POLITECNICO DI TORINO

Corso di Laurea Magistrale
in Ingegneria Matematica

Tesi di Laurea Magistrale

A Pre-Processing Framework for Mitigating Representation Bias in Machine Learning Classification Algorithms

Relatori
Dr. Francesco Della Santa
Dr. A.A.A. (Hakim) Qahtan

Candidato
Annalisa Deiana

Anno Accademico 2023-2024

Abstract

The use of Machine Learning (ML) algorithms in decision-making processes has significantly increased in recent years, providing alternatives to human decisions, which are frequently affected by bias. However, ML algorithms can also exhibit bias, leading to discrimination against individuals or groups based on sensitive attributes such as gender or race. This bias often arises from the imbalanced representation of demographic groups in the training data. Mitigating representation bias during the training phase is crucial to ensure fair application of the ML models in decision-making processes.

This thesis presents a pre-processing framework designed to address representation bias by oversampling minority groups, thereby creating a balanced and fair dataset for model training. The proposed framework identifies subgroups with lower imbalance ratios and employs the DBSCAN clustering algorithm to classify points as core, border, or noise. Subsequently, the SMOTE-NC oversampling algorithm generates synthetic samples through interpolation between border points and border/core points until each group attains the highest balance ratio. The performance and fairness of the proposed method are evaluated using standard performance and fairness measures. Experimental results indicate that the framework significantly improves fairness while maintaining a minimal loss in predictive performance compared to other existing methods.

Contents

Chapter 1

Introduction

In recent years, Machine Learning (ML) algorithms have become increasingly important, permeating various aspects of everyday life. For instance, these algorithms provide personal recommendations for movies and songs and influence critical decisions, such as credit loan approvals, candidate selection in hiring processes, and determining the freedom of defendants in the criminal justice system [52]. ML algorithms are particularly valuable in decision-making processes because they can offer objective decisions. Unlike humans, who may have prejudices against specific groups leading to biased decisions, ML algorithms are designed to be free from opinions or prejudices, ideally producing unbiased outcomes. However, this is not always the case. ML algorithms can also produce biased decisions if they are trained on biased datasets—a concept encapsulated by the phrase "bias in, bias out" [51].

One prominent example is the recruiting engine used by Amazon in 2014 to review job applicants [18]. By 2015, it became evident that the algorithm was biased against female applicants, resulting in the exclusion of many women candidates. This bias arose because the algorithm was trained on historical resumes submitted to the company, which were predominantly from male candidates. Consequently, the algorithm learned to favor male-associated attributes, effectively encoding the gender bias present in the training data. Another example is the COMPAS tool used in the United States criminal justice system [58]. This software is employed to decide whether a defendant should be released on bail or kept in custody before trial, based on a risk score indicating the likelihood of reoffending. Analyzing the dataset used to train COMPAS revealed that African-American males were more likely to be classified as high risk, leading the algorithm to predict higher chances of reoffending for this group compared to others.

1.1 Motivation

As machine learning algorithms increasingly influence decision-making, it is crucial to eliminate the factors leading to biased outcomes. While achieving fair predictions remains a primary goal for researchers, an ultimate method has yet to be found due to varying definitions and measurements of fairness. The impossibility theorem [45] highlighted this challenge by proving the incompatibility of certain fairness measures, leading to the difficulty of defining a single method for mitigating ML bias.

Most existing bias mitigation methods focus on datasets with a single sensitive attribute, dividing the population into privileged and unprivileged groups. However, this approach often fails to capture the complexity of real-world scenarios, where individuals are characterized by multiple intersecting sensitive attributes, resulting in diverse groups. Additionally, many methods attempt to mitigate bias by oversampling to achieve an equal number of samples in each group. This strategy can be problematic, as highly imbalanced datasets often lead to the creation of a large number of synthetic samples, resulting in overly artificial datasets and potential overfitting [64].

Motivated by these challenges, this work aims to mitigate representation bias arising from unbalanced datasets, where some groups are underrepresented. The focus is on considering all possible subgroups defined by multiple sensitive attributes and reducing the ratio between the number of positive and negative examples to ensure that each subgroup has an equal likelihood of receiving a positive prediction from the classifier. The goal is to achieve a uniform ratio of positive to negative instances across all subgroups, thereby promoting fairness in ML outcomes.

1.2 Glossary

This section introduces the most important terms and definitions used throughout the thesis, which will frequently appear in the later chapters.

- **Fairness**: The absence of prejudice or favoritism towards an individual or a group based on their intrinsic or acquired traits [52].
- **Bias**: Systematic and unfair discrimination against an individual or a group of individuals in favor of others [28].
- **Protected/Sensitive Attributes**: Attributes of an individual that should be irrelevant in decision-making processes. According to Article 21 of the European Union's Agency for Fundamental Rights [1], these include sex, race, religion, ethnic or social origin, age, among others.
- **Groups and Subgroups**: Groups in a dataset are defined by all possible combinations of sensitive attributes. Subgroups further divide each group into two, based on the label (positive/negative).

- **Imbalance Ratio (IR)**: Computed for each group, this is the number of instances with positive label divided by the number of instances with negative label. It represents the acceptance rate for each group.
- **Privileged Group**: The group with the highest IR among all the defined groups.
- **Unprivileged Group**: The group with the lowest IR among all the defined groups.
- **Skewed Groups**: Groups where the distribution of positive and negative instances is significantly imbalanced.

1.3 Thesis outline

The rest of this work is structured as follows. Chapter 2 reviews related works, exploring the sources of bias, the methods for measuring it, and various techniques proposed in the literature for its mitigation. Chapter 3 provides the theoretical background, presenting the problem formally and explaining the methods used. Chapter 4 introduces the proposed framework to address representation bias. In Chapter 5, we evaluate the proposed framework by comparing its effectiveness against other techniques aimed at reducing bias. Finally, Chapter 6 concludes our work and present the limitations and the future work directions.

Chapter 2

Related work

This chapter presents the related work that have contributed to the theoretical and practical foundations of achieving fairness in machine learning. First, it explores the possible sources of bias in machine learning, distinguishing between bias originating from the data and bias arising from the algorithms. Next, it introduces various metrics for quantifying bias, including group fairness measures, individual fairness measures, and subgroup fairness measures, along with discussions on the impossibility theorem and the concept of discrimination. Finally, the chapter summarizes methods for mitigating bias, categorizing them based on the stage at which they are applied: pre-processing (before training), in-processing (during training), and post-processing (after training).

2.1 Sources of Bias in Machine Learning

To effectively mitigate bias in a dataset, it is crucial to understand its origins. This necessitates an exploration of the different types of bias in machine learning. According to Friedman et al. [28], bias in computer systems is defined as systematic and unfair discrimination against an individual or a group of individuals in favor of others, thus serving as a source of unfairness. In this context, bias in machine learning can broadly be categorized into two main types: bias originating from the data and bias originating from the algorithms. Data bias arises during the data collection process and is often context-dependent, making it difficult to completely eliminate. On the other hand, algorithmic bias results from the decisions made by users when handling and processing this data [6].

2.1.1 Bias from Data

Data bias emerges during the data collection process. It is important to carefully select the dataset used to train a classifier because if the dataset contains bias, this will be reflected in biased predictions. Mayson et al. [51] summarize this phenomenon with the phrase "bias in, bias out" exemplifying that if the outcome to be predicted, such as an arrest, happens more often to black individuals than white individuals in the training

dataset, future decisions are likely to reflect this trend as well.

Historical bias falls into this category. It occurs when the dataset projects a type of unfairness that existed in the past but no longer reflects current reality [62]. The dataset could be well-constructed, with appropriate feature selection and perfect sampling, yet still contain bias from past decisions [52]. An example of historical bias is found in word embeddings used in natural language processing. A recent study [30] showed that an embedding model trained on data from a specific decade reflects the reality of that decade, which might now be considered biased. For instance, words representing job occupations such as "nurse" or "engineer" are more correlated with female and male genders, respectively. Using a dataset affected by historical bias to train a model can lead to unfair predictions because it reflects stereotypes that, although once accurate, are now considered unfair.

Another type of data bias is Representation Bias, which occurs when some subgroups are not well-represented in the dataset, causing the data to fail to represent the entire population [62, 6]. This bias arises from the sampling process during data collection [52]. Representation Bias can occur when the population used for training differs from the population for which the model will be used. For instance, a model trained with individuals from Boston may not perform well when analyzing individuals from Rome [62]. Representation Bias also arises when one or more subgroups are underrepresented, meaning the amount of data representing these minorities is insufficient compared to the majority subgroups. Consequently, the model does not have enough data to learn about the underrepresented subgroups, leading to unfair predictions. An example is the ImageNet dataset [19], which contains 1.2 million labeled images. Around 45% of the images were taken in the United States, while only about 3% were taken in India and China, causing the model to perform poorly on images from these countries [62].

Measurement bias arises from how the features that constitute the dataset are chosen, computed, and measured [52]. This bias can occur if different measurement methods are applied across different subgroups. For example, in a factory setting, if some locations are monitored more frequently than others, the locations with increased monitoring will appear to have more errors, not because they actually do, but due to the higher level of scrutiny [62]. Measurement bias can also arise from using attributes that are proxies for sensitive attributes, leading to unfair decisions towards specific subgroups. Alternatively, proxy attributes may oversimplify a concept, such as using GPA to represent a "successful student," which does not account for the social differences between individuals [62].

2.1.2 Bias from Algorithms

Algorithmic bias occurs when an algorithm introduces bias that was not present in the input data [52]. It results from the decisions made by developers and users when processing and handling data, such as the choice of features, model selection, and the tuning of hyperparameters [46]. Algorithmic bias can amplify existing data bias or introduce new

bias, leading to unfair outcomes even if the data itself is unbiased [57]. Additionally, algorithmic outputs can influence user behavior, creating a feedback loop that perpetuates and even increases bias over time [15].

Aggregation bias occurs when a single model is applied across the entire dataset, failing to account for the diverse characteristics of different subgroups [62]. This approach assumes homogeneity within the population, not taking into account the unique features and patterns that could characterize the subgroups. Consequently, the model's predictions may be accurate for the majority group but significantly biased for minority or underrepresented groups. In some cases, the model may not accurately represent any group within the dataset [62].

While aggregation bias arises from the use of a single model that is expected to fit the entire population, ignoring subgroup differences, Learning bias emerges from the algorithmic design choices that shape the model, such as the selection of a learning function or regularization techniques [7]. For instance, if the objective function optimized by the algorithm during training is not carefully selected, it can unintentionally amplify differences between subgroups. For example, an objective function that prioritizes accuracy as the primary measure may enhance this measure at the expense of other fairness measures, such as disparate impact [62]. This occurs due to the trade-off between fairness and accuracy, where improving one often leads to a reduction in the other [57].

Algorithmic bias can also arise from feedback loops. These occur when decisions made using a trained machine learning model influence the subsequent data collected for future training iterations, potentially reinforcing and amplifying any existing bias [15]. In the work of Baeza-Yates [4], this is referred to as Bias on User Interaction, which is influenced by presentation bias and ranking bias. For instance, in web searches, users are more likely to click on the pages they see on the screen and tend to prefer the top-ranked results [4]. This behavior creates a feedback loop, where top-ranked pages receive more clicks, further increasing their popularity and reinforcing their high rankings.
Another example of feedback loops, introduced by bias in user interaction beyond the web field, is examined in the work of Lum et al. [49]. In this study, arrest data were used to train a predictive policing model aimed at preventing crime before it occurs. These systems predict potential crime hotspots based on historical arrest records. Consequently, areas with higher crime rates receive more police attention. However, because arrests are more likely in areas with a heavy police presence, this creates a feedback loop. The model continually predicts higher crime rates in these over-policed areas, leading to increased police deployment and further arrests, perpetuating a cycle of bias and over-policing [15].

2.2 Fairness Measures

To mitigate bias in datasets and achieve fairness in machine learning, it is crucial to first define what fairness means and how it can be measured. However, there is no universally accepted definition [60]. While people often have an intuitive sense of what is fair, these

intuitions can vary significantly across different situations and individuals.
According to [52], fairness in decision-making can be defined as the lack of prejudice or favoritism towards an individual or a group based on their intrinsic or acquired traits. This notion of fairness implies that decisions should be free from bias related to an individual's sensitive attributes—those characteristics that are irrelevant in specific decision-making contexts. Article 21 of the European Union's Agency for Fundamental Rights [1] explicitly prohibits discrimination based on protected attributes such as sex, race, religion, ethnic or social origin, age, among others.
The challenge of defining fairness is further complicated by the recent formulation of the impossibility theorem [45], which suggests that it is generally impossible to meet all fairness criteria at the same time.
As a result, various fairness measures have been proposed in the computer science literature to address different aspects of bias and discrimination. These measures will be explained in more details in the following subsections.

2.2.1 Group Fairness Measures

The goal of group fairness is to ensure that machine learning algorithms treat different demographic groups equitably, particularly those defined by sensitive attributes such as race, gender, or age. Once these protected groups are identified, group fairness measures aim to achieve parity in certain statistical metrics across these groups [15]. However, this parity does not need to be perfect: imposing strict parity can actually reduce an algorithm's accuracy [9].

In the literature, many group fairness measures have been proposed, each one of them trying to define a different statistical metric to be equalized between groups. In the book of Barocas et al. [5], the authors divide these fairness measures based on three criteria, that can be generalized to score functions using simple (conditional) independence statements. The first criterion introduced in the book is *Independence*, and it requires the prediction of an outcome to be statistically independent of the sensitive attributes. This means that an individual's membership in a specific demographic group should not influence their chances of receiving a favorable outcome [11]. An example of the measures in this category is the demographic parity measure, also known as statistical parity measure. A classification algorithm satisfies this fairness measure if the probability of a positive (or negative) prediction is independent of the sensitive attribute, therefore is the same across groups [21]. However, demographic parity is not sufficient on its own because it only ensures equal outcomes across groups, without considering the underlying reasons or individual circumstances that might influence those outcomes. For instance, in one example reported in [21], statistical parity allows for the selection of unqualified individuals from the unprivileged group merely to meet parity requirements.
To address these shortcomings in demographic parity, the conditional statistical parity measure has been introduced. This fairness measure refines demographic parity by conditioning on a set of relevant non-sensitive attributes, therefore permitting a set of legitimate attributes to affect the outcome [16].
Another fairness measure that fall into the Independence criterion is disparate impact

[23]. This measure is defined as the ratio of the probability of a positive outcome for the unprivileged group over the probability of a positive outcome for the privileged group. According to the 80% rule, a dataset exhibits disparate impact if this ratio is less than 0.8 [27].

The second criterion introduced in the book by Barocas et al. [5] is *Separation*, which focuses on equalizing error rates across demographic groups. Separation is a fairness criterion that requires the prediction of an outcome to be statistically independent of the sensitive attributes, conditional on the true outcome. This means that within each stratum defined by the true target variable, the prediction should not depend on the sensitive attributes. Therefore, Separation ensures that error rates (both false positive and false negative rates) are equal across different demographic groups. In binary classification, the true positive rate is defined as the probability of predicting a positive outcome when the actual outcome is positive, while the false positive rate is defined as the probability of predicting a positive outcome when the actual outcome is negative. The false negative rate is one minus the true positive rate.
Measures fall into this category include the equalized odds measure, which is a fairness measure that requires both the false positive rate and the false negative rate (and consequently the true positive rate) to be equal across different demographic groups. This ensures that individuals with positive and negative target labels receive similar prediction outcomes, regardless of their membership in a privileged or unprivileged group [34].
A possible relaxation of equalized odds is to consider only one of the two error rate equalities, rather than both. Equal Opportunity, also known as Equality of Opportunities, focuses on ensuring equal false negative rates across different demographic groups. This means that the probability of assigning a positive outcome to individuals in the positive class should be the same for both privileged and unprivileged groups[52]. On the other hand, Predictive Equality aims to equalize the false positive rates across demographic groups. This measure ensures that the likelihood of incorrectly predicting a positive outcome (when the true outcome is negative) is the same for all groups, regardless of their sensitive attributes [16].

The third and last criterion introduced in the book by Barocas et al. [5] is *Sufficiency*, which requires that individuals receiving the same decision from the model have equal probabilities of the actual outcome, regardless of their sensitive attributes. This criterion ensures parity in the likelihood of the true outcome for people with the same prediction. Predictive parity, also known as test fairness, is a fairness measure that aligns with the Sufficiency criterion by ensuring equal positive predictive values (PPV) across different demographic groups. This means that the ratio of correctly predicted positive outcomes (true positives) to all predicted positive outcomes is equal for both privileged and unprivileged groups [14].

This subsection has covered several key group fairness measures. These measures aim to ensure fair outcomes across different demographic groups and are crucial in mitigating bias in machine learning models. While other fairness measures exist, the ones discussed here are the most relevant for this thesis and are widely used in practice.

It is important to note that, as will be discussed in subsection 2.2.4, an impossibility theorem has been formulated, which states that not all three fairness criteria — Independence, Separation, and Sufficiency — can be satisfied simultaneously. This highlights the inherent trade-offs and complexities involved in achieving fairness in machine learning.
The next subsection will explore individual fairness measures, focusing on ensuring fair treatment at the individual level rather than across groups.

2.2.2 Individual Fairness Measures

While group fairness measures focus on achieving statistical parity between demographic groups, the main idea behind individual fairness measures is to ensure that similar individuals are treated similarly. This concept was first introduced by Dwork et al. [21], who formalized this principle using a Lipschitz condition on the classifier. Specifically, the authors defined a classifier as a mapping from individuals to distributions over outcomes. They then framed this as an optimization problem, introducing a utility loss function that the classifier aims to minimize to be considered fair.
Dwork et al. defined a distance metric among individuals and used the Lipschitz condition as a constraint in the optimization problem. This constraint ensures that the distance between two individuals in the decision space does not exceed their distance in the input space. One shortcoming of this approach is the challenge of defining a distance metric that accurately reflects the concept of similarity between individuals in the input space.
Zemel et al. [66] proposed the concept of *Consistency* to compute the similarity between individuals. Specifically, Consistency is computed by comparing the predicted label of each individual with those of its nearest neighbors, based on the idea that close individuals should receive similar predictions.

In the work of Kusner et al. [47], the concept of individual fairness is defined as counterfactual fairness. According to this definition, the outcome of a classifier should remain unchanged when comparing two individuals with identical features but different sensitive attributes. This notion is sometimes referred to as Fairness Through Unawareness or blindness in the literature. Under this framework, a classifier is considered fair if it does not explicitly use protected attributes in the decision-making process [63].
However, a significant drawback of this approach is that it fails to account for the presence of proxy attributes, which are variables correlated with the sensitive attributes. Consequently, even if protected attributes are not explicitly included in the training or decision process, other attributes that are closely related to them may still introduce bias. For example, a zip code can act as a proxy for race, as certain areas may be predominantly inhabited by specific racial groups. This means that decisions based on zip code can inadvertently lead to racial discrimination.

At this point, one might consider developing a method that excludes both sensitive attributes and their proxy attributes from the decision-making process. This concept has been explored by Kamiran et al. [40] and is known as *suppression*. In this approach, the authors aim to identify attributes that are highly correlated with protected attributes and exclude both these and the protected attributes from the training of the classifier.

However, this raises questions about defining "highly correlated" attributes, determining appropriate thresholds for exclusion, and recognizing the potential loss of valuable information when many features are removed from the input space.
Another approach, introduced by Zemel et al. [66], aims to address the problem of information loss that occurs when certain features are deleted. The authors propose finding an intermediate representation of the data that retains as much information from the original features as possible while simultaneously obfuscating any information related to the protected attributes. This "cleaned" representation of the data is then used to train a classifier, ensuring that the model remains fair by not being influenced by sensitive attributes.

Joseph et al. [38] proposed an individual fairness measure using the contextual multi-armed bandit framework. Their method ensures that less qualified individuals are not favored over more qualified ones, regardless of sensitive attributes. Each demographic group, defined by sensitive attributes like race, is represented as an arm in the bandit problem. Pulling a lever, in this context, means choosing an individual from a given group. The key idea is that the algorithm should maximize the cumulative reward obtained after each lever pull, thus ensuring fair treatment among individuals based on their qualifications rather than protected attributes. One drawback of this method is that it does not account for the societal structures that may cause certain individuals to be less qualified due to insufficient resources.

In this subsection, the most important and well-known notions of individual fairness measures have been presented along with their drawbacks. Despite their promise, a significant limitation is determining what it means for two individuals to be similar. This involves defining a suitable distance metric, which is inherently subjective and context-dependent. Consequently, while valuable, individual fairness measures must be applied cautiously, considering their inherent challenges.

2.2.3 Subgroup Fairness Measures

The third category of fairness measures is known as Subgroup Fairness Measures, which can be seen as an intermediate approach between group fairness and individual fairness. As explained in the previous subsections, group fairness typically considers a fixed number of demographic groups, often based on a single sensitive attribute, while individual fairness focuses on ensuring fairness at the individual level. Subgroup fairness measures aim to ensure statistical parity across numerous subgroups, defined by a structured class of functions over the protected attributes [44]. This approach seeks to balance the broader reach of group fairness with the granularity of individual fairness, providing a more nuanced method for addressing fairness across diverse and overlapping subgroups.

The concept of subgroup fairness was first introduced by Kearns et al. [44], who highlighted the issue of fairness gerrymandering. This occurs when statistical measures are balanced across high-level, pre-defined groups, but the classifier remains unfair when considering subgroups formed by intersections of sensitive attributes. For instance, they

provide an example of a classifier that gives positive predictions only to individuals identified as Black Men or White Women. While the classifier appears fair when considering only the gender attribute, with men and women receiving positive predictions 50% of the time, it becomes clear that it is unfair when examining the intersection of gender and race. To achieve subgroup fairness, Kearns et al. propose selecting a statistical constraint, such as the false negative rate, and ensuring that this constraint is equalized across the numerous subgroups within the dataset. One drawback of the gerrymandering approach of fairness is its scalability. With numerous protected attributes, the number of potential subgroups grows combinatorially, posing significant computational challenges and making it difficult to enforce fairness constraints efficiently across all intersections.

Another work similar to [44] is presented by Hébert-Johnson et al. [69], where the authors introduce the concept of *multicalibration*. This measure of algorithmic fairness ensures accurate predictions for all subpopulations, which are defined by a specified class of computations. Multicalibration effectively addresses bias that emerge during the learning process, ensuring that predictions remain fair across various overlapping subgroups.

2.2.4 Impossibility Theorem

The impossibility theorem in fairness states that it is impossible to simultaneously satisfy the three group fairness criteria - Independence, Separation, Sufficiency - except in highly constrained special cases [45]. This theorem emphasizes the inherent trade-offs involved in algorithmic fairness. Each of these criteria has distinct definitions and implications, leading to conflicts when attempting to achieve all three at once.

Kleinberg et al. [45] presented the Impossibility Theorem as a resolution to the debate on defining fairness in algorithmic classification. They argue that fairness definitions are context-dependent, with no single correct definition, and that all are valid yet incompatible. Furthermore, the authors introduce two special cases in which the three fairness criteria can be simultaneously satisfied.
The first case is *perfect prediction*, where individuals' labels are known with certainty. In this scenario, the prediction is independent of the sensitive attribute (Independence) because it perfectly matches the true class label, which is not influenced by the sensitive attribute. Both the true positive rate (TPR) and false positive rate (FPR) are consistently 1 and 0, respectively, for all groups, ensuring equalized odds (Separation). Since the predictions are perfectly accurate, the positive predictive value (PPV) is 1 for all groups, meeting the sufficiency criterion. Thus, perfect prediction achieves all three fairness measures without conflict.
The second case is *equal base rates*. When the two groups have the same proportion of members in the positive class (i.e., the base rate is the same for both groups), the predictions align with these base rates for each group. This ensures that predictions are independent of the sensitive attribute, satisfying Independence. Additionally, the true positive rate (TPR) and false positive rate (FPR) are equal across groups, meeting the criteria for equalized odds (Separation). Finally, since the predictions reflect the actual

base rates, the positive predictive value (PPV) remains consistent across groups, satisfying Sufficiency. Therefore, equal base rates allow all three fairness criteria to be met simultaneously.
Kleinberg et al. [45] also proved that these are the only two cases in which the criteria can be satisfied concurrently. In all other scenarios, a trade-off between the three criteria is necessary, as shown by Barocas et al. [5]. In their book, the authors provide a mathematical proof that the criteria are mutually exclusive.

The Impossibility Theorem was also introduced in the work of Chouldechova et al. [14]. The authors proved the incompatibility of the three criteria in two steps. First, they examined how predictive parity (PPV) conflicts with error rates (such as false positive rate (FPR) and false negative rate (FNR)) when base rates differ between groups. They illustrated this through a specific equation connecting PPV, base rate, FPR, and FNR, showing that different base rates prevent equal FPR and FNR across groups, even if predictive parity is satisfied. Second, Chouldechova et al. [14] demonstrated the incompatibility between disparate impact and error rates (FPR, FNR). They showed that differences in false positive and false negative rates can lead to disparate impact, particularly when high-risk assessments result in more severe consequences. By examining the expected differences in penalties across groups under a simple risk-based policy, they highlighted how such policies can cause unequal treatment, thus proving that fairness in error rates and disparate impact cannot be achieved simultaneously.

2.2.5 Discrimination

Having established the basic definitions of fairness in machine learning, it is important to explore the concepts of explainable and unexplainable discrimination. According to Mehrabi et al. [52], discrimination arises from human prejudice and stereotypes associated with sensitive attributes. However, there are situations where considering these protected attributes in decision-making is necessary. For instance, in the medical field, treating men and women differently based on their gender is essential due to biological differences; treating them identically could be harmful.

Kamiran et al. [41] introduced the concept of explainable discrimination, which acknowledges that differentiating between subgroups is sometimes justified through relevant attributes and therefore considered legal. Discrimination is deemed justifiable when the differences between subgroups can be logically explained. The authors illustrate this with the Adult dataset, where women, on average, have lower annual incomes than men because they tend to work fewer hours per week. If this explainable difference is not accounted for, a classifier trained on the dataset might incorrectly predict lower salaries for men to balance the perceived disparity. This could ultimately harm men and result in reverse discrimination.

On the other hand, unexplainable discrimination occurs when the different treatment between subgroups cannot be justified by relevant attributes and is therefore considered

illegal. This type of discrimination is often based on bias, prejudice, or systemic inequalities. Zhang et al. [68] divide unexplainable discrimination into two categories based on whether the protected attributes are considered explicitly or not.
As the authors report, direct discrimination, also known as disparate treatment [59], occurs when an individual is treated less favorably explicitly because of their protected attributes. This type of discrimination is overt and intentional. For example, rejecting a female candidate for a job in favor of a less qualified male candidate simply because of her gender is direct discrimination. Indirect discrimination, also known as disparate impact [59], occurs when sensitive attributes are not explicitly used in decision-making, but the outcome still results in unfair treatment of the unprivileged groups [68]. This type of discrimination is often unintentional and harder to detect, typically arising from the use of proxies for sensitive attributes. For instance, using an individual's zip code to make decisions such as granting a loan could lead to indirect discrimination if the zip code correlates with race.

2.3 Bias Mitigation Algorithms

Having explored the different sources of bias in machine learning and their measurement, it is essential to discuss how these bias can be mitigated to achieve fairer outcomes. Bias mitigation algorithms are techniques designed to reduce or eliminate bias in machine learning models. These techniques can be broadly categorized into three types: pre-processing, in-processing, and post-processing methods.

2.3.1 Pre-processing

Pre-processing methods aim to modify the data before training a model to minimize bias and ensure that the feature space is not influenced by sensitive attributes [5]. One of the benefits of these techniques is that they are implemented early in the development process, making them independent of the used ML model, as they are applied before the model is trained [61].
Several techniques can be used to modify the data prior to training. One such technique is *fairness through unawareness*, which involves excluding protected attributes from the prediction process [29]. However, this method is insufficient to ensure fairness, as it does not account for the potential presence of proxy attributes that could still reflect characteristics of the sensitive attributes [57]. Consequently, while fairness through unawareness may remove direct bias, it may not fully eliminate indirect bias that can be inferred from other features.

Another approach, known as *Relabelling*, aims to modify the ground truth labels in the training set to ensure the dataset satisfies specific fairness criteria [20]. Within this approach, one pre-processing technique is *Massaging* the dataset, which involves changing some labels in the data to remove dependencies between the class label and the sensitive attributes [10].

Calders and Kamiran [10] explain the Massaging method in their paper. To achieve independence between the class label and the sensitive attributes, certain data point labels are modified: in the privileged group, some positive class labels are changed to negative (demotion candidates), and in the unprivileged group, some negative class labels are changed to positive (promotion candidates). The modified instances are not chosen randomly but are selected using a ranker algorithm. This algorithm ranks instances based on their probability of belonging to the positive (desired) class; the higher an instance is ranked, the more likely it is to be classified as positive. Using this ranker, promotion candidates are sorted in descending order, while demotion candidates are sorted in ascending order. The top-ranked instances from each list are then chosen for label changes, ensuring the modifications promote fairness in the dataset. As the authors themselves noted, the Massaging method is quite intrusive since it changes the ground truth labels of the dataset [10].

Another category of pre-processing methods to mitigate bias is known as *Resampling*. Resampling methods adjust the composition of the training data by either removing or duplicating specific samples. This process involves increasing the representation of underrepresented samples or decreasing the representation of overrepresented ones, thereby balancing the dataset to mitigate bias [20].
One method within this category is *Preferential Sampling* [39], introduced by Calders and Kamiran, to address the drawbacks of the Massaging technique. The main idea behind Preferential Sampling is to select data objects that are the best possible choices for eliminating discrimination in the dataset, specifically focusing on those near the decision boundary. For the unprivileged group, boundary samples with positive label are duplicated, while those with negative label are removed. Conversely, for the privileged group, boundary samples with positive label are removed, and those with negative label are duplicated. A ranking algorithm, similar to the one used in the Massaging method, is employed to identify these borderline objects, ensuring that the dataset is adjusted to promote fairness effectively.
A less sophisticated resampling method is *Uniform Sampling* [40], which operates similarly to Preferential Sampling. However, instead of focusing on borderline samples, Uniform Sampling randomly selects samples from the dataset to be removed or duplicated.

Undersampling and *Oversampling* are two specific subcategories of Resampling. Undersampling involves removing certain samples from the majority class to reduce its prevalence, while Oversampling involves generating new synthetic samples for the minority class to increase its representation. Both methods aim to balance the dataset and mitigate bias [53]. One basic form of oversampling involves randomly duplicating instances from the minority class. However, this simple duplication can lead to overfitting because the model might learn specific details from the repeated samples rather than general patterns. Additionally, more replication does not effectively shift the decision boundary to address bias [13]. To overcome these drawbacks in our research, SMOTE (Synthetic Minority Over-sampling Technique), introduced by Chawla et al. [13], is used. SMOTE creates synthetic samples by interpolating between existing minority instances, producing more diverse and generalized data points rather than merely replicating the existing ones.
Other versions of SMOTE have been proposed, such as Borderline-SMOTE [33]. In this

method, the authors oversample the minority class using the same interpolation technique as SMOTE, but they specifically focus on borderline samples—those with more than half of their nearest neighbors belonging to the majority class. Another advanced version of SMOTE is ADASYN (Adaptive Synthetic Sampling Approach for Imbalanced Learning) [35]. ADASYN generates synthetic samples through interpolation but adapts the process based on the density of minority samples. This technique focuses on instances that are harder to learn, i.e., those with fewer similar neighbors, effectively oversampling more where the density of minority samples is lower.

To mitigate bias, the authors of the Massaging technique introduced another pre-processing method known as *Reweighting* [10]. This approach aims to be less intrusive than Massaging and Sampling, as it does not involve changing labels, or removing or duplicating instances from the dataset. Instead, it assigns a weight to each instance. Specifically, higher weights are assigned to samples from the unprivileged group with positive labels and from the privileged group with negative labels.

2.3.2 In-processing

In-processing methods involve modifying the learning algorithm itself to enforce fairness during the training process [52]. These techniques depend on the used classification algorithm and include approaches such as using ensembles, developing novel or adjusted algorithms, or adding a regularization term to the loss function to mitigate bias [20]. For a comprehensive understanding, the categorization of in-processing methods introduced by Hort et al. [37] will be used throughout this subsection, as it effectively encompasses a wide range of in-processing algorithms.

The first two categories introduced by Hort et al. are called *Regularization* and *Constraints*, both of which involve modifying the loss function of the classification algorithm. The *Regularization* category adds a term to the loss function to penalize certain types of discrimination, while the *Constraints* category imposes constrains on the loss function to ensure that specific fairness measures are met [37].
For instance, Kamiran et al. [42] proposed a regularization approach for Decision Tree models. Traditional Decision Tree loss functions focus solely on accuracy, optimizing splits to improve overall model accuracy. Kamiran et al. introduced a regularization term to the loss function that accounts for discrimination. This term ensures that leaf splits are allowed only when they minimize discrimination, effectively transforming the decision tree into a discrimination-aware classifier.
An example of a constraints-based in-processing method is the work of Zafar et al. [65], where the authors address the challenge of maximizing model accuracy while adhering to boundary constraints that act as proxies for the disparate impact fairness measure. They apply this approach to both logistic regression and support vector machine models. The key proxy measure introduced is called decision boundary covariance, which quantifies the relationship between sensitive attributes and the classifier's decision boundary. By incorporating this measure into the training process, they adjust the model to reduce unfairness while maintaining as much accuracy as possible.

The third category of in-processing methods is known as *Adversarial Learning*, which involves the use of two competing classification algorithms. Specifically, the primary classification model is trained to predict the ground truth labels, while the adversarial model is trained to predict the sensitive attribute based on the classifier's predictions. The competition between the two models aims to ensure that the adversarial model cannot accurately determine the sensitive attribute, thereby promoting fairness by reducing the dependency of the classifier's predictions on sensitive attributes [20].
An example of the Adversarial Learning method is demonstrated in the work by Zhang et al. [67], where a prediction model is trained to predict the label while simultaneously preventing an adversarial model from predicting a protected attribute. This approach considers three fairness metrics: Demographic Parity, Equality of Odds, and Equality of Opportunity. Notably, this method is model agnostic and can be applied to any gradient-based learning model, making it versatile for various regression and classification tasks.

The fourth category of in-processing methods is known as *Compositional* approaches, which address bias by training multiple classification models. These methods either use one model for each population group (e.g., privileged and unprivileged) or employ an ensemble technique where the final prediction is made by aggregating the votes from the different classifiers [37].
In their paper, Dwork et al. [22] propose the use of decoupled classifiers, training a different classifier for each population group defined by sensitive attributes. The goal is to minimize a joint loss function that considers all classifiers collectively. For instance, using different classifiers for majority and minority group, the approach seeks to optimize these classifiers together to minimize the overall loss. This method ensures that the combined model provides fair and accurate predictions across all groups by effectively finding the global optimum of the joint loss function, provided the loss function is weakly monotone.

The last category of in-processing methods is *Adjusted Learning*, which achieves fairness by modifying the learning procedures of standard machine learning algorithms [20].
An example of this approach is demonstrated in the work by Noriega-Campero et al. [55], where the authors propose an active learning framework to train decision trees. Initially, the model is trained using only a subset of features, and then the algorithm iteratively collects additional information about individuals within a predefined information budget. This process ensures that more features are gathered for groups or individuals that are harder to classify. By incorporating an adaptive, iterative feature acquisition strategy, this framework modifies the traditional learning process of decision trees, dynamically adjusting the features collected to improve both accuracy and fairness in predictions.

2.3.3 Post-processing

Post-processing methods adjust the predictions or decision rules after the model has been trained to correct any bias that may exist [37]. These methods are particularly useful when the learned model is treated as a black box, making it impossible to modify the training data or the learning algorithm [52]. By altering the model's outputs, post-processing

techniques aim to achieve fairness without interfering with the underlying model structure or training process. These methods can be categorized into *Input correction*, *Classifier correction* and *Output correction* [37].

The Input correction approaches modify the testing data by adding a pre-processing layer to an already trained algorithm [20]. An example of this method is demonstrated in the work by Adler et al. [2], where the authors audit black-box models to study the indirect influence of sensitive attributes on the output. They achieve this by training the black-box model and then applying a pre-processing step to the test set, which involves obscuring the influence of certain features by finding the minimal perturbation necessary to achieve fairer classifications.

Classifier correction approaches involve taking a trained classifier and deriving a related, fairer classifier from it [20]. An example of this method is introduced in the work by Hardt et al. [34], where the authors aim to develop a classifier that is fair with respect to Equalized Odds and Equal Opportunity, starting from a potentially unfair trained classifier. They accomplish this by defining an optimization problem in which a loss function, dependent on both the original and the desired fair classifier, is minimized subject to constraints representing the fairness measures. This approach allows for the derivation of a new, fairer classifier without the need for retraining.

The last category of post-processing methods is Output Correction, which focuses on correcting the predicted labels [37]. Kamiran et al. [43] propose a framework that adjusts the prediction labels near the decision boundary, where the classifier is most likely to make biased decisions. They define the critical region as the area where labels are assigned with a probability close to 0.5. To reduce discrimination, instances within this critical region are relabeled: positive for those belonging to the unprivileged group and negative for those in the privileged group, while keeping the labels outside the critical region unchanged.

2.4 Addressing Limitations in Bias Mitigation Techniques

Existing bias mitigation methods often use excessive oversampling to match group sizes, leading to artificial datasets and overfitting. The proposed framework addresses these issues by equalizing the Imbalance Ratio (IR) of unprivileged groups to that of the most privileged group, thus reducing the need for excessive oversampling. It employs SMOTE-NC to generate diverse synthetic instances, mitigating overfitting risks.

An innovative aspect of this framework is the use of DBSCAN (Density-Based Spatial Clustering of Applications with Noise) to select critical points for oversampling. This approach targets crucial areas for sample generation and automates parameter selection, enhancing both the efficiency and effectiveness of the process.

Chapter 3

Theoretical Background

This chapter formally formulates the problem by defining the dataset's composition and the measures used to evaluate the fairness and performance of the model. It begins with a theoretical and graphical introduction to the thesis's aim of mitigating representation bias through oversampling. Following this, the theoretical components of the framework are introduced to explain their functionalities. These components include the DBSCAN algorithm, which is used to select points for oversampling, performed using the SMOTE-NC algorithm. Gower's distance is employed as a precomputed metric for DBSCAN to handle both numerical and categorical attributes. Finally, the Logistic Regression classifier is introduced as the chosen classification algorithm due to its inherent capability to represent a decision boundary. In this work, we aim at shifting the decision boundary to reduce bias.

3.1 Problem Statement

The purpose of this thesis is to develop a pre-processing framework to address representation bias, which arises when groups within a population, defined by sensitive attributes (such as race and gender), are unbalanced. This work assumes that the original labels in the dataset are accurate, thus there is no need to alter them.
Representation bias often results from the sampling process during data collection [52], leading to datasets where some groups are skewed or have have less number of instances compared to others. These skewed groups are more likely to receive unfair predictions from the model. For example, in a hiring process, if the dataset that is used to train the classifier is skewed such that female applicants receive, in proportion to their number, less favorable outcomes compared to male applicants, the classifier will likely learn this pattern. Consequently, male applicants would have a higher likelihood of being hired than female applicants.
The aim of this work is to mitigate the effect that representation bias in the training dataset has on the predictions of the classification model. This will be achieved by defining a pre-processing algorithm to oversample the skewed groups, thereby creating a more balanced and fair dataset for training the model.

Next, the problem will be formalized. The dataset used for training is defined as $D = \{X, S, Y\}$, where X represents the set of non-sensitive attributes, S represents the set of sensitive attributes, and Y represents the actual labels of the samples. For simplicity, one binary sensitive attribute will be considered in the theoretical part (e.g. gender), but the approach can be easily extended to multiple sensitive attributes. Furthermore, this analysis will focus on binary classification problems, where $Y \in \{0, 1\}$, with label 0 indicating an unfavorable outcome and label 1 indicating a favorable outcome.
To train the classification model, the dataset D is divided into a training set D_{train} and a testing set D_{test}. The training set D_{train} is used to train the classifier, while the testing set D_{test} is used to evaluate its performance. In particular, the trained model will take D_{test} as input, and produce the predicted labels $\hat{Y} \in \{0, 1\}$.

Before training the classifier, a pre-processing step is performed, which consists of identifying the privileged group and oversampling the other groups accordingly.
The binary sensitive attribute $S \in \{0, 1\}$ splits the dataset D into two groups: the privileged one ($S = 1$) denoted as $priv$, and the unprivileged one ($S = 0$) denoted as $unpriv$. To determine which group is privileged and which is unprivileged, the Imbalance Ratio (IR) is computed. For each group $G_i, i \in \{0,1\}$ with cardinality $|G_i|$, the Imbalance Ratio is defined as the number of samples with positive label $|G_i^+|$, divided by the number of samples with negative label $|G_i^-|$:

$$IR_i = \frac{|G_i^+|}{|G_i^-|}.$$

The privileged group is defined as the one with the highest Imbalance Ratio $G_{priv} = G_k \; s.t. \; IR_k > IR_j, j \in \{0, 1\}$ since we consider a single binary sensitive attribute. If we have n subgroups, then $G_{priv} = G_k \; s.t. \; IR_k > IR_j, j \in \{0, 1, \ldots, n\}$. The IR is then used to set the oversampling target for all other groups. Specifically, each group will be oversampled to match the IR of the privileged group. This ensures a balanced data distribution across different categories, addressing the issue of skewed representation. For instance, in the hiring example, the training dataset will be resampled so that female and male candidates have the same IR, which effectively makes it appear that both groups have the same likelihood of being hired.

To evaluate the performance of the trained classifier in terms of both fairness and accuracy, various performance measures $A = \{A_{m_1}, A_{m_2}, A_{m_3}, ..., A_{m_n}\}$ and fairness measures $F = \{F_{m_1}, F_{m_2}, F_{m_3}, ..., F_{m_n}\}$ are used. performance measures assess the accuracy of the classifier's predictions; in this work, *Accuracy*, *Balanced Accuracy*, and *F1-Score* will be utilized. Fairness measures, on the other hand, evaluate the fairness of the classifier based on specific fairness definitions. In this work, *Disparate Impact*, *Equalized Odds*, *Equal Opportunity*, and *Consistency* are employed.

As a summary, this thesis will address the following issues:

- Quantifying how representation bias in datasets leads to unfair decisions.

- Mitigating representation bias using synthetic data generated through a proposed pre-processing method.
- Evaluate the proposed framework against existing bias mitigation techniques.
- Assessing the proposed framework's in terms of fairness and performance.

3.2 Performance and Fairness Measures

This section presents the performance and fairness measures previously mentioned, which are used to assess classifier's outcomes. To provide a foundation for understanding these metrics, it is useful to first introduce the concepts of true positives, true negatives, false positives, and false negatives.
In binary classification, each instance in the dataset has a ground truth label $Y \in \{0, 1\}$ that the classifier aims to predict as $\hat{Y} \in \{0, 1\}$. In this setting, four types of outcome are possible [25]. When the classifier correctly predicts a positive instance as positive ($Y = 1$ and $\hat{Y} = 1$), this is counted as a *true positive* (TP). Conversely, when the classifier correctly predicts a negative instance as negative ($Y = 0$ and $\hat{Y} = 0$), this is counted as a *true negative* (TN). When the classifier's prediction is incorrect, two scenarios arise: if the true label is positive but predicted as negative ($Y = 1$ and $\hat{Y} = 0$), it is counted as a *false negative* (FN); if the true label is negative but predicted as positive ($Y = 0$ and $\hat{Y} = 1$), it is counted as a *false positive* (FP). These definitions will be used to define the measures.

3.2.1 Performance Measures

Performance measures are used to evaluate how accuracy are the classifier's predictions. This work utilizes three such measures: Accuracy, Balanced Accuracy, and F1-Score.

- **Accuracy**: this performance measure is defined as the ratio of correctly classified instances (the sum of true positives and true negatives) to the total number of instances in the dataset:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

  The desired score for accuracy is one, indicating perfect accuracy where all samples are classified correctly. Accuracy provides an overall measure of how well the model is predicting across the entire dataset, without distinguishing between instances from privileged or unprivileged groups [32]. However, accuracy can be a misleading evaluation measure, particularly in the case of highly imbalanced datasets. For example, consider a fraud detection dataset consisting of 10,000 transactions, with only 200 being fraudulent. If a classifier trained on this dataset labels every transaction as non-fraudulent, it will achieve an Accuracy of 98% (9800/10000 = 0.98), which is very close to the desired score. Despite this high accuracy, the classifier fails to identify any fraudulent transactions, demonstrating poor performance in detecting the minority class.

- **Balanced Accuracy**: This evaluation metric is particularly useful for imbalanced datasets, addressing the limitations of Accuracy. The ideal Balanced Accuracy score is one, indicating perfect performance. The metric is defined as the arithmetic mean of *Recall* and *Specificity*, which consider both the correctly classified positive instances and the correctly classified negative instances. Recall is calculated as the number of correctly predicted positive instances out of all positive instances: $Recall = \frac{TP}{TP+FN}$, and Specificity is calculated as the number of correctly predicted negative instances out of all negative instances: $Specificity = \frac{TN}{TN+FP}$. Therefore, Balanced accuracy is defined as:

$$Balanced\ Accuracy = \frac{Recall + Specificity}{2}$$

  Continuing with the example of fraud detection, where a classifier labels all 10,000 transactions as non-fraudulent, the classifier results in $TP = 0$, $FP = 0$, $FN = 200$, and $TN = 9{,}800$. Despite a high overall accuracy, this classifier achieves a Balanced Accuracy of 50%, highlighting its poor performance in detecting fraudulent transactions.

- **F1-Score**: The F1-Score is a performance measure defined as the harmonic mean of *Precision* and *Recall*. Recall, as defined previously, measures the proportion of correctly predicted positive instances out of all actual positive instances. Precision is defined as the number of correctly predicted positive instances out of all predicted positive instances: $Precision = \frac{TP}{TP+FP}$.
  The F1 Score uses the harmonic mean to balance the trade-off between Precision and Recall [32], ensuring that both metrics are given equal weight. The desired value is 1, and it is calculated as follows:

$$F1\text{-}Score = \frac{2}{Precision^{-1} + Recall^{-1}} = \frac{2TP}{2TP + FP + FN}$$

3.2.2 Fairness measures

Fairness measures are used to evaluate the fairness of a classifier's predictions. This subsection introduces three group fairness measures—Disparate Impact, Equalized Odds, and Equal Opportunity—along with an individual fairness measure, Consistency.

- **Disparate Impact**: This fairness measure is defined as the ratio of the positive prediction rate for the unprivileged group to the positive prediction rate for the privileged group:

$$Disparate\ Impact = \frac{P[\hat{Y} = 1|S = 0]}{P[\hat{Y} = 1|S = 1]}$$

  The desired score for Disparate Impact (DI) is one, indicating equal acceptance rates across groups. However, the "80 percent rule" relaxes this requirement, stating that predictions can be considered free of disparate impact if the ratio falls between 0.80 and 1.25 (= 1/0.80) [27]. A DI less than 1 indicates that the privileged group is more likely to receive positive predictions compared to the unprivileged group, whereas a DI greater than 1 indicates the opposite.

- **Equalized Odds**: Equalized Odds, introduced by Hardt et al. [34], is a fairness measure ensuring that groups within a population have the same false positive rates (FPR) and false negative rates (FNR). This metric guarantees that individuals receive similar prediction outcomes regardless of their membership in a privileged or unprivileged group. The formal definition of Equalized Odds is as follows:

$$P[\hat{Y} = 1|S = 1, Y = 0] = P[\hat{Y} = 1|S = 0, Y = 0]$$

$$P[\hat{Y} = 1|S = 1, Y = 1] = P[\hat{Y} = 1|S = 0, Y = 1]$$

  The first equation ensures equality of the FPR across groups, while the second equation ensures equality of the true positive rate (TPR) across groups. Consequently, since $FNR = 1 - TPR$, it also ensures equality of the FNR.
  In this work, the Average Odds Difference (AOD) is used to compute the Equalized Odds value. AOD is defined as the arithmetic mean of the differences in TPR and FPR between the privileged and unprivileged groups. The desired value for AOD is zero, and it is calculated as follows:

$$AOD = \frac{(FPR_{unpriv} - FPR_{priv}) + (TPR_{unpriv} - TPR_{priv})}{2}$$

- **Equal Opportunity**: This fairness measure, also introduced by Hardt et al. [34], is a relaxation of Equalized Odds. While Equalized Odds requires both the false positive rate (FPR) and the true positive rate (TPR) to be equal across groups, Equal Opportunity focuses solely on the equality of TPR for both privileged and unprivileged groups. The formal definition is:

$$P[\hat{Y} = 1|S = 1, Y = 1] = P[\hat{Y} = 1|S = 0, Y = 1]$$

  In this work, the Equal Opportunity Difference (EOD) is used to compute this fairness measure. The goal is to achieve an EOD value of zero. The EOD is defined as:

$$EOD = TPR_{unpriv} - TPR_{priv}$$

- **Consistency**: This individual fairness measure, introduced by Zemel et al. [66], emphasizes that two individuals similar with respect to a specific task should be treated similarly. Consistency is calculated by comparing the predicted labels for an individual with those of its nearest neighbors. The formula for Consistency is:

$$Consistency = 1 - \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - \frac{1}{n\_neighbors}\sum_{j \in N_{n\_neighbors}(x_i)}\hat{y}_j|$$

  In this formula, n represents the total number of instances in the dataset, $n\_neighbors$ is the number of nearest neighbors considered (default is 5), and $\hat{y}_i$ is the model's prediction for instance x_i. The desired value for consistency is one, indicating that similar individuals are treated similarly.

3.3 Improving Fairness Theoretically

This section aims to discuss how fairness can be theoretically improved by considering the Equal Opportunity Difference (EOD). Specifically, it seeks to provide a theoretical proof that oversampling the unprivileged group to achieve the same imbalance ratio across population groups can mitigate representation bias. This mitigation leads to a shift in the decision boundary, resulting in an increase in positive predicted values for the unprivileged group and a decrease for the privileged group, thereby reducing the EOD (i.e., making the True Positive Rates similar across groups). Additionally, this section aims to establish theoretical boundaries for the changes in positive predicted values for both groups.

First, let's recall the definition of Equal Opportunity Difference (EOD). EOD is a measure of fairness that evaluates the difference in True Positive Rates (TPR) between privileged and unprivileged groups. The closer this difference is to zero, the better, as it indicates reduced bias and more equitable acceptance rates across groups. In this section, the absolute value of the difference will be considered, as the primary interest is in differences that do not deviate significantly from zero.
The absolute value of EOD is defined as:

$$\text{EOD} = |\text{TPR}_{\text{priv}} - \text{TPR}_{\text{unpriv}}| < \xi \tag{3.1}$$

where ξ is a value close to zero (e.g., 0.3). The True Positive Rate for the privileged group is given by:

$$\text{TPR}_{\text{priv}} = \frac{\text{TP}_{\text{priv}}}{\text{TP}_{\text{priv}} + \text{FN}_{\text{priv}}} \tag{3.2}$$

and the True Positive Rate for the unprivileged group is:

$$\text{TPR}_{\text{unpriv}} = \frac{\text{TP}_{\text{unpriv}}}{\text{TP}_{\text{unpriv}} + \text{FN}_{\text{unpriv}}} \tag{3.3}$$

To ensure fairness, the goal is to decrease the value of EOD by a quantity $0 < \delta < \xi$ such that:

$$\text{EOD} < \xi - \delta \tag{3.4}$$

3.3.1 Graphical Illustration

To illustrate the problem, consider a synthetic dataset with privileged and unprivileged groups, each having two non-protected attributes and one binary protected attribute, where a binary prediction is made ($S \in \{0, 1\}, Y \in \{0, 1\}$). Assume $\text{TPR}_{\text{priv}} > \text{TPR}_{\text{unpriv}}$ and focus on decreasing the EOD by increasing $\text{TPR}_{\text{unpriv}}$. Increasing $\text{TPR}_{\text{unpriv}}$ is preferable to reducing TPR_{priv}, as the latter would result in some privileged individuals receiving a negative label despite deserving a positive one, which is unethical.
By increasing $\text{TPR}_{\text{unpriv}}$, the number of false negatives ($\text{FN}_{\text{unpriv}}$) will decrease by the same amount, as the sum of True Positives (TP) and False Negatives (FN) equals the total number of actual positives ($Y = 1$), which remains constant.

To achieve the increase in $\text{TPR}_{\text{unpriv}}$, the decision boundary of the classifier needs to be shifted in favor of the unprivileged group. This can be accomplished by oversampling the positive unprivileged instances so that both groups achieve the same Imbalance Ratio.

Figure 3.1 illustrates the synthetic training dataset before oversampling (left) and after oversampling (right). In these figures, blue and yellow dots represent the negative and positive instances of the privileged group, respectively, while red and green dots represent the negative and positive instances of the unprivileged group, respectively. The line between the two shaded areas indicates the decision boundary learned by the classifier from the training data. The red area represents the predicted negative zone, while the blue area represents the predicted positive zone. These images show how the decision boundary shifts after the oversampling of the positive unprivileged samples (green dots).

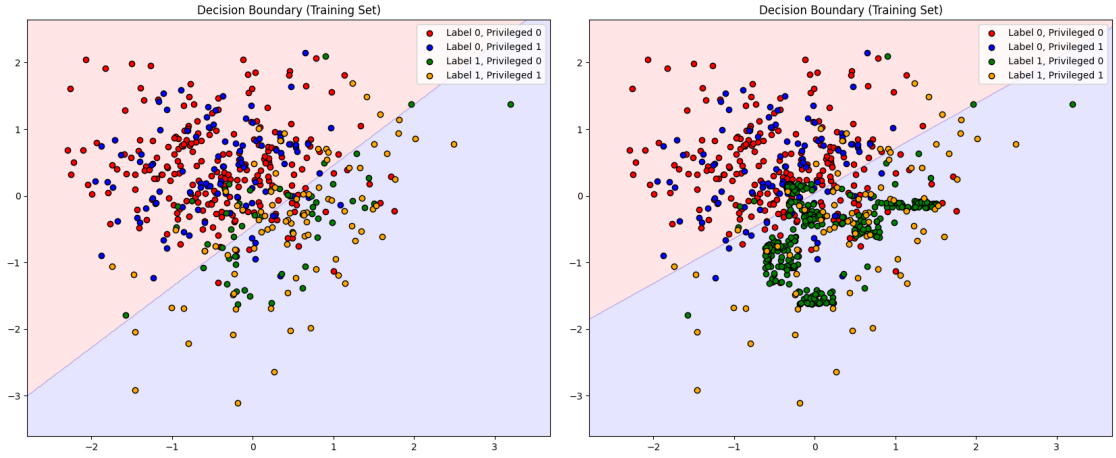

Figure 3.1: Comparison of decision boundaries before (left) and after (right) oversampling the training set.

In Figure 3.2, the same test set used to evaluate the classifier's performance is shown alongside the decision boundaries. The image on the left displays the decision boundary for the test dataset before oversampling, while the image on the right shows the decision boundary for the test dataset after oversampling. Initially, the Disparate Impact was 0.625 and the Equal Opportunity Difference was 0.0664. After oversampling, the DI decreased to 0.594, and the EOD decreased to 0.043, moving closer to zero. It is evident that, after oversampling, more positive unprivileged instances (green dots) fall within the predicted positive area (blue area). Note the decrease in DI and the increase in EOD when shifting the decision boundaries.
With a visual understanding of this process, the theory behind it can now be introduced and complemented with a numerical example.

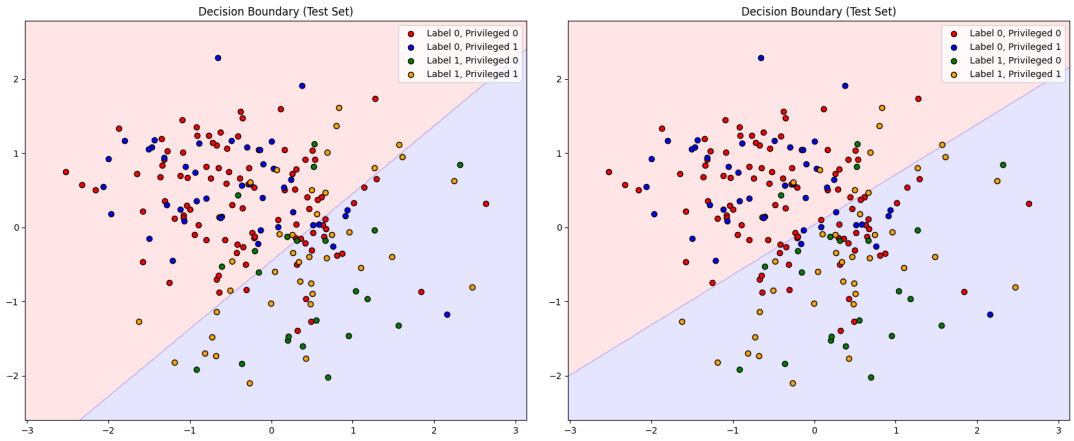

Figure 3.2: Comparison of decision boundaries before (left) and after (right) oversampling the test set.

3.3.2 Theoretically decreasing the EOD

To study this problem theoretically, consider decreasing the Equal Opportunity Difference (EOD) by an amount δ. This requires reducing the distance between the True Positive Rates (TPR) of the privileged and unprivileged groups, and it can be achieved either by decreasing the True Positives of the privileged group or by increasing the TP of the unprivileged group. Starting from the definition of EOD in equation (3.1), and the definitions of TPR for the privileged (equation (3.2)) and unprivileged (equation (3.3)), the problem of decreasing EOD can be rewritten as follows (denoting the privileged group as p, and the unprivileged group as u):

$$-(\xi - \delta) < \frac{TP_p - \lambda}{TP_p + FN_p} - \frac{TP_u + \epsilon}{TP_u + FN_u} < \xi - \delta$$

This can be studied through three theoretical cases:

Case 1: $\lambda = 0$, $\epsilon \neq 0$:
In this scenario, only the number of True Positives for the unprivileged group is increased, while the values for the privileged group remain unchanged. Under this hypothesis, the upper and lower bounds for ϵ can be derived as follows:

$$\begin{cases} \epsilon > (TP_u + FN_u)(TPR_p - TPR_u - \xi + \delta) \\ \epsilon < (TP_u + FN_u)(TPR_p - TPR_u + \xi - \delta) \end{cases} \tag{3.5}$$

Case 2: $\epsilon = 0$, $\lambda \neq 0$:
In this scenario, only the True Positives for the privileged group are decreased, while the values for the unprivileged group remain unchanged. Under this hypothesis, the lower

and upper bounds for λ can be derived as follows:

$$\begin{cases} \lambda > (TP_p + FN_p)(TPR_p - TPR_u - \xi + \delta) \\ \lambda < (TP_p + FN_p)(TPR_p - TPR_u + \xi - \delta) \end{cases} \tag{3.6}$$

Case 3: $\epsilon \neq 0$, $\lambda \neq 0$:
In this case, both a decrease in the number of True Positives for the privileged group and an increase in the number of TP for the unprivileged group are considered. Assuming that the only unknowns in these inequalities are λ and ϵ, the inequalities can be rewritten with λ as a dependent variable of the independent variable ϵ:

$$\begin{cases} \lambda > (\mathrm{TP}_p + \mathrm{FN}_p)\left(\mathrm{TPR}_p - \xi + \delta - \mathrm{TPR}_u - \frac{\epsilon}{\mathrm{TP}_u + \mathrm{FN}_u}\right) \\ \lambda < (\mathrm{TP}_p + \mathrm{FN}_p)\left(\mathrm{TPR}_p + \xi - \delta - \mathrm{TPR}_u - \frac{\epsilon}{\mathrm{TP}_u + \mathrm{FN}_u}\right) \end{cases} \tag{3.7}$$

Since all values except λ and ϵ are known, λ is constrained by two linear functions of ϵ.

Identifying the upper and lower bounds for these adjustment factors is crucial to avoid adverse effects. For instance, an excessively large shift in the decision boundary could result in one group having a disproportionately high TPR compared to the other, creating an inverted fairness issue.

3.3.3 Numerical Example

Consider the initial values for the True Positives and False Negatives of both privileged and unprivileged groups:

$$\begin{aligned} \mathrm{TP_p} &= 80, \quad \mathrm{FN_p} = 20 \\ \mathrm{TP_u} &= 50, \quad \mathrm{FN_u} = 50 \end{aligned}$$

The initial True Positive Rates are calculated as follows:

$$\begin{aligned} \mathrm{TPR_p} &= \frac{80}{80 + 20} = 0.8 \\ \mathrm{TPR_u} &= \frac{50}{50 + 50} = 0.5 \end{aligned}$$

The initial Equal Opportunity Difference is:

$$\mathrm{EOD} = 0.8 - 0.5 = 0.3 \tag{3.8}$$

Thus, $\xi = 0.3$. To decrease the EOD by at least $\delta = 0.1$, consider Case 1 where $\lambda = 0$ (only increasing the TP for the unprivileged group). The following bounds are obtained:

$$\begin{cases} \epsilon > (50 + 50)(0.8 - 0.5 - 0.3 + 0.1) = 10 \\ \epsilon < (50 + 50)(0.8 - 0.5 + 0.3 - 0.1) = 50 \end{cases} \tag{3.9}$$

Analyzing values of ϵ outside these boundaries:

- For $\epsilon = 5$: $EOD = |0.8 - \frac{(50+5)}{(50+50)}| = 0.25$
  EOD does not decrease by $\delta = 0.1$;
- For $\epsilon = 55$: $EOD = |0.8 - \frac{(50+55)}{(50+50)}| = 0.25$
  Again, EOD does not decrease by the desired amount;
- For $\epsilon = 40$: $EOD = |0.8 - \frac{(50+40)}{(50+50)}| = 0.1$
  EOD decreases by more than $\delta = 0.1$.

Similar considerations apply for Case 2, where $\epsilon = 0$ and only λ changes (decreasing the TP for the privileged group):

$$\begin{cases} \lambda > (80+20)(0.8 - 0.5 - 0.3 + 0.1) = 10 \\ \lambda < (80+20)(0.8 - 0.5 + 0.3 - 0.1) = 50 \end{cases} \tag{3.10}$$

In Case 3, where both $\epsilon \neq 0$ and $\lambda \neq 0$, the system of inequalities (3.7) becomes:

$$\begin{cases} \lambda > (80+20)(0.8 - 0.3 + 0.1 - 0.5 - \frac{\epsilon}{50+50}) \\ \lambda < (80+20)(0.8 + 0.3 - 0.1 - 0.5 - \frac{\epsilon}{50+50}) \end{cases} \tag{3.11}$$

Simplifying further:

$$\begin{cases} \lambda + \epsilon > 10 \\ \lambda + \epsilon < 50 \end{cases} \tag{3.12}$$

By adjusting these parameters within the given limits, the EOD can be effectively reduced without overcompensating for either group. This ensures a balanced approach to fairness, preventing a scenario where one group's TPR becomes disproportionately high relative to the other, thereby maintaining a more equitable decision boundary.

3.4 DBSCAN Algorithm

As discussed in the previous section, oversampling is employed to shift the decision boundary of the classifier in favor of the unprivileged group. A crucial step in this process is determining which points to select for oversampling. This section discusses the DBSCAN algorithm, which is used to identify the instances to be oversampled, as will be explained in more detail in the next chapter.
Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is a clustering algorithm first introduced by Ester et al. [24]. Its uniqueness lies in its ability to create clusters of arbitrary shape based on the notion of density. Additionally, DBSCAN does not require prior knowledge of the number of clusters, as it automatically determines them based on the density of points.
Since clusters are defined as regions with higher point density, a distance function is required to measure the distance between points, allowing for the identification of these

dense regions. As noted by Ester et al. [24], the algorithm works with any distance function. In this work, Gower's distance is employed [31] (Section 3.5) because it can handle samples with both numerical and categorical features.
Once the notion of distance is defined, the DBSCAN algorithm requires two parameters from the user:

- **eps**: This is the radius used to determine the *eps-neighborhood* of a point. For a point p in a dataset D, all points within a distance less than or equal to *eps* from p are considered neighbors of p, i.e., $N_{eps}(p) = \{q \in D | d(q,p) \leq eps\}$.
- **MinPts**: This is the minimum number of *eps-neighbors* (including the point itself) a point must have to be considered a core point [56].

With these two parameters, points can be categorized into three types:

- **Core Points**: Points with a number of neighbors greater than or equal to *MinPts*. These points form the central part of the clusters.
- **Border Points**: Points with fewer neighbors than *MinPts* but within the neighborhood of a core point. These points lie on the edges of clusters.
- **Noise Points**: Points with fewer neighbors than *MinPts* and not within the neighborhood of a core point. These points are considered outliers and lie outside clusters.

The classification of points into these categories depends on the chosen values for *eps* and *MinPts*. For instance, smaller *eps* values lead to more isolated clusters.
Figure 3.3 illustrates how DBSCAN categorizes the points in the dataset into these three types. In this example, there is only one cluster. The green points are core points, each having at least $MinPts = 3$ points (including itself) within its neighborhood, represented by the circle around it. The blue point is a border point, with less than *MinPts* neighbors (including itself) but within the neighborhood of a core point. The red point is a noise point, having no other points within its neighborhood, thus being an outlier. It is important to note that a point can be classified as noise even if it has border points within its neighborhood.

3.5 Gower's Distance

In this work, Gower's distance is utilized as a precomputed distance measure in the DBSCAN algorithm. This enables DBSCAN to effectively cluster data that includes both numerical and categorical features.
Gower's distance, introduced by J. C. Gower [31], is a metric designed to quantify the distance between two individuals. It is particulary significant for its ability to handle datasets comprising both numerical and categorical attributes, unlike more traditional metrics such as Euclidean or Manhattan distance, which typically only handle numerical attributes.
In his paper, Gower introduces a similarity matrix in which each entry represents the similarity between two individuals, with values ranging from 0 (very dissimilar) to 1 (very

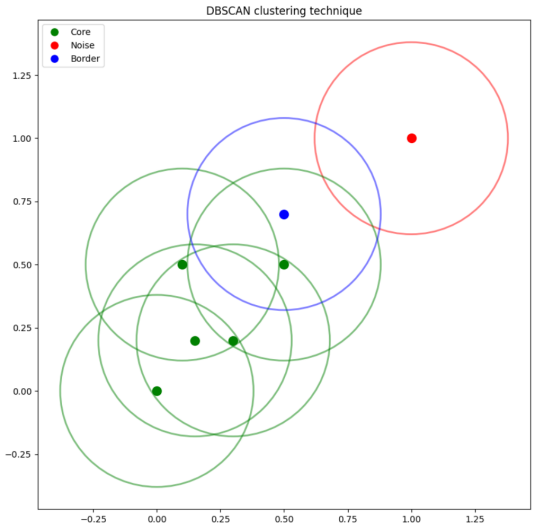

Figure 3.3: DBSCAN classification with MinPts = 3 and eps = 0.38. Green points are core points with at least three neighbors. The blue point is a border point, within the neighborhood of a core point. The red point is a noise point, with no neighbors, thus classified as an outlier.

similar).
Given two individuals in a population defined by m features (both categorical and numerical) $\boldsymbol{x}_i = x_{i1}, x_{i2}, ..., x_{im}$ and $\boldsymbol{x}_j = x_{j1}, x_{j2}, ..., x_{jm}$, the entry in the matrix representing their similarity is defined as:

$$S_{Gower}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{1}{m} \sum_{k=1}^{m} s_{ijk}$$

where the values s_{ijk} are the Gower similarity scores, computed differently based on the nature of the feature. Specifically:

- For **numerical** features, the Manhattan distance is used. The similarity score is defined as:
$$s_{ijk} = 1 - \frac{|x_{ik} - x_{jk}|}{R_k}$$
where R_k is the range of the numerical feature, i.e., the maximum value minus the minimum value.

- For **categorical** features, the similarity score is 1 if the two features have the same value, and 0 otherwise:
$$s_{ijk} = \begin{cases} 1 & x_{ik} = x_{jk} \\ 0 & x_{ik} \neq x_{jk} \end{cases}$$

The Gower's distance matrix is derived from the similarity matrix as follows:

$$d_{Gower} = 1 - S_{Gower}$$

where values of d_{Gower} close to zero represent similar individuals, while values close to one represent individuals that are dissimilar.

To better understand the computation of the Gower's distance matrix, consider the following example:
Given three individuals in a population, defined by $m = 4$ features (two numerical and two categorical):

$$x_1 = [10, 9, \text{Male}, \text{White}]$$
$$x_2 = [5, 20, \text{Female}, \text{White}]$$
$$x_3 = [2, 5, \text{Female}, \text{Black}]$$

The distance between individuals x_1 and x_2 is computed as follows:

$$d_{Gower}(x_1, x_2) = 1 - \frac{1}{4}\left[\left(1 - \frac{|10-5|}{10-2}\right) + \left(1 - \frac{|9-20|}{20-5}\right) + 0 + 1\right] \approx 0.59$$

3.6 Oversampling with SMOTE-NC

After identifying the core, border, and noise points for each subgroup using DBSCAN, the next step is to create new synthetic samples between the border points and the border/core points. For this purpose, SMOTE-NC will be used to perform the oversampling. Oversampling is one of the pre-processing bias mitigation techniques introduced in Section 2.3. It plays a crucial role in this work as it is used to create synthetic samples to address the representation bias present in the original dataset.
SMOTE (Synthetic Minority Over-sampling Technique) is an oversampling algorithm introduced by Chawla et al. [13]. SMOTE was developed to overcome the limitations of classical resampling strategies, which often duplicate minority points already present in the dataset, potentially leading to overfitting. Instead, Chawla et al. proposed an innovative approach that interpolates between existing minority points to generate new synthetic samples, thereby enriching the dataset without simply replicating existing data.

The details of how SMOTE works are as follows. Consider a numerical dataset composed of a majority group and a minority group. SMOTE begins by determining the number of new points that need to be added to the minority group, typically to equalize the number of points with those in the majority group. For each minority sample $\boldsymbol{x}_i$, the algorithm identifies the 5 nearest neighbors within the minority group and randomly selects one of these neighbors, $\boldsymbol{x}_j$. A new synthetic sample is then created by interpolating between the original minority point and the selected neighbor.
Consider the two points $\boldsymbol{x}_i = x_{i1}, x_{i2}, ..., x_{im}$ and $\boldsymbol{x}_j = x_{j1}, x_{j2}, ..., x_{jm}$ consisting of m numerical features. The *k-th* feature of the new synthetic point is defined as an interpolation between the *k-th* feature of the selected point, x_{ik}, and the *k-th* feature of the neighbor, x_{jk}, as follows:

$$x_{syn,k} = x_{ik} + \alpha(x_{jk} - x_{ik}) \tag{3.13}$$

where α is a random number between 0 and 1. Thus, the new synthetic point lies on the line segment connecting the two points, at a position determined by the randomly chosen

α. Figure 3.4 illustrates the process of creating a new synthetic point using SMOTE. The black point represents the selected minority point. Its 5 nearest neighbors, also from the minority group, are connected to this main point with dashed lines. One of these neighbors is then chosen, and the blue synthetic point is created along the line segment connecting the selected minority point and the chosen neighbor.

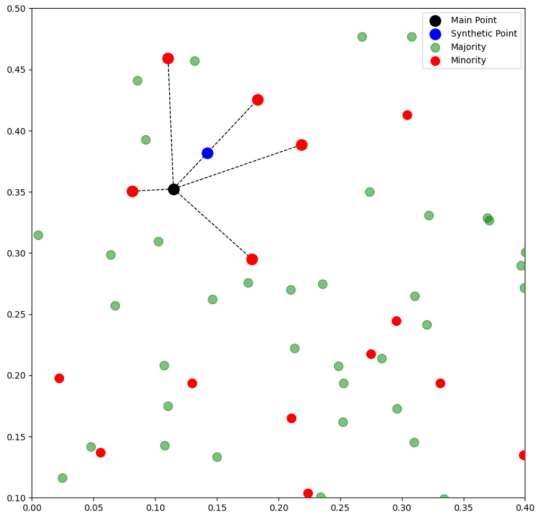

Figure 3.4: Application of the SMOTE algorithm on a dataset with a majority group (green points) and a minority group (red points). A minority point is selected (black point), and a new synthetic point (blue point) is created through interpolation.

When the dataset contains both numerical and categorical attributes, the SMOTE-NC algorithm, proposed by Chawla et al. [13], is employed to handle both types of attributes. This algorithm is utilized in this work to perform oversampling. Numerical features are interpolated in the same manner as in SMOTE. For categorical attributes, the value assigned is the one that occurs most frequently among the 5-nearest neighbors. For example, if the categorical attribute is 'Gender' and the 5-nearest neighbors have values 'M', 'M', 'F', 'M', and 'F', then the synthetic point will have 'M' for the 'Gender' attribute, as it is the most frequent value.

3.7 Logistic Regression

Once the training dataset is oversampled, it can be used to train a classification algorithm. This allows for the computation of fairness measures and evaluation metrics by testing the classifier's performance on the test set. Logistic Regression is selected for this study because it inherently represents the concept of a decision boundary, which this work aims to shift.
Logistic Regression, first introduced by D. R. Cox [17], is a classification algorithm used to analyze the dependency of a binary response variable on one or more explanatory

variables. Cox's work specifically examined how different values of explanatory variables lead to different binary responses.
Let Y be the binary response variable, and $\mathbf{x} = x_1, x_2, ..., x_k$ the explanatory variables. Cox proposed the logistic law to link the probability of the response being one, $\pi(\mathbf{x}) = P[Y = 1|\mathbf{x}]$, to the values of the explanatory variables as follows:

$$\text{logit}(\pi(\mathbf{x})) = \ln\left(\frac{\pi(\mathbf{x})}{1-\pi(\mathbf{x})}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_k x_k$$

where $\beta_j, j = 0, ..., k$ are the regression coefficients for the explanatory variables. Specifically, β_j quantifies the change in the log odds for a one-unit change in x_j, keeping all other explanatory variables constant, and assuming $x_0 = 1$.
The above formula can be rewritten in terms of probabilities as follows:

$$\pi(\mathbf{x}) = \frac{e^{\beta_0+\beta_1 x_1+\beta_2 x_2+...+\beta_k x_k}}{1 + e^{\beta_0+\beta_1 x_1+\beta_2 x_2+...+\beta_k x_k}}$$

$$1 - \pi(\mathbf{x}) = \frac{1}{1 + e^{\beta_0+\beta_1 x_1+\beta_2 x_2+...+\beta_k x_k}}$$

The regression coefficients are estimated by maximizing the likelihood function, which represents the plausibility of the model. Consider a dataset with n independent observations $(\mathbf{x}_i, Y_i), i = 1, ..., n$, each composed of the vector of explanatory variables and the binary response. Since the binary response can only take two values based on the explanatory variables, it follows a Bernoulli distribution $Y_i \sim Bernoulli((\pi(\mathbf{x}_i)))$ [26]. The likelihood function is expressed as:

$$\text{L}(\beta|\mathbf{x}) = \prod_{i=1}^{n} \pi(\mathbf{x}_i)^{Y_i} \left(1 - \pi(\mathbf{x}_i)\right)^{1-Y_i} = \prod_{i=1}^{n} \frac{\pi(\mathbf{x}_i)}{1-\pi(\mathbf{x}_i)}^{Y_i} \left(1 - \pi(\mathbf{x}_i)\right)$$

Using the definitions of $\pi(\mathbf{x}_i)$ and $1-\pi(\mathbf{x}_i)$, and the vector representation $\beta = [\beta_0, \beta_1, ..., \beta_k]$ and $\mathbf{x}_i = [x_0, x_1, ..., x_k]'$, the likelihood function can be written as:

$$\text{L}(\beta|\mathbf{x}) = \prod_{i=1}^{n} \left(e^{\beta \mathbf{x}_i}\right)^{Y_i} \left(\frac{1}{1 + e^{\beta \mathbf{x}_i}}\right)$$

To simplify the optimization, the log-likelihood function is considered because logarithms are strictly increasing monotonic functions. The log-likelihood function is then:

$$\mathcal{L}(\beta|\mathbf{x}) = \ln \text{L}(\beta|\mathbf{x}) = \sum_{i=1}^{n} Y_i \beta \mathbf{x}_i - \sum_{i=1}^{n} \ln\left(1 + e^{\beta \mathbf{x}_i}\right)$$

The derivative of the log-likelihood function with respect to a regression coefficient β_j is computed and set to zero to find the maximum likelihood estimates:

$$\frac{\partial \mathcal{L}(\beta|\mathbf{x})}{\partial \beta_j} = \sum_{i=1}^{n} \left[Y_i - \pi(\mathbf{x}_i)\right] x_{ij} = 0 \qquad j = 1, ..., k$$

Since these equations are typically infeasible to solve analytically, approximation methods such as the Newton-Raphson Iterative Algorithm are used to find acceptable values for each β_j. This algorithm iteratively updates the regression coefficients until convergence is reached [26].

The linear combination of the explanatory variables and their corresponding coefficients, $\beta\mathbf{x}$, represents the decision boundary in the feature space. This decision boundary is what the model uses to separate the classes, and adjusting it is crucial for addressing representation bias. Once the regression coefficients are defined, the probability of an event $\mathbf{x}$ can be computed, and a response value assigned. In general, when the probability is greater than a certain threshold (typically 0.5), a positive response ($Y = 1$) is assigned; otherwise, a negative response ($Y = 0$) is assigned.

Chapter 4

Proposed Framework for Mitigating Representation Bias

This chapter presents a comprehensive framework designed to mitigate representation bias in datasets. The framework comprises several key steps, beginning with data preparation to ensure the dataset is standardized and ready for analysis. This initial step involves the selection and refinement of attributes and rows, followed by standardization and encoding to facilitate accurate analysis.

Next, the framework considers all possible combinations of binary sensitive attributes and binary label to define subgroups within the dataset. These subgroups are used to stratify the dataset into training and testing sets, ensuring proportional representation. The imbalance ratios (IR) of these groups are then computed to identify the most privileged group, defined by the highest IR. Subsequently, oversampling is performed on all other groups to achieve the same IR as the most privileged group.

To execute the oversampling process, the DBSCAN algorithm is applied to each subgroup with positive label to identify core, border, and noise points. Following this, the SMOTE-NC algorithm generates new synthetic points by interpolating between border points and their neighboring core or border points. These new synthetic samples are then integrated into the training dataset, which is subsequently used to train a classifier. The performance of this classifier, quantified through both evaluation metrics and fairness measures, is compared to the performance of a classifier trained on the original dataset (prior to oversampling).

This proposed framework is visually represented in the flowchart in Figure 4.1, illustrating the sequential steps from data preparation to the final evaluation.

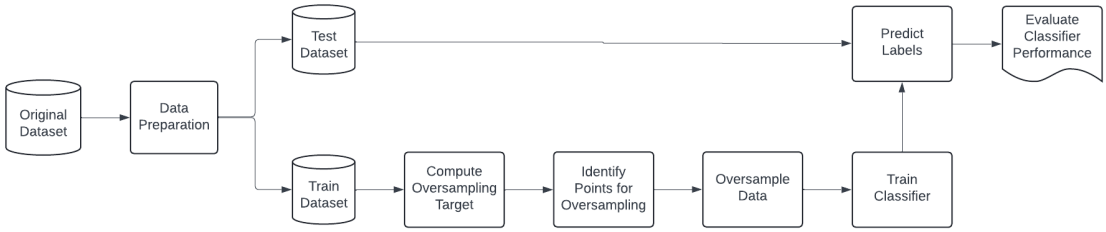

Figure 4.1: Flowchart of the proposed framework

4.1 Data Preparation and Imbalance Ratio Computation

The original dataset is prepared and refined through a comprehensive data preparation process to ensure it is ready for further analysis. This step involves several key tasks aimed at adjusting the dataset.

- **Attribute and Row Selection**: During data preparation, a careful selection of attributes and rows is performed. Attributes that are redundant or highly correlated with others are removed, as they do not provide extra value. Additionally, rows with missing values are excluded to ensure the integrity of the dataset.
- **Standardization and Encoding**: To standardize the dataset, categorical attributes are encoded, and numerical attributes are subjected to Z-score normalization. This ensures that all data is on a common scale, facilitating more accurate analysis.
- **Binary Sensitive Attributes and Binary Labels**: The set of sensitive attributes (e.g., Gender, Race) and a binary label (e.g., hired: Y/N) must be provided. This study focuses on binary sensitive attributes. If a protected attribute has more than two values, the dataset is either filtered to include only the two most common groups or modified to combine categories. For example, if 'Race' includes 'Caucasian', 'African-American', and 'Hispanic', the dataset might be filtered to only 'Caucasian' and 'African-American', or combined into 'Caucasian' and 'Non-Caucasian'.
- **Group and Subgroup Identification**: With binary sensitive attributes and binary labels, distinct groups and subgroups within the dataset can be identified. Groups are defined by all possible combinations of the sensitive attributes. If there are n binary sensitive attributes, this results in 2^n different groups. For instance, with two sensitive attributes such as Gender (M/F) and Race (W/B), there are $2^2 = 4$ groups: (M, W), (M, B), (F, W), and (F, B).
  Subgroups are then formed by splitting each group based on the binary label, resulting in 2^{n+1} subgroups. In the previous example, considering a binary label with values 1 (favorable outcome) and 0 (unfavorable outcome), this yields $2^{2+1} = 8$ subgroups: (M, W, 1), (M, W, 0), (M, B, 1), (M, B, 0), (F, W, 1), (F, W, 0), (F, B, 1),

and (F, B, 0).

- **Splitting into Training and Test Sets**: The prepared dataset, now standardized and without missing values, is split into 70% train and 30% test sets with stratification based on the identified subgroups. This ensures that each subgroup is proportionally represented in both sets.
- **Imbalance Ratio (IR) Computation**: To determine the most privileged group, the Imbalance Ratio (IR) is calculated for each group. The IR is the ratio of the subgroup with positive label to the subgroup with negative label. For example, the IR for the (M, W) group is computed as:

$$\mathrm{IR}_{(\mathrm{M,W})} = \frac{|(\mathrm{M}, \mathrm{W}, 1)|}{|(\mathrm{M}, \mathrm{W}, 0)|}$$

The group with the highest IR is considered the most privileged, and its imbalance ratio ($\mathrm{IR}_{\max}$) serves as the target for oversampling other groups. For each unprivileged group, defined as a pair of subgroups (PU, NU) with the same sensitive attributes but opposite labels, the oversampling target is calculated as:

$$\text{oversampling\_target} = (\mathrm{IR}_{\max} * |\mathrm{NU}|) - |\mathrm{PU}|$$

This ensures that after oversampling, each group achieves the same imbalance ratio as the most privileged group.

4.2 Identifying Border Points for Oversampling Using DBSCAN

With the training dataset prepared, subgroups identified, and the most privileged group determined, the next step involves performing oversampling. The objective is to adjust the dataset so that each group ultimately achieves the same Imbalance Ratio (IR) equal to $\mathrm{IR}_{\max}$. For each group, defined as a pair of subgroups (PU, NU), the aim is to oversample points from PU (those with positive label) to increase the IR of the group. DBSCAN is applied within each subgroup PU to identify the points for oversampling.

As discussed in Section 3.4, DBSCAN is a density-based clustering algorithm requiring a distance metric to define density, an **eps** value to determine the neighborhood of a point, and a **MinPts** value to define core points. In this work, for each subgroup PU, the Gower's distance matrix (Section 3.5) is calculated and used as a precomputed metric for the DBSCAN algorithm. This approach takes both numerical and categorical attributes into account in the distance calculations.
For the **MinPts** parameter, the natural logarithm of the total number of samples in the subgroup is used. This choice adapts automatically to the size of the subgroup, effectively filtering out noise points while remaining robust and sensitive to the subgroup's specific characteristics.
The **eps** value is selected to allow for the formation of a unique cluster with some noise

points. This eps value is calibrated so that a slight decrease would result in the formation of multiple clusters, as shown in Figure 4.2. The primary idea is to treat each subgroup as a single cluster while identifying core, border, and noise points within that cluster. This approach facilitates the selection of border points for oversampling, based on the rationale that border points are more likely to be misclassified. Specifically, oversampling border points helps shift the classifier's decision boundary in favor of the unprivileged group under analysis.

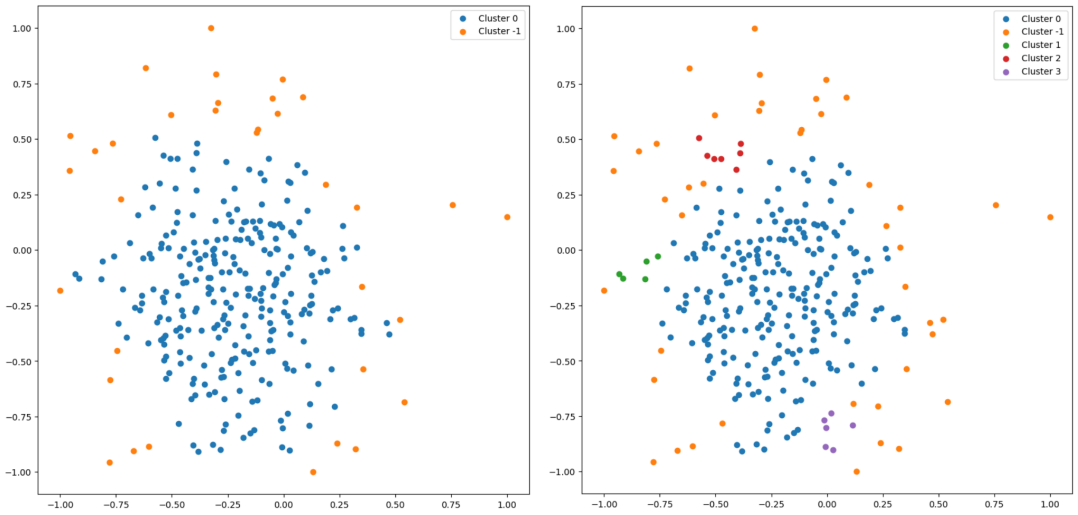

Figure 4.2: Comparison of clusters created by DBSCAN with the same MinPts but different epsilon values. Using eps = 0.13 (left) results in the formation of a single cluster with some noise points, while a slight decrease to eps = 0.12 (right) leads to the formation of multiple clusters.

To determine the **eps** value that allows for the creation of a single cluster but leads to multiple clusters if decreased, a binary search is implemented. The procedure is detailed in Algorithm 1. Specifically, the optimal **eps** value is searched within a specified interval between eps_{min} and eps_{max}. In each iteration, the midpoint eps_{mid} of the interval is computed and used to create clusters with the DBSCAN algorithm (using the same distance matrix and **MinPts**, varying only **eps**). Based on the number of clusters created and their characteristics, the interval is adjusted by either increasing eps_{min} or decreasing eps_{max}, until the difference between them is less than the specified step.

At the end of this procedure, each subgroup PU is represented by a single cluster, within which points are classified as core, border, or noise points. The border points and their nearest neighbors will then be selected for the oversampling process.

4.3 Oversampling Border Points with SMOTE-NC

After using the DBSCAN algorithm to identify core, border, and noise points for each subgroup PU, the next step is to perform the oversampling process. New synthetic points are created for each unprivileged subgroup PU to ensure that, by the end of the oversampling, each group achieves the same Imbalance Ratio (IR) as the most privileged group. This approach aims to mitigate representation bias, resulting in a dataset where each group has an equal acceptance rate.

For each subgroup PU, the oversampling process continues until the number of new synthetic samples exceeds the oversampling target, defined as:

$$\text{oversampling\_target} = (\text{IR}_{\text{max}} * |\text{NU}|) - |\text{PU}|$$

The new points are then added to the training dataset to increase the number of samples with positive labels for each unprivileged group.

The creation of new synthetic points is executed using the SMOTE-NC algorithm (as detailed in Section 3.6), which interpolates between border points and their neighboring border/core points. Specifically, for each subgroup PU, a border point is selected, and its 5 nearest neighbors are identified, excluding noise points to retain only core and border points. The choice of 5 nearest neighbors follows the methodology established by Chawla et al. [13], the authors of SMOTE and SMOTE-NC. If valid neighbors are present, one is randomly selected. A new synthetic point is then generated using SMOTE-NC interpolation: categorical attributes are assigned the most frequent value among the valid neighbors, and numerical attributes are determined through linear interpolation between the selected border point and its chosen neighbor.
The oversampling procedure is detailed in Algorithm 2.

After the oversampling process, the new synthetic samples are integrated into the training dataset, ensuring that each group achieves the same IR. This balanced dataset will then be used to train a classifier. The test dataset is subsequently used to evaluate the trained classifier by comparing the predicted labels with the actual labels.
The performance of this classifier will be assessed against that of a classifier trained on the original dataset (before oversampling), using both evaluation metrics and fairness measures.

Algorithm 1 Find Optimal Epsilon

Input: PU, categorical attributes cat_attr, MinPts min_samples, distance_matrix, eps_step, eps_{min}, eps_{max}
Output: optimal epsilon value

function FIND_EPS(PU, cat_attr, min_samples, distance_matrix, eps_step, eps_{min}, eps_{max})

function CLUSTER_COUNT(eps)
dbscan ← DBSCAN(eps, min_samples, 'precomputed')
clusters ← dbscan.fit_predict(distance_matrix)
return #clusters, label of clusters ▷ Label -1 indicates noise points
end function

while eps_{max} - eps_{min} > eps_step **do**
$\text{eps}_{\text{mid}} \leftarrow \frac{1}{2}(\text{eps}_{\text{min}} + \text{eps}_{\text{max}})$
n_clusters_mid, labels_mid ← CLUSTER_COUNT(eps_{mid})

if n_clusters_mid == 1 **then**
if -1 in labels_mid **then**
$\text{eps}_{\text{min}} \leftarrow \text{eps}_{\text{mid}}$ ▷ Only noise points, increase eps_{min}
else
$\text{eps}_{\text{max}} \leftarrow \text{eps}_{\text{mid}}$ ▷ Only core points, decrease eps_{max}
end if

else if n_clusters_mid > 2 **then**
$\text{eps}_{\text{min}} \leftarrow \text{eps}_{\text{mid}}$ ▷ More than two clusters, increase eps_{min}

else
$\text{eps}_{\text{max}} \leftarrow \text{eps}_{\text{mid}}$ ▷ Exactly two clusters, fine-tune further
end if
end while
return eps_{max}
end function

Algorithm 2 SMOTE-DBSCAN Framework

Input: train dataset D_{train}, categorical attributes cat_attr, imbalance ratio IR_{max}, sensitive attributes S, label Y
Output: new synthetic samples for D_{train}

function OVERSAMPLE_GROUPS(D_{train}, cat_attr, IR_{max}, S, Y)
    synthetic_samples ← []
    subgroups ← each combination of S and Y in D_{train}
    paired_subgroups ← pairs of subgroups with same S and opposite Y
    for each pair (PU, NU) in paired_subgroups **do**
        IR ← $\frac{|PU|}{|NU|}$
        synthetic_points ← CUSTOM_SMOTE_DBSCAN(IR, IR_{max}, PU, NU)
        synthetic_samples ← synthetic_samples ∪ synthetic_points
    end for
    return synthetic_samples
end function

function CUSTOM_SMOTE_DBSCAN(IR, IR_{max}, PU, NU)
    oversampling_target ← (IR_{max} × |NU|) - |PU|
    distance_matrix ← GOWER_MATRIX(PU, cat_attr)
    min_samples ← ln(|PU|)
    eps ← FIND_EPS(PU, cat_attr, min_samples, distance_matrix)
    dbscan ← DBSCAN(eps, min_samples, 'precomputed')
    clusters ← dbscan.fit_predict(distance_matrix)
    core_points, border_points, noise_points ← DBSCAN results
    synthetic_points ← []
    current_index ← 0
    while |synthetic_points| < oversampling_target **do**
        point_A ← border_points[current_index]
        neighbors_A ← 5 nearest neighbors of point_A from distance_matrix
        valid_neighbors_A ← neighbors_A not noise_points
        if valid_neighbors_A is not empty **then**
            point_B ← random select from valid_neighbors_A
            synthetic_point ← []
            for each attribute col in PU **do**
                if col is in cat_attr **then**
                    synthetic_point[col] ← most frequent value in valid_neighbors_A[col]
                else
                    α ← random number between 0 and 1
                    synthetic_point[col] ← point_A[col] + α(point_B[col] - point_A[col])
                end if
            end for
            synthetic_points ← synthetic_points ∪ synthetic_point
        end if
        increment current_index
    end while
    return synthetic_points
end function

Chapter 5

Evaluation

This chapter presents a comprehensive evaluation of the proposed DBSCAN-SMOTE framework for mitigating representation bias. The evaluation is structured into several sections to ensure a thorough analysis. Initially, the datasets used in the experiments are described, detailing their characteristics and the distribution of sensitive attributes. Following this, the evaluation metrics and fairness measures are reiterated to provide clarity on the assessment criteria. The core of the chapter then focuses on the experiments conducted to determine the framework's performance, comparing it with alternative approaches and existing bias mitigation algorithms.

5.1 Datasets

The proposed framework addresses the representation bias present in the original datasets by performing oversampling to enhance fairness. The resulting oversampled dataset is then used to train a classification model, whose performance is evaluated and compared to that of a classifier trained on the original dataset.
This section describes the three datasets used to perform the analysis. Each dataset meets the requirements of this work, specifically containing binary sensitive attributes and a binary label. The characteristics of each dataset, such as the number of samples, features, and the distribution of sensitive attributes, are detailed. The three well-known datasets used are the *German Credit* dataset [36], the *COMPAS Recidivism* dataset [58], and the *Adult* dataset [8].

- **German Credit dataset**:
  The German Credit dataset comprises 1,000 samples, each representing an individual who has taken a credit from a bank [36]. Individuals are classified as good or bad credit risks based on attributes such as job status, credit history, and age, among others. Each individual in the dataset is described by 20 features of both categorical and numerical types.
  The binary sensitive attributes in this dataset are 'Gender' and 'Age'. The 'Gender' attribute is categorized as Male and Female. The 'Age' attribute is binarized by classifying individuals aged 25 years and older as Adults, and those younger than

25 years as Young. The label indicates the binary classification of good versus bad credit risks.
Considering all possible combinations of sensitive attributes and the binary label, the dataset consists of 4 groups, further divided into 8 subgroups: (Male, Adult, good), (Male, Adult, bad), (Male, Young, good), (Male, Young, bad), (Female, Adult, good), (Female, Adult, bad), (Female, Young, good), and (Female, Young, bad).
The Imbalance Ratio (IR) for each of the 4 groups is as follows:

Group	IR
(Male, Adult)	2.767
(Male, Young)	1.556
(Female, Adult)	2.098
(Female, Young)	1.360

It is observed that the (Male, Adult) group is the most privileged, as it has the highest IR, while the (Female, Young) group is the most unprivileged, with the lowest IR. The objective of the proposed framework is to oversample each group of the German dataset so that, ultimately, each imbalance ratio approximates 2.767.

- **COMPAS Recidivism dataset**:
The COMPAS Recidivism dataset contains criminal records of defendants from Broward County from 2013 and 2014 [58]. The original dataset includes 7,214 samples, detailing past crimes, types of offenses, and time spent in jail. Following the feature selection reported in the AI Fairness 360 documentation [3], the dataset is reduced to 9 attributes (5 categorical, 4 numerical) and a binary label.
The binary label indicates the likelihood of recidivism, with 1 representing non-recidivism and 0 representing recidivism. The binary sensitive attributes are 'Race' and 'Sex'. The 'Race' attribute is filtered to include only 'Caucasian' and 'African-American', resulting in 6,150 samples
The dataset is divided into 4 groups based on combinations of sensitive attributes and binary labels, further divided into 8 subgroups: (Caucasian, Female, 1), (Caucasian, Female, 0), (Caucasian, Male, 1), (Caucasian, Male, 0), (African-American, Female, 1), (African-American, Female, 0), (African-American, Male, 1), and (African-American, Male, 0). The Imbalance Ratios (IR) for each group are:

Group	IR
(Caucasian, Female)	1.842
(Caucasian, Male)	1.464
(African-American, Female)	1.655
(African-American, Male)	0.838

The most privileged group is (Caucasian, Female), with the highest IR, while the most unprivileged group is (African-American, Male), with the lowest IR. This indicates that African-American males are more likely to be predicted as recidivists

Group	IR
(White, Male)	0.479
(White, Female)	0.140
(Black, Male)	0.235
(Black, Female)	0.064

compared to Caucasian females. The goal of the proposed framework is to equalize the IR across all groups to match that of the most privileged group.

- **Adult dataset**:
  The Adult dataset is the largest dataset used in this work, containing 48,842 samples. Each row describes an individual using 14 attributes, such as educational status, occupation, and native country, to predict their annual income.
  The sensitive attributes are 'Race', which is binarized by selecting instances with values either 'White' or 'Black', and 'Sex', which takes values 'Male' or 'Female'. The label predicts whether a person's annual income exceeds \$50K (favorable outcome) or is less than \$50K (unfavorable outcome).
  As with the other datasets, this dataset is divided into 8 subgroups based on combinations of sensitive attributes and the binary label: (White, Male, $> 50K$), (White, Male, $\leq 50K$), (White, Female, $> 50K$), (White, Female, $\leq 50K$), (Black, Male, $> 50K$), (Black, Male, $\leq 50K$), (Black, Female, $> 50K$), and (Black, Female, $\leq 50K$). The Imbalance Ratios (IR) for each group are:

  The IR is particularly low for all groups in this dataset. The most privileged group is (White, Male), while the most unprivileged group is (Black, Female), indicating that black females are less likely to be predicted an income higher than \$50K compared to white males. The objective of the bias mitigation method is to oversample such that each group achieves an IR approximately equal to 0.479, the IR of the most privileged group.

Table 5.1 summarizes the main characteristics of the three selected datasets.

Characteristics	German Credit	COMPAS Recidivism	Adult
Number of Rows	1,000	6,150	48,842
Number of Features	20	9	14
Sensitive Attributes	Gender, Age	Race, Sex	Race, Sex
Favorable Outcome	Good Credit Risk	Did Not Recidivate	Income > \$50K
Unfavorable Outcome	Bad Credit Risk	Recidivated	Income ≤ \$50K
Most Privileged Group	(Male, Adult)	(Caucasian, Female)	(White, Male)
Most Unprivileged Group	(Female, Young)	(African-American, Male)	(Black, Female)
Max IR	2.767	1.842	0.479

Table 5.1: Summary of characteristics for the German Credit, COMPAS Recidivism, and Adult datasets. The table includes the number of rows, number of features, sensitive attributes, definitions of favorable and unfavorable outcomes, and the most privileged and unprivileged groups for each dataset. It also lists the maximum Imbalance Ratio observed.

5.2 Evaluation Metrics and Fairness Measures

To recall the steps of the proposed framework, once a dataset has been selected, it undergoes a Data Preparation phase, as detailed in Section 4.1. Subsequently, for each unprivileged group, border points are identified using the DBSCAN algorithm and then oversampled using SMOTE, as detailed in Sections 4.2 and 4.3, respectively, until the Imbalance Ratio (IR) of the most privileged group is achieved.
The new oversampled training dataset is utilized to train a classification model. This model is subsequently evaluated on the test dataset, where the predicted labels are compared to the ground truth labels to assess the classifier's performance.
The evaluation metrics and fairness measures employed to assess the quality of the predictions were introduced in Section 3.2 and are briefly summarized here to formalize their application in this evaluation part.

The evaluation metrics used in this work include:

- **Accuracy**: This metric quantifies the proportion of correctly predicted labels out of the total predictions, regardless of the subgroup. It simply compares all predicted labels to the original labels, providing a score based on the number of correct predictions. However, accuracy can be misleading for highly imbalanced datasets, as it relies heavily on the true positive rate.

- **Balanced Accuracy**: Unlike plain accuracy, balanced accuracy considers both true positive and true negative rates, making it more suitable for imbalanced datasets. It averages the recall obtained on each class.

- **F1-Score**: The F1-score is the harmonic mean of precision and recall, providing a balance between the two metrics. It is particularly useful for imbalanced datasets as it considers both false positives and false negatives.

The group fairness measures introduced in Section 3.2 were based on the consideration of a single binary sensitive attribute, resulting in one privileged group and one unprivileged group. These measures compared the outcome probabilities between these two groups. However, in this work, we are considering multiple binary sensitive attributes, which lead to the formation of multiple groups rather than just two. By computing the Imbalance Ratio (IR), it is possible to identify the most privileged group among these multiple groups. Consequently, we can apply the group fairness measures to various combinations of these groups, thereby extending the analysis beyond the binary privileged and unprivileged classification.
The group fairness measures used in this work are:

- **Disparate Impact Ratio (DI Ratio)**: This measure computes the ratio of the rate of favorable outcomes for the unprivileged group to that of the privileged group. A DI Ratio close to 1 is preferable, indicating fairness.

- **Average Odds Difference (AOD)**: AOD is the arithmetic mean of the differences in true positive rate (TPR) and false positive rate (FPR) between the privileged and

unprivileged groups. It encapsulates the concept of Equalized Odds, suggesting that different groups should have similar error rates.

- **Equal Opportunity Difference (EOD)**: This measure focuses on equalizing the true positive rate among groups. It is a relaxation of AOD, only considering the true positive rates.

Lastly, the individual fairness measure Consistency is recalled as:

- **Consistency**: This individual fairness measure evaluates whether similar individuals receive similar predictions, considering all labels (not just those of specified groups).

5.3 Experiments

This section presents the experiments conducted to evaluate the proposed DBSCAN-SMOTE method.
First, the effectiveness of using the Imbalance Ratio as the oversampling target is verified, showing that it yields better results compared to the widely-used approach of oversampling to achieve equal sample sizes for each group. This justifies the choice of the IR-based method. Next, the proposed framework is compared with two other frameworks developed during the thesis writing process, demonstrating why DBSCAN-SMOTE was selected over the alternatives. Following this, the proposed framework is benchmarked against existing methods in the literature, showing competitive results in terms of fairness measures and evaluation metrics. Finally, the DBSCAN-SMOTE framework's ability to mitigate bias across all groups within the dataset is illustrated.
The results reported in the tables are the averages of 10 runs. Each run splits the original dataset into 70% for training and 30% for testing, using the same test set for each method. The datasets used for these experiments are the German, COMPAS, and Adult datasets, each split using stratification based on the subgroups within the datasets.
The group fairness measures—DI Ratio, AOD, and EOD—are computed by comparing the most privileged group against the most unprivileged group, except in the final experiment, where each pairwise comparison is evaluated.

5.3.1 Computing the Number of Examples for Oversampling

To validate the effectiveness of using the highest Imbalance Ratio (IR_{max}) as the oversampling target, this experiment compares the results obtained with the proposed framework based on the IR against those obtained by oversampling to equalize the sample sizes for positive and negative subgroups. This comparison aims to determine whether targeting the IR leads to better results than simply equalizing the sample sizes across subgroups.

In more detail, the DBSCAN-SMOTE method is applied in both scenarios. For each subgroup, border points are selected as explained in the previous chapter, and new synthetic

points are generated following the previously outlined steps. The key difference between the two cases lies in the oversampling target. In the IR-based approach, the most privileged group, identified as the one with the highest IR, serves as the oversampling target for the other groups. Conversely, in the equal size-based approach, the most numerous group is identified, and all other groups are oversampled to match this group's size. Specifically, each positive-labeled subgroup (PU) is adjusted to have the same number of samples as the positive subgroup of the most numerous group, and each negative-labeled subgroup (NU) is adjusted to have the same number of samples as the negative subgroup of the most numerous group.

In the German Credit dataset, the group with the highest Imbalance Ratio coincides with the most numerous group. Therefore, in both oversampling scenarios, every group will achieve a ratio of positive to negative samples approximately equal to 2.767, but with different sample sizes in the two different experiments.
The sensitive attributes and the label are encoded such that the favorable values are represented as 1 and the unfavorable values as 0. Specifically, Male = 1 and Female = 0, Adult = 1 and Young = 0, and Good Credit Risk = 1 and Bad Credit Risk = 0. For example, (Male, Young, Good Credit Risk) is encoded as (1, 0, 1).
Table 5.2 summarizes the counts and the ratios of positive to negative instances for each group in the original training dataset (first column), in the training dataset after oversampling to equalize sample sizes (second column), and in the training dataset after oversampling based on the IR (third column).

Group	**Orig. Count**	**Ratio**	**Equal Size Count**	**Ratio**	**IR Count**	**Ratio**
(1,1,1)	321	2.767	321	2.767	321	2.767
(1,1,0)	116		116		116	
(1,0,1)	28	1.556	321	2.767	50	2.778
(1,0,0)	18		116		18	
(0,1,1)	107	2.098	321	2.767	142	2.784
(0,1,0)	51		116		51	
(0,0,1)	34	1.360	321	2.767	70	2.800
(0,0,0)	25		116		25	

Table 5.2: Comparison of counts and positive-to-negative instance ratios for each group in the German dataset. The table presents the original training dataset (first column), the dataset after oversampling to equalize sample sizes (second column), and the dataset after oversampling based on the Imbalance Ratio (IR) (third column).

The three datasets—the original dataset, the dataset oversampled to equalize sample sizes, and the dataset oversampled based on the IR—were each used to train a Logistic Regression model. These three trained classifiers were then tested on the same test set, and the results are presented in Table 5.3.

From these results, it can be noticed that the Disparate Impact Ratio (DI Ratio) and the Equal Opportunity Difference (EOD) are more favorable for the IR-based method compared to the equal size-based method. Although the Average Odds Difference (AOD) is slightly better for the equal size-based model, all the values are very close to each other. The values for Accuracy, Balanced Accuracy, and F1 Score are lower than those of the original dataset, as expected, since the classifier is being adjusted to improve fairness, which can lead to some misclassifications. However, the metrics for the IR-based method are higher than those for the equal size-based method, indicating that the IR-based method has a better fairness-accuracy trade-off. This means that the IR-based method improves fairness without losing too much accuracy, even compared to the original dataset.

Approach	**DI Ratio**	**AOD**	**EOD**	**Consi.**	**Acc.**	**Bal. Acc.**	**F1 Score**
Orig.	0.695	-0.202	-0.136	0.857	**0.771**	**0.693**	**0.844**
Equal Size	0.962	**0.032**	-0.028	0.843	0.729	0.654	0.813
IR	**0.998**	0.065	**-0.007**	**0.873**	0.757	0.660	0.839

Table 5.3: Comparison of Logistic Regression models trained on the original dataset, the dataset oversampled to equalize sample sizes, and the dataset oversampled based on Imbalance Ratio (IR) for the German dataset.

In the Adult dataset, the group with the highest imbalance ratio coincides with the most numerous group. Consequently, both oversampling strategies result in groups having the same ratio of positive samples to negative samples, but with different sample sizes.
The sensitive attributes and the label are encoded with the same convention as before, where 1 represents a favorable outcome and 0 represents an unfavorable outcome. Specifically, White = 1 and Black = 0, Male = 1 and Female = 0, Income $>$ \$50K = 1 and Income $\leq$ \$50K = 0.
Table 5.4 reports the counts and ratios for the original dataset and the oversampled datasets. The three datasets are then used to train three Logistic Regression classifiers, which are subsequently tested on the same test dataset. The results are reported in Table 5.5.
The results show that the fairness measures are better for the IR-based method compared to the equal size-based method. Accuracy is nearly identical between the two oversampling methods, while Balanced Accuracy and F1 Score are slightly better for the equal size-based method.

In the COMPAS dataset, the group with the highest Imbalance Ratio does not coincide with the most numerous group. In fact, in this dataset, the most numerous group is the one considered the most unprivileged by the IR. Therefore, when oversampling using the IR-based method, the groups will ultimately have the IR of the most privileged group. In contrast, when oversampling using the equal size-based method, all the groups will have the same number of samples as the most unprivileged group, maintaining its ratio of positive to negative samples.

Group	Orig. Count	Ratio	Equal Size Count	Ratio	IR Count	Ratio
(1,1,1)	6126	0.479	6126	0.479	6126	0.479
(1,1,0)	12787		12787		12787	
(1,0,1)	1019	0.140	6126	0.479	3497	0.479
(1,0,0)	7299		12787		7299	
(0,1,1)	286	0.235	6126	0.479	583	0.480
(0,1,0)	1215		12787		1215	
(0,0,1)	88	0.064	6126	0.479	657	0.479
(0,0,0)	1371		12787		1371	

Table 5.4: Comparison of counts and positive-to-negative instance ratios for each group in the Adult dataset. The table presents the original training dataset (first column), the dataset after oversampling to equalize sample sizes (second column), and the dataset after oversampling based on the Imbalance Ratio (IR) (third column).

Approach	DI Ratio	AOD	EOD	Consi.	Acc.	Bal. Acc.	F1 Score
Orig.	0.081	-0.119	-0.162	**0.952**	**0.794**	**0.644**	**0.455**
Equal Size	0.387	-0.056	-0.065	0.927	0.773	0.641	0.454
IR	**0.430**	**-0.029**	**-0.029**	0.933	0.778	0.637	0.444

Table 5.5: Comparison of Logistic Regression models trained on the original dataset, the dataset oversampled to equalize sample sizes, and the dataset oversampled based on Imbalance Ratio (IR) for the Adult dataset.

The sensitive attributes and the label are encoded such that the favorable values are represented as 1 and the unfavorable values as 0. Specifically, Caucasian = 1 and African-American = 0, Female = 1 and Male = 0, did not recidivate = 1 and did recidivate = 0. For example, (Caucasian, Male, did not recidivate) is encoded as (1, 0, 1).
Table 5.6 summarizes the counts and the ratios of positive to negative instances for each group. Note how the ratios differ between the equal size-based oversampling and the IR-based oversampling, reflecting the different group choices for defining the oversampling target in each case.
A Logistic Regression model was trained using each of the three datasets: the original dataset, the dataset oversampled to equalize sample sizes, and the dataset oversampled based on the IR. These trained classifiers were subsequently tested on the same test set, with the results presented in Table 5.7.
From these results, it is evident that the fairness measures are significantly better for the IR-based oversampling method compared to the equal size-based method. Although both methods result in groups having equal imbalance ratios, there are key differences that justify the superior fairness of the IR-based method.
Firstly, the IR-based method focuses on aligning all groups to the imbalance ratio of the most privileged group, which inherently has a more favorable distribution of positive to

Group	Orig. Count	Ratio	Equal Size Count	Ratio	IR Count	Ratio
(1,1,1)	256	1.842	969	0.838	256	1.842
(1,1,0)	139		1156		139	
(1,0,1)	782	1.464	969	0.838	984	1.843
(1,0,0)	534		1156		534	
(0,1,1)	283	1.655	969	0.838	315	1.842
(0,1,0)	171		1156		171	
(0,0,1)	969	0.838	969	0.838	2130	1.843
(0,0,0)	1156		1156		1156	

Table 5.6: Comparison of counts and positive-to-negative instance ratios for each group in the COMPAS dataset. The table presents the original training dataset (first column), the dataset after oversampling to equalize sample sizes (second column), and the dataset after oversampling based on the Imbalance Ratio (IR) (third column).

Approach	DI Ratio	AOD	EOD	Consi.	Acc.	Bal. Acc.	F1 Score
Orig.	0.555	-0.389	-0.295	0.883	**0.655**	**0.643**	**0.718**
Equal Size	0.620	-0.271	-0.204	0.861	0.647	0.640	0.692
IR	**0.963**	**-0.028**	**-0.018**	**0.975**	0.558	0.527	0.704

Table 5.7: Comparison of Logistic Regression models trained on the original dataset, the dataset oversampled to equalize sample sizes, and the dataset oversampled based on Imbalance Ratio (IR) for the COMPAS dataset.

negative samples. This approach ensures that acceptance rates across different groups are more similar to those of the privileged group, directly addressing disparities in treatment between privileged and unprivileged groups.
Additionally, the high amount of oversampling required for the equal size-based method can introduce noise and reduce the classifier's ability to generalize, potentially leading to higher false positive or false negative rates. In contrast, the IR-based method's targeted approach of adjusting the imbalance ratio to match the most privileged group ensures that the classifier learns to treat all groups more equitably.
In conclusion, from this experiment, it can be stated that the IR-based method is preferred because it requires less oversampling while effectively equalizing the acceptance rates across all groups to match those of the most privileged group. This approach improves fairness more efficiently and results in superior fairness metrics, making it a more effective strategy for mitigating bias.

5.3.2 Finding Border Points for Oversampling

During the development of the proposed DBSCAN-SMOTE framework, two alternative methods based on the Imbalance Ratio (IR) were created and tested. Each of these methods aimed to achieve the same IR as the most privileged group by focusing on oversampling in challenging areas, such as border points or regions with lower density. This subsection compares the results obtained from the three proposed bias mitigation methods.

The first alternative method is inspired by the labeling technique introduced by Napierala et al. [54]. In their work, samples are labeled based on the number of neighbors from the same group. Specifically, considering the 5 nearest neighbors, a sample is labeled as *Safe* if 4 or 5 neighbors belong to the same group, *Borderline* if 2 or 3 neighbors belong to the same group, *Rare* if only 1 neighbor belongs to the same group and this neighbor has either 0 or 1 neighbors from the same group (otherwise it is counted as Borderline), and *Outlier* if it has no neighbors from the same group.
In this alternative, Gower's distance is used to compute the 5 nearest neighbors for each sample, which are then labeled according to Napierala et al.'s method.
The most privileged group is identified by its IR, and other groups are oversampled to match this IR. For each subgroup requiring oversampling, a randomly selected Borderline sample and one of its nearest neighbors are used to generate a new synthetic sample, following the DBSCAN-SMOTE method detailed in Section 4.3. Ultimately, each group achieves the same IR as the most privileged group, with new synthetic points created at the borders of each subgroup.

The second alternative method focuses on oversampling in less dense areas to achieve the same IR for each group, equal to that of the most privileged group. First, the dataset is clustered using k-means clustering [50], excluding the sensitive attributes and the label from the clustering process to find the natural groupings.
For each group, the required number of synthetic samples is calculated to ensure that each group reaches the same IR. New synthetic samples are generated based on the density within each cluster: the average distance between samples in each cluster is computed for each subgroup, and more samples are added in clusters with greater average distances, thus targeting areas where the data are less dense. This ensures that new synthetic samples are created in regions where the existing samples are more sparsely distributed.

The results obtained with the three oversampled datasets are compared, along with the results obtained using the original dataset for the training. Table 5.8 summarizes the results obtained using the German dataset, Table 5.9 summarizes the results obtained using the COMPAS dataset, while Table 5.10 summarizes the results obtained using the Adult dataset.
As shown in the tables, the selected framework (DBSCAN-SMOTE) generally performs better in terms of both evaluation metrics and fairness measures. Specifically, DBSCAN-SMOTE shows improved DI Ratio, AOD, EOD, and Consistency across the datasets compared to the other methods. It also maintains a competitive accuracy, balanced accuracy, and F1 score, indicating that it effectively balances fairness and classification

performance.
These results highlight the advantages of DBSCAN-SMOTE, which motivated its selection as the preferred framework for bias mitigation in this research. The DBSCAN-SMOTE method not only equalizes the IR across groups but also strategically creates synthetic samples in challenging border areas, resulting in fairer and more accurate outcomes.

Method	DI Ratio	AOD	EOD	Consi.	Acc.	Bal. Acc.	F1 Score
Orig.	0.675	-0.234	-0.110	0.863	**0.764**	**0.681**	**0.840**
k-means	1.014	0.063	0.050	0.882	0.751	0.645	0.837
Labeling	1.033	0.077	0.054	**0.889**	0.745	0.631	0.834
DBSCAN-SMOTE	**1.005**	**0.053**	**0.037**	0.882	0.748	0.640	0.835

Table 5.8: Performance comparison of Logistic Regression models trained using the original dataset, k-means-based oversampling, labeling-based oversampling, and DBSCAN-SMOTE for the German dataset.

Method	DI Ratio	AOD	EOD	Consi.	Acc.	Bal. Acc.	F1 Score
Orig.	0.551	-0.390	-0.300	0.880	**0.652**	**0.640**	**0.714**
k-means	0.886	-0.077	-0.055	0.928	0.604	0.581	0.713
Labeling	0.882	-0.081	-0.059	0.928	0.604	0.581	0.713
DBSCAN-SMOTE	**0.969**	**-0.023**	**-0.016**	**0.977**	0.554	0.523	0.702

Table 5.9: Performance comparison of Logistic Regression models trained using the original dataset, k-means based oversampling, labeling-based oversampling, and DBSCAN-SMOTE for the COMPAS dataset.

Method	DI Ratio	AOD	EOD	Consi.	Acc.	Bal. Acc.	F1 Score
Orig.	0.089	-0.111	-0.147	**0.953**	**0.794**	**0.640**	**0.457**
k-means	**0.543**	0.027	0.075	0.929	0.769	0.630	0.432
Labeling	0.537	0.006	0.025	0.933	0.775	0.638	0.447
DBSCAN-SMOTE	0.468	**-0.005**	**0.017**	0.933	0.778	0.639	0.448

Table 5.10: Performance comparison of Logistic Regression models trained using the original dataset, k-means based oversampling, labeling-based oversampling, and DBSCAN-SMOTE for the Adult dataset.

5.3.3 Comparison with Existing Bias Mitigation Algorithms

To evaluate the effectiveness of the proposed DBSCAN-SMOTE framework, its performance is benchmarked against several established bias mitigation methods from the literature. The selected methods for comparison include Fair-SMOTE, Reweighing, and Remedy, each employing distinct approaches to addressing bias in imbalanced datasets. This

subsection provides a comparative analysis of the results obtained using the DBSCAN-SMOTE framework and these existing techniques.

Fair-SMOTE [12] is a pre-processing bias mitigation technique designed to address bias arising from imbalanced datasets and improper data labeling. The process begins by dividing the training dataset into subgroups based on sensitive attributes and label. It then identifies the largest subgroup and uses SMOTE to oversample the other subgroups, ensuring that each subgroup ultimately has an equal amount of data, matching the size of the largest subgroup. Fair-SMOTE tackles biased data labels through a technique called "situation testing," which involves flipping the values of sensitive attributes for each sample and checking if the prediction changes. By the end of this process, the oversampled training dataset has the same number of samples in each subgroup, ensuring each group has an equal number of positive and negative labels (i.e., an imbalance ratio of 1 for each group). However, Fair-SMOTE requires substantial oversampling in cases of highly imbalanced datasets, potentially leading to a significant increase in synthetic data and not accounting for the original data distribution, which can result in an artificial dataset.

Reweighing [10] is a pre-processing bias mitigation technique that adjusts the importance of each instance in the dataset by assigning weights, rather than generating synthetic data. This method assigns higher weights to positive instances from unprivileged groups and negative instances from privileged groups. The process starts by defining privileged and unprivileged groups based on sensitive attributes. Each instance is then assigned a weight based on its group membership and label, ensuring that the learning algorithm appropriately considers the less represented groups. By incorporating these weights during model training, Reweighing mitigates bias without altering the original data distribution. This approach avoids the drawbacks of oversampling, such as the creation of numerous synthetic samples, and preserves the integrity of the original dataset.

Remedy, a recently developed technique introduced by Lin et al. and published in the 2024 IEEE 40th International Conference on Data Engineering (ICDE) [48], is considered for comparison because it closely aligns with the scope of the proposed DBSCAN-SMOTE framework. Remedy views the dataset as a hierarchical structure based on sensitive attributes and introduces the concept of an "Implicit Biased Set (IBS)", defined as regions where the imbalance ratio is significantly lower than that of neighboring regions. The neighborhood for these regions is determined using a specified distance metric. Remedy aims to mitigate bias within the IBS, employing strategies such as oversampling. In this work, Remedy is considered with bias mitigation performed via oversampling: after identifying the IBS, oversampling is applied based on the imbalance ratio to balance the representation within these biased regions.

The results of the comparison between the proposed DBSCAN-SMOTE framework and the bias mitigation methods from the literature, using the three datasets and logistic regression as the classification model, are reported in Table 5.11. This analysis demonstrates that the DBSCAN-SMOTE method achieves competitive outcomes in terms of fairness measures compared to other established methods, while maintaining reasonable

Dataset	Method	DI Ratio	AOD	EOD	Consi.	Acc.	Bal. Acc.	F1 Score
German	Orig.	0.736	-0.178	-0.087	0.861	**0.762**	**0.677**	**0.840**
	Fair-SMOTE	1.109	0.148	0.063	0.831	0.717	0.658	0.799
	Reweighing	0.946	**0.017**	**0.012**	0.864	0.756	0.672	0.835
	Remedy	1.038	0.100	0.062	0.861	0.755	0.674	0.834
	DBSCAN-SMOTE	**1.035**	0.082	0.053	**0.884**	0.751	0.641	0.837
COMPAS	Orig.	0.544	-0.400	-0.312	0.876	**0.651**	**0.639**	0.712
	Fair-SMOTE	0.708	-0.234	-0.182	0.900	0.634	0.618	0.713
	Reweighing	0.812	-0.125	-0.096	0.898	0.631	0.614	**0.715**
	Remedy	1.401	0.255	0.239	0.866	0.609	0.594	0.689
	DBSCAN-SMOTE	**0.962**	**-0.032**	**-0.022**	**0.975**	0.553	0.521	0.701
Adult	Orig.	0.086	-0.113	-0.151	0.952	**0.795**	**0.636**	**0.438**
	Fair-SMOTE	0.159	-0.076	-0.064	0.940	0.778	0.625	0.418
	Reweighing	0.244	-0.013	**-0.001**	0.959	0.794	0.623	0.402
	Remedy	0.292	0.018	0.039	**0.974**	0.793	0.592	0.317
	DBSCAN-SMOTE	**0.454**	**0.001**	0.029	0.934	0.780	0.634	0.437

Table 5.11: Comparison of Logistic Regression results using Fair-SMOTE, Reweighting, Remedy and DBSCAN-SMOTE methods for the German, COMPAS, and Adult datasets.

performance levels.
Specifically, the Disparate Impact (DI) Ratio values obtained using the DBSCAN-SMOTE framework are consistently the closest to one across all datasets evaluated. This indicates a strong performance in terms of fairness. For the German dataset, although the DI Ratio values obtained with all four bias mitigation methods are very close to one, DBSCAN-SMOTE still achieves the best value. For the Adult dataset, none of the proposed bias mitigation techniques surpass the 80% threshold necessary to be considered fair (DI Ratio > 0.80). However, the DI Ratio value achieved by DBSCAN-SMOTE is still the highest and significantly improved compared to the original dataset's near-zero value, demonstrating the positive impact of the oversampling strategy.
In addition to DI Ratio, DBSCAN-SMOTE also shows favorable outcomes in other fairness measures, such as Average Odds Difference (AOD), Equal Opportunity Difference (EOD), and Consistency. These improvements highlight the method's effectiveness in increasing fairness.
However, these gains in fairness often result in a slight decrease in performance metrics, such as Accuracy, Balanced Accuracy, and F1 Score, when compared to the original dataset. This performance trade-off occurs because the oversampling process forces the classifier to adjust the decision boundary in favor of unprivileged groups, which can lead to some misclassifications. Despite this, the decrease in performance metrics is generally within acceptable limits.
Overall, the DBSCAN-SMOTE framework demonstrates its capability to mitigate representation bias effectively, achieving comparable results to other bias mitigation techniques, while the performance loss remains within a tolerable range.

5.3.4 Ensuring Fairness Among the Different Subgroups

In previous experiments, the group fairness measures—DI Ratio, AOD, and EOD—were computed by comparing the most privileged group against the most unprivileged group, defined by the original Imbalance Ratio (IR). While this approach highlights fairness improvements in the most extreme cases, it is also essential to evaluate fairness across all groups within the datasets.
In this experiment, the DI Ratio, AOD, and EOD are calculated for every possible pairwise comparison between groups. Each dataset comprises four distinct groups, resulting in a total of six pairwise comparisons per dataset.
This comprehensive evaluation assesses the effectiveness of the proposed framework in mitigating bias across all group interactions, not just between the most and least privileged groups.

The pairwise comparison results obtained with the DBSCAN-SMOTE framework using the German dataset are presented in Table 5.12. Each group is denoted in the format (Gender, Age), where Male = 1 and Female = 0, Adult = 1, and Young = 0. In each comparison, the first group listed is more privileged than the second group, meaning it has a higher Imbalance Ratio and is considered the privileged group for the computation of the fairness measures.

Comparison	**DI Ratio**	**AOD**	**EOD**
(1, 1) vs (0, 0)	1.035	0.082	0.053
(0, 1) vs (0, 0)	1.139	0.147	0.089
(0, 1) vs (1, 0)	1.156	0.146	0.076
(1, 1) vs (0, 1)	0.919	-0.064	-0.036
(1, 1) vs (1, 0)	1.050	0.082	0.040
(1, 0) vs (0, 0)	0.993	0.001	0.013

Table 5.12: Pairwise comparison results using the DBSCAN-SMOTE framework for the German dataset.

For the COMPAS dataset, the pairwise comparison results obtained with the proposed framework are presented in Table 5.13. As before, the first group listed is more privileged than the second. The groups are denoted in the format (Race, Sex), where Caucasian = 1 and African-American = 0, Female = 1, and Male = 0.
The results for the pairwise comparison using the Adult dataset are presented in Table 5.14. In each comparison, the first group listed is more privileged than the second. The groups are denoted in the format (Race, Sex), where White = 1 and Black = 0, Male = 1, and Female = 0.
Tables 5.12, 5.13 and 5.14 demonstrate that the proposed framework effectively mitigates bias across different groups within each dataset. This is achieved by equalizing the Imbalance Ratio (IR) across all groups, ensuring that each group has the same rate of acceptance regardless of its sensitive attributes.

Comparison	DI Ratio	AOD	EOD
(1, 1) vs (0, 0)	0.962	-0.032	-0.022
(0, 1) vs (0, 0)	0.962	-0.032	-0.025
(0, 1) vs (1, 0)	0.994	-0.005	-0.002
(1, 1) vs (0, 1)	1.000	-0.000	0.003
(1, 1) vs (1, 0)	0.994	-0.006	0.000
(1, 0) vs (0, 0)	0.967	-0.026	-0.023

Table 5.13: Pairwise comparison results using the DBSCAN-SMOTE framework for the COMPAS dataset.

Comparison	DI Ratio	AOD	EOD
(1, 1) vs (0, 0)	0.454	0.000	0.029
(0, 1) vs (0, 0)	0.713	0.013	0.020
(0, 1) vs (1, 0)	0.951	-0.009	-0.040
(1, 0) vs (0, 0)	0.753	0.023	0.060
(1, 1) vs (0, 1)	0.645	-0.013	0.009
(1, 1) vs (1, 0)	0.602	-0.022	-0.032

Table 5.14: Pairwise comparison results using the DBSCAN-SMOTE framework for the Adult dataset.

Specifically, the DI Ratio values are all close to one for the German and COMPAS datasets and show significant improvement for the Adult dataset, even exceeding the 80% threshold in one comparison. Additionally, the AOD and EOD values are close to zero in every comparison across the three datasets, highlighting the effectiveness of the proposed framework in promoting fairness.

Chapter 6

Conclusions

6.1 Summary

This work proposed a framework to mitigate representation bias, which arises when certain groups in a dataset are imbalanced or skewed compared to others. A key assumption throughout this work is that the original labels in the dataset are unbiased; hence, the fairness of the assigned labels for individuals is not questioned. Focusing solely on representation bias, the objective was to create new synthetic instances for underrepresented groups so that each group achieves the same Imbalance Ratio, thereby ensuring an equal acceptance rate.
Unlike previous bias mitigation techniques, the proposed framework considers all combinations of sensitive attributes to define the groups within the dataset. Each group is further divided into subgroups based on their binary labels (positive or negative). By oversampling each group to reach the maximum Imbalance Ratio, the decision boundary of the classifier is shifted in favor of underrepresented groups, ensuring that each group has the same likelihood of receiving a positive prediction.

The framework comprises several stages. First, a data preparation process standardizes the data, followed by a 70%-30% split into training and testing sets. Next, each subgroup with positive labels undergoes the DBSCAN algorithm to identify core, border, and noise points within a single cluster. Oversampling is then performed using the SMOTE-NC method, interpolating between a border point and its neighboring border/core point. By the end of the oversampling process, each group achieves an Imbalance Ratio equal to the highest one in the dataset.
The proposed framework was evaluated by comparing it to itself with a different oversampling target and to other bias mitigation algorithms. The results demonstrate that this model performed better overall. By considering other bias mitigation techniques from the literature, the effectiveness of the method was proven, achieving competitive results in terms of fairness measures with minimal loss in evaluation metrics. Additionally, the efficacy of oversampling each group was shown by computing fairness measures in pairwise comparisons, demonstrating that the framework mitigates bias across all groups in the

dataset, not just the most extreme cases.

6.2 Limitations

Despite its strengths, this work has several limitations. One major assumption made throughout this thesis is that the original labels in the dataset are unbiased. Consequently, these labels were assumed to be accurate and were not altered. This is a significant assumption because it overlooks the fact that labeled datasets often originate from historical data where decisions were made by individuals. There is no guarantee that these labels were fair, as they may reflect human bias.
Another limitation is the requirement for the dataset to have binary sensitive attributes. This restricts the framework's applicability, as it may not handle multi-valued or more complex protected attributes effectively. Addressing this limitation would be essential for broader applicability in real-world scenarios where attributes are not always binary.
Additionally, the proposed framework heavily relies on the DBSCAN clustering algorithm, which depends on the chosen distance metric and two critical parameters: epsilon (eps) and minimum points (MinPts). Changes in these parameters or the distance metric can lead to different results, potentially affecting the robustness and consistency of the framework. The distance metric also influences the computation of the five nearest neighbors in the oversampling step, which is crucial for generating synthetic samples.

6.3 Possible Future Work Directions

Starting from on the observed limitations, several directions for future work can be proposed. One significant area for future research is the analysis of the original labels to assess whether they present bias. This approach would involve evaluating and correcting bias arising from historical data collections, ensuring the training data itself is fair.
Another important direction is to consider multi-valued sensitive attributes instead of relying solely on binary sensitive attributes. Developing techniques to handle more complex sensitive attributes would enhance the framework's applicability and effectiveness in real-world scenarios.
Additionally, the dependency on the DBSCAN clustering algorithm and the choice of distance metric represent areas for potential improvement. Exploring more sophisticated distance metrics, such as the Heterogeneous Value Difference Metric (HVDM), could be beneficial. HVDM, for instance, computes distances between categorical attributes in a more nuanced manner than simply considering them as 1 if equal and 0 otherwise. This metric could improve the quality of the generated clusters and the computation of the five nearest neighbors for the oversampling step.

Bibliography

[1] Charter of fundamental rights of the european union. *Official Journal of the European Union*, C 303, 2007. Article 21 - Non-discrimination.

[2] P. Adler, C. Falk, S. A. Friedler, G. Rybeck, C. Scheidegger, B. Smith, and S. Venkatasubramanian. Auditing black-box models for indirect influence. *Knowledge and Information Systems*, 54:95–122, 2018.

[3] AI Fairness 360. Compasdataset. `https://aif360.readthedocs.io/en/latest/modules/generated/aif360.datasets.CompasDataset.html`. Accessed: 2024-06-27.

[4] R. Baeza-Yates. Bias on the web. *Communications of the ACM*, 61(6):54–61, 2018. ISSN 0001-0782. doi: 10.1145/3209581.

[5] S. Barocas, M. Hardt, and A. Narayanan. *Fairness and Machine Learning: Limitations and Opportunities*. MIT Press, 2023.

[6] J. Baumann, A. Castelnovo, R. Crupi, N. Inverardi, and D. Regoli. Bias on demand: A modelling framework that generates synthetic data with bias. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '23, pages 1002–1013, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701924. doi: 10.1145/3593013.3594058.

[7] J. Baumann, A. Castelnovo, R. Crupi, N. Inverardi, and D. Regoli. Supplementary material for the paper: Bias on demand: A modelling framework that generates synthetic data with bias. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency (FAccT '23)*. Association for Computing Machinery, 2023. doi: 10.1145/3593013.3594058.

[8] B. Becker and R. Kohavi. Adult. UCI Machine Learning Repository, 1996. DOI: https://doi.org/10.24432/C5XW20.

[9] S. Buijsman. Navigating fairness measures and trade-offs. *AI and Ethics (Online)*, 2023. ISSN 2730-5953.

[10] T. Calders, F. Kamiran, and M. Pechenizkiy. Building classifiers with independency constraints. In *2009 IEEE International Conference on Data Mining Workshops*, pages 13–18. IEEE, 2009. ISBN 1424453844.

[11] A. Castelnovo, R. Crupi, G. Greco, D. Regoli, I. G. Penco, and A. C. Cosentini. A clarification of the nuances in the fairness metrics landscape. *Scientific Reports*, 12 (1):4209, 2022. ISSN 2045-2322.

[12] J. Chakraborty, S. Majumder, and T. Menzies. Bias in machine learning software: Why? how? what to do? In *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ESEC/FSE 2021, page 429–440, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450385626. doi: 10.1145/3468264.3468537.

[13] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16: 321–357, June 2002. ISSN 1076-9757. doi: 10.1613/jair.953.

[14] A. Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big Data*, 5(2):153–163, 2017. ISSN 2167-6461.

[15] A. Chouldechova and A. Roth. The frontiers of fairness in machine learning. *arXiv.org*, 2018. ISSN 2331-8422.

[16] S. Corbett-Davies, E. Pierson, A. Feller, S. Goel, and A. Huq. Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 797–806, New York, NY, USA, 2017. ACM. ISBN 1450348874.

[17] D. R. Cox. The regression analysis of binary sequences. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 20(2):215–232, 1958.

[18] J. Dastin. Insight - amazon scraps secret ai recruiting tool that showed bias against women. *Reuters*. URL `https://www.reuters.com/article/world/insight-amazon-scraps-secret-ai-recruiting-tool-that-showed-bias-against-women-idU`

[19] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

[20] J. Dunkelau and M. Leuschel. Fairness-aware machine learning: An extensive overview. *An Extensive Overview*, pages 1–60, 2019.

[21] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, pages 214–226, New York, NY, USA, 2012. ACM. ISBN 1450311156.

[22] C. Dwork, N. Immorlica, A. T. Kalai, and M. Leiserson. Decoupled classifiers for fair and efficient machine learning. 2017.

[23] T. U. EEOC. *Uniform guidelines on employee selection procedures*. 03 1979.

[24] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, pages 226–231. AAAI Press, 1996.

[25] T. Fawcett. Introduction to roc analysis. *Pattern Recognition Letters*, 27:861–874, 2006. doi: 10.1016/j.patrec.2005.10.010.

[26] R. Febrianti, Y. Widyaningsih, and S. Soemartojo. The parameter estimation of logistic regression with maximum likelihood method and score function modification. In *Journal of physics: Conference series*, volume 1725, page 012014. IOP Publishing, 2021. doi: 10.1088/1742-6596/1725/1/012014.

[27] M. Feldman, S. A. Friedler, J. Moeller, C. Scheidegger, and S. Venkatasubramanian. Certifying and removing disparate impact. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD'15, page 259–268, 2015.

[28] B. Friedman and H. Nissenbaum. Bias in computer systems. *ACM Transactions on Information Systems*, 14(3):330–347, 1996. ISSN 1046-8188. doi: 10.1145/230538.230561.

[29] P. Gajane and M. Pechenizkiy. On formalizing fairness in prediction with machine learning. 2018.

[30] N. Garg, L. Schiebinger, D. Jurafsky, and J. Zou. Word embeddings quantify 100 years of gender and ethnic stereotypes. *Proceedings of the National Academy of Sciences*, 115(16), April 2018. ISSN 1091-6490. doi: 10.1073/pnas.1720347115.

[31] J. C. Gower. A general coefficient of similarity and some of its properties. *Biometrics*, 27(4):857–871, 1971. ISSN 0006-341X, 1541-0420.

[32] M. Grandini, E. Bagli, and G. Visani. Metrics for multi-class classification: an overview. 2020.

[33] H. Han, W.-Y. Wang, and B.-H. Mao. Borderline-smote: A new over-sampling method in imbalanced data sets learning. In *Advances in Intelligent Computing*, pages 878–887. Springer Berlin Heidelberg, 2005. ISBN 3540282262.

[34] M. Hardt, E. Price, and N. Srebro. Equality of opportunity in supervised learning. *arXiv.org*, 2016. ISSN 2331-8422.

[35] H. He, Y. Bai, E. A. Garcia, and S. Li. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, volume 10, pages 1322–1328. IEEE, 2008. ISBN 1424418208.

[36] H. Hofmann. Statlog (German Credit Data). UCI Machine Learning Repository, 1994. DOI: https://doi.org/10.24432/C5NC77.

[37] M. Hort, Z. Chen, J. M. Zhang, M. Harman, and F. Sarro. Bias mitigation for machine learning classifiers: A comprehensive survey. *ACM Journal on Responsible Computing*, 1(2):1–52, 2024.

[38] M. Joseph, M. Kearns, J. Morgenstern, and A. Roth. Fairness in learning: Classic and contextual bandits. *arXiv.org*, 2016. ISSN 2331-8422.

[39] F. Kamiran and T. Calders. Classification with no discrimination by preferential sampling. In *Proc. 19th Machine Learning Conf. Belgium and The Netherlands*, volume 1. Citeseer, 2010.

[40] F. Kamiran and T. Calders. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, 33(1):1–33, 2012. ISSN 0219-1377.

[41] F. Kamiran and I. Žliobaitė. Explainable and non-explainable discrimination in classification. 3:155–170, 2013.

[42] F. Kamiran, T. Calders, and M. Pechenizkiy. Discrimination aware decision tree learning. In *2010 IEEE international conference on data mining*, pages 869–874. IEEE, 2010. doi: 10.1109/ICDM.2010.50.

[43] F. Kamiran, A. Karim, and X. Zhang. Decision theory for discrimination-aware classification. In *2012 IEEE 12th international conference on data mining*, pages 924–929. IEEE, 2012. doi: 10.1109/ICDM.2012.45.

[44] M. Kearns, S. Neel, A. Roth, and Z. S. Wu. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In J. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2564–2572. PMLR, 10–15 Jul 2018.

[45] J. Kleinberg, S. Mullainathan, and M. Raghavan. Inherent trade-offs in the fair determination of risk scores. *arXiv.org*, 2016. ISSN 2331-8422.

[46] N. Kordzadeh and M. Ghasemaghaei. Algorithmic bias: Review, synthesis, and future research directions. *European Journal of Information Systems*, 31(3):388–409, 2022. doi: 10.1080/0960085X.2021.1927212.

[47] M. J. Kusner, J. R. Loftus, C. Russell, and R. Silva. Counterfactual fairness. *arXiv.org*, 2018. ISSN 2331-8422.

[48] Y. Lin, S. Gupta, and H. V. Jagadish. Mitigating subgroup unfairness in machine learning classifiers: A data-driven approach. In *Proceedings of the 2024 IEEE 40th International Conference on Data Engineering (ICDE)*. IEEE, 2024.

[49] K. Lum and W. Isaac. To predict and serve? *Significance*, 13(5):14–19, 2016. doi: 10.1111/j.1740-9713.2016.00960.x.

[50] J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press, 1967.

[51] S. G. Mayson. Bias in, bias out. *Yale Law Journal*, 128(8):2218–2473, 2019.

[52] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys*, 54(6):1–35, 2021. ISSN 0360-0300.

[53] R. Mohammed, J. Rawashdeh, and M. Abdullah. Machine learning with oversampling and undersampling techniques: Overview study and experimental results. In *2020 11th international conference on information and communication systems (ICICS)*, pages 243–248. IEEE, 2020.

[54] K. Napierala and J. Stefanowski. Types of minority class examples and their influence on learning classifiers from imbalanced data. *Journal of Intelligent Information Systems*, 46(3):563–597, June 2016. ISSN 1573-7675. doi: 10.1007/s10844-015-0368-1.

[55] A. Noriega-Campero, M. A. Bakker, B. Garcia-Bulle, and A. S. Pentland. Active fairness in algorithmic decision making. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society (AIES '19)*, pages 77–83, New York, NY, USA, 2019. Association for Computing Machinery. doi: 10.1145/3306618.3314277.

[56] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[57] D. Pessach and E. Shmueli. A review on fairness in machine learning. *ACM Computing Surveys*, 55(3):51:1–51:44, 2022. ISSN 0360-0300. doi: 10.1145/3494672.

[58] ProPublica. Compas recidivism risk score data and analysis. ProPublica, 2016. Retrieved from https://www.propublica.org/datastore/dataset/compas-recidivism-risk-score-data-and-analysis.

[59] A. Romei and S. Ruggieri. A multidisciplinary survey on discrimination analysis. *The Knowledge Engineering Review*, 29(5):582–638, 2014.

[60] N. A. Saxena. Perceptions of fairness. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '19, page 537–538, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450363242. doi: 10.1145/3306618.3314314. URL `https://doi.org/10.1145/3306618.3314314`.

[61] M. H. Shahrezaei, R. Loughran, and K. M. Daid. Pre-processing techniques to mitigate against algorithmic bias. In *2023 31st Irish Conference on Artificial Intelligence and Cognitive Science (AICS)*, pages 1–4. IEEE, 2023. ISBN 9798350360219.

[62] H. Suresh and J. Guttag. A framework for understanding sources of harm throughout the machine learning life cycle. In *Proceedings of the 1st ACM Conference on Equity and Access in Algorithms, Mechanisms, and Optimization (EAAMO '21)*, pages 1–9. Association for Computing Machinery (ACM), 2021. doi: 10.1145/3465416.3483305.

[63] S. Verma and J. Rubin. Fairness definitions explained. In *2018 IEEE/ACM International Workshop on Software Fairness (FairWare)*, pages 1–7, New York, NY, USA, 2018. ACM. ISBN 9781450357463.

[64] B. W. Yap, K. A. Rani, H. A. A. Rahman, S. Fong, Z. Khairudin, and N. N. Abdullah. An application of oversampling, undersampling, bagging and boosting in handling imbalanced datasets. In T. Herawan, M. M. Deris, and J. Abawajy, editors, *Proceedings of the First International Conference on Advanced Data and Information Engineering (DaEng-2013)*, pages 13–22, Singapore, 2014. Springer Singapore. ISBN 978-981-4585-18-7.

[65] M. B. Zafar, I. Valera, M. G. Rodriguez, and K. P. Gummadi. Fairness constraints: Mechanisms for fair classification. In *Artificial intelligence and statistics*, pages 962–970. PMLR, 2017.

[66] R. S. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork. Learning fair representations. In S. Dasgupta and D. McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 325–333, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR.

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

[68] L. Zhang, Y. Wu, and X. Wu. A causal framework for discovering and removing direct and indirect discrimination. *arXiv.org*, 2016. ISSN 2331-8422.

[69] Úrsula Hébert-Johnson, M. P. Kim, O. Reingold, and G. N. Rothblum. Calibration for the (computationally-identifiable) masses. *arXiv.org*, 2018. ISSN 2331-8422.