} + \epsilon\right). \tag{5}$$

Hence: $100\epsilon N_{G_p} + \gamma\left(\xi + \delta\right) N_{G_u} = \delta N^+_{G_p} N_{G_u}$

Consequently, we can distinguish between three special cases: 1) $\epsilon = \gamma$: we need to increase the positive predictions from the unprivileged group by $\epsilon = \frac{\delta N^+_{G_p} N_{G_u}}{100 N_{G_p} + (\xi+\delta) N_{G_p}}$ and decrease the positive predictions from the privileged group by the same number. 2) $\gamma = 0$: we need to increase the positive predictions from the unprivileged group by $\epsilon = \frac{\delta N^+_{G_p} N_{G_u}}{100 N_{G_p}}$ while keeping the same number of positives from the privileged group. 3) $\epsilon = 0$: we need to decrease the positive predictions from the privileged group by $\gamma = \frac{\delta N^+_{G_p} N_{G_u}}{(\xi+\delta) N_{G_u}}$ while keeping the same number of positives from the unprivileged group.

Proposition 1. *Since the number of instances (records) from the unprivileged group is significantly smaller than the number of instances from the privileged group, it can be easily shown that increasing the number of positive records of the unprivileged group while keeping the number of positives from the privileged group unchanged will incur the minimum number of changes (case* **2***: $\gamma = 0$).*

Our solution: based on Proposition 1, mitigating the bias in the outcome of the ML models can be, preferably, achieved by increasing the number of positive predictions for instances from the unprivileged group. In this work, we use SMOTE [4] for generating synthetic examples from the minority groups. We improve the quality of the generated instances by interpolating the most similar instances by clustering the instances before applying SMOTE. In this way, we make sure that there are enough examples with the positive label from the unprivileged group to train unbiased models. We expect this solution to increase the probability of the positive predictions for the unprivileged group.

Effects of increasing the DIR value: it is clear that improving the DIR measure would affect the performance measures. For example, with an oracle classifier (that has 100% accuracy), then according to Eq. (3), increasing DIR by δ will have the following effects: i) True Positives (TP) will be decreased by γ (i.e. $TP' = TP - \gamma$). ii) True Negatives will be decreased by ϵ (i.e. $TN' = TN - \epsilon$); iii) The False Positives (FP) will be increased by ϵ ($FP' = FP + \epsilon$) and the False Negatives (FN) will be increased by γ ($FN' = FN + \gamma$). Thus, the oracle classifier's accuracy will be decreased by $\left(\frac{\gamma+\epsilon}{|D|}\right)$. If F_1' is the new $F1\text{-}Score$, then $F_1' = \frac{2*(TP-\gamma)}{2*TP+FN+FP+\epsilon-\gamma}$. For the case of perfect classifier, $F_1 = 1$ and $F_1' = \frac{2(TP-\gamma)}{2TP-\gamma+\epsilon}$. The decrease in the $F1\text{-}Score$ will be $1 - \frac{2(TP-\gamma)}{2TP-\gamma+\epsilon}$.

4 The SynthFair Framework

The SynthFair framework consists of three main steps: i) data preparation step where we identify the subgroup IDs of each example; ii) synthetic data generation step; and iii) classification step. The three steps are summarized in Algorithm 1.

4.1 Data Preparation

In the preparation step, we identify the subgroup IDs, add this information as a new variable to the dataset, and split the dataset into training and testing with stratification. We assume that the sensitive attributes and the label are binary attributes. In our experiments, we have two binary sensitive attributes and one binary class label in every dataset. The subgroup that has unfavorable values in both sensitive attributes is considered the most unprivileged subgroup, whereas the subgroup with favorable values for both sensitive attributes is considered the most privileged subgroup. The other subgroups that have different combinations of favorable and unfavorable values for different sensitive attributes are interpreted as both privileged and unprivileged subgroups. We believe that reducing the bias between the most privileged/unprivileged subgroups will minimize the bias between all the subgroups in the dataset.

After adding the subgroup ID variable, the sensitive attributes are removed from the dataset since the new subgroup IDs contain information regarding these sensitive attributes. Moreover, if a dataset contains a set of numerical variables,

Algorithm 1 The SynthFair Framework

Input: data $D = \{x_1, ..., x_n\}$, train-test split ratio ρ, sensitive attributes S, label Y, Strategy, SDG $\in$ {SMOTE, cGANs}
Output: Fairness and Performance measures' values.

for each $\boldsymbol{x} \in D$ **do**
  $G_x \leftarrow$ subgroup$(\boldsymbol{x}, S, Y)$ // Identify subgroup ID
end for
$A_{G_x} \leftarrow \{G_x, \forall x \in D\}$ // Create an attribute for subgroup IDs
$D_{train}, D_{test} \leftarrow Split(D, train, test, \rho)$
if SDG_Tech == SMOTE **then**
  $CS =$ SMOTE_Gen(D_{train}) // Cluster set CS from SMOTE algorithm
else
  cGAN_model = cGANs(D_{train})
  $ma = \underset{A_{G_x} \subseteq D_{train}}{\text{argmax}} |A_{G_x}|$
  for A_{G_x} in D_{train} **do**
    $A'_{G_x} \leftarrow cGAN\_model.generate\,(A_{G_x}, ma)$
    $D'_{train} \leftarrow D_{train} \cup A'_{G_x}$
  end for
end if
if Strategy == 1 **then**
  $D'_{train} \leftarrow \bigcup_{C_i \in CS} C'_i$
  Train a model M using D'_{train}
end if
if Strategy == 2 **or** Strategy == 3 **then**
  for i = 1 to m **do**
    train a model M_i using C'_i data
  end for
end if
labels $\leftarrow \{\}$
for $\boldsymbol{x} \in D_{test}$ **do**
  labels $\leftarrow$ labels $\cup$ {($\boldsymbol{x}$, class($\boldsymbol{x}$))}
end for
for subgroup in subgroups **do**
  Calculate fairness and performance measures
end for

these variables are standardized (scaled to a specific range) in both training and test sets separately. This step ensures that no variables can dominate the calculation of distance metrics during the various steps.

4.2 Synthetic Data Generation (SDG)

From the subgroup IDs, we identify the subgroup that has the maximum number of examples. After that, we generate synthetic examples from the other subgroups such that the final training set contains the same number of examples from each subgroup. We deploy two methods for generating the synthetic exam-

ples, namely, the Synthetic Minority Oversampling Technique (SMOTE) [4] and the conditional Generative Adversarial Networks (cGANs) [28].

SMOTE: creates new synthetic examples by linearly interpolating two examples that are close to each other in the feature space and belong to the same class. The generated examples are produced along the lines that connect the existing examples. A major drawback of SMOTE is that it might introduce the artificial minority class examples too deeply in the majority class space. Because of that, the interpolated examples that are used to generate the new example must be close enough to each other. Therefore, we cluster the training data into groups such that the used examples for oversampling are close to each other.

Clustering: for clustering the training data, we use the **fuzzy c-means clustering** [23], which is a soft clustering algorithm that allows each example in a dataset to be assigned to more than one cluster. In fuzzy clustering, each example belongs to a cluster with a certain probability which adds up to 1 in total. In fuzzy c-means, clusters are formed based on a distance measure (such as Euclidean distance), which is used to calculate (and minimize sum of) the distances between the examples and the assigned cluster centroids. Thus, it is important to apply standardization on the numerical features of the datasets to prevent an unjustified domination these features as we mentioned earlier in the data preparation step. Since fuzzy c-means requires the number of clusters to be given as an input, we run the fuzzy c-means multiple times using a predefined set of values for the number of clusters. In each run, we compute the **fuzzy partition coefficient** (FPC) and the **silhouette score**. We choose the number of clusters that yields the best combination of these two values. Let $CS = \{cs_1, \ldots, cs_k\}$ be the set of clusters. After partitioning the training set into k clusters CS, using the cluster memberships of training examples, we generate synthetic examples to balance the subgroups within each cluster.

Classification: in this step, a classification algorithm of choice or multiple classification algorithms are trained according to one of three strategies:

Stra1) Using Single Classification Model: After generating the required examples in each cluster, the clusters are concatenated together to form a single large training set. Using this balanced dataset, a classifier is trained and the labels are predicted based on its decisions.

Stra2) Using Cluster Membership for Prediction: The data in each cluster is used to train a different classifier. Examples from the test set are initially assigned to the cluster that returns the highest membership probability. Then, the label of a given test example that is assigned to cluster cs_j is decided based on the outcome of the classifier that was trained using the data in the cluster cs_j.

Stra3) Using Weighted Cluster Memberships for Prediction: Similar to the Stra2, we train multiple classifiers using the data in each cluster (one classifier per

cluster). However, instead of choosing one classifier in Stra2, all the trained classifiers are taken into consideration when predicting the class label of a test example. First, the fuzzy c-means model is used to retrieve the membership probabilities of the test example to the clusters that include examples with the same subgroup ID as the test example. After that, we use the membership probabilities as a weight for the predicted class label from each corresponding classifier. That is, for a test example e_i with a subgroup id r, and $CS_r \subseteq CS$ be the subset of clusters that include examples from the subgroup r, $\tilde{Y}(e_i) = \sum_{cs_j \in CS_r} w(e_i, cs_j) C_j(e_i)$, where $w(e_i, cs_j) = \frac{p(e_i \in cs_j)}{\sum_{cs_q \in CS_r} p(e_i \in cs_q)}$ and $C_j(e_i)$ is the outcome of classifier C_j, which was trained on cs_j, for e_i. Finally, we assign the class label based on the value of $\tilde{y}(e_i)$ such that $\tilde{y}(e_i) > 0.5 \implies \widehat{Y}(e_i) = +ve$, and $\widehat{Y}(e_i) = -ve$, otherwise.

CGAN Because of the high performance of the Generative Adversarial Networks [10] in creating synthetic data, we used a specific type of GANs that is called cGAN [28]. With cGANs, a set of conditions can be specified to decide the number and the shape of the synthetic examples as they can model complex distributions. In our framework, we specify the conditions on the subgroups IDs to generate examples from all subgroups except the subgroup with the max number of examples in the training set. Since the training process requires a large number of examples to learn from, we use all the examples in the initial training set to train the cGANs. For this reason, we do not use the previous strategies. Instead we train a single model using the new training set with the same number of examples from each subgroup.

5 Evaluation

To evaluate the performance of the SynthFair framework in terms of fairness and performance, we compared SynthFair with two bias mitigation methods. We used three benchmark datasets that are widely used for evaluating the fairness algorithms. The different methods are evaluated in terms of the Disparate Impact Ratio (DIR), Equalized Odds Difference (EOD) and consistency (Cons.) as fairness measures. For measuring the classification performance, we report the results for the accuracy and the F1-Score.

5.1 Datasets

We have used three datasets that are widely used in the fairness domain. We have chosen the German Credit dataset as a representative of small datasets and UCI Adult dataset as a representative of relatively large datasets [7] while the COMPAS dataset is obtained from ProPublica Data Store[3]. "Charge description" column in the COMPAS dataset and "native country" column in the Adult

[3] Article and dataset are available at [propublica] [accessed 20 Aug. 2024]

dataset were removed to reduce the dimensionality of the datasets after applying the one-hot encoding of the categorical variables. All the datasets contain two binary sensitive attributes and a binary decision label in our experiments. The details regarding each dataset can be found in Table 3.

5.2 Benchmark Methods

Our framework is compared to three different benchmark methods. The first benchmark is a standard logistic regression algorithm with no bias mitigation. The second one is a pre-processing technique, which is the Learning Fair Representations (LFR) from [32]. Finally, the third benchmark is an in-processing technique, which is the Adversarial Debiasing, introduced in [33]. The LFR is trained with logistic regression classifier. The Adversarial Debiasing also works with built-in logistic regression classifier. For SynthFair, when generating the synthetic data using SMOTE, we compare three classification models, namely, Logistic Regression (LR), Random Forests (RF) and Gradient Boosting Trees (GBT). When the data is generated using cGANs, we report only the best results that we get using one of the three classifiers or a neural network classifier[4].

Dataset	**Adult**	**German**	**COMPAS**
Domain	Income	Credit approval	Criminal risk
Attributes	14	20	51
Instances	48842	1000	7918
S. Attributes	Gender, Race	Age, Gender	Gender, Race
Privileged Group	(Male, Caucasian)	(≥ 40, Male)	(Female, Caucasian)
Labels (+ve, -ve)	($\geq 50k$, $< 50k$)	(Approved, Not)	(No Rec., Rec.)

Table 3: Characteristics of the datasets that are used in the experiments. The used abbreviations: (S. Attribute = Sensitive attributes, and Rec. = Recidivate).

5.3 Experimental Setup

We have implemented our framework in Python and imported AIF360 library[5] to execute the benchmark methods. We collected the values for the fairness measures that were explained in Section 3 and two performance measures {Accuracy (Acc.), and the F1 Score (F1)}. However, due to space limitations, we report the results of EOD, DIR and Cons. We have conducted three main experiments in total and used ten-fold cross validation with the randomized train/test split per dataset and reported the averages of the results over the different runs[6]. In the first experiment, the three different strategies that are implemented with SynthFair framework are compared with each other on the German Credit dataset

[4] The implementation of our framework can be found in the [GitHub Repo.]
[5] https://github.com/Trusted-AI/AIF360
[6] More results and detailed discussions are available at [Fairness Thesis]

to find the most optimal strategy (see Table 4). The most unprivileged (a1: 0, a2: 0) and privileged (s1: 1, s2: 1) subgroups are used to calculate the measures. In the second experiment, every possible subgroup combination as privileged and unprivileged groups is compared with each other to check the improvement in fairness measures between these subgroups. In the experiment, only the subgroup having a favorable value for both sensitive attributes (a1:1, a2:1) is always privileged, and the subgroup with values (a1:0, a2:0) is always unprivileged in the comparisons. The other subgroups can be compared as both privileged and unprivileged groups (see Table 5). In the third experiment, the benchmark methods are compared against SynthFair framework on the three datasets. All the fairness measures are calculated by comparing the most unprivileged (a1:0, a2:0) with the most privileged (a1:1, a2:1).

Cl.	**Tech.**	**EOD**	**DIR**	**Cons.**	**Acc.**	**F1**
LR	N M	0.252	0.631	0.835	0.759	0.837
	Stra1	0.075	0.768	0.776	0.701	0.773
	Stra2	0.053	0.833	0.746	0.698	0.775
	Stra3	0.021	0.897	0.749	0.685	0.759
	GANs	0.014	0.843	0.862	**0.793**	0.814
RF	N M	0.112	0.811	0.834	0.756	**0.839**
	Stra1	0.023	0.899	0.814	0.749	0.831
	Stra2	0.020	0.954	0.794	0.744	0.827
	Stra3	0.016	0.911	0.801	0.749	0.832
	GANs	0.068	0.928	**0.930**	0.723	0.832
GBT	N M	0.096	0.793	0.811	0.757	0.835
	Stra1	**0.006**	0.881	0.796	0.738	0.814
	Stra2	0.030	0.956	0.775	0.727	0.808
	Stra3	0.049	**0.972**	0.777	0.731	0.811
	GANs	0.021	0.955	0.880	0.706	0.816

Table 4: Comparing the performance of Logistic Regression (LR), Random Forests (RF) and Gradient Boosting Trees (GBT) on German Credit dataset. The comparison is between the most privileged/Unprivileged subgroups (i.e., a1:0, a2:0 vs. a1:1, a2:1).

5.4 Results and Analysis

Comparing Classification Strategies for the Optimal Framework Construction: The averaged results on all datasets with different classifiers show that the third strategy (SynthFair with Stra3) performs the best among other strategies in achieving a high DIR. It also causes the minimal loss in performance measures among other strategies (see the results with German dataset on Table 4). Even though it looks like Random Forest classifier is not the best combination with SynthFair with Stra3 according to Table 4, it is the most consistent classifier with our framework in terms of providing high fairness scores (EOD, and DIR) while having minimal trade-off in other fairness and performance measures

Subgroups (G_p vs. G_u)	EOD	DIR	DPD	PPD	Cons.
A: 0, S: 0 vs A: 1, S: 0	0.02	0.95	0.04	0.15	0.80
A: 1, S: 0 vs A: 0, S: 1	0.02	1.20	0.07	0.10	0.80
A: 0, S: 1 vs A: 1, S: 1	0.06	0.89	0.10	0.10	0.80
A: 0, S: 0 vs A: 0, S: 1	0.04	1.06	0.02	0.05	0.80
A: 1, S: 0 vs A: 1, S: 1	0.04	0.97	0.03	0.00	0.80
A: 0, S: 0 vs A: 1, S: 1	0.02	0.91	0.07	0.15	0.80

Table 5: Evaluating SynthFair using the pairwise comparison of the subgroups using the German dataset. Sensitive attributes are (A = age and S = sex).

Data	Tech.	EOD	DIR	Cons.	Acc.	F1
German	Original DF	-	0.748	0.682	-	-
	LR	0.252	0.631	0.835	**0.759**	**0.837**
	LFR	0.123	0.764	**0.985**	0.650	0.745
	Adv. Deb.	0.362	0.570	0.983	0.683	0.798
	SynthFair	**0.022**	**0.914**	0.803	0.743	0.827
Adult	Original DF	-	0.235	0.848	-	-
	LR	0.260	0.248	0.937	**0.816**	**0.821**
	LFR	**0.087**	**0.463**	0.975	0.720	0.520
	Adv. Deb.	0.238	0.088	**0.999**	0.794	0.506
	SynthFair	0.126	0.370	0.845	0.794	0.793
COMPAS	Original DF	-	0.688	0.675	-	-
	LR	0.481	0.429	0.967	**0.677**	**0.710**
	LFR	0.270	0.641	**0.999**	0.647	0.666
	Adv. Deb.	0.485	0.420	0.998	0.664	0.696
	SynthFair	**0.151**	**0.712**	0.796	0.627	0.657

Table 6: Comparing SynthFair trained with random forests classifier against logistic regression (LR), Learning Fair Representation (LFR), and adversarial Debiasing (Adv. Deb.). Both LFR and Adv. Deb. use logistic regression.

when all of the experimented datasets are considered. Moreover, it shows more consistent results that can be explained by the small standard deviation values that we observed during the experiments but could not present here due to space limitations. Thus, we recommend using SynthFair with the Stra3 strategy and Random Forest classifier. The cGANs show promising results. We recommend SynthFair with Stra3 over cGANs because of the high computational cost and the inconsistent results on the other datasets. Improving the computational complexity and the quality of the generated data for cGANs is left for future work.

Effect of SynthFair on All Subgroups: The results of using SynthFair with Random Forest classifier on German dataset show that all the EOD values are lower than 0.06, all the DIR are above the threshold of 0.8, and the DP Differ-

ences are also smaller than 0.1, which means that SynthFair improves fairness satisfactorily in this dataset for all possible combinations of privileged and unprivileged subgroups. Having values greater than 1.0 in DIR means that the subgroup considered as the unprivileged group is actually more privileged than the subgroup considered as the privileged group in the equation. For example, in Table 5, the DIR on the second row is 1.119, which means that the subgroup "age:1, sex:0" is more privileged than the subgroup "age:0, sex:1". However, since the value is smaller than 1.2, it is still considered as satisfactorily fair.

Comparing SynthFair with Baseline Methods: The results indicate that SynthFair with the third strategy successfully decreases the EOD and increases the DIR consistently. Depending on the severeness of the bias in the dataset, DIR does not always reach the minimum threshold, which is 0.8. However, our solution outperform the other benchmarks in most of the cases in terms of both EOD and DIR, which can be seen in Table 6. Only in the Adult dataset, LFR outperforms the SynthFair in terms of the DIR by 2%. Furthermore, SynthFair with Random Forest yields the minimum loss when all of the performance measures in the experiments are compared to other mitigation techniques (LFR and Adversarial Debiasing) in most cases. It is found that the other benchmarks perform better at achieving a higher Consistency score, although our framework does not cause a significant decrease in this score, which is not more than a 0.1 decrease in most of the cases. The standard deviation scores reveal that our results in different randomized runs provide consistently similar improvements in results compared to other benchmark methods. It should be noted that Adversarial debiasing algorithm has a significantly low score of 0.88 in DIR, because it could not predict any positive outcomes for the unprivileged subgroup in several runs.

6 Conclusion

We studied the bias problem in ML as data imbalance problem, where there is an unequal representation of the different subgroups in terms of positive and negative outcomes in the datasets. We proposed the SynthFair framework as a pre-processing bias mitigation technique that has a minimum explicit intervention to the machine learning pipeline since it changes neither the original class labels of a dataset, nor any classification algorithm's training structure. Our solution provides consistent improvements in achieving higher fairness measures among different subgroups while maintaining the classifier's performance. SynthFair can be integrated with different clustering, oversampling, and classification algorithms to find a customized solution that works best for any given dataset. The directions to improve the SynthFair framework include but are not limited to studying more synthetic data generation techniques that might generate high-quality examples, proposing a method for determining the quality of the synthetic examples, and improving the computational complexity of the cGANs as well as the quality of their generated examples.

Bibliography

[1] Angwin, J., Larson, J., Mattu, S., Kirchner, L.: Machine bias. ProPublica (2016), https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

[2] Berk, R., Heidari, H., Jabbari, S., Kearns, M., Roth, A.: Fairness in criminal justice risk assessments: The state of the art. Sociological Methods & Research **50**(1), 3–44 (2021)

[3] Calders, T., Kamiran, F., Pechenizkiy, M.: Building classifiers with independency constraints. In: (ICDM Workshops'09)

[4] Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: synthetic minority over-sampling technique. J. of A.I. research **16**, 321–357 (2002)

[5] Chouldechova, A.: Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. Big data **5**(2), 153–163 (2017)

[6] Chouldechova, A., Roth, A.: A snapshot of the frontiers of fairness in machine learning. Communications of the ACM **63**(5) (2020)

[7] Dua, D., Graff, C.: UCI machine learning repository (2017), http://archive.ics.uci.edu/ml

[8] Dwork, C., Hardt, M., Pitassi, T., Reingold, O., Zemel, R.: Fairness through awareness. In: Proceedings of the 3rd innovations in theoretical computer science conference. pp. 214–226 (2012)

[9] EEOC., T.U.: Uniform guidelines on employee selection procedures (03 1979)

[10] Engelmann, J., Lessmann, S.: Conditional wasserstein gan-based oversampling of tabular data for imbalanced learning. Expert Systems with Applications **174**, 114582 (2021)

[11] Feldman, M., Friedler, S.A., Moeller, J., Scheidegger, C., Venkatasubramanian, S.: Certifying and removing disparate impact. In: (KDD'15). pp. 259–268

[12] Hardt, M., Price, E., Srebro, N.: Equality of opportunity in supervised learning. In: NIPS'16

[13] Hort, M., Chen, Z., Zhang, J.M., Harman, M., Sarro, F.: Bias mitigation for machine learning classifiers: A comprehensive survey. ACM J. Responsib. Comput. (nov 2023)

[14] Kamiran, F., Calders, T.: Classification with no discrimination by preferential sampling. In: Proc. 19th M.L. Conf. Belgium and The Netherlands (2010)

[15] Kamiran, F., Calders, T.: Data preprocessing techniques for classification without discrimination. Knowledge and Information Systems **33**(1), 1–33 (2012)

[16] Kamiran, F., Karim, A., Zhang, X.: Decision theory for discrimination-aware classification. In: (ICDM'12)

[17] Kamishima, T., Akaho, S., Sakuma, J.: Fairness-aware learning through regularization approach. In: (ICDM Workshops'11)

[18] Kilbertus, N., Rojas-Carulla, M., Parascandolo, G., Hardt, M., Janzing, D., Schölkopf, B.: Avoiding discrimination through causal reasoning. In: (NIPS'17)

[19] Kleinberg, J., Mullainathan, S., Raghavan, M.: Inherent trade-offs in the fair determination of risk scores. In: 8th Innovations in ITCS'17

[20] Letzter, R.: Amazon just showed us that 'unbiased' algorithms can be inadvertently racist. Insider (04 2016), https://www.businessinsider.com/how-algorithms-can-be-racist-2016-4?international=true&r=US&IR=T

[21] Pleiss, G., Raghavan, M., Wu, F., Kleinberg, J., Weinberger, K.Q.: On fairness and calibration. In: (NIPS'17)

[22] Ristanoski, G., Liu, W., Bailey, J.: Discrimination aware classification for imbalanced datasets. In: (CIKM'13)

[23] Ross, T.J., et al.: Fuzzy logic with engineering applications, vol. 2. Wiley Online Library (2004)

[24] Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature Machine Intelligence **1**(5), 206–215 (2019)

[25] Salimi, B., Rodriguez, L., Howe, B., Suciu, D.: Interventional fairness: Causal database repair for algorithmic fairness. In: (SIGMOD'19). pp. 793–810 (2019)

[26] Soper, S.: Amazon to bring same-day delivery to roxbury after outcry. Bloomberg (04 2016), https://www.bloomberg.com/news/articles/2016-04-26/amazon-to-bring-same-day-delivery-to-roxbury-after-outcry

[27] Verma, S., Rubin, J.: Fairness definitions explained. In: 2018 IEEE/ACM international workshop on software fairness (fairware). pp. 1–7 (2018)

[28] Xu, L., Skoularidou, M., Cuesta-Infante, A., Veeramachaneni, K.: Modeling tabular data using conditional gan. In: (NeurIPS'19)

[29] Yan, S., Kao, H.t., Ferrara, E.: Fair class balancing: Enhancing model fairness without observing sensitive attributes. In: (CIKM'20)

[30] Zafar, M.B., Valera, I., Gomez Rodriguez, M., Gummadi, K.P.: Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In: Proceedings of WWW'17. pp. 1171–1180

[31] Zafar, M.B., Valera, I., Rogriguez, M.G., Gummadi, K.P.: Fairness constraints: Mechanisms for fair classification. In: Artificial Intelligence and Statistics. pp. 962–970 (2017)

[32] Zemel, R., Wu, Y., Swersky, K., Pitassi, T., Dwork, C.: Learning fair representations. In: (ICML'13)

[33] Zhang, B.H., Lemoine, B., Mitchell, M.: Mitigating unwanted biases with adversarial learning. In: Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society. pp. 335–340 (2018)