School of Computer Science
The University of Auckland
New Zealand

Balancing Utility and Fairness against Privacy for Sensitive Data

Andrew Chester

September 2021

Supervisors:

Dr. Yun Sing Koh

Dr. Jörg Wicker, Mr. Junjae Lee (advisor)

A thesis submitted in partial fulfillment of the requirements of
Master of Science

Abstract

Machine learning models continue to replace human decision making in high-stakes environments. In fields as diverse as healthcare, college acceptance and loan approval these algorithms can have significant effects on peoples lives. Privacy has long been a concern when dealing with sensitive information. Recently, fairness in machine learning has become a significant concern as well. Previous research has established a well known trade-off between privacy and accuracy. More recently, research has investigated the trade-off between fairness and accuracy. The focus of this thesis is the interaction between these two concepts. This is a recent field of study and most research focuses on the interaction of an individual de-identification mechanism and the subsequent bias mitigation methods which follow a similar methodology.

In this thesis we investigate the complex interaction between privacy, fairness, and accuracy through multiple de-identification and bias mitigation mechanisms. As motivating examples we address the general privacy cases as well as the healthcare field. We consider a scenario where a company that has access to sensitive records, be they medical or financial, would like to publish this data for users to analyse. The company operates under a regulatory environment which requires them to de-identify the data to protect the privacy of the individuals in the data. Additionally, the company would like to ensure that the data is not biased towards any particular groups.

We first developed a technique to assess the level of fairness loss and accuracy loss due to the de-identification process. We developed two novel measures to assess this loss. We applied this to multiple levels of de-identified data to assess the impact that de-identification has on these measures. Following this we adapted this technique to assess the fairness gain and the accuracy loss due to bias mitigation on the original and de-identified data. We adapted our novel measures to apply them to this comparison. Finally, we adapted a bandit based hyper-parameter optimisation mechanism to assess the hyper-parameters of the mitigation mechanisms and hyper-parameters to achieve a good trade-off between fairness and accuracy on de-identified data.

Acknowledgements

I would like to thank and acknowledge the following people.

The **Precision Driven Health** Partnership for funding this work. Special thanks to Kelly Atkinson for coordinating with me on this. **Junjae Lee** for his helpful insight into how to display my research in a practical application. **Quan Sun** for his feedback and suggestions of relevant material. My supervisors: **Dr. Jörg Wicker** for his levity and humorous discussions and most of all, **Dr. Yun Sing Koh** for her incredible support and patience throughout this year, combined with enough pressure to motivate me. This was critical to completing this thesis.

List of Publications

- The content of Chapter 3 was published in:
  Andrew Chester, Yun Sing Koh, Jörg Wicker, Quan Sun, and Junjae Lee. Balancing Utility and Fairness against Privacy in Medical Data (SSCI), 2020.

- The content of Chapter 4 was accepted in:
  Andrew Chester, Yun Sing Koh, and Junjae Lee. Understanding Effects of Mitigation on De-identified Data (IEA/AIE), 2021.

Contents

Glossary

Anoymization Field of research with techniques to prevent individuals who are the subjects of the data from being identified..

Fairness In the context of this work, the equal apportionment of positive and negative outcomes between privileged and unprivileged individuals in the protected attributes..

Generalisation The consolidation of narrow values into broader categories in categorical attributes or broader ranges in numeric data..

Perturbation The addition of noise to data values or values counts to produce uncertainty as to the inclusion of individual records or not..

Privacy Guarantees that the individuals who are the subjects of the data cannot be re-identified..

Privileged Class The class or range in the protected attribute which receives higher positive outcomes..

Protected Attribute Group The attribute(s) on which the user wants to achieve fair outcomes..

Suppression The removal of records, attributes or values form a dataset..

Unprivileged Class The class or range in the protected attribute which receives lower positive outcomes..

List of Figures

List of Algorithms

List of Tables

1

Introduction

Privacy and fair decision making have become significant concerns for all stakeholders when dealing with sensitive data and implementing machine learning processes. Privacy in machine learning broadly entails preventing data users from being able to identify individuals in the data. Fair decision making in machine learning is concerned with the even or proportional distribution of perceived positive outcomes between individuals or groups. Machine learning models by nature use a statistical distribution to discriminate between outcomes. However, given historical biases or sampling biases, this discrimination can cause disparities on Protected Attribute Group groups. This can result in undue advantage to giving Privileged Class classes and disproportionate disadvantage to Unprivileged Class classes. This unequal situation has lead to the establishment of laws such as the General Data Protection Regulation (GDPR) or the inclusion of provisions into existing laws such as the Health Insurance Portability Accountability Act (HIPAA), which govern the use and distribution of sensitive personal information.

In the last couple of decades, a research community has developed around Privacy-Preserving Data Mining/Publishing (PPDM/PPDP). The community aims to develop algorithms that simultaneously preserve privacy and maximise accuracy. This community has developed numerous mechanisms to preserve privacy during the publication and analysis of sensitive data. Due to this body of work's enormity, this research limits its scope to two main methods, k-anonymity Anoymization [75] and differential privacy [26].

Recently, there have been high profile examples of algorithms that are contain biases among certain demographic groups. These examples highlight the potential for algorithms to be unfair. Fairness, however, is a complex subject as the concept of fairness can vary from different perspectives. The branch of machine learning research known as Fair Machine Learning (Fair-ML) has developed to understand better and address this subject. This branch of research aims to develop metrics to measure bias, and methods to correct this bias while maintaining the predictive accuracy of the final model. This branch of research has produced numerous methods for bias mitigation. This thesis focuses on group fairness and the equality of positive outcomes between different classes in various demographic groups.

1.1 Motivation

For a long time, the issue of privacy has been a significant concern in many human-centric applications of machine learning due to the sensitive nature of the data involved [21]. This thesis investigates the complex association and interaction between the processes designed to preserve privacy and mitigate unwanted bias. The implications of this interaction can affect any area which employs machine learning. Our motivating examples look at both general and medical contexts.

Companies such involved in Health care management or financial management have access to extensive patient or customer records, which include information such as diagnoses, comorbidities, or account balances and demographics. This aggregation of data presents an enticing opportunity. Subsequently, machine learning processes can be applied to this data in order to understand health implications of whole populations.

In most cases, laws regulate that these raw records are shared only with the particular institutions which have a medical necessity to use them. These regulations create a need for de-identification. For healthcare management companies to release the data, they must have some degree of certainty that the data they are releasing will not allow the users to identify who the patients are and their diagnoses. Simultaneously, the publishers of the data want to ensure the data they are releasing does not contain biases that will unfairly disadvantage one class or another. Given that the de-identification processes objective is to make each individual indistinguishable from a set amount of other individuals in the data set, class imbalance and minority populations can have a significant effect on these processes.

Additionally, certain conditions are statistically more prevalent in some attribute classes than in others, which results in outcome disparities. For example, when assessing the likelihood of sickle-cell disease, it is sensible to send populations of African descent for further testing more often given that the disease is statistically more prevalent in those communities.

This discerning triage method aligns with the idea of maximising utility of medical resources. The result though, is a trade-off between attempting to catch all true positive cases - to care for patients, and to avoid false negatives - to minimise unnecessary costs. These goals lead to a question of equal balanced accuracy in disease detection rates meaning equal true positive and true negative ratios between classes.

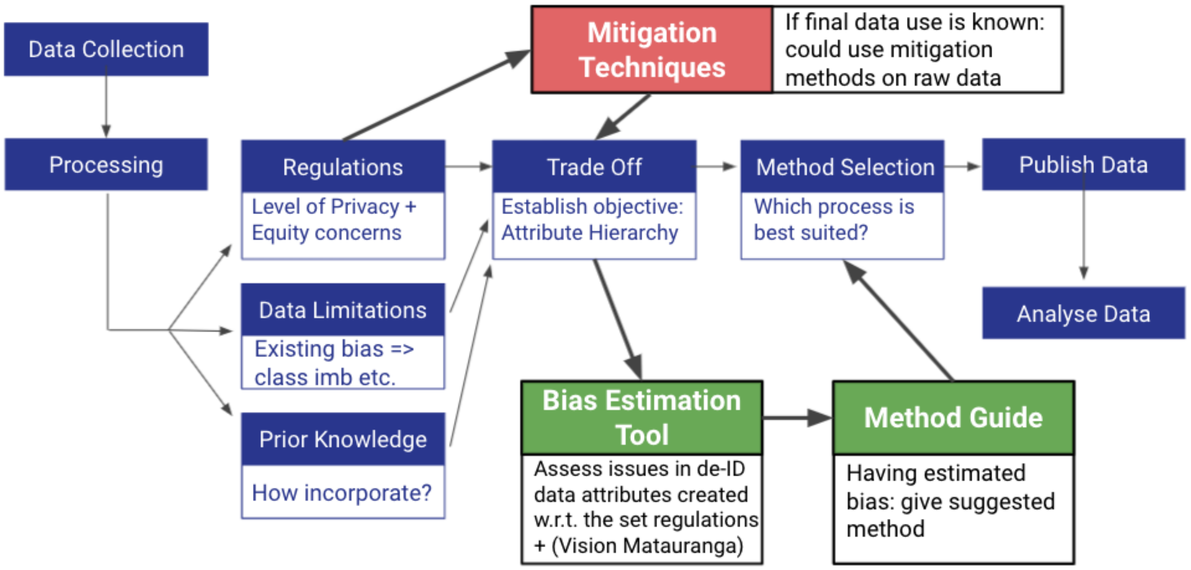

Data Collection
Processing
Regulations
Level of Privacy + Equity concerns
Data Limitations
Existing bias => class imb etc.
Prior Knowledge
How incorporate?
Mitigation Techniques
If final data use is known: could use mitigation methods on raw data
Trade Off
Establish objective: Attribute Hierarchy
Method Selection
Which process is best suited?
Publish Data
Analyse Data
Bias Estimation Tool
Assess issues in de-ID data attributes created w.r.t. the set regulations + (Vision Matauranga)
Method Guide
Having estimated bias: give suggested method

Figure 1.1: Schematic for application of bias estimation and mitigation.

This setting differs from general fairness domains such as housing loan approval or college admittance. In these scenarios, the different populations are presumed to have an equal underlying target distribution. The goal would be to create equal proportions of perceived positive outcomes.

1.2 Problem Statements

To assess the effects of de-identification mechanisms on levels of unwanted bias and assist data publishers in implementing effective mitigation mechanisms, we aim to answer the following questions:

1. To what extent does the level of de-identification and other data characteristics affect the accuracy and fairness of predictive models? How can this be used to maximise the utility of the data pre-processing pipeline?

2. How does the level of de-identification and level of existing bias and class imbalance affect the bias mitigation process? How can we use this information to inform the bias mitigation method selection?

3. Given the complexity of the interaction between different data characteristics and different de-identification and bias mitigation mechanisms, can we use hyper-parameter optimisation to achieve the three-way compromise of privacy, fairness, and accuracy?

1.3 Objectives

The aims of our research are:

1. To develop a mechanism that can assess the level of bias introduced and loss to accuracy resulting from the de-identification process.

2. To develop a mechanism that can assess the improvement to bias and the loss to accuracy resulting from bias mitigation processes.

3. To develop a bias mitigation optimisation mechanism that can determine the best de-identification and bias mitigation mechanism to achieve near optimal trade-off between fairness and accuracy on de-identified data.

1.4 Contributions

The main contributions of our work are:

1. Understand the relationships between de-identification and the fairness and accuracy trade-off and a framework to assess these effects.

2. Analyse the effects of mitigation on de-identified data with different characteristics and a framework to assess these effects.

3. Develop a hyper-parameter optimisation mechanism that can efficiently find a set of hyper-parameters for a mitigation mechanism to achieve an effective trade-off between fairness and accuracy on de-identified data.

1.5 Overview of Research

This thesis discusses the interaction of two broad topics, that of privacy and fairness. Figure 1.2 illustrates the data processing pipeline and highlights where these two concepts fit in. We outline the context of our motivation example in Figure 1.1.

The initial phase of the research investigated and compiled the related work on fairness and privacy. We developed the bias assessment framework to be implemented on data to which a company has access. This data will contain any inherent bias or sampling bias, that are present from the collection phase. Additionally, the company can perform de-identification to the extent that meets the privacy requirements set out by their regulatory body. The company can apply this framework to the data before or after de-identification. This framework allows the company to determine the levels of unwanted bias and determine the additional bias that has arisen through the de-identification process. This allows a decision to be made on the level of de-identification to perform, if the fairness lost is unacceptable.

Alternatively, the company can perform bias mitigation using the mitigation optimisation framework to address fairness. Then the company can use the mitigation assessment framework to assess the mitigation process's effectiveness by assessing the level of fairness and accuracy before and after the mitigation process.

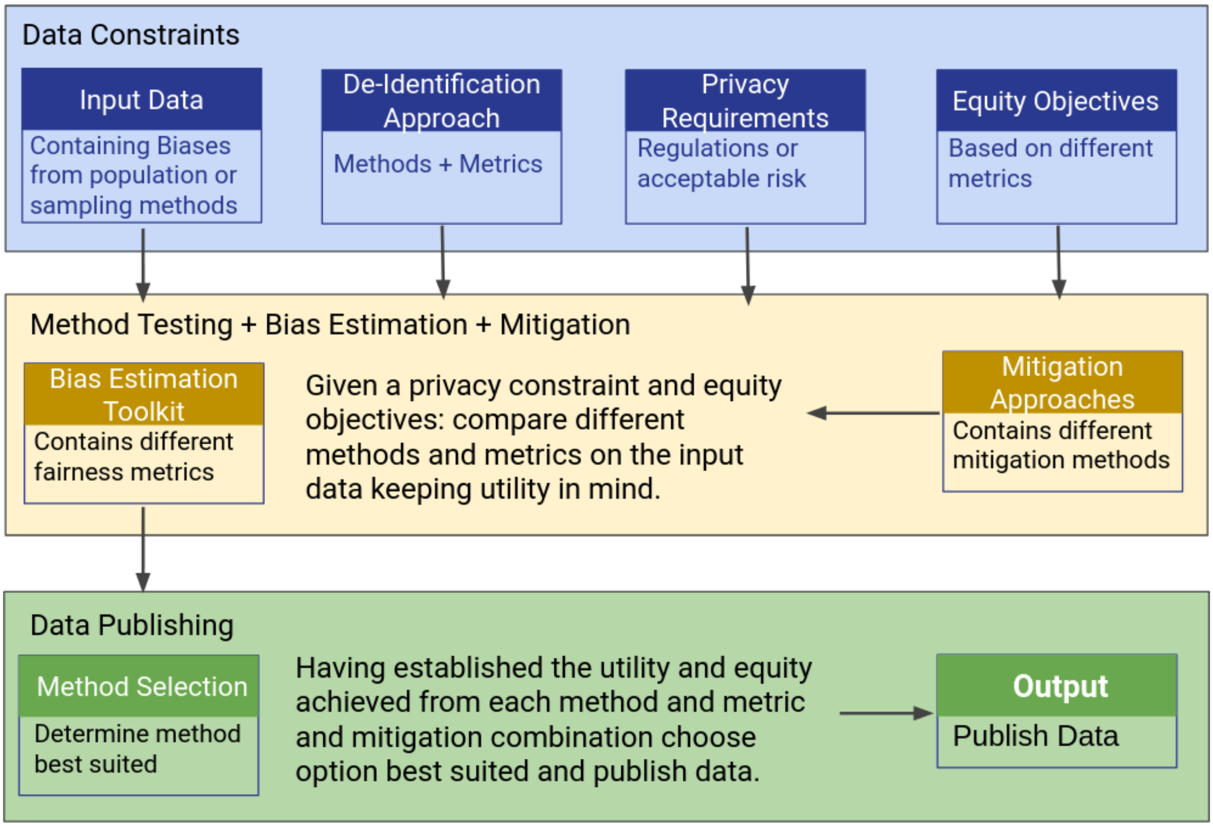

Data Constraints
Input Data
Containing Biases from population or sampling methods
De-Identification Approach
Methods + Metrics
Privacy Requirements
Regulations or acceptable risk
Equity Objectives
Based on different metrics
Method Testing + Bias Estimation + Mitigation
Bias Estimation Toolkit
Contains different fairness metrics
Given a privacy constraint and equity objectives: compare different methods and metrics on the input data keeping utility in mind.
Mitigation Approaches
Contains different mitigation methods
Data Publishing
Method Selection
Determine method best suited
Having established the utility and equity achieved from each method and metric and mitigation combination choose option best suited and publish data.
Output
Publish Data

Figure 1.2: Private, fairness aware data processing pipeline.

1.6 Structure of this Thesis

- Chapter 2 provides a background on the concept of data privacy, the issues of bias in machine learning, and recent work relating to data de-identification and bias mitigation.

- Chapter 3 describes experiments conducted to assess the effects of de-identification on the level of fairness and accuracy of models developed from this data. We developed this work into the bias assessment framework.

- Chapter 4 describes experiments conducted to assess the effects of bias mitigation on de-identified data and the effects of data characteristics on the fairness and accuracy of models developed from this data. These experiments inform the other part of the bias assessment framework.

- Chapter 5 introduces the bandit based mitigation hyper-parameter optimisation mechanism (Mitiband). A mechanism for determining the optimal mitigation mechanism and hyper-parameters for trading-off fairness and accuracy in de-identified data.

- Chapter 6 concludes the thesis, points out future directions and adds some final reflections and remarks.

- Appendix: this contain material for reproducability including: datasets, Python files, and Jupyter notebooks.

2 Background and Related Work

This chapter provides background and discusses the related work for this thesis. Section 2.1 provides an overview of the premise of this research. Section 2.2 outlines some of the sources of bias. Section 2.3 will discuss the rationale, mechanics, and effects of the de-identification process. Section 2.3.3 will discuss the trade-off between privacy and utility and the metrics used. Section 2.4 details the three primary de-identification methods and the effects on fairness. Furthermore, this section will describe some of the metrics used for determining fairness. Section 2.5 outlines mitigation methods for dealing with different forms of bias. Section 2.7 discusses some of the open challenges in the area. Finally, Section 2.8 will conclude this chapter.

2.1 Overview

Due to the increased scrutiny around data publishing and machine learning processes, PPDM/PPDP and Fair-ML have become active areas of research [64, 12]. As research into privacy preservation progressed, attackers exposed vulnerabilities in specific methods; thus, numerous methods have been devised to address these issues [33]. These methods fall into three main categories: Generalisation and Suppression, anatomisation and permutation, and Perturbation and differential privacy.

The research community developed various metrics to assess the effects of these procedures. These metrics can measure the information lost in the de-identified dataset or compare the quality of models built using the original data with those built using the de-identified data. Although numerous metrics exist, they all focus on the quality of the results over the whole dataset [64]. None of these metrics explicitly address the differential loss in information, nor the difference in accuracy, between different attributes classes in the data.

This gap leads to the issues of fairness. Fairness in machine learning is concerned with biases present in the data, the model, or the predicted outcomes. Early research on biases was conducted for fraud detection and rare attribute detection. This research aimed to improve the ability of models to detect target attributes on minority (low proportion) classes [35]. More recently, work has gone into dealing with the issues of fairness between groups and individuals. Numerous metrics and methods have been developed in the Fair-ML community and tool-kits have been developed to implement these. For example, Bellamy et al. [13] established a comprehensive tool-kit AIF360 of bias mitigation processes focused on dealing with fairness in machine learning. This tool-kit included numerous metrics and mitigation methods.

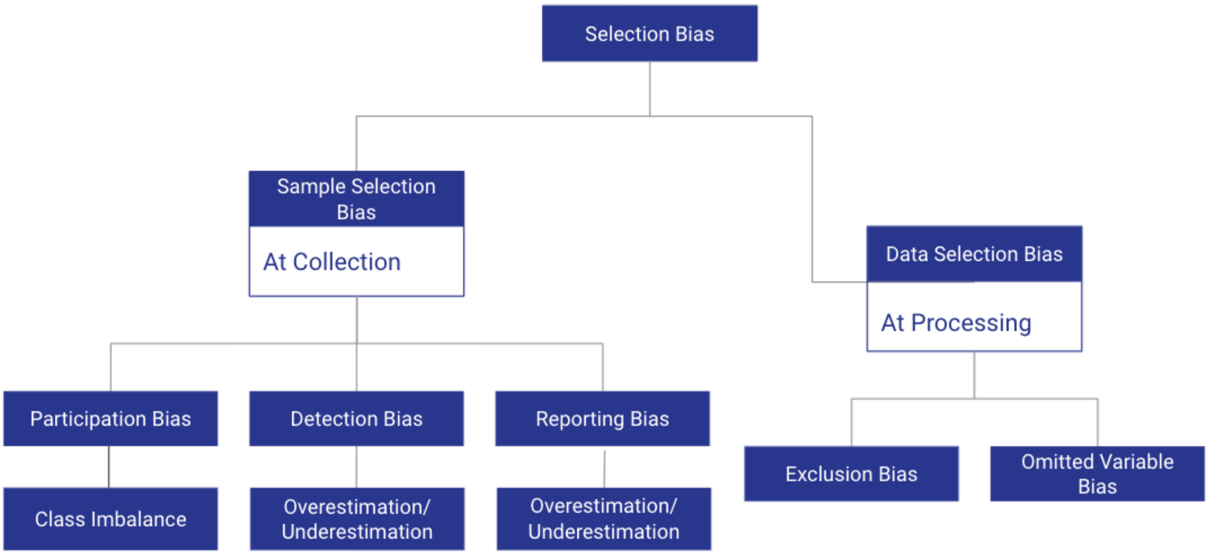


Figure 2.1: Map of bias sources.

2.2 Sources of Bias

The classes and types of bias that can arise during data processing are diverse, these are categorised in Figure 2.1. Thus, bias is a broad term that varies from field to field and applies to most aspects of research and industry. Additionally, the research around bias in the field of machine learning and particularly the medical field has dissected bias into many different scenario-specific forms [19]. However, one fundamental way to define bias in a statistical sense is the persistent or systematic error expected to be made by a learning algorithm when trained on any given training set [24] error-bias. Alternatively, bias can represent fairness. There are numerous ways to assess fairness in machine learning. However, for this survey, fairness refers to the differential benefits of outcomes that different attribute classes receive. This discussion will focus on five relevant types of bias. These are: participation bias leading to class imbalance (Section 2.2.1), exclusion bias leading to loss of patient class (Section 2.2.4), reporting and response bias (Section 2.2.3), detection bias (Section 2.2.5), and omitted variable bias (Section 2.2.6). Finally, this section will discuss the issues of fairness.

2.2.1 Participation Bias

Participation bias describes the disparity in relative number of individuals from different classes participating studies, recorded in analysed events, or featured in the data. This bias can be prevalent in healthcare studies [82]. An example of this is that healthcare is not universally accessible in all countries. Therefore, lower-income individuals may not seek out a doctor in time to address and therefore record the progress of a particular condition that can secondarily affect racial groups found in lower-income areas [82].

This disparity leads to class imbalance which is can be significant particularly in regards to its effect on the de-identification process. As discussed in Section 2.3, some of the de-identification processes seek to anonymise individuals from a group through generalisation. Given a small sample size, a minority group may require a further generalisation of other attributes that will diminish the data's predictive ability on those minority classes. This differential generalisation results in an imbalance in terms of predictive ability towards the majority classes as well. The disparity in detection ability of majority and minority ethnicity individuals in facial recognition systems [42] is an example of this.

2.2.2 Detection Bias

Detection bias describes a scenario where the rates of detection for certain target outcomes might be effected by certain other feature values or co-morbidities in the medical context. This leads to either overestimation or underestimation. An example of this is the issue of syndemics. Given the prevalence of diabetes among obese individuals, physicians may be more inclined to test for diabetes on obese patients. This could over inflate the statistics for diabetes for obese people due to the greater likelihood of observing the condition resulting form greater testing rates. Conversely, given a condition which is harder to identify given another condition might underestimate the correlation between these to conditions. An example of this is between obesity and prostate cancer, where the larger size on the patients may make accurate biopsies harder to achieve thus underestimating the diagnoses of prostate cancer [70].

2.2.3 Reporting or Response Bias

In the machine learning context, reporting bias or response bias describes how there is a general tendency for people to under-report negative or socially stigmatised behaviours and over-report socially favourable activities [66]. An example of this would be for a person at the doctor to report drinking less alcohol than they do and reporting exercising more often than they do.

Reporting bias can cause overestimation or underestimation. These two terms describe a situation where the relationship between the independent and dependent attributes are systematically either above or below the real rates.

2.2.4 Exclusion Bias

The second significant contributor to class imbalance is exclusion bias. This can result in the loss of patient classes; this situation is similar to class imbalance, except the minority class has been excluded entirely. Loss of patient class can arise from all the same issues discussed that lead to class imbalance. This term can also vary from field to field. For this discussion, exclusion bias relates to excluding individuals or classes from research or removal from datasets before analysis. This type of bias can occur in both the collection and processing phase and the analysis phase.

In the collection phase, exclusion bias can also be for strategic purposes or due to oversight. To achieve specific results, researchers can use strategic exclusion. Clinical studies are often highly selective on acceptance into trials to remove additional variables from the study to derive more repeatable results, or for safety reasons like excluding pregnant woman from studies [39]. Thus, particular populations or individuals may be excluded from a study to achieve the desired outcome. Alternatively, exclusion can result from other inherent biases that affect the researchers at the collection phase. Researchers' own biases can affect how they conduct studies and the questions they want to answer. This bias affects the scope of research towards answering questions that researchers would like to know [80].

In the processing phase, specific issues can cause exclusion bias to arise. When dealing with raw data, it is common to perform varied processes to clean the data. This process can include removing outlier individuals and classes if they appear not to fit the model and look like noise [65]. Therefore, they may be excluded to improve the model's predictive ability on the remaining classes. However, depending on these outliers' nature, they may represent real but rare pattern or attribute occurrence. Thus, the process creates a bias against understanding the implications of rare attributes.

The issue of exclusion bias is indirect in regards to de-identification. Suppose exclusion bias occurs before the de-identification phase. In that case, the datasets being de-identified have no evidence that the class existed, and the de-identification process will similarly exclude those classes. This bias prevents the ability of members of this class to benefit from the research conducted. Alternatively, the research conducted cannot take the uniqueness of those classes into account. Additionally, excluding these classes precludes the use of many mitigation techniques due to the removal of all instances.

2.2.5 Detection Bias

The other influential bias which also relates to overestimation and underestimation is detection bias. The medical field has put much research into this type of bias. Detection bias refers to a situation where the mechanism or impetus for investigation may disproportionately fall on a particular group. De Jong et al. [22] provide an example of this type of bias when analysing the correlation between diabetes and cancer. They found that accounting for detection bias lowers this correlation, reducing overestimation.

The effect of overestimation and underestimation on the de-identification process could be influential. The de-identification process often amalgamates different classes in the independent variables together. This amalgamation could have unexpected results on the proportions of the dependent variables.

2.2.6 Omitted Variable Bias (OVB)

OVB is the omission of variables relevant to the predictive ability of models derived from its constituent data. OVB can arise in the collection phase due to several factors, including lack of insight into dependency relationships or due to technical challenges and lack of capability for detecting a specific variable [18]. OVB can be affected by the processing phase in much the same way as exclusion bias. In the k-anonymity model, when an attribute is generalised into a larger majority class, that attribute's effects must be attributed to the majority class [73].

If the omitted variables are relevant to a particular population but not another, the statistical significance of that variable might be missed and the effect of that variable will skew the results of other remaining variables and diminish the quality of analysis on that particular attribute as well. Additionally, if there are multiple omitted variables, attempts to account for one variable could increase bias since the different omitted variables might be counteracting each other [73].

2.3 Data Privacy

It is necessary to understand the de-identification mechanics and bias types that can arise to understand how biases affect the de-identification process. This section will cover four main aspects: first, the rationale for the de-identification process; second, the privacy models and metrics used to assess the level of privacy and utility; third, the data characteristic which leads to error-bias and loss of utility; fourth, the issues of fairness and the metrics used to assess this.

2.3.1 Rationale for De-identification

De-identification is a process used to preserve privacy when publishing data either as micro-data or as a statistical distribution. The de-identification process takes the raw data from a table, T, as input and outputs the de-identified table, T'. There are numerous methods to achieve this, and each method produces a different output. However, the inputs are generally similar; each table has a set of records representing individuals with a set of attributes containing demographic and healthcare or other useful information. In a table, if any row has the same value for all attributes then these individuals are considered part of an equivalence class: given $(r, r') \in T$ if $\forall a \in A$ — $r_a = r'_a$ then r and r' form an equivalence class.

There are four distinct categories into which these attributes fall. Firstly, direct-identifiers DIDs are attributes that are unique to each individual; these include Names, Phone Numbers, and ID numbers. In Table 2.1, the attribute birth-date is a direct identifier. Secondly, quasi-identifiers (QIDs) are attributes that can form a unique vector to each individual. For example, in Table 1: <Gender, Postal-Code> is not a QID since there are at least two individuals in each equivalence class. However, <Age, Postal-code> forms a QID, since only one individual is 33 and one who is 60 with Postal code 0627. Next, there are sensitive attributes. These are attributes which an individual may not want to be publicly known in Table 2.1 diagnosis is the sensitive attribute. Finally, there are non-sensitive attributes. In Table 2.1, blood pressure (BP) is a non-sensitive attribute since it is not publicly available and is not common knowledge.

Table 2.1: Medical Record with Direct and Quasi Identifiers

ID	Name	Date of Birth	Gender	Age	Ethnicity	Post-Code	BP	Diagnosis
001	Alice	25/12/1970	Female	50	European	1010	120/75	Heart Disease
002	Betty	05/05/1971	Female	50	European	1010	101/70	Alzheimer's
003	Charli	25/04/1998	Female	22	European	0627	150/90	Heart Disease
004	David	31/12/1919	Male	100	European	0627	85/50	Dementia
005	Emma	14/02/1998	Female	22	Māori	0627	103/77	Flu
006	Fetu	01/01/1920	Male	100	Samoan	0627	110/70	Dementia
007	Gary	04/03/1987	Male	33	Chinese	0627	105/62	Flu
008	Han	02/01/1960	Male	60	Chinese	0627	75/50	Flu
Direct-identifiers			Quasi-identifiers				Non-sensitive	Sensitive

2.3.2 De-identification - Privacy Models

It is important to understand the framework which guides these processes for preserving privacy to understand the purpose of different de-identification techniques. The first metric is k-anonymity [74]. There are further definitions which extend k-anonymity though the original idea is the same; the metric sets out a minimum number of records that need to be in each equivalence class. This metric protects against record linkage attacks, but not attribute linkage attacks. These attacks focus on narrowing down the sensitive attributes a record might have based on its equivalence class. The l-diversity model answers this problem and sets out the minimum number of unique sensitive attributes associated with each equivalence class to prevent attribute linkage attacks [3]. When releasing statistical distributions, attackers can infer whether an individual is in a class or not using a probabilistic attack. This vulnerability led to the creation of the t-closeness metric [61]. The metric sets out a minimum threshold by which the distribution on a sensitive attribute in any equivalence class can differ from that sensitive attribute's overall distribution.

2.3.3 Effects of De-identification on Utility

There is a well-established dilemma between privacy and utility see Section 2.3. As such, these privacy models and de-identification methods led to a multitude of different metrics designed to measure their effects. Multiple surveys of de-identification processes have found that in almost all cases, regardless of the metric or method used, some reduction in the predictive quality of models will result when trained on the de-identified datasets [33, 30, 64]. It is, therefore, necessary to assess the frameworks used for evaluating the utility of these de-identification processes. There are two general categories; how the data is affected (data-metrics) and how further analysis is affected (analysis-metrics).

Data Metrics

Data metrics measure changes to the dataset as a result of the de-identification process. The first two metrics do not take the distribution into account. Although they account for all individual record generalisations, they treat them as equally significant. This equality among attributes is not likely to be the case given the number of attributes that could exist. For example, when generalising from <*lawyer*> to professional would likely have a lower impact than generalising from professional to <*any-job*>. However, both previous metrics treated these the same. The issue of determining the differential cost of generalising different categorical attributes is an open question [30].

1. Minimal distortion (MD) [75] is a simple penalty-based system that increments a distortion counter every time any value is generalised. This metric is essentially a

taxonomy tree where each leaf in the tree represents an attribute value, and the parent nodes are the combined terms of their child nodes.

2. Loss metric (LM) [44] which, given an attribute generalisation operation, gives a ratio of how many other nodes a record value is indistinguishable from the original number of leaf nodes that existed. All records with this attribute have this metric applied; then, the LM metric is the average of all records. This method also applies to numerical attributes and take the ratio of the new sub-domain to the previous full domain. For example, with age, if the <Age> attribute is generalised from <5> to <1-5> and the original domain is capped at 100, then the LM ratio would be 5/100.

3. Information Loss (i-Loss) is very similar to the LM metric in its analysis and the result for each generalised value. However, this process assigns each attribute a weighting. This weighting can mitigate the disadvantage of the generalisation for the whole dataset. This procedure does not solve the issue of each generalisation having differential cost, but the user can weight each attribute differently according to the significance the user places on it. As with LM, i-Loss takes the average of the values of all records with the attribute-weightings accounted.

4. Classification metric (CM) Iyengar et al. [44] was proposed to deal with the cost of individual generalisation operations. The CM charges a penalty every time a record value is generalised or suppressed into a group in which it is not the majority class. Intuitively, this metric applies to classification analysis in that the generalised record from the minority class will be classified with the majority class, resulting in a classification error.

5. Discernibly Metric (DM) [72] was an existing metric that dealt with the issue of distinguishing individual generalisation operations. To this end, DM charges a penalty to each record based on the size of its generalised class at each operation. For example, if the super-class is of size c, then each record generalised into this super-class will have a penalty of c. This metric is a direct measure of the effects of k-anonymity.

Trade-off Metrics

Other metrics, when addressing only the utility gained, ignore the privacy loss. Thus, if the operation that gains the most information also reduces the privacy so much so that no further specialisation operation is possible, the final result may be sub-optimal from a privacy perspective [33]. The trade-off metrics compare each generalisation operation, (s), individually as a relation between both sides, the utility gained, and the privacy lost. Information Gain to Privacy Loss (IGPL) is an iterative specialisation example, where, starting with a fully generalised set, iteratively specialises general classes into more specific classes. Each operation gains more specific information but will lose some privacy. Conversely, the metric can be used in reverse as well for generalisation operations (g) Information Loss to Privacy gain (ILPG) where each generalisation operation losses information but gains privacy. [33]:

$$IGPL(s) = \frac{IG(s)}{PL(s)+1} \qquad ILPG(g) = \frac{IL(g)}{PG(g)+1} \tag{2.1}$$

The ILPG function measures the quality of each iterative generalisation performed in the k-anonymity process. The way to measure IG and IL, or PG and PL depends on additional metrics used to determine the quality of the information.

Analysis metrics

The methods of determining utility through analysis are comparing the results of analytical processes on the original data and the de-identified data. Different data mining techniques work better on different types of data. Therefore, the composition of the input data has a significant impact on the quality of the analysis. As such, this should be taken into account when performing the de-identification process. This section illustrates these metrics through comparison of data mining on the original and de-identified data; three different data mining techniques; clustering, classification, and association rule mining provide the examples.

Clustering: metrics employ comparisons between the cluster created on the original data to that created on the de-identified data. Mis-classification error can also be applied to the clustering scenario, as shown in Equation 2. This metric measures the number of data points k that were grouped in a different cluster in the de-identified set D', than they were in the original set D; and takes this number over the number of over-all points N.

$$M_E = \frac{1}{n} x \sum_{i=1}^{k} (|Cluster_i(D)| - |Cluster_i(D')|) \tag{2.2}$$

Many methods use statistical metrics which can rely on measuring false positives and false negatives. Statistical metrics require reference to determine the accuracy of the cluster. The user can create reference by removing a class attribute, then performing the cluster and reassigning the class attribute. Subsequently, the majority class on that attribute define the cluster. If the record is not part of the majority class, it is considered to be irrelevant. A reference point is deemed to be external to the process, so metrics using this method are called external metrics. Examples of the metrics include the Rand Index (RI), F-Measure, Silhouette co-efficient, and Davies-Bauldin Index (BDI).

Classification is a broad set of procedures used for predicting attribute values on future instances. Decision trees filter records into partitions depending on their attributes. Thus, it can determine which attributes are best at allocating records into specific partitions. For example, taking Table 2.1, given a particular protected attribute like <Diagnosis> the other attributes like Blood Pressure (BP), Age, and Gender form a decision tree to determine the <Diagnosis>. To determine a diagnosis, attributes need to have a boundary that differentiates their values for the target class. A binary class, like <Gender>, is simple to split. However, other attributes require a decision as to where the 'splitting point' is located. There are numerous metrics to determine these criteria, including Information Gain and Gini Index.

Since the primary objective, in this case, is predicting future occurrences or values, one of the standard techniques for assessing a classifier's quality is prediction accuracy. Effectively measuring the percentage of correctly predicted class values on future records, or for testing values with their class attribute hidden. This metric can be inverted to measure the number of incorrectly predicted class values, equating to the error rate. Then the error rate on the de-identified data can be compared with the error rate on the original data. These metrics allow a more in-depth insight into the correlation between particular attributes and the target attribute. This insight can indicate which attributes can be generalised to a higher degree or even entirely removed while maintaining predictive accuracy.

Association Rule Mining: For de-identification Atallah et al. [7] proposed the Association Rule Hiding method, which mines the non-sensitive rules and hides the sensitive rules. It achieves this aim by suppressing the item sets which generate the rules. However, this process may hide non-sensitive rules at the same time. One metric used to assess the effects of this process is the misses cost or Missed Pattern (MP) and the Artifact pattern (AP).

$$MP = \frac{\# \sim Rp(D) - \# \sim Rp(D')}{\# \sim Rp(d)} \qquad AP = \frac{|P'| - |P \cap P'|}{|P'|} \tag{2.3}$$

Here D represents a dataset, and D' is the de-identified dataset. $\# \sim R_p(x)$ represents how many non-sensitive patterns are found in the dataset x. The desire is to have all non-sensitive patterns found and to have $MP = 0$. Conversely, AP measures the number of artifact patterns present in the de-identified data that were not present in the original data. Here P and P' represent the set of patterns found in D and D'. As with MP, the desire is to have $AP = 0$, meaning no false patterns have emerged.

This section covered the issues of error bias as it relates to the utility of the de-identified dataset. Given that the way utility is measured can differ depending on the situation or the desired outcome, the focus here was on the different metrics used to assess utility. These metrics can measure the data losses or the losses to specificity on the data, which result in reduced information available in the published data. Alternatively, classification metrics measure the decreased accuracy or increased error rate that arises from the analysis performed on the de-identified data. Additionally, the trade-off metrics highlight the dilemma that arises between improving privacy and maximising utility.

2.4 Fairness in Machine Learning

As mentioned, the issue of data utility or error-bias is a well-established dilemma in the de-identification process. As discussed in Section 2.3, the de-identification process requires altering the data in some way, which results in a trade-off between privacy and utility. The same is valid for privacy and fairness. Additionally, this pursuit of fairness can also be at odds with utility.

This section will first outline some of the metrics used to assess fairness. Then, the discussion will cover the issues of bias that can arise from the most common types of de-identification. Additionally, we highlight how different levels of privacy affect these biases. To analyse fairness-bias, we divide the field into three categories. First, generalisation and suppression reduce the uniqueness of a given entry in a table. Second, anatomisation and permutation disassociate the QIDs from the sensitive attributes. Third, perturbation and differential privacy only releases a statistical representation of the data and no micro-data.

2.4.1 Metrics to Assess Fairness

The challenge with establishing fairness metrics is that regardless of the metric used there is almost no scenario in which the outcomes of any process could be entirely fair to all attribute classes [13]. Bellamy et al. [13] recently published a comprehensive tool-kit with a review of many metrics to assess fairness-bias.

1. Statistical Parity Difference: This metric assesses the outcome of decision rules among sensitive attributes. This metric can be used for binary classification or multi-variate classification as well [14].

2. Equal Opportunity Difference: This is a metric of the difference in the true positive rate among different attributes classes usually described as the privileged and unprivileged. This metric measures the benefit that each attribute class receives; it works on classification tasks and can be used to compare the input and output data [13].

3. Average Odds Difference This is another metric relating to true positives and take the difference between the true positive rate and the false positive rate and compares the privileged to the unprivileged classes [13].

4. Disparate Impact This metric assesses the probability of receiving a favourable outcome depending on the attribute class [11]. This metric can be applied to the input data or the output data from a classifier.

2.4.2 Generalisation and Suppression

Generalisation has the goal of reducing the specificity of each record. Therefore, there is an inherent loss involved in performing these processes. The issues arise when inherent biases in the sampled data are either reflected or exacerbated by the process of generalisation. To assess these processes, there are many different methods, implementations and metrics by which to accomplish the generalisation and thus it is necessary to understand the complex interaction that occurs between these implementation metrics, attribute hierarchies, utility goals, and fairness goals.

Effects due to class imbalance

One of the critical aspects affecting fairness is class imbalance. Whether this results from natural differences in population proportions or the result of biases that affect the sampling process, class imbalance can have a sizeable impact on fairness. When the intention is to make all individuals in a dataset anonymous, the process differs depending on class proportions and can result in different generalisation levels for particular populations.

Table 2.2: Medical Record with Quasi Identifiers

Gender	Age	Ethnicity	Postal-Code	Diagnosis
Female	50	European	1010	Heart Disease
Female	50	European	1010	Alzheimer's
Female	22	European	0627	Heart Disease
Male	100	European	0627	Dementia
Female	22	Māori	0627	Flu
Male	100	Samoan	0627	Dementia
Female	33	Chinese	0627	Flu
Male	60	Chinese	0627	Flu

The procedure for generalisation involves taking attributes of the QID and removing some level of detail. Taking a single attribute specifically, imagining a situation where all values in an attribute are the same, would require no generalisation. Conversely, if each value was unique from all other values, some generalisation would need to be performed on each attribute. Given both these cases have equal application of generalisation across the individuals in this attribute, there would be no explicit fairness-bias. However, this is likely not the case. If there is an attribute class, for example, <Ethnicity> in Table 2.2, with eight records and there are two minority classes (Māori) and (Samoan) each with

$\frac{1}{8}$ of the group. Additionally, there is (Chinese), which has $\frac{2}{8}$ while the majority class (European) has $\frac{4}{8}$. Here the majority class may meet the privacy threshold of k while the minority class does not. While the majority class can remain un-generalised since both (Māori) and (Samoan) are unique values. It would be necessary for them to be generalised into a larger group for example (Māori and Pasifika), thus meeting the threshold. At the same time, the value (Chinese) may or may not meet the privacy threshold of k depending on whether $k = 2$ or $k > 2$. When published, this modified data set would lack the detail to perform group-specific analysis on Māori or Samoan individuals. While European and Chinese individuals would retain attribute specific analysis.

Effects due to attribute correlation

Despite this potential for fairness-bias to arise, the issues of class imbalance and differential generalisation of majority and minority groups in an attribute class may not always have a substantial impact on the outcome benefit or cost. Another crucial factor that must be present, the attribute that is being generalised must correlate to the target attribute, and the values in the sensitive attribute must differ. If the attribute has little correlation or no difference on the target attribute, the fairness-bias resulting from differential generalisation should have little added effect on the predictive ability of models trained on a dataset produced from such a generalisation.

Table 2.2 provides a small example. If the objective is to predict the rate of (Dementia) across the <Ethnicity>; the rate of dementia for (European) is $\frac{1}{4}$, for (Chinese) it is $\frac{0}{2}$, and for (Māori) $\frac{0}{1}$ while for (Samoan) it is $\frac{1}{1}$. Among the (Māori) and (Chinese) records, none have a diagnosis of (Dementia). At the same time, (Dementia) does appear in other <Ethnicity> groupings. Thus <Ethnicity> may have an impact on rates of (Dementia). Thus, generalising (Māori/Chinese) together will retain the same predictive rates for (Dementia). On the other hand, going back to the previous example where (Māori) and (Samoan) are grouped into (Maroi/Pasifika), the new dementia rate would be $\frac{1}{2}$ overestimating the rate for (Māori) by 50% and underestimating the rate for (Samoan) by 50%. These differentials represent measurement errors in the predictive ability of each group.

Effects of k-anonymity

The next issue that affects fairness-bias is the level of privacy that is required. Using k-anonymity if the value of k us set to $k = 1$, there is no need to generalise at all. Then the process introduces no additional bias. Using the data and scenario of detecting (Dementia) as the previous example, if $k = 2$ then values for (Māori) are overestimated, and the values for (Samoan) are underestimated. However, if the value of k is increased to $k = 3$, then one possible grouping would be (European) with a (Dementia) rate of $\frac{1}{4}$ and (Non-European) with a (Dementia) rate of $\frac{1}{4}$. This grouping would produce accurate predictions of 1/4 for (Europeans). Meanwhile, both (Chinese and Māori) are overestimated by 25% while (Samoan) is underestimated by 75%. Compare this to the previous example where again $k = 2$ it can be seen that the underestimation for the (Samoan) group increased. However, this reduces the overestimation for the (Māori) group. The final result here is: for 2 groups or 3 records they are overestimated by 25%, and for 1 group or 1 record it is underestimated by 75%. For an overall distortion shown by the added average error rate in Equation 2.4.

$$(0.25 * (3) + 0.75 * (1))/4 = 0.375 \tag{2.4}$$

$$(0.15 * (4) + 0.6 * (1))/5 = 0.24 \tag{2.5}$$

An alternative grouping for $k = 3$ would be to group (European/Samoan) giving a (Dementia) rate of $\frac{2}{5}$ and (Māori/Chinese) with a (Dementia) rate of $\frac{0}{3}$. This grouping still fulfils $k = 3$ anonymity; however, the target attribute distortions are very different. For (Māori/Chinese), the results have no distortion. For (European/Samoan) There are four records or 1 class that is overestimated by 15% and one record and one class underestimated by 60%. Using overall distortion as a metric, shown in Equation 2.5. Also, this alternate grouping produces a lower maximum distortion experienced by any particular group 60% as opposed to 75%

In addition to k-anonymity, there are additional anonymity metrics that set stricter conditions, which meet higher levels of privacy to address different kinds of attacks. Attribute attacks are where an attacker does not determine who an individual record is but can infer the value of a sensitive attribute based on the group that their victim is a part of, and the proportions of the sensitive attributes. For example, if an attacker knows their victim is European, then from Table 1: the attribute <European> indicates only three possible conditions (Heart Disease, Obesity, Dementia). This situation can remain even after generalisation; for example, in the (Māori/Chinese) group, all individuals would have (Flu) for <diagnosis>. Thus the attacker would know the diagnosis for all three individuals.

Effects of l-diversity

To address this, Machanavajjhala et al. [62] proposed the l-diversity model. This model ensures that there are at least l distinct well-represented values in the sensitive attribute for each group, so the attacker cannot infer with a high likelihood that their victim has any particular condition. This metric is directly working against the predictive ability of a model. If there is a group at high risk of dementia, so high that all individuals in the category have dementia, this would imply a strong correlation between the attribute and the target attribute. However, this would not meet even the lowest l-diversity condition $l = 2$. Thus, the class would need to be split up and mixed with different values on the sensitive attribute, thus losing this strong association.

Additionally, this further favours the majority classes since the majority class is larger and more likely to have a sufficient diversity of values in the sensitive attribute field. While the minority class, which have already been generalised to meet k-anonymity may, need to be further generalised. Take the example of generalising to (European/Samoan) and (Māori/Chinese). This example exhibits less overall and less group-specific distortion. However, the sensitive value for (Māori/Chinese) are all the same; thus, it would no longer be an acceptable generalisation. This situation creates a need to resort back to the (European) and (Non-European) grouping, which has a more significant overall and group-specific distortion.

One approach proposed to decide the level of l is to set $l = k$, thus matching the level of anonymity achieved by the anonymisation operations. This approach can have limitations, though, specifically for binary classifiers when only two values are possible. A compromise metric $(c - l)$-diversity sets the maximum and minimum threshold for the most common and least common values, respectively.

As mentioned l-diversity has limitations in that it does not take the distribution of sensitive attributes into account. For example, if the objective is to classify dementia again using a simple binary classifier of YES or NO for an attribute <Dementia>, Imagine a table with 1000 records with a QID set of <Age, Gender, Ethnicity>. Assuming dementia occurs at a rate of 50 in 1000 but has a negligible occurrence under 50 and has a continually higher percentage as age increases. Presume ages are divided by ten-year increments, and for ages (50-59) dementia has a rate $\frac{10}{100}$ and for (60-69) a rate of $\frac{20}{40}$. If the groups are split along ten-year increments, then an attacker could infer that an individual in their 60's has a 50% likelihood of having dementia. This scenario shows how the original assumption an attacker would have is that there is a $\frac{1}{20}$ chance an individual has dementia. However, given the distribution, they can see that the probability reduces to $\frac{1}{2}$.

Effects of t-closeness

Li et al. [61] proposed t-closeness to address this issue. This metric sets out a threshold t by which the proportion of a sensitive attribute in any equivalence class can differ from the overall distribution. This condition sets out limits to the composition of each equivalence class but also depending on the occurrence rate of a rare sensitive attribute could limit the number of equivalence classes due to limited instances of values to split among different equivalence classes, further distorting the results. The value of t could be determined in many ways; Li et al. proposed the Earth Mover Distance (EMD). For a simple explanation, assume a ratio of 4 : 1, meaning the proportion in a class can be no more than 4 times greater than in the overall population, setting $t = 4 \quad \forall a < 0$. For example, in the previous situation attempting to determine rates of dementia, if the overall rate of dementia in the population is 5% with $t = 4$ then the rate of dementia in an equivalence class must be less than 20%. From this, the equivalence class for (60-69) does not meet this requirement. One method to deal with this is to adjust the partitioning. If "under 60" rates are lower and "over 70" rates taper off, the partitions could be shifted to split up this group which contains the highest likelihood of dementia. Instead, having (55-64) with a rate of 14% and (65-74) with a rate of 20%. If this method was not sufficient to reach t-closeness, then the <Age> might have to be generalised further by possibly including all ages above (60) or by suppressing the attribute. This scenario could result in a further increase in the error rate for the more generalised class.

This metric focuses directly on specific groups, not the population as a whole; thus, this method can disadvantage minority groups, including age brackets prone to rare conditions or even common conditions distributed unevenly. Another example, predicting the occurrence of a condition like sickle-cell anemia, the condition is most prominent among people of African descent [38]. Suppose this research were performed where people of African descent are a minority group. In that case, this method could preclude discovering such explicit correlation as having the condition solely found in one equivalence class would violate the t-closeness principle.

Effects of an attribute hierarchy

These simplistic examples explain how the interaction of class imbalance, correlation, and privacy level affect fairness. However, for a realistic example, it is necessary to understand these examples with more numerous QID sets. Instead of working on a singular attribute, all of these aspects need to be addressed on sets of attributes with differing class imbalances and differing correlation levels. This challenge is where the idea of attribute hierarchies comes in.

One way to achieve this is to use metrics that favour one group or another during the generalisation process creating a hierarchy that determines the order in which attributes will be generalised. This process could entail determining that it is preferable to suppress <Ethnicity> before <Age> when determining the likelihood of dementia, this may result in an improvement in predictive ability as the data would indicate dementia correlates to a high degree with particular age groups. However, if there is also a correlation, though not as strong, with <Ethnicity>, this might be lost if the <Ethnicity> attribute is preferentially generalised. This default generalisation would favour the majority ethnicity as their instances would be more prominent in the training data.

Additionally, different diagnoses or conditions would likely see other correlations among different attributes meaning the metrics and attribute hierarchy for researching different conditions would differ. Furthermore, if the final analytical work is unknown, then there is no definitive way to determine which is the best hierarchy. Suppose the objective is to reduce bias to marginalised populations. In that case, the setting up of a hierarchy is like using a biased metric to guide the algorithm to counteract the natural algorithmic bias against minority populations.

2.4.3 Anatomisation and Permutation Methods

The methodology of anatomisation proposed by Tao et al. [76] and permutation proposed by [87] both involve separating QIDs from the sensitive attributes. Anatomisation releases the data in separate tables; one with QIDs (QIT) and the other with sensitive attributes (ST). Both tables also contain a special attribute called GroupID. This attribute is what links feature attribute groups to target attribute groups. All records in a single group in the QID will have a GroupID which matches a GroupID in ST with a corresponding set of attributes. Going back to the idea of l-diversity, if the GroupID in ST has l distinct values, then the likelihood of matching a record in QIT to a sensitive attribute in ST is $1/l$. To adjust the privacy level, GroupID can be expanded in QIT and therefore expanded in ST, resulting in a greater l.

The process starts with a table of patient records and diagnoses, then entails partitioning the QID groups from the patient records so that no more than $1/l$ records has the same diagnosis for each group. Then remove all sensitive attributes and replace them with a GroupID. Then, create a second table ST with two columns GroupID and diagnosis count, which contains the number of instances of each diagnosis in the corresponding QID. Since this does not involve changing the QIDs or the sensitive attributes in any way, there are fewer opportunities to introduce fairness-bias directly. Xiao et al. [84] showed that this method could produce better results for aggregate queries. Intuitively this method retains more specific domain values. However, the method requires a process similar to generalisation, which partitions the QIDs into groups.

The algorithm *Anatomise* [84] takes in a table of micro-data and an operator chosen parameter l defined similarly to l-diversity. First, we hash all record-tuples in the original table into buckets based on their diagnosis. It starts an iterative process of making groupings from the buckets; each iteration results in a new QID-group. While there are at least l non-empty buckets, the algorithm selects a random tuple from the l buckets with the most tuples. The process assigns the tuples from all buckets until there are less than l non-empty buckets. If there are remaining tuples, those tuples are assigned to previously created groups that do not include that tuple's sensitive attribute.

Since this method always groups together the most common conditions, it cannot guarantee an equitable distribution. Consider a table of 1,000 micro-data health records

Table 2.3: Example of Anatomisation

(a) Diagnosis Bucket

Diagnosis	Count	QID Tuples
Flu	300	$(T_1...T_{300})$
Indigestion	250	$(T_1...T_{250})$
Insomnia	200	$(T_1...T_{200})$
Heart Disease	145	$(T_1...T_{145})$
Dementia	50	$(T_1...T_{50})$
Alzheimer's	45	$(T_1...T_{45})$
Sickle-Cell A.	10	$(T_1...T_{10})$

(b) Iteration 1

Diagnosis	Count	QID Tuples
Flu	100	$(T_1...T_{300})$
Indigestion	50	$(T_1...T_{250})$
Insomnia	0	$(T_1...T_{200})$
Heart Disease	145	$(T_1...T_{145})$
Dementia	50	$(T_1...T_{50})$
Alzheimer's	45	$(T_1...T_{45})$
Sickle-Cell A.	10	$(T_1...T_{10})$

(c) Iteration 2

Diagnosis	Count	QID Tuples
Flu	50	$(T_1...T_{300})$
Indigestion	0	$(T_1...T_{250})$
Insomnia	0	$(T_1...T_{200})$
Heart Disease	95	$(T_1...T_{145})$
Dementia	50	$(T_1...T_{50})$
Alzheimer's	45	$(T_1...T_{45})$
Sickle-Cell A.	10	$(T_1...T_{10})$

(d) Iteration 3

Diagnosis	Count	QID Tuples
Flu	0	$(T_1...T_{300})$
Indigestion	0	$(T_1...T_{250})$
Insomnia	0	$(T_1...T_{200})$
Heart Disease	45	$(T_1...T_{145})$
Dementia	0	$(T_1...T_{50})$
Alzheimer's	45	$(T_1...T_{45})$
Sickle-Cell A.	10	$(T_1...T_{10})$

(e) Iteration 4

Diagnosis	Count	QID Tuples
Flu	0	$(T_1...T_{300})$
Indigestion	0	$(T_1...T_{250})$
Insomnia	0	$(T_1...T_{200})$
Heart Disease	35	$(T_1...T_{145})$
Dementia	0	$(T_1...T_{50})$
Alzheimer's	35	$(T_1...T_{45})$
Sickle-Cell A.	0	$(T_1...T_{10})$

(f) Sensitive Attribute

GroupID	Diagnosis	Count
Group1	(Flu Indigestion Insomnia Heart Disease)	635
Group2	(Flu Indigestion Heart Disease)	150
Group3	(Flu Heart Disease, Dementia, Alzheimer's)	185
Group4	(Heart Disease Alzheimer's Sickle-Cell A.)	30

with QID of <Age, Sex, Ethnicity> and the sensitive attribute as <Diagnosis>. First performing the bucketisation step with diversity set to $l = 3$ gives Table 2.3(a). Then iteratively assigning the record-tuples into groups would proceed as follows. In the first iteration, there are 200 sets of 3 individuals that can have (Flu Indigestion Insomnia). Removing these would result in Table 2.3(b). In the second iteration, there are 50 sets of 3 individuals that can have (Flu Indigestion Heart Disease), this would result in

Table 2.3(c). In the third iteration, there are 50 sets of 3 individuals that can have (Flu Heart Disease, Dementia) and leaves Table 2.3(d). In the fourth iteration, there are 10 sets of 3 individuals that can have (Heart Disease Alzheimer's Sickle-Cell A.), which results in Table 2.3(e). This instantiation leaves two diagnosis buckets remaining, each with 35 individuals. Then the process places these into QID groups that do not contain their diagnosis. In this example, the (Heart Disease) records are added to Group1, and the (Alzheimer's) records are added to Group3, as shown in Table 2.3(f).

This example demonstrates that this method would place all the instances with the rare disease sickle-cell anemia in one category. Since this condition is affected by the <Ethnicity> attribute, grouping should have a corresponding change in the proportions of values on the <Ethnicity> attribute, thus, retaining the correlation. If the rare-condition remains after the last iteration, all these instances will be allocated to the same group. This situation may not always be the case. If there were many rare conditions, it might result in the splitting up of the groups. Since the method retains all QID information, checking if a rare condition is positively correlated to a specific trait in the QID set, for example, <Ethnicity> or <Age>, then the user could create a threshold to leave all rare conditions below a certain point. This threshold would ensure they end up grouped to retain that correlation. This scenario shows that in some cases, this method of de-identification could be less biased against rare conditions than generalisation.

They also show that there is an eligibility condition that needs to be met: at most n/l tuples may be associated with the same sensitive attribute where n is the size of the table at any stage in the iteration process. This condition could present issues when dealing with common diagnoses like flu or cold. Furthermore, this example presupposes a single diagnosis, where there might be multiple diagnoses for an individual, which might have predictive factors that this particular scenario does not consider. Additionally, the separation of QIDs and sensitive attributes into different tables presents challenges for many standard statistical analysis approaches that prevents this technique's broader applicability.

2.4.4 Perturbation Differential Privacy

The perturbation method adds noise by replacing values in the data, resulting in a dataset that no longer represents real individuals but should retain the statistical information. Expressly, the data publisher only retains the statistical properties they desire. Explicit altering the data and preserving specific correlations leaves open the possibility for fairness-bias to arise. Similarly, differential privacy produces a statistical distribution of the original data.

Perturbation

One perturbation method uses additive noise. This method involves manipulating sensitive attribute values s by adding r a random value selected from a chosen distribution [1]. Additionally, Fuller [32] demonstrated how the process could retain some statistical information like means and correlations. However, Kargupta [52] noted that when the correlation is high and low noise is added, then it is possible to make close estimations to the actual values. This limitation indicates that in scenarios with high correlation, the noise would have to be increased to meet the same privacy threshold. This added noise would have the effect of averaging out the correlation between classes. Thus, reducing the predictive ability of the models on groups which have a high correlation to a particular condition.

In synthetic data generation, this method requires building a statistical model of the data and then sampling points to generate new data points for publication. Alternatively, Aggarwal et al. [4] proposed a method called condensation. This method involves condensing the records into groups while maintaining the mean and correlations for the attributes. Then extracting statistical information from these groups to generate data points that can be published. The need to choose methods for grouping results is another opportunity for bias to arise and results in the same issues as the grouping phase in the anatomisation approach.

Data swapping is virtually the same process as permutation, where the values are non-synthetic. The process simply swaps the sensitive attributes for the QIDs. Vigneswari et al. [78] showed that in some instances, non-synthetic perturbation not only provides better privacy but also results in fewer lost rules when performing association rule mining.

Differential Privacy

Differential privacy publishes statistical information about a dataset. The novel insight is to look at the distributions and distort the data so that an attacker could not be sure if any individual were in the data set or not [26]. Differential privacy uses a perturbation method to account for shortcomings of the perturbation methods as found by Agrawal et al. [5]. They found that general perturbation methods do not consider the underlying knowledge an adversary might have or what an adversary might be able to infer.

This metric for establishing privacy involves calibrating noise according to the sensitivity of the distribution to an individual record. With more massive data sets, each individual will contribute less and less to the statistical distribution that will be published. Therefore the process can add less noise. However, in the case of a rare condition which is more common in minority populations, preserving correlations would be difficult since the instances of these rare conditions would contribute more to the statistical distribution. Therefore, the process must add more noise to ensure that these individuals do not uniquely contribute to a particular distribution. Thus, this process would result in fairness-bias against minority populations with rare conditions.

This section covered the varied effects that three fundamental approaches to de-identification have on fairness-bias. Each of these approaches also has multiple implementations, which vary in the way in which they perform operations. Additionally, various algorithms use heuristics to guide these algorithms; they can use many different metrics to determine these heuristics. These factors and the various privacy levels required interact to make assessing bias a complex problem.

2.5 Bias Mitigation Strategies

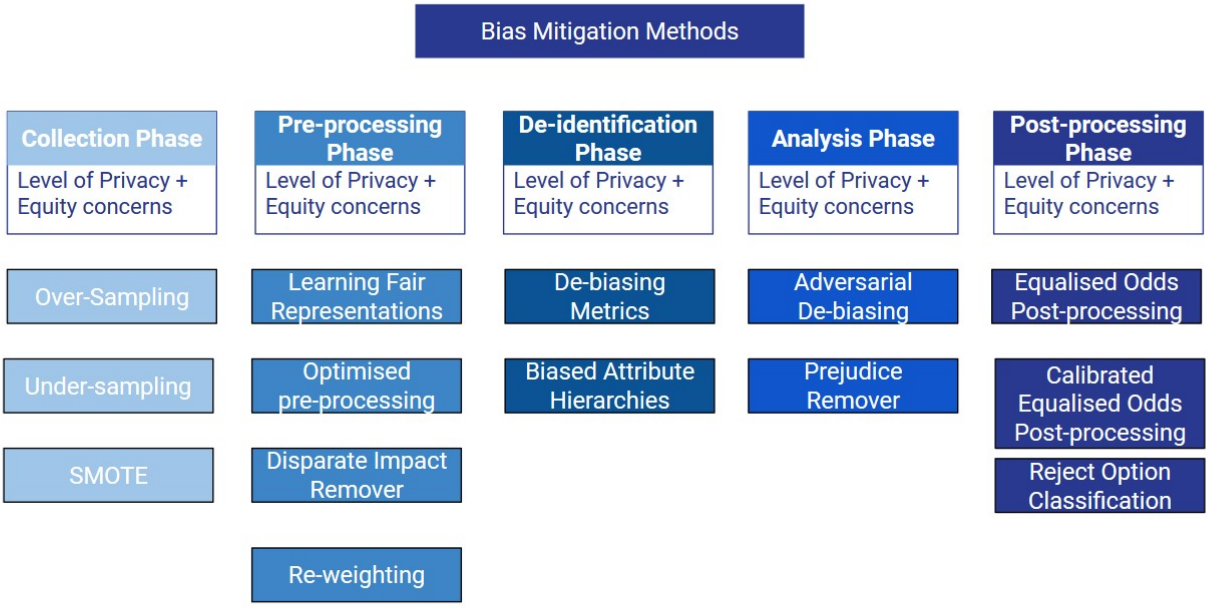

Bias Mitigation Methods
Collection Phase
Level of Privacy + Equity concerns
Pre-processing Phase
Level of Privacy + Equity concerns
De-identification Phase
Level of Privacy + Equity concerns
Analysis Phase
Level of Privacy + Equity concerns
Post-processing Phase
Level of Privacy + Equity concerns
Over-Sampling
Under-sampling
SMOTE
Learning Fair Representations
Optimised pre-processing
Disparate Impact Remover
Re-weighting
De-biasing Metrics
Biased Attribute Hierarchies
Adversarial De-biasing
Prejudice Remover
Equalised Odds Post-processing
Calibrated Equalised Odds Post-processing
Reject Option Classification

Figure 2.2: Bias mitigation techniques by phase.

The methods for bias mitigation are best explained based on the phases where bias arises. This discussion splits this into five sections: the collection phase, the pre-processing phase, the de-identification phase, the processing/analysis phase, and the post-processing phase.

2.5.1 Mitigating Bias in Collection Phase

In the collection phase, the main concern is class imbalance. Two general techniques address this issue: under-sampling and over-sampling. Under-sampling can then be divided into two methods again. This sampling is achieved by preferentially selecting individuals in the minority class over the majority class. Alternatively, replicating instances from the minority class which generates synthetic data. The former method has the benefit of maintaining only real instances, yet only works when sufficient examples of the minority class exist [58]. The latter method, which includes Synthetic Minority Over Sampling (SMOTE) works even with small sample sizes but involves duplicating examples or creating synthetic examples which are not real individuals. This synthesised data can pose challenges in some research where records must represent actual people [33].

Alternatively, under-sampling involves removing majority class examples to balance out the proportions in the collected data. This procedure is similar to algorithmic oversampling, where it is effective only if sufficient examples of minority and majority classes exist. It only alters the distributions and does not require synthetic data; thus, it retains only real examples [58]. However, these results in regards to mitigation performance are not universally effective to all methods of analysis. Japkowicz et al. [47] found that oversampling was more effective than under-sampling for C5.0, but the over-or-under-sampling were equally effective for MLP and SVM techniques. Additionally, general methods for improving model performance, like feature selection, can also help to mitigate the effects of class imbalance [58].

2.5.2 Mitigating Bias in Pre-Processing

There are four main techniques used in the pre-processing step [13]. First, learning fair representations [85] the user defines the protected attributes. The process involves encoding the data in a way that obfuscates the protected attribute while still creating a suitable encoding. Second, optimised pre-processing [16] can take in metrics like group fairness, and individual distortion and create a probabilistic transformation that modifies the data to be more fair based on the metrics chosen. Third, disparate impact remover [29] is similar in that it modifies the feature values, but it maintains a rank-ordering of the attributes. Fourth, re-weighting [49] assigns different weights to attributes to achieve fairness.

2.5.3 Mitigating Bias in the De-identification Process

Fung et al. [33] mention there are examples of instances in the de-identification process where generalisation can improve the accuracy of a model by generalising away some noise. The example given is where the suppression of <birth-date> and replacement with just <age> can reduce the noise in the data. Given a sufficient understanding of the dataset would enable the data publisher to utilise an attribute hierarchy to generalise the more noisy attributes first. Additionally, this attribute hierarchy provides a method to address fairness-bias issues as well by preferentially generalising majority classes first. However, this might conflict with generalising the noisiest first.

The discussion on fairness Section 2.4 also mentioned that many de-identification processes could utilise heuristics to guide the algorithms. Heuristics provide another method similar to attribute hierarchy to determine the order for attribute generalisation. A biased metric could be used to mitigate the inherent bias that exists in a given dataset. However, this method could conflict with utility in some cases when the metric that favours fairness does not favour overall utility.

2.5.4 Mitigation Bias in Processing/Analysis Phase

In the analysis phase, a broad array of methods can be used to mitigate bias. In regards to class imbalance, the field of imbalanced class learning includes methods such as cost-sensitive learning, hybrid and ensemble approaches. Cost-sensitive approaches use a method of assigning different weights to the event of misclassification on the class of interest. This method can also be used as part of an ensemble or hybrid approach which combines multiple classifiers built on the one data set [58]. Examples of these ensemble approaches include bagging and boosting. Bagging creates multiple learners based on different sample feature groupings; then they create a complete classifier. Alternatively, boosting uses multiple training sample sets and iteratively chooses different weights for the classifiers. The user can combined these methods with data sampling methods, and tailor these hybrid approaches to improve the classification for minority classes [58].

Recent work by Krasanakis et al. [56] establishes a framework of types of unfairness and set up metrics by which to assess this bias and apply this framework to the process of adaptive sensitive re-weighting. This approach is similar to the hybrid and ensemble approaches just mentioned. However, instead of determining the biases that need addressing on the sample proportions, it then uses the derived framework to guide the interactive re-weighting approach.

There have been approaches using re-weighting of training samples and recently approaches using adversarial learning methods when addressing misclassificaiton. Adversarial learning has become a popular field of research and has also been applied to bias mitigation [86, 81]. These approaches address bias in models where there is a known or suspected bias against a particular attribute. The model predictor is trained to learn the target attribute while the adversary is trained to learn the protected attribute, which has the known or suspected bias. The objective here is to maximise the predictor's ability to determine the target attribute and minimise adversary's ability to predict the protected attribute. Zhang [86] found this approach produced a provably less biased model and still maintained effectiveness on the primary prediction task. Additionally, this work covered both simplistic and complex models and found that the simplistic adversary was effective in both scenarios.

2.5.5 Mitigation in Post-Processing

There are three primary post-processing mitigation techniques: equalised odds post-processing [36], calibrated equalised odds post-processing [68] and reject option classification. Equalised odds post-processing makes the probability of a positive outcome fairer. This process adjusts labels on the output data using derived probabilities to determine the changes. Calibrated equalised odds addresses the tension between fairness and utility. This method relaxes the conditions of fairness, only addressing one metric at a time, for instance, creating equal true positives or equal true negatives. Reject option classification addresses the distribution of the target attribute on the protected attribute. It then adds favourable outcomes (e.g. credit approval) to the disadvantaged class and unfavourable outcomes (e.g. credit denial) to the advantaged class, for example, within a band around the decision boundary.

2.5.6 Bias Detection and Mitigation Tool-kit

Many of the current tool-kits deal only with bias detection and provide no techniques for mitigating such bias. In comparison, other tool-kits address both bias detection and bias mitigation. This section discusses both types of tool-kits.

Bias Detection Tool-kit

First, this section will discuss some well-known bias detection mechanisms. FairML [2] is as an auditing tool for predictive models by quantifying the relative effects of various inputs on a model's predictions. FairTest [77], on the other hand, approaches the task of detecting biases in a dataset by checking for associations between predicted labels and protected attributes. This methodology also provides a way to identify the input space regions where an algorithm might incur unusually high errors. Aequitas [71] is an open-source bias and fairness audit tool-kit. This tool-kit enables users to seamlessly test models for several bias and fairness metrics on multiple population sub-groups. Aequitas looks for biased actions or outcomes that are based on false or skewed assumptions about various demographic groups.

Bias detection and bias mitigation tool-kit

Some tool-kits can address both bias detection and bias mitigation. Themis-ML [10] is a repository that provides a few fairness metrics, such as mean difference, and some bias mitigation algorithms [49, 50]. Fairness Comparison [31] is one of the more extensive libraries. This tool-kit includes several bias detection metrics and bias mitigation methods [29, 51, 15]. Fairness comparison is a test-bed to allow different bias metrics and algorithms to be compared in a consistent way; it also allows the addition of additional algorithms and datasets. AI Fairness 360 (AIF360) [13] is a tool-kit designed to help facilitate the transition of fairness research algorithms for use in an industrial setting and to provide a common framework for fairness researchers to share and evaluate algorithms. The AIF360 Python package implements techniques from 8 published papers from the broader algorithm community. This package includes over 71 bias detection metrics and 9 bias mitigation algorithms.

2.6 Datasets and Metrics

This section covers aspects of the experimentation used throughout the other chapters. First, we detail the datasets on which the experiments are performed. Second, we discuss the various metrics used in the course of our experimentation and analysis.

2.6.1 Datasets

Adult [54]

The (Adult) Census Income dataset was extracted from the 1994 US Census. There are 48842 records, some values are missing, and with removal of records with missing values there are 45222 records. There are 14 attributes. The attribute chosen for fairness comparison was gender. The privileged class is the (male) values which make up 67% of the records the other 33% are (female) values the unprivileged class. The imbalance ratio is 2 : 1 for the favoured class. The target for classification tasks is a binary value indicating whether the individual makes over $50,000 per year. Over $50,000 is the positive outcome and makes up 24% of the records. Under $50,000 is the negative outcome and makes up 76% of the records. We use processed version of this dataset for all of our experiments, which employs four of the attributes along with the target. The four attributes are:

- Age: continuous variable between 16 and 100. This was binned to 10 year brackets: eg. (0-9) , (20-29).
- Gender: binary variable (male) or (female).
- Ethnicity: categorical variable with five value, binarised to (white), (non-white).
- Education-num: continuous, trimmed to 6, . . . , 12 and greater than 12 or less than 6.
- Income-per-year (target): this is a binarised values indicating whether the individual makes more than $50,000 pre year.

German [25]

The Statlog (German) Credit Scoring dataset is commonly used for tests involving imbalanced class learning. There are 1000 credit application records with no missing values. There are 20 attributes. The attribute chosen for the fairness comparison was gender. The privileged class is the (male) value and makes up 69% of the records. The unprivileged class is the (female) value and makes up 31% of the records creating an imbalance of close to 2 : 1 for the favoured class. The target of the binary classification task is to determine whether the person is worthy of credit. The positive outcome is an approval of credit, making up 70% of the target outcomes. The negative outcome is a decline of credit, making up 30% of the target outcomes. All of our experiments use the same pre-processing as in AIF360 [12]. Five attributes are used:

- Age: a continuous variable binarised to (under 25) or (25 and over).
- Gender: a binary variable male or female.
- Credit history: a binary variable either delayed payments or no history/paid debts.
- Savings: a categorical variable (greater than 500), (500 or less), or (unknown).
- Employment: a categorical variable (4 years or less), (more than 4 years)
- Credit (target): binary variable either credit (approval) or (denial).

COMPAS [6]

The Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) two year recidivism scores is a popular dataset for bias and fairness in machine learning research. This dataset contains the criminal history, jail and prison time and demographics and a two year follow up of whether the defendant re-offended. There are 7214 records. There are missing values, with incomplete records removed, there were 6167 records. There are 52 attributes. The attribute chosen for the fairness comparison was ethnicity. The privileged class is the (white) value and makes up 40% of the records. The unprivileged class is the (non-white) value and makes up 60% of the records creating an imbalance of 2 : 3 for the favoured class. The target of the binary classification task is to predict whether an inmate is going to re-offend. The positive outcome negative prediction making up 53% of the predictions. The negative outcome is a positive prediction to re-offend. These make up 47% of the predictions. All of our experiments use the same pre-processing as in AIF360 [12]. We utilise 5 of the 52 features:

- Age: a continuous variable. This variable was binned to three values (<25), (25-45) and (>45).

- Gender: a binary variable (male) or (female).
- Ethnicity: a categorical variable with five values, binarised to (white), (non-white).
- Number of priors: a categorical variable (0), (1-3), (>3)
- Charge degree: a binary variable (felony) or (misdemeanour)
- Recidivism (target): a binary variable of whether the defendant re-offended

MEPS [25]

3 sets The Medical Expenditure Panel Survey (MEPS) dataset is used for a real-world medical data case study. The 2015 file contains data from rounds 3,4,5 of panel 19 (2014) and rounds 1,2,3 of panel 20 (2015). The 2016 file contains data from rounds 3,4,5 of panel 20 (2015) and rounds 1,2,3 of panel 21 (2016). This dataset contains 15830 patient records from hospital stays with no missing values. There are 139 attributes in the pre-processed data with some values already onehot encoded. The attribute chosen for the fairness comparison was ethnicity which is a composite attribute. The privileged class is the (white) value which indicates white and non-Hispanic and makes up 36% of the records. The unprivileged class is the (non-white) value and makes up 64% of the records creating an imbalance of close to 1 : 2 for the favoured class. The target of the binary classification task is to predict whether a patient will have high healthcare utilisation which indicates any type of treatment provided, including office visits, outpatient visits, emergency room visits, inpatient nights, and home health visits. This value has an average of roughly 10 for the surveyed population. The positive outcome is a negative prediction for high utilisation, and these make up 87% of predictions. The negative outcome is a positive prediction for high utilisation. These predictions make up 17% of the predictions. All of our experiments use the same pre-processing as in AIF360 [12]. We utilise all attributes in this dataset to mimic a real-world scenario. These attributes are onehot encoded and represent two aspects, first the household portion (demographic attributes) including:

- Age: a continuous variable measured in years.
- Gender: a binary variable (male) or (female).
- Ethnicity: a binary variable (white), (non-white).
- Region: a categorical variable with four values indicating in which geographic region the patient lives.
- Marry: a categorical variable indicating the patient's marital status.

- Hon-discharge: a categorical variable indicating the circumstance of the patient's discharge from the military.

- Employment: a categorical variable indicating the patient's employment status.

- Utilisation (target): a continuous variable made from a composite of healthcare utilisation values.

Additionally, the insurance portion of the data details the presence or degree of medical conditions such as diabetes, asthma, and mental health etc. This is a comprehensive list of medical conditions and degrees of symptoms.

2.6.2 Metrics

Here we detail all standard metrics used in all chapters. We utilise three accuracy metrics:

1. Accuracy: simply the percent of labels correctly predicted.

2. Balanced accuracy: 0.5*(True positive rate + True negative rate).

3. True-positive-rate: percent of correct predictions on the positive labels.

We utilised two fairness metrics: disparate impact, and TP-ratio.

1. Disparate Impact: this is a measure of the percent predicted positive of the unprivileged class / the percent predicted positive in the privileged class. Values between $[0, 1]$ indicate the severity of bias towards the privileged group, between $[1, \infty]$ indicate unprivileged group's level of bias.

2. TP-ratio: similar to the Average odds difference but is a ratio difference instead.

Lastly, we employed two efficiency metrics:

1. Percent-positive-outcomes: this is simply the number of positive outcomes / the number of overall predictions. In the context of a medical diagnosis, this metric implies the cost in terms of additional procedures or tests performed.

2. Run-time: is the difference between the cpu clock-time at the beginning and end of all necessary procedures being measured. This was used primarily to compare Mitiband to Random Search and Grid Search.

2.7 Challenges and Opportunities

There are several remaining challenges to be addressed in the bias literature. Among them are:

- Codifying definitions of fairness. There are numerous proposed definitions of what constitute bias and fairness from a machine learning perspective. Because of this, it is nearly impossible to understand how one bias mitigation technique would fare under a different definition of bias. Standardising the ability to mitigate bias in a particular manner remains a complex and open problem.

- From utility to fairness. There is often a trade-off between utility and fairness. Understanding the tension between the two factors is crucial. Operationalising the trade-off between the factors when dealing with de-identification remains an exciting future direction.

- End-to-end bias mitigation tool. A rich area for future research is the development of processes and tools for bias mitigation. Current tools focus heavily on the development of algorithmic methods to assess and mitigate biases in individual ML models. The compounding effects of utilising de-identification alongside an ML model require a rethink of applying bias mitigation techniques more effectively in the broader system design.

- De-identification method guide. In addition to the concept of end-to-end bias mitigation for de-identification, a vital aspect of this could incorporate a method guide that could help determine which de-identification approach and method would be most appropriate. Given the type and nature of the input data and the set of criteria, for example, privacy level and bias mitigation objectives, knowing which techniques are best suited would be beneficial. This guide would create a simplified procedure to guide data publishers in meeting or weighing up varied privacy, utility and fairness objectives.

2.8 Summary

Although the development of big data presents excellent research opportunities, there are limitations to utilising many types of data. De-identification provides a way to address privacy concerns, however, it comes with challenges as well. This survey covered the two major concerns arising from the de-identification process: maintaining utility and ensuring fairness. The first section reviewed the types of error-bias and the issues that lead to fairness-bias. The second and third section catalogued the fundamental types of de-identification—this covered specific methods showing how they lead to error-bias and showed the metrics used to assess utility. The fourth section discussed how these same processes could result in unfair or inequitable outcomes and the metrics used to measure fairness. This section used extensive scenarios utilising different methods. This section showed that the different methods have different levels of fairness-bias depending on the level of privacy, and the correlation between the feature and target attributes. Finally, the fifth section detailed some specific mitigation techniques to deal with bias in the collection, processing and analysis phase.

Whenever there is a choice about how to alter data, there is a potential for bias. De-identification inevitably requires altering the data in some way. Thus, the issues of bias will continue to arise when intending to publish sensitive data. Privacy protection has seen increased concern in the public consciousness, and legislation has begun to address some of these circumstances. Fairness has also become an issue of public concern, and there are beginning to be initiatives to address this issue. Both privacy and fairness are essential issues to address to make better use of the data that is becoming available.

3

Bias Assessment

This chapter explains the bias assessment framework and utilises this to investigate bias on various data. Section 3.1 discusses the motivation for this line of research. Section 3.2 discusses the methodology in developing the framework to assess the effects of de-identification on accuracy and fairness. Section 3.3 outlines the algorithmic procedures in the bias assessment framework. Section 3.4 outlines the datasets, and the experiments. Section 3.5 discusses the results. We summarise this chapter in Section 3.6.

The objective of this chapter was to investigate the effects of de-identification on bias and fairness. We developed a bias assessment framework to analyse the level of fairness and accuracy in original and de-identified data. We utilised this framework to experiments with three real-world datasets as well as artificial data. We employed various metrics and developed two novel measures to compare the results of these tests.

3.1 Motivation

Previous research has analysed the effects of de-identification in the privacy and utility trade-off; this work focused on tuning the noise applied in the differential private algorithm [34]. Our objective is to investigate the privacy fairness trade-off. Additionally, we assess this trade-off on both a differentially private and a generalisation/suppression method.

The contributions of this chapter are three-fold. First, we develop a framework for

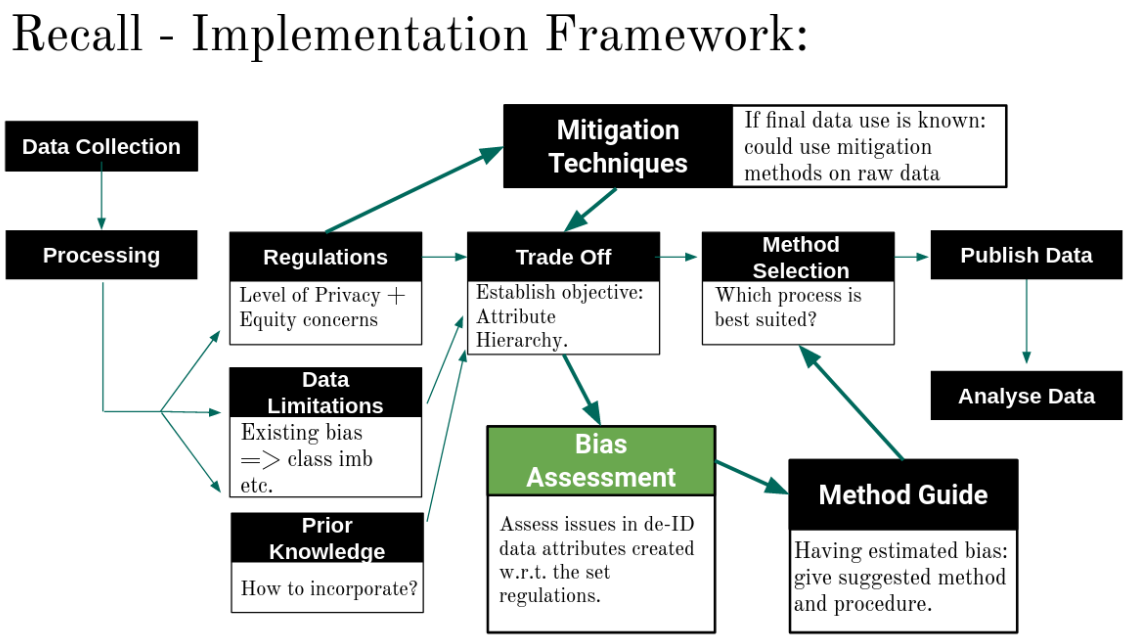

Recall - Implementation Framework:
Data Collection
Processing
Regulations
Level of Privacy + Equity concerns
Data Limitations
Existing bias => class imb etc.
Prior Knowledge
How to incorporate?
Mitigation Techniques
If final data use is known: could use mitigation methods on raw data
Trade Off
Establish objective: Attribute Hierarchy.
Bias Assessment
Assess issues in de-ID data attributes created w.r.t. the set regulations.
Method Selection
Which process is best suited?
Method Guide
Having estimated bias: give suggested method and procedure.
Publish Data
Analyse Data

Figure 3.1: Data analysis pipeline.

analysing the effects of de-identification on bias; we developed two novel measures for this. Second, we investigate how issues that affect the accuracy and fairness of models in general will affect models created from de-identified data. To this end, experiments were run on data with differing levels of: class imbalance, positive target outcomes, and target outcome imbalance. Third, we assess the effects of de-identification on three real-world datasets.

Figure 3.1 displays our data analysis pipeline. We highlight where the bias assessment framework can be used for bias assessment. We place this bias assessment process at a stage where we can use the results of the assessment to guide the following steps.

3.2 Bias Assessment Framework

This section discusses the methodology used to develop the bias assessment framework. Additionally, we outline the assessment measures we employ. Our objective was to measure the fairness and utility loss as a result of the de-identification process.

3.2.1 Methodology

Formally, we start with an initial dataset D. This dataset contains a protected class C and remaining attributes X. There is a binary target Y described in Equation 3.2. The target Y is either a positive outcome $Y^+ \subset Y$ or a negative outcome $Y^- \subset Y$ and the protected class C is either unprivileged $C_0 \subset C$ or privileged $C_1 \subset C$. This framework can be extended trivially to assess bias across multi-variate protected class $(C_1, \ldots, C_n)$ and can be extended to include a continuous target, provided there is a user-defined threshold τ such that: $\forall\ y \geq \tau : y \in Y^+$.

$$D = (C, X, Y) \tag{3.1}$$

The original dataset might have some level of bias against the unprivileged class, for instance:

$$bias(D) = \frac{Pr(y \in Y^+ | c \in C_0)}{Pr(y \in Y^+ | c \in C_1)} \leq 1 \tag{3.2}$$

In the de-identification pre-processing framework the data is then de-identified. This step is skipped when performing in-processing de-identification. This step creates a new dataset $\bar{D}$ which contains a transformation of the attributes $\bar{X}$ and for the purposes of analysis must contain the original attribute C. Depending on the de-identification methodology the target may remain unaltered leaving Y or it may result in altered $\bar{Y}$:

$$\bar{D} = (C, \bar{X}, \bar{Y}) \tag{3.3}$$

If Y is transformed to $\bar{Y}$ this may lead to a different level of bias defined by:

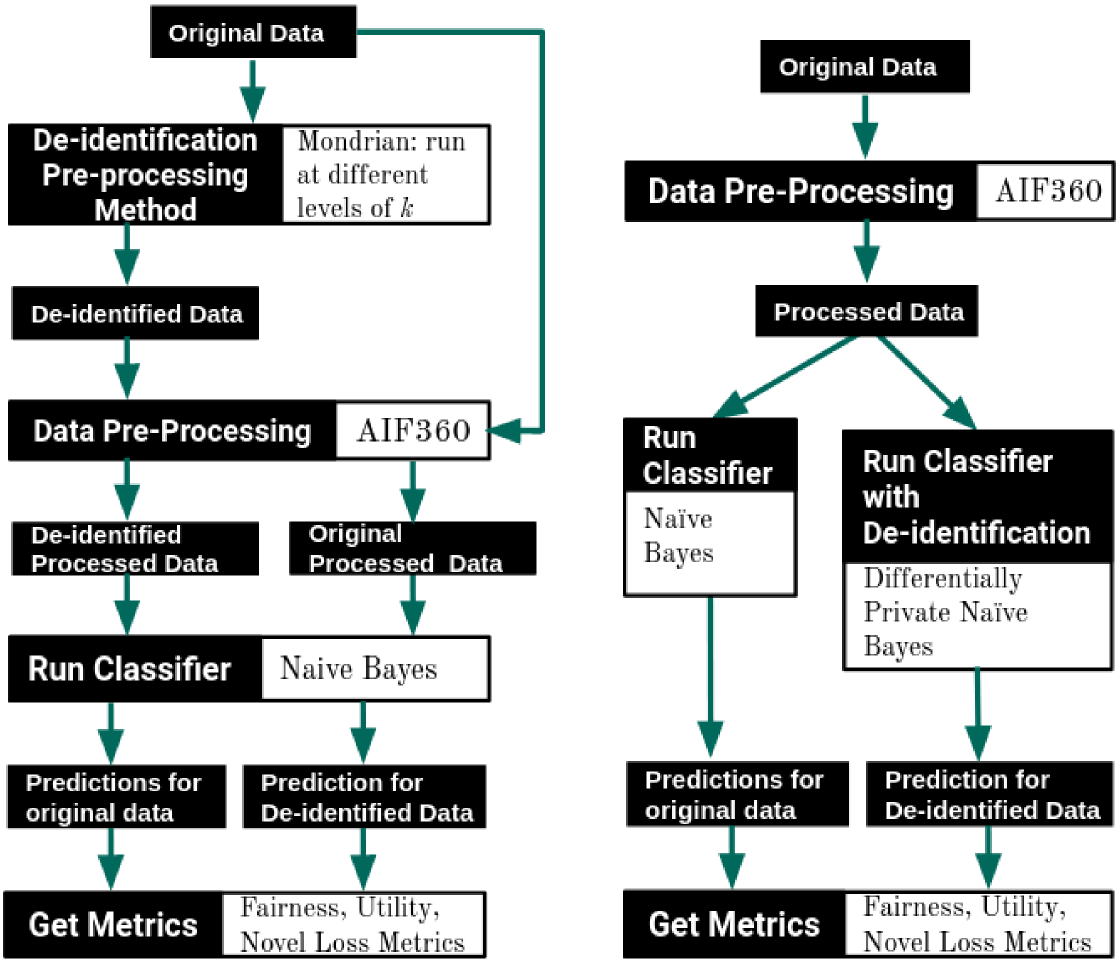
$$bias(\bar{D}) \in \frac{Pr(\bar{y} \in \bar{Y}^+ | \bar{c} \in \bar{C}_0)}{Pr(\bar{y} \in \bar{Y}^+ | \bar{c} \in \bar{C}_1)} \tag{3.4}$$

The difference between $bias(D)$ and $bias(\bar{D})$ is the original change in fairness which may increase or decrease the bias. However, The change in fairness in the training data will also affect models created using the new data. Thus, this framework applies pre-processing and model creation to assess this change. The data pre-processing is performed to prepare it for creating a model. The pre-processing is applied to both the original and de-identified data according to the users specifications; we model ours as in AIF360 [12].

In the pre-processing de-identification framework Figure 3.2(a) two models are created using a classifier. One is trained on the processed original data, and the other is trained on the processed de-identified data. Using the in-processing de-identification framework Figure 3.2(b) two models are created using a classifier. Both models are trained on the processed original data. However, one is trained using a standard classification model, while the other is trained using the private classification model. In our case, we use a differentially private classifier. Both the pre-processing and in-processing de-identification frameworks create two models, an original model M and a de-identified model $\bar{M}$.

The original and de-identified classifiers are then tested using a test set from either the original or de-identified data respectively. This produces two sets of predictions P with $(P^+, P^- \subset P)$ and P^* with $(P^{*+}, P^{*-} \subset P^*)$.

We can apply extensive metrics to these results. We first measure accuracy described


Original Data
De-identification Pre-processing Method
Mondrian: run at different levels of k
De-identified Data
Data Pre-Processing
AIF360
De-identified Processed Data
Original Processed Data
Run Classifier
Naive Bayes
Predictions for original data
Prediction for De-identified Data
Get Metrics
Fairness, Utility, Novel Loss Metrics
Original Data
Data Pre-Processing
AIF360
Processed Data
Run Classifier
Naïve Bayes
Run Classifier with De-identification
Differentially Private Naïve Bayes
Predictions for original data
Prediction for De-identified Data
Get Metrics
Fairness, Utility, Novel Loss Metrics

(a) Pre-Processing De-identification (b) In-Processing De-identification

Figure 3.2: Bias assessment framework for pre-processing and in-processing de-identification methodologies.

in Chapter 2. We also develop two novel measures to assess privacy fairness loss described in Section 3.2.2, and privacy utility loss described in Section 3.2.3.

3.2.2 Privacy-Fairness Loss Measure

The set of equations from Equation 3.5 to Equation 3.7 measures the increase in bias that arises as a result of the de-identification process. First, we assess the predictions from the original processed dataset with predictions P. We measure ρ, which is the percentage of outcomes predicted as positive. Then β is the percentage of positive outcomes in the ground truth. Therefore, δ gives the ratio of positive outcomes lost between the predicted and the actual positive outcomes.

$$\delta = \rho/\beta : \quad \rho = \frac{|P^+|}{|P|} : \forall p \in P, c \in C, \quad \beta = \frac{|Y^+|}{|Y|} : \forall p \in P, c \in C \tag{3.5}$$

We can then measure the δ_0 for the unprivileged class and δ_1 for the privileged class.

The difference between these Δ is the fairness lost on that set of predictions.

$$\Delta^P = \delta_0 - \delta_1 \mid \delta_0 \subset \delta : \ c \in C_0, \ \ \delta_1 \subset \delta : \ c \in C_1 \tag{3.6}$$

This same formula is then applied to the de-identified processed dataset using predictions P^* to derive Δ^{P^*}. These values can be compared directly as in the plots in our experiments or we can derive a single privacy-fairness-loss value:

$$\text{privacy-fairness-loss} = \Delta^{P^*} - \Delta^P \tag{3.7}$$

For example, the test set could have a rate of 40% positive outcome on the privileged class and 20% positive outcome on the unprivileged class. However, the predicted positive rate for the privileged class is 30% and the predicted positive rate for the unprivileged is 5%. Thus, the percent of positives predicted for the original data is:

$$\Delta^P = \delta_0 - \delta_1 = -0.5 : \quad \delta_0 = \frac{\rho_0}{\beta_0} = \frac{0.05}{0.20} = 0.25, \quad \delta_1 = \frac{\rho_1}{\beta_1} = \frac{0.30}{0.40} = 0.75. \tag{3.8}$$

where as the percent positives predicted for the de-identified data is:

$$\Delta^{P*} = \delta_{0*} - \delta_{1*} = -0.46 : \quad \delta_0 = \frac{\rho_{0*}}{\beta_{0*}} = \frac{0.03}{0.15} = 0.20, \quad \delta_1 = \frac{\rho_{1*}}{\beta_{1*}} = \frac{0.20}{0.30} = 0.66. \tag{3.9}$$

Thus, privacy-fairness-loss $= -0.46 - (-0.5) = +0.04$ and the fairness difference has reduced slightly from the original predictions to the de-identified predictions. However, the overall positive predictions are lower for both the privileged and unprivileged classes.

3.2.3 Privacy-Utility Loss Measure

To determine the specific effect that increased privacy has on overall utility, we use accuracy, and measure the relative error rate. We measured the loss to account for the loss which arises as the privacy increases. This metric is measured as area between the overall accuracy on the original data and accuracy of de-identified data, denoted as Privacy-Utility loss.

$$a_{overall} - a_{deid} \tag{3.10}$$

where $a_{overall}$ is the accuracy before de-identification and a_{deid} is the accuracy after de-identification process. If it is 0, then there is no loss in the de-identification process. We applied this measure to both the privileged and unprivileged classes to indicate the differential effects.

3.3 Algorithm Discussion

This section first discusses the methodology of the algorithmic processes used in our algorithm. Then we discuss Algorithm 1 which displays the full process and outlines the order of procedures in this implementation. The order and inclusion of certain procedures varies depending on the specific use case.

We built upon methods in the AIF360 package and developed a framework to assess the level of bias arising from the de-identification process. Additionally, we assessed various de-identification methods and selected Mondrian [59] and IBM Differential Privacy [40] for testing. We applied de-identification in two manners. With the Mondrian method, the data set is de-identified at the beginning of the bias assessment framework. Alternatively, with the IBM differential privacy, the de-identification process is part of the classifier and is applied during the classification stage of the bias assessment framework.

This framework employs five standard packages. These were Numpy [37], Pandas [83], Scikit-learn [67], Sci-py [79], and Matplotlib [41]. The framework utilises a series of procedures displayed in two stages and the overall process is described in Algorithm 1. Additionally, the privileged and unprivileged classes values and the positive and negative label values are inputs.

We use three generic procedures in the framework: re-run *get_dataframes* once using arguments *orig_loc* to get the original dataframe df_deid_0 zero indication no de-identification. Then we run the procedure k times using $deid_k\_loc$ to get df_deid_k for each de-identification level in K. Additionally, the argument $feature\_names$ defines the features to keep for all dataframes. These dataframes are added to all_dfs.

Next we iterate though all $df_k \in all\_dfs$. For each df, the *split_data* procedure splits the data into train and test sets using the Sklearn split function. The size of the test set is $test\_size$. The $target$ argument is used to ensure the proportions of the target in the training and test set are close.

The procedure *scale_numeric* scales the numeric data for the training and test sets using either the StandardScaler or MinMaxScaler. This function takes in the dataframe df, then scales the numeric attributes *numeric* using the scaler type $scaler$. The *deid* variable is a flag to run the process for de-identified data, and if there is a hierarchy key *hk* this can be applied. This procedure produces and produces X_train_scaled and X_test_scaled.

Then we employ a custom procedure for the categorical attributes of the scaled training and testing data X_train_scaled and X_test_scaled. The procedure *onehot_encode* uses Pandas performs the pre-processing on categorical attributes *categorical* of the dataframe df. This process includes label binarisation, feature selection, and one hot encoding. We performed this process in the manner described in AIF360 [12]. However, de-identified data is flagged with *deid*. If the values have no hierarchy key *hk*, and are generalised into a selection of possible classes, *e.g.*, (a, b, d), a novel multi-hot encoding is performed to assign a 1 to each column for a, b, and d. This procedure produces *X_train_enc* and *X_test_enc*.

Following this, the classification task is performed using a $classifier$ on processed data. This process trains a model with the training set *X_train_enc* and *y_train*. The model is tested with *x_test_enc* and creates predictions *df_pred*. This framework uses Sklearn. Thus, any classifier in Sklearn can be used to compare the original and pre-processed de-identified dataset. However, with model-based de-identification (IBM differential privacy), for comparison on the same model, the framework is limited to those models design for de-identification methods.

Then *get_metrics* takes in the dataframe with prediction *df_pred* and the set of protected attributes *protect_dict* and the set of metrics met_dict. This procedure runs through all the protected attributes and the metrics, then returns a dictionary, $results_k$, of all values of the metrics. This dictionary is finally added to the set of dictionaries, *results*, for all de-identification levels.

Following this the *plot_metrics* procedure can be used to plot the results using the Matplotlib library. This procedure takes in the dictionary of results from the original data *results*, the dictionary of protected attributes *protect_dict*, and the dictionary of metrics *met_dict*. The procedure plots all the metrics for all the protected attributes of the original and k de-identified data so that the user can compare these results.

3.4 Experiments

In this section, we discuss the datasets used and the experimental setup. The objective of experiments is two-fold. First, we explored the effects of data characteristics on bias in de-identified data. This included class imbalance, the ratio of positive outcomes for both the privileged, and ratio of positive outcomes for the unprivileged classes. Second, we investigated the effects of de-identification on bias in real-world datasets. To assess the effects of in these experiments, we used four metrics accuracy, disparate impact as described in Chapter 2, and the two novel measures described in Section 3.2.

3.4.1 Datasets

The datasets we used for these experiments were the **Adult** dataset and the **German** dataset described in Chapter 2. Additionally, for the medical data case study we utilised the Medical Expenditure Panel Survey (**MEPS)** dataset described in Chapter 2.

To investigate the effects of imbalanced classes, we designed a synthetic dataset gen-

Algorithm 1 Bias assessment algorithm

```
Input: orig_loc - file location of original data, $deid_k$_loc - file location of de-identified data, features - features to keep, target - target attribute, $C_1$, $C_0$, $Y^+$, $Y^-$, test_size - size of test set, numeric - numeric attributes, scaler - type of scaler, categorical - categorical attributes, classifier - type of classifier, protect_dict - dictionary of protected classes, met_dict - dictionary of metrics
Output: dictionary of metrics (plot)
df_deid$_0$ ← get_dataframes(df_orig_location, features) all
all_dfs.add(df_deid$_0$)
for k ∈ K do
    df_deid$_k$ ← get_dataframes(deid$_k$_loc, features)
    all_dfs.add(df_deid$_k$)
end for
for df$_k$ ∈ all_dfs: $k$ = de-identification level do
    X_train, X_test, y_train, y_test ← split_data((df$_k$, test_size)
    X_train_scaled ← scale_numeric(X_train, numeric, scaler, is_deid, has_hk)
    X_train_enc ← onehot_encode(X_train_scaled, categorical, is_deid, has_hk)
    X_test_scaled ← scale_numeric(X_test, numeric, scaler, is_deid, has_hk)
    X_test_enc ← onehot_encode(X_test_scaled, categorical, is_deid, has_hk)
    df_pred ← run_classifier(X_train_enc, X_test_enc, y_train, y_test, target, classifier)
    results$_k$ ← get_metrics(df_pred, protect_dict, met_dict)
    results.add(results$_k$)
end for
plot ← plot_metrics(results, protect_dict, met_dict
```

erator to manipulate the Adult dataset. **Synthetic Data Generator.** The synthetic data generator creates datasets through the selective sampling of the Adult dataset. The samples represent different class imbalance levels between the privileged and unprivileged class and different levels of fairness. This distribution is achieved by selectively sampling the positive outcome percentage for the privileged and unprivileged classes. This process allowed us to control the dataset generated and systematically investigate the effects of class imbalance and pre-existing disparities in positive outcomes. In the dataset generated, we sub-sample 8,000 rows from the original Adult dataset. For class imbalance, we started at no imbalance $1:1$, up to $1:100$. Our positive outcome ratio is set between 50% to 1%.

3.4.2 Experimental Setup

Both experimental procedures use the Mondrian k-anonymity algorithm [59] and the IBM differential privacy approach [26, 40], as the underlying de-identification techniques.

Setup using k-anonymity. The Mondrian k-anonymity method [59] is a top-down approach that iteratively partitions the dataset into smaller partitions provided they are larger than size k. This method can accept a hierarchy key that determines the broader categories into which the values are generalised. For example, the attribute values *unmarried* and *divorced* could both be mapped to unmarried. No hierarchy key was used for these tests as there is no defined hierarchy prescribed for these datasets. The result of the generalisation process is creating a partition in which each attribute can be any value within a given range or set. All possible values were then multi-hot encoded for each member of that partition.

We used this pre-processing procedure on the original dataset as well. Then, to determine the effects of k-anonymity and model the relationship between the de-identification level with accuracy and fairness, we performed the Mondrian algorithm choosing $k = 4$, $k = 8$, and $k = 16$. This set gives a range for possible privacy levels required. We trained our models on each of these datasets, using Gaussian naïve Bayes. Initially, we evaluated it against synthetic datasets with varying imbalance ratio and positive outcome ratios. We then evaluated the model's performance on the real-world data and in the medical data case study.

Setup using Differential Privacy. To assess the differential privacy method, we used the method devised in [40]. This is an open-source python package. We employed the differentially private naïve Bayes function to compare these results against those of the same classifier used on the k-anonymity private models. We first performed the pre-processing procedures on the original dataset and then chose ϵ at $0.1, 1.0, 5.0, 10$. The selection of ϵ is an open area of research [55], thus, we selected the set to evaluate the effects given several possible levels of de-identification.

3.5 Results

We discuss the results in two sections based on the research aims. First, we investigated the affects of data characteristics on bias levels in de-identified data. Second, we investigated the effects of de-identification on bias using real-world datasets.

3.5.1 Effects of Data Characteristics on de-identification.

Figure 3.3 shows the results of using k-anonymity to de-identify data and produce a naïve Bayes model. Figure 3.4 shows the results of using a differential privacy method to de-identify the data and produce a naïve Bayes model. In the figures first column, we show the accuracy and disparate impact of the results based on imbalance ratio, privileged positive, and privileged class. The graphs in the second column of the figure show the privacy-utility loss, and the third column shows the privacy-fairness loss. The standard deviations of the results are reasonably small. For readability purposes, we have not included it in the graphs below. We provide the full results along with standard deviations and significance test in the following link: https://github.com/andrewj-rc/Balancing-Utility-and-Fairness-Against-Privacy-in-Medical-Data.

From Figure 3.3 we observed that when using k-anonymity, the accuracy deteriorates with higher k and that the effects are more pronounced on the unprivileged class. This observation is true for higher imbalance, as well as lower unprivileged and privileged positive outcomes. We observed that the utility loss is higher for the unprivileged than the privileged class for all three parameters. There is a greater differential observed between the privileged and unprivileged classes with lower positive outcomes for unprivileged and privileged with this data. On the fairness measure we observed a high prediction rate for positive outcomes for both privileged and unprivileged. This tendency decreases with higher class imbalance and higher unprivileged and privileged positive outcomes.

Figure 3.4 shows that using differential privacy, the accuracy for the privileged class has a decreasing trend. Whereas the unprivileged class has an increasing trend with a higher imbalance ratio. Similarly, we observe that both privileged and unprivileged accuracy tend to decrease with a high percent of privileged and unprivileged positive outcomes. The utility for the privileged class shows a tendency to be higher than that of the unprivileged class regardless of the imbalance or privileged and unprivileged positive percentages. For the fairness measure, we observed that in this data, the unprivileged class has a higher fairness measure than the privileged class.

3.5.2 Effects of de-identification on bias.

Tables 3.1, 3.2, and 3.3 present the results of the test on the real-world datasets. The first column indicates the data on which we tested the model, either original data (not de-identified) or de-identified and the de-identification methodology. The proceeding columns present accuracy, disparate impact, and fairness and utility measures.

Table 3.2 presents the German dataset results. As the de-identification level increases, we observed a loss in accuracy when tested on the original data for both de-identification methodologies. At the same time, the disparate impact tends to increase with higher levels of de-identification for both. With the Fairness Loss metrics for both privileged and unprivileged, initially the rates are positive. However, they both have a decreasing positive rate. For the utility loss with the privileged class, the trend is decreasing for higher k-anon and increasing for lower ϵ. Conversely, with the unprivileged class, the level of loss has an increasing trend for both methods with high k. When testing against de-identified data, we observe an increase in utility with higher k. However, for the unprivileged class, this trend is not seen.

Table 3.1 presents the Adult dataset results. Firstly, assessing the accuracy; for k-anonymity, when using the original test data, we observed a drop in accuracy from the original data at all levels of k. This trend is not seen when tested on the de-identified data.

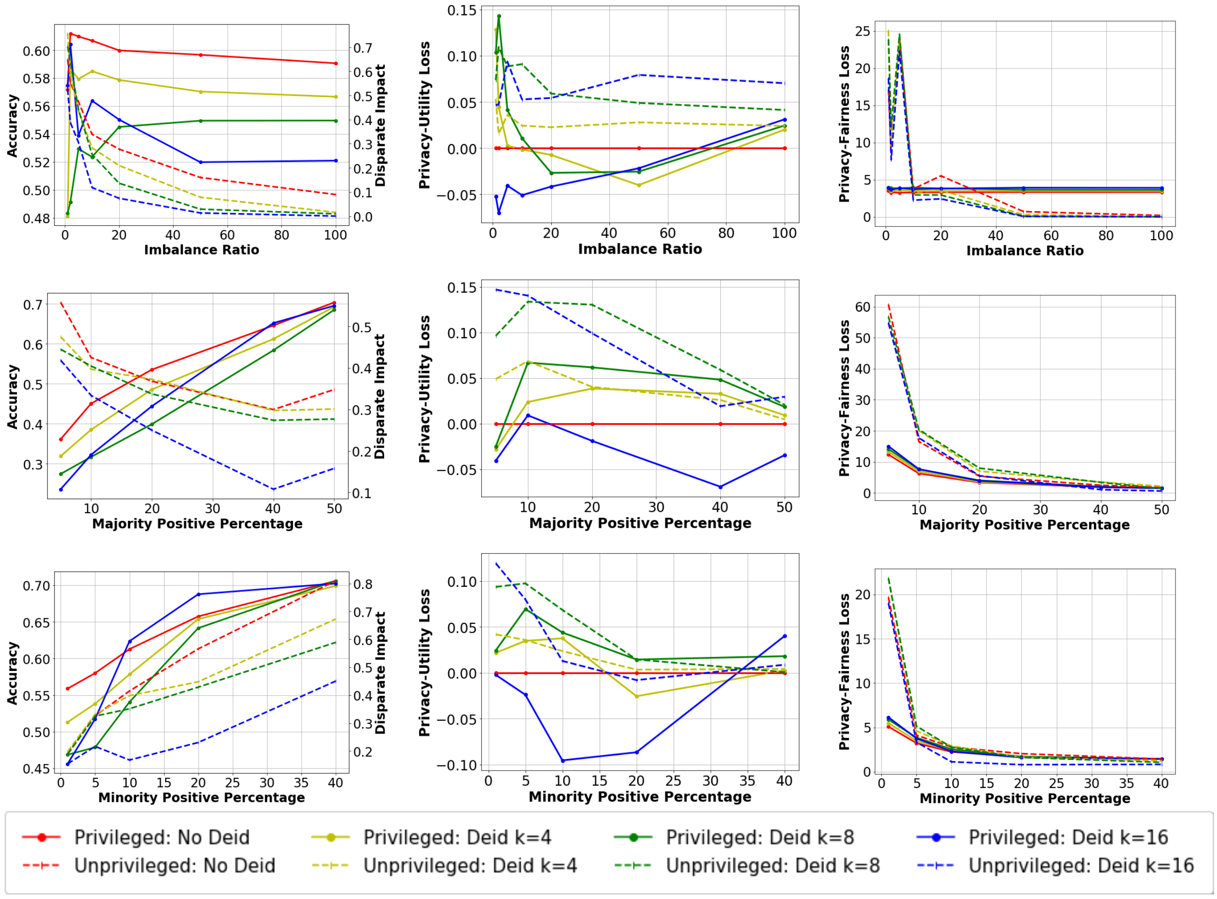


Figure 3.3: Results for naïve Bayes model using k-anonymity.

Additionally, we do not see this trend with the differential privacy method with all levels of ϵ. For disparate impact with k-anonymity using the original test data, we observed an increase with higher k. When testing on the de-identified data, we see the opposite trend. The Fairness measure for privileged and unprivileged classes has much more variability when tested on both the original and de-identified test data. This variability is true for k-anonymity and differential privacy. For k-anonymity, tested on the original data, the unprivileged class tends to have a high fairness measure. We observe the opposite on most cases for the unprivileged class. With utility loss for k-anonymity when tested against the original test data we see there is a loss in accuracy with all levels of k. When tested on the de-identified data we observe the opposite. There is an increase in the utility with higher levels of k using k-anonymity. However, using differential privacy on the de-identified data, results in a marginal decease in utility with lower ϵ.

3.5.3 Medical Data Case Study

This study uses the MEPS HC-181 full year 2015 consolidated data described in Chapter 2. For the QID set, 22 onehot encoded features representing six attributes were selected (Age,

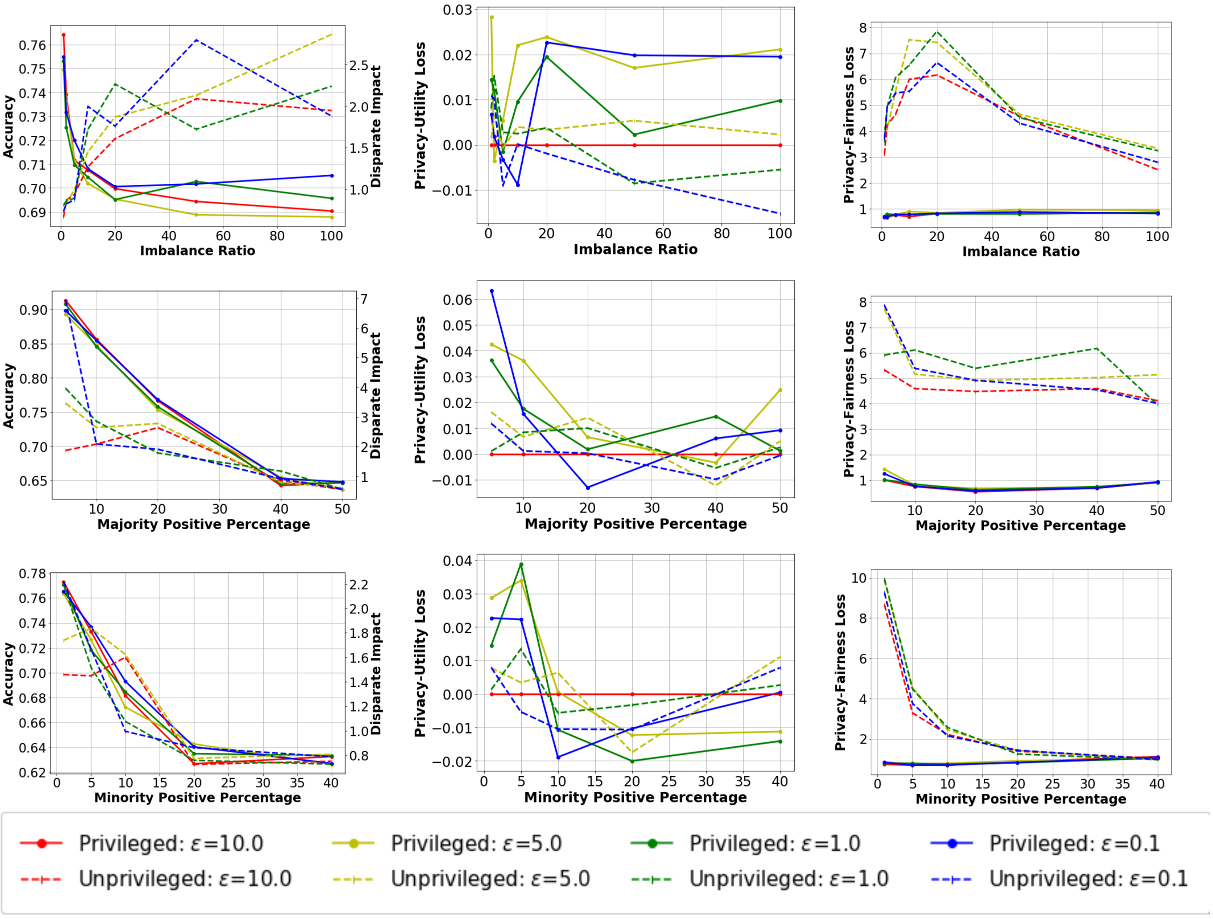


Figure 3.4: Results for naïve Bayes model using differential privacy.

Table 3.1: Results on the Adult Dataset

	De-ID Level	Acc	Std Dev.	Disparate Impact	Fairness Loss Priv	Fairness Loss Unpriv	Utility Loss Priv.	Utility Loss Unpriv
Original	None	0.76	0.02	0.54	0.92	1.25	0.00	0.00
	k=4	0.63	0.06	0.85	1.27	2.54	0.18	0.11
k-anonymity	k=8	0.66	0.10	0.87	0.90	1.89	0.14	0.08
	k=16	0.66	0.08	1.27	0.64	1.94	0.14	0.07
De-ID	None	0.75	0.02	0.58	1.06	1.57	0.00	0.00
	k=4	0.78	0.01	0.41	1.02	1.10	-0.02	-0.03
k-anonymity	k=8	0.77	0.02	0.34	1.08	0.93	-0.02	-0.02
	k=16	0.77	0.01	0.30	1.33	1.03	-0.03	-0.02
Original	ϵ=10	0.73	0.08	0.53	1.05	1.62	0.00	0.00
	ϵ=5	0.76	0.02	0.44	0.82	0.99	-0.05	-0.02
Differential Privacy	ϵ=1.0	0.74	0.11	0.56	0.97	1.53	-0.01	0.00
	ϵ=0.1	0.73	0.11	0.60	1.13	1.94	0.01	0.00
De-ID	ϵ=10	0.76	0.03	0.50	0.80	1.10	0.00	0.00
	ϵ=5	0.76	0.02	0.43	0.98	1.10	0.01	-0.01
Differential Privacy	ϵ=1.0	0.76	0.02	0.43	1.01	1.16	0.00	0.00
	ϵ=0.1	0.72	0.08	0.53	0.97	1.55	0.05	0.03

Race, Region, Marital Status, Student Status, and Military Status). Recall, the task was to determine whether the level of utilisation would be high (≥10).

Table 3.3 presents the results from the MEPS dataset. We observed an increase in accuracy for k-anonymity when tested on the original test data at low level k. This case is true when tested on the de-identified test data as well. When comparing this to the differential privacy method we observe a drop in accuracy with the first level of de-identification. However, the accuracy tends to increase again with higher de-identification

Table 3.2: Results on the German Dataset

	De-ID Level	Acc	Std Dev.	Disparate Impact	Fairness Loss Priv	Fairness Loss Unpriv	Utility Loss Priv.	Utility Loss Unpriv
Original	None	0.65	0.13	0.86	1.18	1.24	0.00	0.00
	k=4	0.61	0.12	0.97	1.03	1.06	0.02	0.04
k-anonymity	k=8	0.53	0.16	0.84	0.86	0.87	-0.03	0.13
	k=16	0.49	0.15	0.61	0.68	0.66	-0.03	0.16
De-ID	None	0.58	0.10	1.04	0.93	1.10	0.00	0.00
	k=4	0.58	0.15	0.82	0.98	1.06	-0.01	-0.02
k-anonymity	k=8	0.60	0.10	0.98	1.04	1.12	-0.01	-0.03
	k=16	0.61	0.11	1.03	1.11	1.32	0.06	-0.05
Original	ϵ=10	0.63	0.08	0.99	1.11	1.19	0.00	0.00
	ϵ=5	0.57	0.16	0.82	0.98	1.03	0.07	0.06
Differential Privacy	ϵ=1.0	0.55	0.16	0.75	0.89	0.88	0.09	0.08
	ϵ=0.1	0.53	0.16	0.71	0.86	0.92	0.08	0.10
De-ID	ϵ=10	0.54	0.17	1.14	0.74	0.93	0.00	0.00
	ϵ=5	0.55	0.16	0.82	0.86	1.02	0.01	-0.02
Differential Privacy	ϵ=1.0	0.57	0.15	0.85	0.85	0.89	0.00	-0.05
	ϵ=0.1	0.63	0.12	0.88	1.09	1.16	-0.02	-0.12

Table 3.3: Results on the MEPS Dataset

	De-ID Level	Acc	Std Dev.	Disparate Impact	Fairness Loss Priv	Fairness Loss Unpriv	Utility Loss Priv.	Utility Loss Unpriv
Original	None	0.59	0.03	0.70	2.60	3.60	0.00	0.00
	k=4	0.76	0.01	0.63	1.75	2.16	-0.16	-0.17
k-anonymity	k=8	0.74	0.01	0.63	1.82	2.29	-0.15	-0.16
	k=16	0.70	0.02	0.68	2.04	2.74	-0.10	-0.12
De-ID	None	0.59	0.03	0.67	2.70	3.60	0.00	0.00
	k=4	0.77	0.01	2.73	1.58	2.45	0.07	-0.28
k-anonymity	k=8	0.77	0.01	2.85	1.59	2.41	0.07	-0.28
	k=16	0.78	0.01	2.97	1.58	2.44	0.09	-0.28
Original	ϵ=10	0.81	0.03	0.68	0.75	1.04	0.00	0.00
	ϵ=5	0.75	0.21	0.68	0.99	1.55	0.06	0.06
Differental Privacy	ϵ=1.0	0.75	0.21	0.75	1.04	1.75	0.07	0.05
	ϵ=0.1	0.82	0.02	0.66	0.71	0.91	-0.01	0.00
De-ID	ϵ=10	0.81	0.03	0.72	0.51	0.69	0.00	0.00
	ϵ=5	0.81	0.07	0.67	0.80	1.12	0.01	0.00
Differental Privacy	ϵ=1.0	0.82	0.02	0.68	0.53	0.70	-0.01	0.00
	ϵ=0.1	0.83	0.03	0.61	0.54	0.71	-0.01	-0.01

levels. The fairness measure indicates that the positive outcomes are predicted at a higher rate when using k-anonymity tested on both the original and de-identified test data. This result is true for both privileged and unprivileged classes. However, when using the differential privacy method, the privileged positive class has a lower prediction rate when tested on original data. Additionally, both the privileged and unprivileged classes have a lower predication rate when tested on the de-identified data. For utility loss, both privileged and unprivileged classes tested on original data have an initial improvement in utility from de-identification. However, this decreases with higher de-identification level. When tested on the de-identified data, the utility loss for the privileged class we observe an increasing trend, while the unprivileged class sees an improvement in utility consistent for all levels of utility. When comparing utility on differential privacy, there is less of a trend for both testing on the original and on the de-identified data. When tested on the original data, there are marginal utility losses. Though when tested on the de-identified data, there is very little loss or gain at any de-identification level.

3.5.4 How to choose a "good" Fairness-Utility balance?

Presume a healthcare company wants to publish their data but must first de-identify the data. They want to publish the data which gives the best accuracy (minimise utility loss) while also preventing discrimination (minimise fairness loss). Taking the results from both the artificial and real-world data episodically, we see that there can exist variation in the tendencies among both the fairness and utility loss. This variation could be a result of numerous data specific characteristics. This variation presents a challenge when trying to build a function for these two measures. Essentially, this is a problem of optimisation, similar to maximising utility and privacy, which is NP-hard [57]. Given a set level of privacy, then a possible naïve approach is to determine a minimally acceptable threshold for either fairness or utility then determine the maximum achievable value for the other parameter. Additionally, hyper-parameter tuning has been applied to differential privacy models for dealing with privacy-utility trade-off [9]. Future work extending this type of research may allow us to better optimise fairness and privacy.

3.6 Summary

The process of de-identification has measurable effects on the relationship between both utility and fairness. We introduced two straightforward measures to quantify the level of fairness loss and the level of utility loss before and after de-identification. Using this process allows us to analyse comparative results between two different de-identification techniques and different privacy levels. We also produced a synthetic dataset generator that will enable us to control the ratio of the classes to simulate different characteristics, which allows us to explore a specific dataset thoroughly. In the experiments, we showed the synthetic data generator applied on the Adult dataset. However, we can extend this generator to other datasets. The synthetic data results gave a clearer understanding of the effects data characteristics have on the de-identification process. The tests on real-world data and the on the medical data case study provided greater insight into the effects the de-identification methods and setup have on the fairness and the utility of models created from such data. Finally, we discussed the issue of determining a balance between privacy, utility, and fairness.

4

Bias Mitigation Assessment

This chapter introduces the mitigation assessment framework and investigates the effects of mitigation on fairness and bias in machine learning. Section 4.1 discusses the motivations for this investigation and the development of this framework. Section 4.2 describes the methodology of the bias mitigation framework. Section 4.3 outlines the algorithmic procedures for this framework. Section 4.4 details the experiments conducted using this framework and Section 4.5 displays the results. Finally, Section 4.6 summarises this chapter.

The objective of this chapter was to investigate the effects of employing bias mitigation to de-identified data. We developed a mitigation assessment framework to analyse these effects. We utilised this framework to assess the impact of mitigation and de-identification on three real-world datasets. We also used this framework to investigate how class imbalance and positive outcome ratios affect the mitigation process using synthetic data. We employ numerous metrics with this framework to assess the accuracy, fairness, and utility of the mitigation procedures.

4.1 Motivation

The interaction of fairness and privacy is an emerging field. Due to the growing concern over privacy and fairness, usage of de-identification and mitigation processes has become more prevalent [63]. This simultaneous usage leads to the question of how these processes interact with one another.

A few recent works have investigated the interaction of privacy and fairness [17, 21, 27, 69]. Zemel et al. [85] first discussed the similarity between the concept of fairness mitigation transformations and differential privacy transformations. Subsequent works investigating privacy and fairness have predominantly focused on this type of relationship and therefore, employed differential privacy over other de-identification methods. Additionally, Iosifidis et al. [43] discuss the effects of high class imbalance on the accuracy and fairness of models.

The contributions of this chapter are three-fold. First, we develop a mitigation assessment framework to analyse the effect of bias mitigation on de-identified data. Second, we investigate the impact of bias mitigation on three real-world datasets with different levels of de-identification applied. Third, given that imbalance is a possible source of bias, we conduct further investigations to understand the implications data characteristics have on the mitigation procedure.

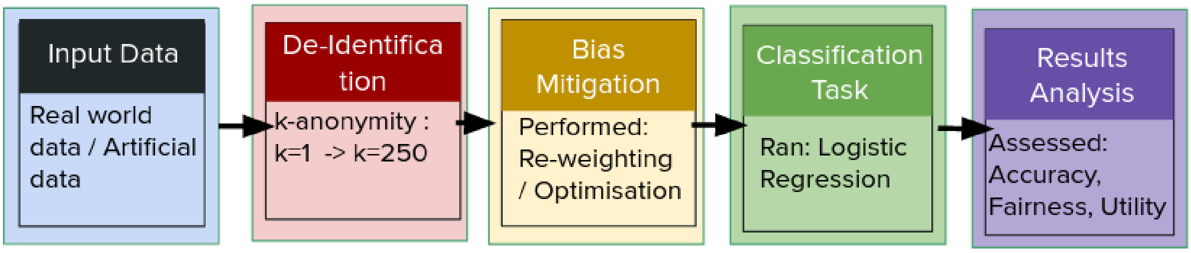

Input Data
Real world data / Artificial data
De-Identifica tion
k-anonymity : k=1 -> k=250
Bias Mitigation
Performed: Re-weighting / Optimisation
Classification Task
Ran: Logistic Regression
Results Analysis
Assessed: Accuracy, Fairness, Utility

Figure 4.1: Mitigation in the data processing pipeline.

4.2 Mitigation Assessment Framework

This section discusses the methodology and the work related to the development of the mitigation assessment framework. Our objective was to measure the fairness and utility loss as a result of the de-identification process.

The preliminary phase applies multiple levels of mitigation to the original data. Recall our definitions from Chapter 3. We have stated with a dataset D with protected class C other variables X and target Y.

$$D = (C, X, Y) \tag{4.1}$$

The original bias is measured as:

$$bias(D) = \frac{Pr(y \in Y^+ | c \in C_0)}{Pr(y \in Y^+ | c \in C_1)} \leq 1 \tag{4.2}$$

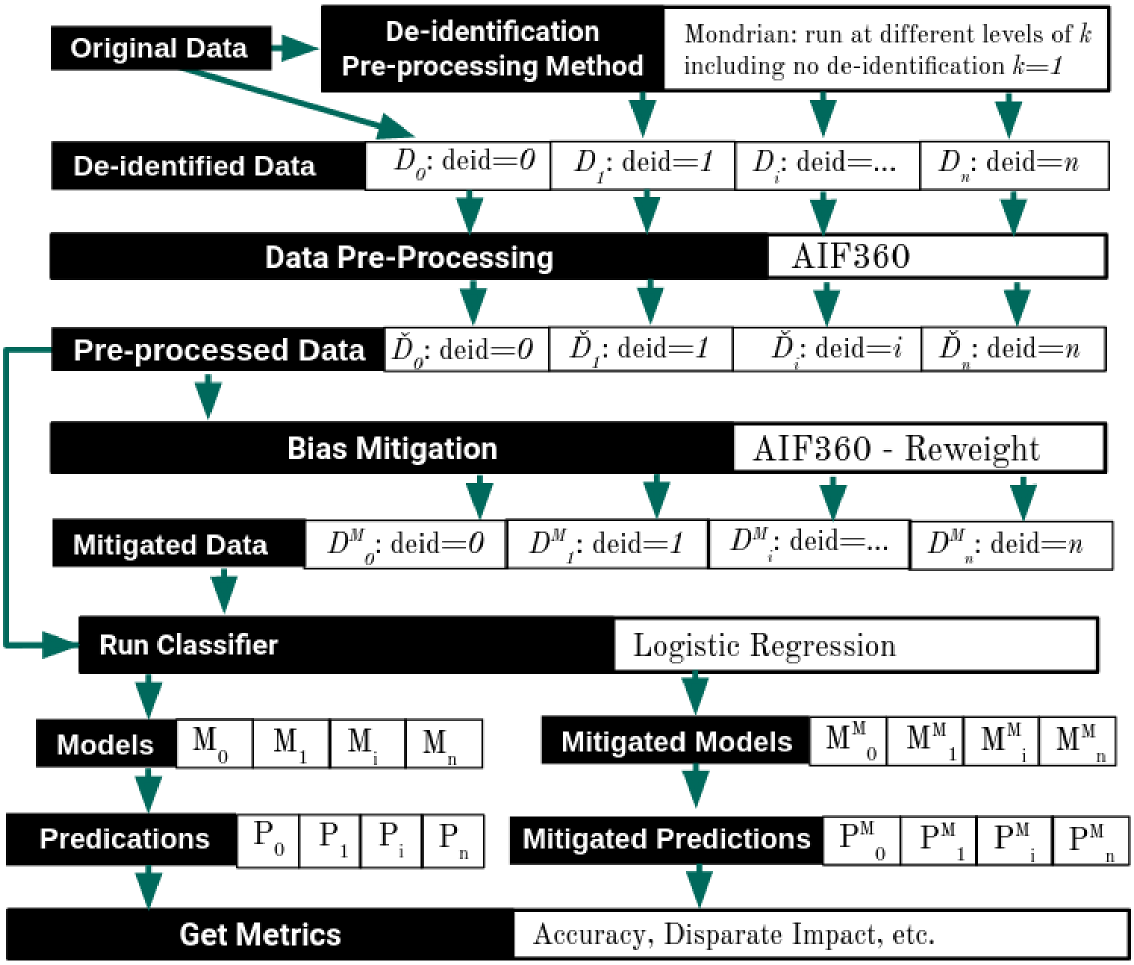


Figure 4.2: Bias assessment framework: comparing effects of mitigation on multiple levels of de-identification.

We perform de-identification using a pre-processing technique at various levels of de-identification. This step creates a dataset $\bar{D}_k$ for each $k \in K$. There are $|K|$ levels of de-identification. We consider the original dataset as part of this set of de-identified datasets, with no de-identification applied. K is a set based on the de-identification levels described by the technique used. We employ k-anonymity thus, there are $|K|$ datasets.

If the de-identification method transforms Y into $\bar{Y}$ and or C into $\bar{C}$ this may lead to a different level of bias defined by:

$$bias(\bar{D}) \in \frac{Pr(\bar{y} \in \bar{Y}^{+} | \bar{c} \in \bar{C}_0)}{Pr(\bar{y} \in \bar{Y}^{+} | \bar{c} \in \bar{C}_1)} \tag{4.3}$$

The data pre-processing is performed to prepare it for creating a model. The pre-processing is applied to the all datasets $\bar{D}_i \in n$; we model our prepossessing as in AIF360 [12]. This results in n new dataset $\hat{D}_i \in n$.

Following this, we developed the mitigation assessment portion of the framework. This applies mitigation to all datasets with the various de-identification levels. The output for the new datasets depends on the particular mitigation method. The result could be: a transformed dataset, a transformed statistical distribution, or a weighting for the different values. If the method produces a transformed dataset $\hat{D}_i^g$, with original protected class C, mitigated de-identified attributes $\hat{X}_i^g$, and the original target Y, or de-identified and mitigated target$\hat{Y}_i^g$). If the method produces a weight for different classes, this can be represented as a dictionary of weights W assigned to each class and target combination. If the method produces a statistical distribution, then the representation can vary. The user will need to adjust the inputs to fit the results into this framework.

Following this, the classifier is trained. If the mitigation creates a transformation of the dataset, the framework uses a training set from the new mitigated dataset, described as $\hat{D}_i^g$, to train the model. If the method produces a set of class weights, the classifier can be trained using a training set from D_w which is the original data with the weights W. The framework uses the weights W as an input parameter when fitting the model. If the result is a distribution the process will need to be tailored to the specific outputs requirements. Each case produces two models: one trained on the processed dataset (non-mitigated) M and one trained on the mitigated data M^g.

The processed and mitigated models are then tested using a test set from the de-identified data. This produces two sets of predictions P with $(P^+, P^- \subset P)$ and P^g with $(P^{g+}, P^{g-} \subset P^g)$.

The user can apply extensive metrics to these results. We measure balanced accuracy disparate impact, and utility described in Chapter 2. The results of this analysis can be output using plots for the user to then assess the mitigation and de-identification options.

4.3 Algorithm Discussion

This section outlines the procedures involved in the Mitigation Assessment framework. The framework is developed to compare the de-identified data with the mitigated data and assess the mitigation technique's effects on various metrics. The framework employs a bias mitigation technique and pipes the mitigated dataset and the de-identified dataset into the assessment framework. This allows these two datasets to be compared on the chosen metrics.

We developed this framework using five standard python packages: NumPy [37], Pandas [83], Scikit-learn [67], Sci-py [79], and Matplotlib [41]. In addition to these packages, we utilised the AIF360 package [12] for the bias mitigation implementation. The procedures for the mitigation assessment framework utilise the same pre-processing procedures as described in Chapter 3. Initially, we have an original dataset and de-identified datasets combined to for *deid_dfs*. First, for each dataset we create the training and testing dataset using *get_dataframes* and *split_data*. Then, we perform pre-processing using *scale_numeric* and *onehot_encode*. These procedures for the *run_preprocessing* function.

The mitigation assessment phase runs Algorithm 2, which takes in the privileged and unprivileged class values C_1 and C_0, and the positive and negative target outcome values Y^+ and Y^-. This procedure requires the user to choose a mitigation method *miti_method*, and the parameters for this method *miti_args*. The procedure here can vary depending on the mitigation method used. Additionally, the output will be in different formats depending on the mitigation type. The output is defined as the mitigated result *df_miti*.

The target values can be optimised on the classification threshold if *use_class_thresh* is true. This step is shown in Algorithm 3, where it takes in the dataframe with predictions *df_pred* and the a dictionary of classification thresholds *thresh_dict*, a metric *metric*. This metric is the metric on which the classification threshold is optimised. This procedure returns a set of results *results* optimised on the threshold and the optimum threshold *opt*.

Algorithm 4 details the whole process. The inputs are the mitigated datasets *miti_dfs* and the de-identified datasets *deid_dfs*. Then the pre-processing is performed. The procedures are run on the de-identified and mitigated data. First, *run_preprocessing* prepares the data. Then the classification task is performed using *run_classifier* then results of the assessment are returned using the *get_metrics* procedure. If the user wants to tailor the process, *plot_metrics* can output a plot of the results.

Algorithm 2 Run mitigation

```
Inputs: df - dataframe, $C_1, C_0, Y^+, Y^-$, miti_method - mitigation method type, miti_args parameters for the mitigation method
Output: mitigated dataframe and mitigation model
miti_method ← miti_method.fit(df, $C_1, C_0, Y^+, Y^-$, miti_args)    ▷ fit mitigation method
df_miti ← miti_method.transform(df)
return: df_miti
```

4.4 Experiments

The objectives of the experiments were three-fold. First, we investigated the effects of the re-weighting mechanism on de-identified data. Second, we investigated the impact of different levels of class imbalance, different positive outcome ratios, and the positive outcome difference between privileged and unprivileged classes. Third, we assessed the utility of the mitigation procedure from a medical diagnosis context.

Algorithm 3 Get class threshold

```
Input: $df$ - dataframe, $results$ - dictionary of values from classification procedure, $protect\_dict$ - dictionary of protected attributes, $met\_dict$ - dictionary of metrics, $thresh\_dict$ - dictionary of classification threshold values
Output: dictionary of metric values, optimum classification threshold
mets = array[]                                ▷ instantiate empty 2d-array
for $t \in$ thresh_dict do
    $m \leftarrow$ compute_metric(df, results, protect_dict, met_dict, t)
    mets.add($m$)
end for
results $\leftarrow$ mets
opt $\leftarrow$ max(results[metric]).index
return: results, opt
```

4.4.1 Datasets

Three real-world datasets were used in the tests: the Adult, German Credit Scoring, and MEPS. These datasets are described in Chapter 2

In addition to these standardised datasets, we utilised a Synthetic Data Generator developed in [17]. This data generator creates datasets by selectively sampling from a dataset given a particular protected attribute. We use this generator on the adult dataset, and the protected attribute is gender. The privileged class is male, and the unprivileged class is female. Given the original imbalance and positive outcome ratio of the dataset, there were 1769 unprivileged positive outcomes. Thus, with the positive outcomes set at a maximum of 40% in the unprivileged positive class the size could be up to roughly 4000 and with the imbalance ratio at $1:1$ the maximum synthetic dataset size was 8000. With this size sample the imbalance ratio is set at $[1:1, 2:1, 5:1, 20:1, 50:1, 100:1]$. Additionally, with this size sample we can set the privileged positive percentage (PPP) set at $50, 40, 20, or 10$ and the unprivileged positive percentages (UPP) set at $40, 20, 10, or 5$. This process resulted in 280 unique datasets.

4.4.2 Effects of Mitigation due to De-identification

These test utilised all three datasets: Adult, German, and MEPS. Initially, we processed each dataset similar to the methods used in the AIF360 [13] experiments. For the Adult dataset, the feature set contains gender, race, age, and education years. Income is the target. All of the features form part of the QID set. We reconfigured Mondrian [59] such that the dataset is split first along the protected attribute. This implementation ensures different protected values are not generalised into the same partition, preventing the process from varying the protected attribute distribution. We performed this de-identification procedure using a set of different k-anonymity levels

Algorithm 4 Mitigation assessment algorithm

```
Inputs: miti_dfs - mitigated dataframe, deid_dfs - de-identified dataframes, features - features to keep, target, $C_1, C_0, Y^+, Y^-$, test_size - size of test set, numeric - numeric attributes, scaler - scaling method, categorical - categorical attributes, classifier - classification method, protect_dict - dictionary of protected attributes, met_dict - dictionary of metrics
Output: bias assessment metrics, (plot)
for miti$_k$ ∈ miti_dfs do
    X_train, X_test, y_train, y_test, target ← run_preprocessing(df)
    df_pred ← run_classifier(X_train, X_test, y_train, y_test, target, classifier)
    results$_k$ ← get_metrics(df_pred, protect_dict, met_dict)
    if use_class_thresh then                ▷ if classification threshold should be used
        results$_k$ ← get_classification_threshold(df, results$_k$, protect_dict, met_dict, thresh_dict)
    end if
    miti_results.add(results$_k$)
end for
for deid$_k$ ∈ deid_dfs do
    X_train, X_test, y_train, y_test, target ← run_preprocessing(df)
    df_pred ← run_classifier(X_train, X_test, y_train, y_test, target, classifier)
    if use_class_thresh then                ▷ if classification threshold should be used
        results$_k$ ← get_classification_threshold(df, results$_k$, protect_dict, met_dict, thresh_dict)
    end if
    results$_k$ ← get_metrics(df_pred, protect_dict, met_dict)
    deid_results.add(results$_k$)
end for
plot ← plot_metrics(miti_results, deid_results, protect_dict, met_dict) return plot
```

$k \in \{1, 2, 4, 6, 8, 10, 12, 14, 16, 32, 64, 128, 256\}$. We chose this range to represent a broad scope of possible privacy requirements and to show high granularity of effects.

We performed the re-weighting procedure following the experiments in AIF360 [13]. First, the data was split using a $70 : 30$ training and testing sets. Then the test set was split $50 : 50$ into validation and test sets. We performed ten splits using random seeds and took the mean. The seeds and standard deviations are available in the supplemental material. Then, the re-weight mitigation model is trained using the training set. This model is used to create transformed versions of the training and test datasets.

Following this, we used the original data to train a logistic regression classifier. The classifier was then run on the original training, validation and test datasets to create predictions for each dataset. The original validation set and models predictions from it were used to find the optimal classification threshold from $0.0 - 1.0$ using increments of 0.01. The optimal threshold is at the highest balanced accuracy. We created a second logistic regression model on the transformed training dataset. Then the classifier is run on the transformed training data and the original test data.

We recorded eight metrics from the results of the original and transformed logistic regression models. The first two metrics recorded were accuracy and balanced accuracy, these set a benchmark for the effectiveness of the classifier given the mitigation and de-identification. The following three metrics recorded were: PPP, UPP, and disparate impact. These metrics assess the fairness of the outcomes concerning the opportunity of outcomes. The last three metrics recorded were chosen to equate to a medical testing scenario. The first metric, percentage of positive outcomes for the privileged and unprivileged classes (PPP, UPP) represents the cost in terms of how many patients will receive further testing. The second metric, recall, describes the percentage of true cases accurately caught. The final metric is the novel recall ratio (RR), the percent of true positives for the unprivileged class out of the percent of positive in the privileged class. This measure indicates the differential in cases caught between the privileged and unprivileged classes.

4.4.3 Effects of Mitigation due to Data Characteristics

This investigation aimed to understand the effects that data characteristics have on the efficacy of the mitigation method. We investigated three factors: class imbalance, PPP, and UPP. To measure the efficacy of the methods, we measured three metrics: accuracy, fairness, and utility. We employed the same metrics in these test as with the previous tests.

For these tests, we used the synthetic data generated. We performed ten random splits for each combination of imbalance ratio and positive percentages, and we took the mean. However, given the high imbalance ratios and low positive percentage outcomes, some splits lead to no positive outcomes in the minority class. This situation is not compatible with some metrics. We address this with specific random seeds chosen to ensure that all datasets had at least one individual in the minority class with a positive outcome. These seeds are available in the supplementary material. The pre-processing was performed as in AIF360 [13]. We selected five attributes: gender, race (binarised to white and non-white), age (binned to ten-year increments), gender, and education years. Education years was between 1-13 with less than six and greater then 12 each binned together. The target was income per year binarised to above 50 thousand per year or not. We performed the same mitigation and classification procedures for these tests on the synthetic data as real data.

4.4.4 Utility of Mitigation Mechanism in Medical Context

The concept of utility can vary depending on the context. In this context, utility measures the improvement to RR compared to the loss in overall recall and the cost of overall positive outcomes. These metrics imply a scenario such as a healthcare setting where the objective is to identify as many individuals requiring further treatment as possible while dealing with limited resources and trying to meet the aim of even identification rates between the protected groups.

We performed the same mitigation and classification procedures for these tests on the synthetic data as real data. However, we displayed the results on a range of classification thresholds. The classification threshold shows the contrast in the metrics between using mitigated and non-mitigated data.

4.5 Results

The experiments conducted focused on two aspects. First, we assessed the effects of de-identification level on the accuracy, fairness, and utility of mitigated data. Second, we analysed the affects of different data characteristics on mitigated data.

4.5.1 Effects of de-identification level.

The experiments on accuracy and fairness had a three-fold purpose. First, to assess the accuracy and fairness trade-off of the mitigation procedure with different levels of de-identification. Second, to investigate the cost in regards to overall positive outcomes in relation to recall with respect to different de-identification levels. Third, to investigate the utility of the mitigation procedure in comparison to adjusting the classification threshold for non-mitigated data.

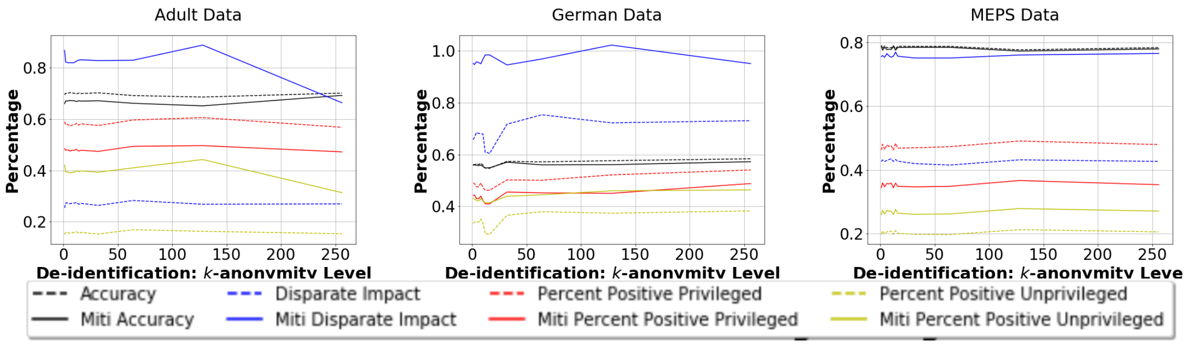


Figure 4.3: Results for accuracy, class positive percentages and disparate impact with and without mitigation on different levels of de-identified data.

Figure 4.3 presents the results for the three data sets comparing the accuracy, disparate impact and positive percentages to the level of de-identification. On the original Adult dataset, initially, with $k = 1$ (no de-identification) the PPP is 0.59 and the UPP is 0.15. Thus, the disparate impact is 0.25 while the accuracy is 0.69. Whereas on the transformed Adult dataset, the PPP is 0.48 and the UPP is 0.42. Thus, the disparate impact is 0.87, and accuracy remains 0.66. As the level of de-identification increases, there is some variation, however, the most significant effect is apparent at high levels of de-identification. With an extreme case of $k = 250$ the original PPP is 0.56 and UPP is 0.15. After mitigation, the PPP decreases to 0.47 while the UPP only increases to 0.31.

These results indicate a reduction in the efficiency of the mitigation process with high de-identification levels. However, with the German and MEPS dataset, this pattern is not as clear. The reduction in PPP and the increase in UPP have some variation with de-identification level. However, the efficacy of the mitigation does not consistently decrease and accuracy for mitigated data only decreases by 0.01 even at the highest de-identification level. On the MEPS dataset, there is minimal variation through different levels of de-identification. The reduction in PPP and the increase in UPP remain consistent at all levels of de-identification as well. These results could be due to the number of attributes included in the QID set for each dataset. With the adult dataset, all four attributes were included in the QID set, whereas, with the German dataset, only 2/4 features were in the QID set. With the MEPS dataset, we included 8/39 of the attributes in the QID set.

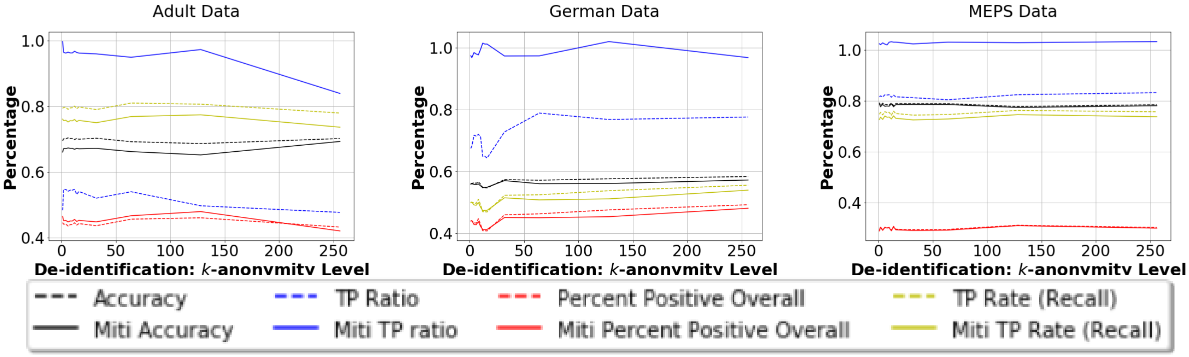
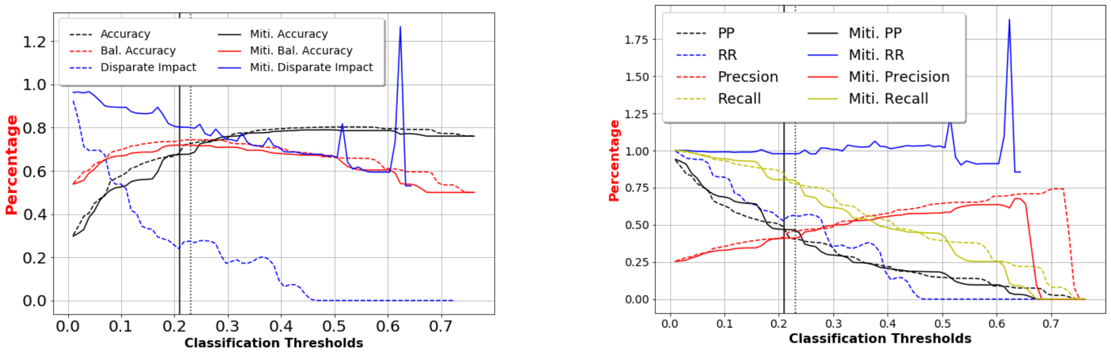


Figure 4.4: Results for accuracy, overall positive rate, class true positive rates, and true positive ratio difference on different levels of de-identified data.

Figure 4.4 displays the results for the overall percent positive, the recall rate and the RR. Initially, at $k = 1$ the recall is 0.79 while the RR is 0.48 and the overall percent positive is 0.44. After re-weighting, the recall reduces to 0.76, and the RR reaches 0.99. The overall positive percentage only increases from 0.44 to 0.46. As the de-identification level increases the effect of mitigation on the loss in recall remains almost constant. However, with high levels of de-identification, the improvement in the RR is less significant. At $k = 250$ the original recall is 0.78 and the RR is still 0.48. After re-weighting, the recall reduces to 0.74, and the RR reaches 0.83. The overall percent positive decreases from 0.43 to 0.42. In comparison, with the non-de-identified data, the recall reduces by 0.03 to improve the RR by 0.51. With de-identification level set at $k = 250$, the recall reduces by 0.04 and this only improves the RR by 0.35.

These results indicate the percent of positive outcomes is not significantly affected by mitigation on the de-identified data. However, with high de-identification levels, the mitigation process becomes less efficient at improving the RR relative to the reduction in recall. These results are similar but less significant on the German and MEPS data sets.



(a) Accuracy-Fairness

(b) Utility-Fairness

Figure 4.5: Fairness trade-off and utility trade-off on classification threshold.

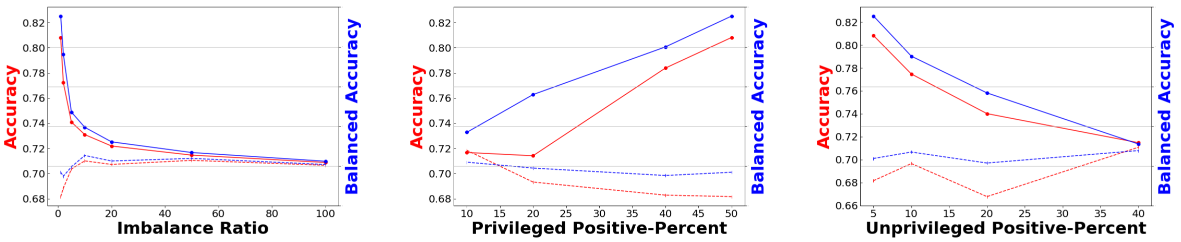
Utility

Figure 4.5(b) displays that initially, for both the original and re-weighted datasets, the recall rate is 1.0, the RR is also 1.0, and the overall positive percentage is 0.95. Thus, the precision is 0.25. These values indicate the situation is completely equal between the classes and that all necessary positive cases have been captured. However, with the positive percentage at 0.95 the cost would be extremely high. As the classification threshold increases, the RR for the original data varies. However, it tends to decrease significantly. With the re-weighted data, the RR generally remains at or above 1.0 up to a threshold of 0.53. Specifically, at the optimal balanced accuracy threshold, for the original data, the recall rate 0.77, the RR is 0.55, the overall positive percentage is 0.4, and the precision is 0.46. At the optimal threshold for the re-weighted data, the recall rate is 0.8, the RR is 0.98, the overall positive percentage is 0.46, and the precision is 0.4. Due to the differing optimal thresholds, the transformation results in an increase in RR of +0.43 and an increase in recall of +0.03. At the same time, there is only an increase of +0.06 in the overall positive percentage and a decrease of −0.06 in the precision.

4.5.2 Effects of imbalanced data.

The purpose of using the synthetic data was to investigate the efficacy and robustness of mitigation methods when dealing with imbalanced data, low positive target outcomes, or a high differential between target outcomes for the protected group's classes.

The pre-processing methodology employed by Bellamy et al. [13] performs binning on many of the QID attributes. This obfuscates much of the effects of the de-identification process. As a result, the effects of de-identification in these test were, in most cases, negligible. Throughout the 280 datasets, 210 were de-identified using four different levels of de-identification $k = \{1, 4, 8, 16\}$ the rest were non-de-identified for comparison. In 144 cases, the de-identification reduced the accuracy with an average reduction of -0.015 and a highest reduction of -0.049. In 66 cases, the de-identification increased the accuracy with an average of 0.017 and a highest increase of 0.06.



(a) Accu. vs. Imbalance (b) Accu. vs. PPP (c) Accu. vs. UPP

Figure 4.6: Results for accuracy and balanced accuracy on synthetic data.

Accuracy:

In these tests, the results showed that the effects of the mitigation process on accuracy on average are minimal. Of the 280 datasets, in 240 cases the mitigation process decreases the accuracy with an average reduction of -0.023%. In 39 cases, the mitigation process increases the accuracy with an average increase of 0.005%. However, in certain cases, the process of mitigation can lead to significant reductions in accuracy.

Figure 4.6 presents the results of three tests. Figure 4.6(a) compares the accuracy and balanced accuracy of the original and mitigated data with respect to the imbalance ratio. We fixed the PPP to 50%, and the UPP to 5%. Initially, with an imbalance ratio of $1 : 1$, we see a reduction in accuracy of 15%. This reduction continually decreases to 1% at a ratio of $100 : 1$. With higher imbalance ratios, the decrease in accuracy and balanced accuracy is lessened. High imbalance ratio means fewer unprivileged examples. Thus, the re-weighting process must augment the outcomes on fewer examples. Therefore, it has a lower impact on the overall accuracy.

Figure 4.6(b) presents accuracy and balanced accuracy in relation to the PPP. We fixed the ratio at $1 : 1$ and the UPP to 5%. To start with, the PPP is at 10% with no loss in accuracy and a 3% loss in balanced accuracy. As the PPP increases, we see a continuous increase in the accuracy lost up to 15% when the PPP is at 50%.

Figure 4.6(c) presents the accuracy and balanced accuracy in relation to the UPP. We fixed the imbalance ratio to $1 : 1$ and PPP to 50%. To start with, the UPP is 5%, and the loss to accuracy is 15%. As the UPP increases, there is a reduction in the accuracy loss. Figures 4.6(b) and 4.6(c) show that it is not the high or low values of the PPP and UPP that affect the accuracy. Thus, the indication is that the differential between the privileged and unprivileged positive outcomes causes a loss to accuracy. Since the re-weighting mechanism is designed to balance the difference in outcomes, the higher the difference in outcomes the more significant the weights the mechanism must apply. Thus, the more significant the effect the process has on the outcomes and the accuracy.

These results indicate that the significant factors determining the change in accuracy are the size of the minority class as a ratio of the whole dataset and the percentage difference in the target outcome of the privileged and unprivileged groups. Intuitively this results from the underlying objective to align the positive outcomes of the privileged and unprivileged groups. Thus, with higher differential, the weights must be altered more and this introduces a greater opportunity for error. Additionally, with high class imbalance, if the minority population is the disadvantaged population, then the number of instances that need to change to align the positive outcomes is much lower. As a result, the weights can be less, so the impact on the overall accuracy is decreased.

Figure 4.5 shows the results of tests performed using a classification threshold from 0.01 – 0.8. These tests had two objectives. First, to analyse the effects the mitigation mechanism has on disparate impact and accuracy. Second, to analyse RR compared to the loss in the overall recall and the cost in overall percentage of positive outcomes. The graphs compare the mitigated data to the original data.

Fairness:

In Figure 4.5(a) the original data starts with a classification threshold of 0.01 and he disparate impact metric is high, meaning the percentage of positive outcomes between groups are similar. As the threshold increases, there is a steep drop in the disparate impact. With the re-weighted data, the decrease in disparate impact is significantly lessened with high classification thresholds. This indicates the mitigation mechanism is effectively increasing fairness over non-mitigated data. With an increased classification threshold, the change in accuracy and balanced accuracy varies slightly. However, the result is always a reduction from the original dataset. This reduction to accuracy and balanced accuracy appears to be minimal compared to the gain in fairness generally.

These tests indicate that the re-weighting mitigation mechanism effectively reduces the fairness bias, while only marginally affecting the accuracy and balanced accuracy. Additionally, when set to an appropriate threshold, the overall positive outcomes and the precision can be minimally affected when improving the RR and recall.

4.6 Summary

We briefly outlined some of the recent work on fairness and privacy in machine learning. We also discussed the motivation to understand the effects of the interaction between these to concepts. The experiments we performed empirically tested the effects of this interaction using three main concepts. First, we investigated the effects of mitigation of different levels of de-identified data using real-world data. Second, we used synthetic data to measure the effects of mitigation, given different data characteristics. Third, we assessed the utility of the re-weighting mitigation mechanism based on the improvement in fairness compared to the loss to accuracy and to the increased cost of positive outcomes. These experiments indicate that the effects of mitigation become less effective with higher levels of de-identification, and that this is dependent on how significant the QID sets is on determining the outcome of a classifier. Additionally, we found that the re-weighting mechanism is more efficient on detests with higher imbalance. Furthermore, we found that the higher the differential between the PPP and UPP, the less efficient the re-weighting mechanism becomes. Finally, we found that given an optimally derived threshold, the re-weighting mechanism effectively improves fairness, while having minimal effect on the accuracy and cost in terms of overall positive outcomes.

5
Mitigation Optimisation

In this chapter we introduce Mitiband (MB) mitigation optimisation mechanism. This mechanism is a novel bandit based approach to search through the mitigation method's hyper-parameter space using multi-metric scoring and user defined limits. The aim is to enable an efficient trade-off between accuracy and fairness. Section 5.1 discusses the motivation for this approach. Section 5.1 outlines existing work in the field of bias mitigation and optimisation. Section 5.3 describes the procedures used in our optimisation approach. Section 5.5 discusses the procedures of the experiments we conducted to test this mechanism. Section 5.6 details the results of the experiments conducted. Section 5.7 concludes this chapter.

The objective of this chapter was to investigate methods to efficiently improve the trade-off between fairness and accuracy for bias mitigation. We build upon both bias mitigation and optimisation techniques to develop this approach. We then experimentally compare our approach with the original mitigation method with other optimisation methods on accuracy, fairness, and utility.

5.1 Motivation

Mitigation methods invariably lead to losses in some metric of accuracy or utility [33]. Like all competing methods, we aim to identify good fairness-accuracy trade-offs. Our objective was to develop a method to optimise the trade-off between the fairness and accuracy metrics on de-identified data. Hyper-parameter optimisation has long been used for the optimisation of model hyper-parameters. Recent research in fair machine learning has used hyper-parameter optimisation with fairness-oriented scoring to improve fairness on machine learning models [20].

The contribution of this chapter is to develop a method to optimise the parameters of the mitigation methods with tailored scoring functions bounded by user defined limits. One of the benefits of this approach is the that it performs successive rounds of searching and assessing; this allows it to actively prune the search space in each iteration. From this the mechanism thoroughly searches the acceptable space using less resources. Additionally, this allows the approach to be tailored to different regulatory scenarios in which the users are operating.

Another benefit of this approach is that the level of bias mitigation can be controlled by the user through the trade-off parameter. This is beneficial if the user is focused more or less on fairness or accuracy. Specifically, when dealing with de-identified data, it allows the user to choose the level of mitigation to apply. For instance, the user might want to mitigate only the extra level of bias introduced from the de-identification process.

In the following section, we formalise the Hyperband algorithm and present how this can be extended to facilitate our approach. We then compare the results of this approach to those achieved with the re-weight method alone and to the fairness levels achieved through random search (RS) and Grid Search (GS).

5.2 Overview

This section outlines the various methodologies that have been built upon to develop the Mitiband mechanism. First, we cover the optimisation framework. Second, we cover the mitigation methodologies.

5.2.1 Optimisation

Optimisation involves selecting a scoring metric and then successively testing and assessing the quality of different parameters based on the scoring metric. Many approaches have been developed, here we outline the approaches significant to the development of our mitigation optimisation approach.

Successive halving (SH) [53, 46] was developed to address the problem of scalability for which optimisation approaches often suffer. The SH approach address scalability by treating the hyper-parameter optimisation problem as a task of finding the best arm in a multi-arm bandit problem. The multi-arm bandit problem deals with allocating a fixed set of resources to competing choices to maximise the outcome measure. The (SH) process involves a deterministic number of rounds, and each round has the same fixed budget. First, the budget is uniformly allocated to each arm (hyper-parameter configuration). Second, the performance is evaluated using the scoring function. Third, the worst-performing half of the configurations are discarded and the process repeats using the remaining configurations. The insight to this approach is that it extrapolates the rank of configurations performances in the early rounds using the low budget and can later get a deeper analysis of these better candidates using the higher budget in later rounds. This system of rounds allows the search space to be guided towards the region of better configurations for the future rounds. This method, however, has two parameters for which there is no clear values, the budget B and the number of sampled configurations n. These two parameters result in a trade-off between evaluating a greater number of configurations (larger n) on a lower budget and fewer configurations (lesser n) on a higher budget B/n. The performance of this approach compares favourably to competing strategies [8, 28, 48, 45].

Hyperband (HB) [60] was developed to address the trade-off between n and B/n by dividing the total budget into different iterations of the trade-off, then calling Successive Halving as a sub-routine in each iteration. This allows the budget for each successive round to be greater. Thus, not wasting budget early and, retaining greater budget for improving the fidelity of the assessment in later rounds. This method takes two parameters, R the maximum number of resources allocated to any configuration and, n the ratio of increase in budget ($n = 2$ for standard SH). Each iteration of SH, called a bracket, takes two parameters the number of configurations n and the resources units allocated r_i. The outer loop searches through possible combinations of (n, r_i) and the inner loop executes SH. The outer loops is performed $s_{max} + 1 = log_n(r)$ times. The inner loop takes B resources. Thus, the total resources used $s_{max}(B)$.

5.2.2 Fairness

Here we cover two alternative approaches to improving model fairness. First, there is optimisation using fairness metrics. Second, there are the mitigation approaches applied to the data before, during, or after processing and model creation. For data publication purposes, we focus on a pre-processing technique.

Fairband (FB) [20] was developed to optimise models based on accuracy and fairness metrics. FB takes advantage of the adaptability of the Hyperband algorithm to develop a fairness aware model tuning. HyperBand is model and metric agnostic. Thus, the user can determine the scoring function on which the model is optimised. Fairband (FB) uses a trade-off metric which acts on two metrics, an accuracy metric a and a fairness metric f. The goal is to maximise a and f for $a, f \in [0, 1]$ or minimise $1 - a$ and $1 - f$. This approach weighs the trade-off metric by the relative importance of the accuracy metric.

This is a method of multi-objective optimisation called *weighted-sum scalerisation* [23]. This defines an optimisation metric o using an $\alpha \in [0,1]$ parameter to trade of the fairness metric q_f and the accuracy metric q_a as in Equation 5.1.

$$o = \alpha * q_a + (1 - \alpha) * q_f \tag{5.1}$$

With this method, α defines the relative importance of the accuracy metric. For example, if $\alpha = 0$, then the model is only measured with the fairness metric. Conversely, if $\alpha = 1$, then the model is measured only based on the accuracy metric. This method does not entail significant resources over simply measuring accuracy as both measures rely on the same predictions. This method of *weighted sum scalerisation* involves determining the value for the α parameter. However, to automate this process, Fairband uses the FB-auto using a heuristic to set α automatically [20]. Fairband assesses the level of difference between the accuracy and fairness metrics. If accuracy is high and fairness is low, the search will be guided toward regions of higher fairness by choosing a (low α), and vice versa. This method employs a proxy metric for the target direction change defined by the difference μ in the average predictive accuracy $\bar{q_a}$ and the average model fairness $\bar{q_f}$ described in Equation 5.2.

$$\mu = \bar{q_f} - \bar{q_a}, \mu \in [-1, 1] \tag{5.2}$$

This equation can be used to direct the value of α, so when $\bar{q_f} < \bar{q_a}$ then μ is negative and decrease α to guide the search to improve fairness. Conversely, if $\bar{f} > \bar{a}$ then μ is positive and the search should be directed to areas of higher accuracy. This change in α should be proportional to μ by some constant $\gamma > 0$. Then we get the integrated equation:

$$\alpha = \gamma * \mu + \epsilon \in R \tag{5.3}$$

Where ϵ is the integration constant. Given that $\mu \in [-1, 1]$ and $\alpha \in [0, 1]$ the only feasible values of γ and ϵ are $\gamma = 0.5$ and $\epsilon = 0.5$. Thus, dynamic α is given by Equation 5.4

$$\alpha = 0.5 * (\bar{q_f} - \bar{q_a}) + 0.5 \tag{5.4}$$

Re-weighting (RW) [49] is a mitigation technique that breaks the data down into four subsets based on the binary privileged class and the positive or negative binary label outcomes. The method takes in a dataset D:

$$D = (C, X, Y) \tag{5.5}$$

The original bias fr the dataset is measured as:

$$bias(D) = \frac{Pr(y \in Y^{+}|c \in C_0)}{Pr(y \in Y^{+}|c \in C_1)} \leq 1 \tag{5.6}$$

From these inputs the data is broken down into four sub-sets:

- *privileged-positive-class (PP)* $= (c, x, y) \ \forall \ c \in C_1, y \in Y^{+}$
- *privileged-negative-class (PN)* $= (c, x, y) \ \forall \ c \in C_1, y \in Y^{-}$
- *unprivileged-positive-class (UP)* $= (c, x, y) \ \forall \ c \in C_0, y \in Y^{+}$
- *unprivileged-negative-class (UN)* $= (c, x, y) \ \forall \ c \in C_0, y \in Y^{-}$

This method assigns a weight to all instances in each of the four classes to make the number of positive and negative labels Y^{+} and Y^{-} are near parity between the two classes C_0 and C_1 on the training set. Higher weights are given to the UP and PN classes while lower weights are given to the UN and PP classes. The functions for determining these weights are described in Equations 5.7 to 5.10:

$$W_{PP} = (|C_1| * |Y^{+}|)/(|C| * |PP|) \tag{5.7}$$

$$W_{PN} = (|C_1| * |Y^{-}|)/(|C| * |PN|) \tag{5.8}$$

$$W_{UP} = (|C_0| * |Y^{+}|)/(|C| * |UP|) \tag{5.9}$$

$$W_{UN} = (|C_0| * |Y^{-}|)/(|C| * |UN|) \tag{5.10}$$

This method results in dataset D_w where W are the instance weights:

$$D_w = (C, X, Y, W) \tag{5.11}$$

This method results in near parity for labels outcomes in the test set in most cases given a good train/test split. One of the drawbacks here is that this can be considered a heavy-handed approach in specific scenarios where there are fairness objectives, but they do not out-weight other metrics. One of the benefits of this approach is that it could be applied to a broader array of non-binary groups and or labels.

5.3 Mitiband

In this section, we first formalise the Mitiband mechanism, then we provide an example scenario of how it would be used. This mechanism incorporates aspects of the two fairness approaches, Fairband and re-weighting. Similar to Fairband, it utilises multi-metric scoring, and it is built upon the Hyperband algorithm. We demonstrate the implementation of Mitiband using the re-weight bias mitigation method. This method does not have explicit hyper-parameters. However, we take the weights generated by the re-weight function as parameters for optimisation. We set the search space between no mitigation on the original data with all weights equal to one and the weights derived front the re-weight mechanism displayed in Figure 5.2. This implementation demonstrates how we can adapt this optimisation method to non-parameterised methods.

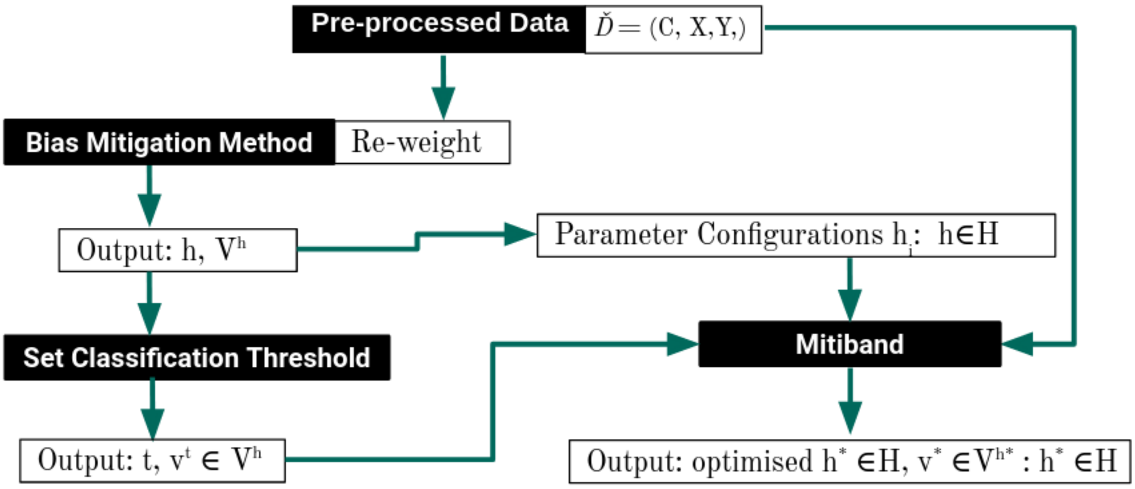


Figure 5.1: Data processing procedure for optimisation of re-weighting using Mitiband.

The preliminary phase of our implementation runs the re-weighting mechanism. This results in class weights W_{PP}, W_{PN}, W_{UP}, W_{UN} and a dataset D_W with:

$$D_w = (C, X, Y, W) \tag{5.12}$$

Following this a classifier is trained and creates model a M_w. The model is then tested using a test-set. This produces predictions P_W:

$$M_w = P_w | D_w \tag{5.13}$$

The predictions are then optimised based on the classification threshold between $[0.01, 0.99]$ with an interval of 0.01. We optimise the outcome based on balanced accuracy, which give optimal threshold t^*.

The optimisation process takes in numerous parameters. Initially, we have an *estimator*. This is the type of model being created. This can be any estimator in the scikit-learn library. With our test, we used the logistic regression estimator.

There is a *resource_parameter*, which is the cost parameter associated with the particular *estimator*. For example, with a decision tree, this can be n_estimators. With our tests, the logistic regression resource parameter is set to the number of iterations before convergence. The resource parameter has a total budget of B.

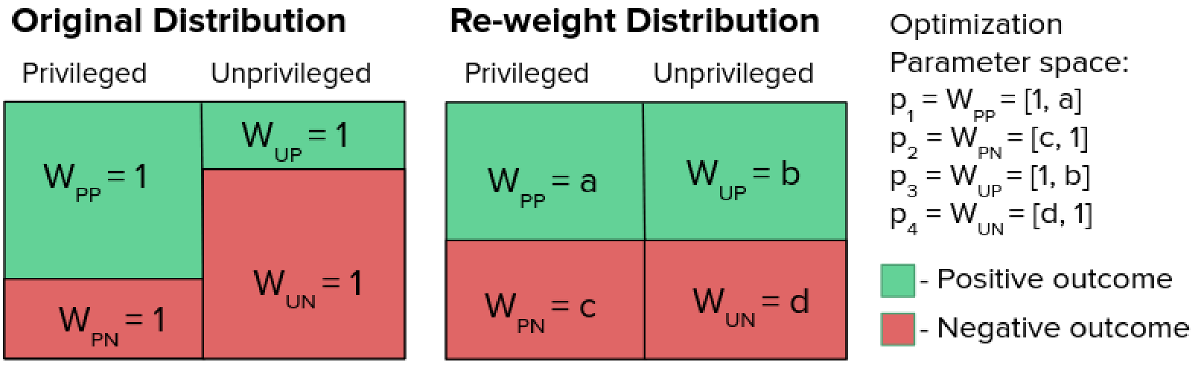


Figure 5.2: Original and re-weight class distributions and search space for Mitiband parameters.

The optimisation is performed on the *parameter_distributions* H. This is a set of parameters $(h_1 \ldots h_n)$ which are part of the model or the mitigation process. For the re-weight mechanism these represent the class weights and have a range from 1 on each sample in the original data to the weight derived from the re-weight mechanism shown in Equation 5.14:

$$H = (h_1, h_2, h_3, h_4) : h_1 \in [W_{PP}, 1], h_2 \in [1, W_{PN}], h_3 \in [1, W_{UP}], h_4 \in [W_{UN}, 1] \quad (5.14)$$

The user can also define a set of metrics Q. This set is a dictionary of metrics by which the mechanism scores the results at each round. These metrics are used to rank the configurations and guide the optimisation process by pruning configurations outside the limits for any metric.

There is also a specific a trade-off metric $\alpha \in [0, 1]$ and a scoring function o. This is a composite metric which takes in an accuracy metric $q_a \in Q$ and fairness metric $q_f \in Q$. The α parameter is describes the relative importance of the accuracy metric:

$$o = \alpha * q_a + (1 - \alpha) * q_f \quad (5.15)$$

The user can also create a set of limits $l \in L$. These define limits for the metrics in Q such that the minimum value for q is $\lfloor v \rfloor$, and the maximum value for q is $\lceil v \rceil$:

$$l = (\lfloor v \rfloor, \lceil v \rceil) \ \forall \ q \in Q \quad (5.16)$$

The η parameter dictates the progression of the mechanism. This is the inverse of the percentage of configurations pruned at each stage. For example, if $\eta = 3$ then $\frac{1}{3}$ of configurations will be pruned each round. Thus, *eta* dictates the rate of convergence to the optimised configuration and the number of rounds.

Finally, there are the R and R_0 parameters. These are the minimum and maximum amount of resources to allocate to the cost parameter applied to each configuration.

Following this, the optimisation process begins. The process is divided into brackets. Since the R defines the max budget for any the parameter configuration in the first round and η is the factor by which the budget is reduced, number of brackets is $s_{max} + 1$, including the final round with one configuration:

$$s_{max} = \lfloor log_{\eta}(R) \rfloor \tag{5.17}$$

In terms of resources the budget for each bracket is B:

$$B = (s_{max} + 1) * R \tag{5.18}$$

Thus the total budget is:

$$total\_budget = (s_{max} + 1) * B \tag{5.19}$$

In each bracket $s \in s_{max}$, successive halving (SH) is run on n sample configurations in H and the minimum resources per configurations is set to r where:

$$n = \lceil \frac{B}{R} * \frac{\eta^s}{s+1} \rceil \tag{5.20}$$

$$r = R * \eta^{-s} \tag{5.21}$$

In each round, all configurations $h \in H$ are assessed on all metrics $q_i \in Q$ and the results are stored in v_i. All configurations where any value v_i for any metric $q \in Q$ is outside of the limits for $l_i \in L$ is discarded leaving the set H^K of acceptable configurations:

$$H^k = \forall h : q \in Q : \lfloor v(h) \rfloor \geq q \leq \lceil q \rceil \tag{5.22}$$

Additionally, in each round the number of configurations is reduced by a factor of η:

$$n = n/\eta \tag{5.23}$$

Thus, the top k configurations are retained and the rest are pruned. If $H^k > k$ then the top k of H^k are retained. However, if $H^k \leq k$ then $k - |H^k|$, random configurations are added. The search space for each $h \in H$ for these random configurations is limited to between the minimum and maximum values of the acceptable parameters of $h \in H^k$ such that:

$$\lfloor h_i \rfloor \geq h \leq \lceil h_i \rceil \ \forall \ h_i \in H^K \tag{5.24}$$

5.3.1 Example Scenario

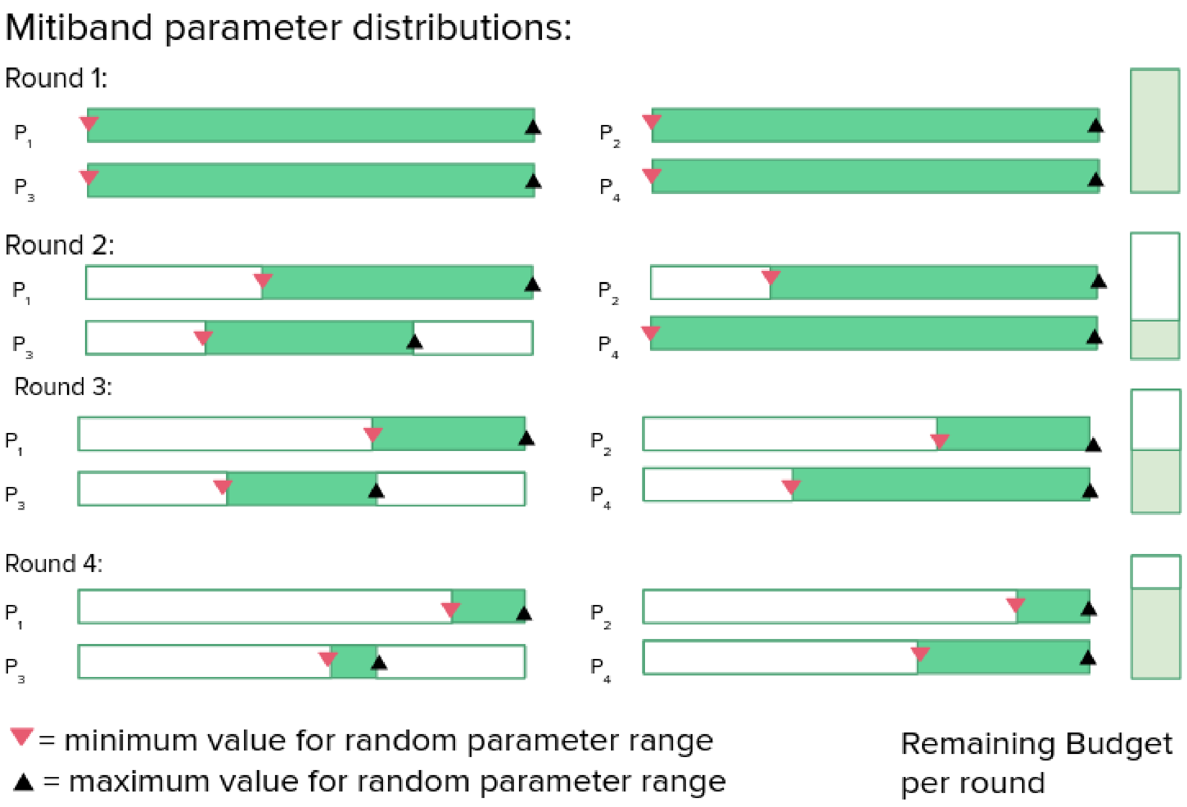

Mitiband parameter distributions:
Round 1:
P_1
P_2
P_3
P_4
Round 2:
P_1
P_2
P_3
P_4
Round 3:
P_1
P_2
P_3
P_4
Round 4:
P_1
P_2
P_3
P_4
▼ = minimum value for random parameter range
▲ = maximum value for random parameter range
Remaining Budget per round

Figure 5.3: Changing parameter distributions and residual budget in successive rounds of Mitiband.

Presume a healthcare company has de-identified some medical data which they would like to publish. A bias assessment has indicated that the disparate impact before de-identification is 0.45 and after it is 0.35. When the data is published, the company could not disclose the exact disparate impact loss to the end-user without undermining the de-identification by revealing information about the original data. If the healthcare company is obligated to meet fairness guidelines, the company might be required to perform bias mitigation on the data. The decision then would be what level of mitigation to perform. The company want to mitigate the original bias, or just mitigate bias back to the original level. This leads to the question of a trade-off between fairness and accuracy. One way to determine this is to run optimisation and set the disparate impact limits to $min = 0.45$ and $max = 1.0$. Then the optimisation can be performed using balanced accuracy as the scoring metric. The first round randomly selects parameters, and the results are analysed using the scoring function. Configurations that produce values outside the limits are pruned. If the number of configuration inside the limits is greater than the budget for that round then take the top k results. If the parameters within the limits is lower, find the maximum and minimum of each parameter of the acceptable values and randomly select values within that range. This method more thoroughly search's the acceptable space. At each round, this space is reduced and the budget per parameter configuration is increased to get a higher fidelity outcome. The final result will be within the selected window and have been tested on an increasingly higher budget. Additionally, the outcome for each configuration can be output as a graph comparing accuracy and fairness. The user can then manually identify a desirable trade-off level from this output graph.

5.4 Algorithm Discussion

This section outlines the algorithmic processes involved with the Mitiband mechanism. The initial step is to define the parameters $H = (h_1 \dots h_n)$ and the search space for each parameter. We derive these parameter distributions by performing the re-weight method in the same manner as described in AIF360 [12]. These results match Section 5.2.2. The outputs from this method are four weights representing parameter distributions $H = (h_1, h_2, h_3, h_4)$. First, R, the maximum resources allocated to a single configuration and η, the budget increase ratio, are used to determine the number of brackets that will take place, s_{max}, and the budget per round B. Then we can get n the number of configurations to choose for each round. The function *get_hyperparameter_configurations* uses the scikit-learn *parameter_sampler* to search through H to get a random selection of n parameter configurations T.

Following this we run successive halving s times with n_i configurations with a budget of r_i per configuration. The function *train_with_budget* creates a model $m_\lambda \in M$ each round using T_λ. Then, we use the *evaluate* function to get the accuracy value of the model A using q_{accu} and the fairness of the model F using q_{fair}. Then we iterate through the metrics Q and use *evaluate* takes the metric q and model m_λ to get a value $v \in V$ for each metric. Then, for each metric q we check if its corresponding value v is within the minimum and maximum limits set by $L[q] = [min, max]$. If v is within the limits we get the objective metric o and add it to O.

After this, we use the function *get_random_configurations* from Algorithm 5. This function takes in T, the current hyper-parameter configurations and assigns it a new set of hyper-parameter settings for the next round. If the number of remaining configurations $|O|$ is greater than k, we retain only the top k scoring configurations of T. If the number of configurations $|O|$ is less than k, an additional $(k - |O|)$ random configurations T_R are selected. This condition is described in Equation 5.25. We select each random configuration H_R from a reduced search space. The search space for these random configurations is within the minimum and maximum values for each parameter h_j of the acceptable search space found so far. This space is the minimum and maximum values for each parameter h_j found in the current configurations $\lambda \in T$. Algorithm 5 describes this selection process.

$$T = \begin{cases} O + get\_configuration(k, l, p)), & \text{if } O < k \\ argsort(O)[0:k], & \text{if } k \geq O \end{cases} \tag{5.25}$$

Algorithm 5 Get random configurations

```
Inputs: $k$ - number of configurations for next round, $O$ - objective metric vales for each configuration tested, $T$ - all tested configurations, $H$ - all configurations ▷ number of configurations for next round, objective metric values, current configurations, all configurations
Output: $T$ set of hyper-parameter configurations for the next round.
if $|O| > k$ then
    $O \leftarrow$ argsort($O$)
    $T \leftarrow T[O[0:k]]$    ▷ T gets top $k$ configuration indexes from $O$
else
    $T \leftarrow O$
    for $i \in \{0, ..., (k - |O|)\}$ do
        $T_R =$ dict $\{h : 0\}$ $\forall h \in H$    ▷ instantiate new dict for a set of parameters
        $H_R \leftarrow []$
        for $h_j \in h\}$ do
            $h^* \leftarrow$rand($min(T[h_j]), max(T[h_j])$)
            $H_R$.add($h^*$)
        end for
        $T_R$.add($H_R$)
    end for
    $T \leftarrow T + T_R$
end if
return: T
```

Algorithm 6 describes the entire process from running re-weight to iterating through the Successive Halving brackets to the multiple rounds of assessing configurations and evaluating limits and selecting random configurations.

5.5 Experiments

This section details the metrics and then the processes used to assess the Mitiband mechanism. The comparisons performed were threefold. First, we compare our mechanism to the original re-weight algorithm (RW) [49] on balanced accuracy and disparate impact. Second, we compare our mechanism on the metric of true positives and TP-ratio to overall true positives. Third, we compare our mechanism to Grid Search (GS) and random search (RS) on the metric on runtime. We performed these experiments on three datasets described in Chapter 2. The first is the Adult dataset [54], the second is the German dataset [25], and the third is the COMPAS dataset [6]. These are all described in Chapter 2.

Algorithm 6 Mitiband algorithm

```
Inputs: $H$ - all configurations, $\eta$ - reduction factor, $R$ - max resources for a configuration, $\alpha$ - trade-off ratio, $q_{accu}$ - accuracy metric, $q_{fair}$ - fairness metric, $Q$ - set of metrics, $L$ - set of limits [max, min] for each metric
Output: best configuration found through the Mitiband process
$s_{max} \leftarrow \lfloor log_{\eta}(R) \rfloor$    ▷ defines number of brackets
$B \leftarrow (s_{max} + 1) * R$    ▷ defines budget per bracket
for $s \in \{s_{max} + 1\}$ do    ▷ iterate through SH brackets, as per [60]
    $n \leftarrow \lceil \frac{B}{R} * \frac{n^s}{s+1} \rceil$
    $r \leftarrow R * n^{-s}$    ▷ choice of n versus B/n trade-off
    $T \leftarrow$ get_hyperparameter_configurations($n$, $H$)
    for $i \in \{0, ..., s\}$ do    ▷ run Successive Halving
        $n_i \leftarrow \lfloor n * n^{-i} \rfloor$    ▷ train $n_i$ configurations
        $r_i \leftarrow r * n^i$    ▷ $r_i$ training budget per config
        $M \leftarrow \{\text{train\_with\_budget}(\lambda, r_i) : \lambda \in T\}$    ▷ train model M
        $A \leftarrow \{\text{evaluate}(q_{accu}, m_\lambda) : m_\lambda \in M\}$    ▷ accuracy score
        $F \leftarrow \{\text{evaluate}(q_{fair}, m_\lambda) : m_\lambda \in M\}$    ▷ fairness score
        $V = []$
        for q ∈ Q do    ▷ iterate through all limits
            $v \leftarrow \{\text{evaluate}(q, m_\lambda) : q_\lambda \in Q\}$    ▷ get scores for each limits metric
            $V.add(v)$
        end for
        if $L[q][min] \leq v \geq L[q][max] \; \forall \; v \in V$ then    ▷ check limits
            $O \leftarrow o : \{\alpha * A[m_\lambda] + (1 - \alpha) * F[m_\lambda] : m_\lambda \in Q\}$    ▷ objective metric
        end if
        $k \leftarrow \lfloor n_i/n \rfloor$
        $T \leftarrow$ get_random_configurations($k, O, T, H$)
    end for
end for
return $\lambda^*$    ▷ configuration with optimal intermediate goal so far
```

5.5.1 Metrics

We chose the metrics to compare with other experimental procedures as well to inform other scenarios. For instance, where the user has de-identified data, which has lost fairness. All metrics used are defined in Chapter 2. We utilised three accuracy metrics: accuracy, balanced accuracy, and true-positive-rate defined in Chapter 2. We utilised two fairness metrics: disparate impact, and TP-ratio. We employed two efficiency metrics: percent-positive-outcome, and runtime defined in Chapter 2.

5.5.2 Comparison to Re-weight on balanced accuracy and disparate impact

For this experimentation, we first run the re-weight algorithm. The re-weight algorithm assigns weights that result in perfect equity of positive and negative outcomes between the unprivileged and privileged classes. Therefore, we take these values as the maximum value by which the user would likely desire to mitigate. Conversely, without any mitigation, the weights of every instance are equal to one. Thus, our search space for $W_{PP} \in [1, W_{PP}]$, for $W_{PN} \in [W_{PN}, 1]$, for $W_{UP} \in [1, W_{UP}]$, and for $W_{UN} \in [W_{UN}, 1]$. These ranges ensures that the correction does not exacerbate the fairness differential, and also does not excessively exceed parity and create a fairness imbalance in the opposite class. The search space is a continuous distribution. However, in cases where there is a discreet distribution given the optimisation technique has a maximum number of configurations set by the resource parameter, the number only needs to result in a greater number of configurations.

The experiments in the AIF360 implementation of the re-weight algorithm used Logistic Regression (LR) as a classifier and were performed on the Adult, German, and COMPAS datasets. The results are optimised using a brute force search over the classification thresholds between [0,1] with a step of 0.01 scored on balanced accuracy. The primary focus of these experiments was assessing the balanced accuracy vs. disparate impact. Our first method of comparison, is to use this classification threshold in the Mitiband mechanism at each stage and assess the scoring based on that result. Second, we ran Mitiband with a base threshold of 0.5 and ran applied the threshold only to the final result to mimic the AIF360 procedure. Significant to our these preliminary tests, was the lack of other metrics so we implemented these experiments to include pertinent other metrics including: **Accuracy**, **Percent-positive-outcomes**, **True-positive-rate**, and **Tp-ratio**. From this, we could assess the quality of the trade-off achieved and the cost associated with each trade-off level regarding other metrics.

5.5.3 Comparison to re-weight on true positive rate and TP-ratio with percent positive

The primary purpose of this comparison is to understand the difference in the results given completely different search spaces with the same or similar metrics. We assessed the range of outcomes that can be achieved from these different algorithms and assess the quality of the outcomes from the different search spaces.

5.5.4 Comparison to Grid Search and Random Search

The Mitiband optimisation approach uses the bandit based Hyperband mechanism. This approach uses the sklearn's BaseSearchCV, which has both for Grid Search and Random Search. These methods represent two alternative comparisons. Grid Search represents a comprehensive search for the optimal solution. The critical comparison with Grid Search is the resource cost to achieve this. We use runtime to highlight this. The other comparison is the Random Search method which has lower resource use but does not guarantee optimality. This comparison highlights the benefit of the successive halving aspect of the Hyperband algorithm to validate that this will find a better solution than simply Random Search. For these experiments, the same procedures were performed as in the comparison with the re-weight algorithm. We simply employed alternative search methodologies.

5.6 Results

This section presents the results and our analysis from the three experimental comparisons. The objective of these three comparisons is to analyse the results of this mechanism compared to benchmarks in the field of optimisation and fair machine learning.

5.6.1 Disparate Impact vs. Balanced Accuracy

We first compared the results achieved through Mitiband to those achieved with the original data and the re-weight mechanism. We performed the pre-processing in the manner of AIF360, and the testing was using Logistic Regression (LR). We performed a fivefold test and took the mean values of each of these tests. Figure 5.6.1 shows the results of tests performed on the original dataset and the re-weight mitigated dataset across the classification threshold optimised to balanced accuracy and the Mitiband dataset optimised on the trade-off metric with balanced accuracy and disparate impact and $\alpha = 0.5$. Thus, an equal trade-off between the two metrics. The most notable difference between these approaches is the search space in which they operate. The original and re-weight algorithm is optimised on a classification threshold. It therefore has a linear search space. Alternatively, the Mitiband mechanism searches through a range for all parameters of the re-weight method giving a multi-dimensional search space.

Since we set the minimum and maximum correction level for each parameter to the original weights derived for the re-weight process, the Mitiband search space is guided to an already optimised region of the search space. The space that Mitiband searches in is already in the upper end of the balanced accuracy. These prior limits result in lower initial values on the fairness metric. However, as the search progresses, Mitiband can discover

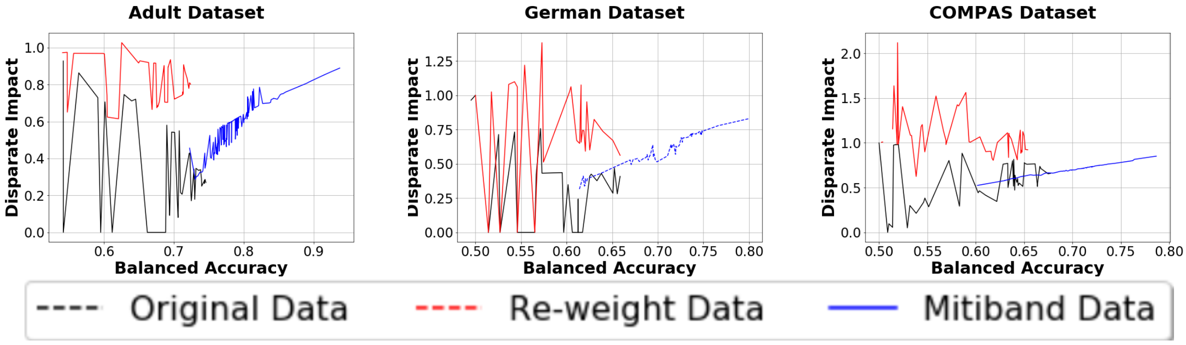


Figure 5.4: Results for level of disparate impact with increasing balanced accuracy.

better parameter settings that improve both the balanced accuracy and the disparate impact. This method leads to levels of fairness that approach 1.0 while also improving balanced accuracy.

It is also evident that the original and the re-weight mitigated data on all datasets can result in values greater than 1.0 for disparate impact. This means that the level of bias mitigation has over compensated for the level of bias due to the parameter values set from the training data. However, with Mitiband the mitigation parameters are not fixed and are searched through to find an optimised setting which is more suitable to the testing data. This approach allows the user to select the range for parameter optimisation, and will depend on whether this is being used as a pre-processing, in-processing, or post-processing technique.

On all three datasets we observe an improvement in balanced accuracy - disparate impact trade-off. However, the level of change and variation differs. This is a result of the difference in the original fairness differential. The Adult dataset has the highest level of disparate impact on the original data 0.29 compared with 0.38 for the German dataset and 0.66 for the COMPAS dataset. Thus higher disparate impact requires more significant weight changes to be applied to the Adult dataset groups. Thus, the search space between no mitigation and full re-weight mitigation was larger. This larger space leads to larger variation at the low end of the Mitiband results on the Adult dataset. In comparison, there is minor variation in the German dataset and minimal variation in the COMPAS dataset. Additionally, with the Adult dataset the variation can be observed to increase on multiple points on the graph. This successive variation is the result of successive rounds in which random configurations are added to the optimisation to utilise the budget and explore the search space. This is less noticeable on the narrower search space of the other two datasets.

5.6.2 True Positive and TP-ratio Difference vs. Percent Positive

We compared the results of Mitiband in the context of medical diagnostics. We compared the true-positive-rate, and the differential in the true positive ratio between the privileged and unprivileged classes TP-ratio. These metrics suppose a medical testing scenario, in which the objective is to maximise the detection rate while minimising false detection and, thus, overall positive outcomes.

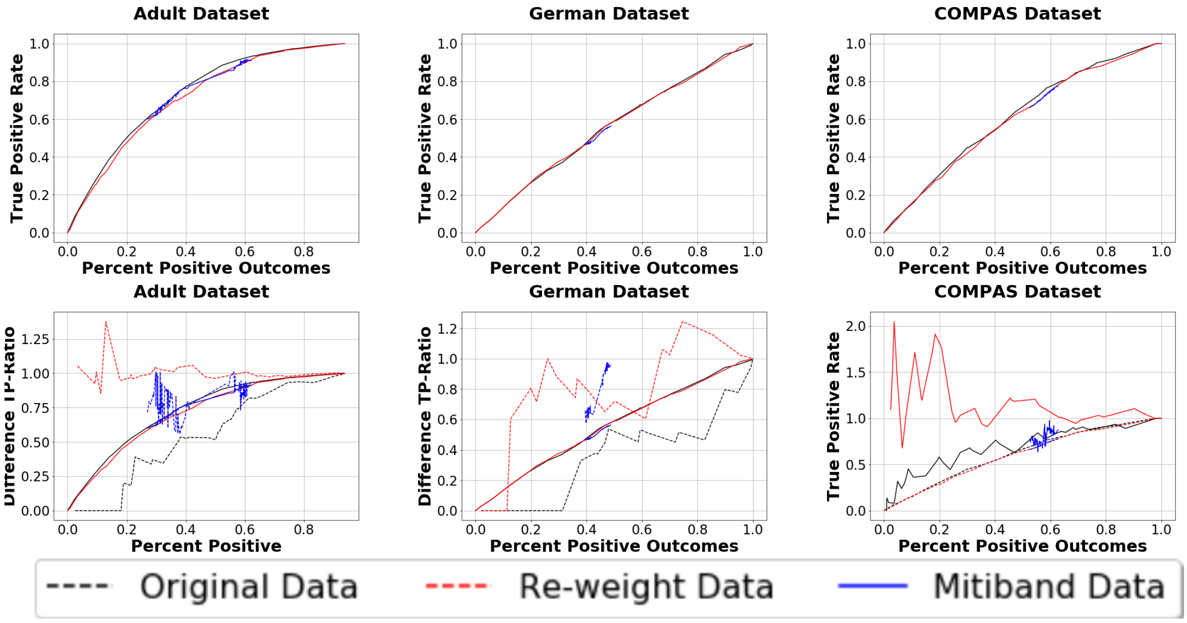


Figure 5.5: Results for true positive rate, and true positive ratio difference with increasing percent positive outcomes.

Figure 5.5 displays the results of these tests. The range of percentage of positive outcomes for the original and re-weight mitigated data is broad. Since AIF360 is optimised for balanced accuracy on the classification threshold between 0.01 and 0.99 with an interval of 0.1. These results range from 0% to 100% positive outcomes. The product is that a large portion of the search space would lack utility or predictive power. The classification / simply finds the most appropriate position in that space. The Mitiband algorithm searches through a narrower space working from the already optimised re-weight values. However, the values which it searches through can represent better fairness accuracy trade-off values.

In all cases, with the original and re-weight mitigated data the increase in positive outcomes leads to an increase in true positives. The results with Mitiband on the Adult dataset indicate that the true positive rate is equal to or grater that the re-weight optimised data at almost all levels of positive outcomes. For the German and COMPAS datasets, the level or true positives for Mitiband is roughly equal to that achieved by the re-weight mitigated results. However, when assessing the TP-ratio, the results of

Mitiband largely fit between those of the original data and the re-weight mitigated data. This results from the search space constraints. The TP-ratio metric is functionally related to disparate impact so the optimisation with $\alpha = 0.5$ for balanced accuracy and disparate impact allows the TP-ratio to converge towards parity. Specifically, with the German dataset, the Mitiband values increase significantly above the re-weight mitigation TP-ratio values.

Additionally, in all datasets the positive outcomes are less common so the model will generally be better at determining the majority outcomes. Thus, with an increase in percent positive outcomes, there is also likely to be an increase in true positives. This increase is likely to also increase the balanced accuracy. However, this will also likely lead to a reduction in accuracy. This is where the optimisation is effective at finding these parameters in the search space which lead to greater fairness while reducing loss to balanced accuracy. However, when dealing with raw accuracy, the trade-off could be more significant.

5.6.3 Runtime

The third metric we employed for comparison was runtime. Runtime is the processing time for the algorithm to take in the inputs and return an optimised solution, a parameter configuration. The runtime is measured by the CPU-clock in seconds to three decimal places. Table 5.1 shows the time complexity for all the algorithms for each of the three datasets.

Table 5.1: Runtime for Different Algorithms

Algorithm	Adult	German	COMPAS
Re-weight	1.4	0.29	0.39
Random Search	33.26 (6.53)	1.08 (0.04)	2.92 (0.13)
Grid Search	111.76 (5.70)	19.16 (0.68)	69.90 (2.28)
Mitiband	43.52 (2.59)	2.64 (.018)	5.93 (0.44)

Mitiband employs an optimisation mechanism, thus, we demonstrate how this search method can be more efficient than simple Random Search or Grid Search. This implementation of Mitiband optimises on the re-weighting procedure. The re-weight procedure is performed before the optimisation phase to obtain the initial weights. Thus, Mitiband will always exceed the time complexity for the re-weight algorithm on its own.

Table 5.1 highlights the difference in runtime for these three different optimisation techniques. Grid Search iterates through all combinations of the parameter values and thus is the longest. Random Search samples a selection of parameter combinations within the search space and evaluates them on the same budget. Mitiband starts by randomly selecting the number of configurations set by the budget then evaluates them on a lower number of iterations. The processing time is higher than random search due to the extra check for multiple parameters, and the greater number of configurations it samples. However, the time is not significantly greater, and it samples significantly more configurations.

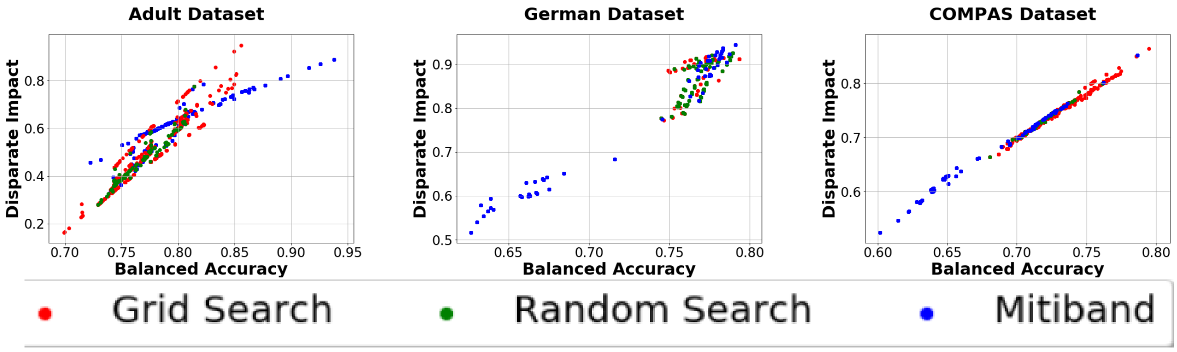


Figure 5.6: Results for level of disparate impact with increasing balanced accuracy on, Grid Search, Random Search, and Mitiband.

Furthermore, to analyse the efficiency of each optimisation method we compared the results of Mitiband to those achieved through Random Search and Grid Search. We compared these three optimisation techniques on how effective they were at finding a trade-off between fairness (disparate impact) and balanced accuracy. Figure 5.6 displays the fairness accuracy trade-off between the three optimisation techniques. Firstly, the Adult dataset results highlight the improvement of the Mitiband method over simple Random Search. The values for random search are spread evenly across the range of search space. Mitiband has some variation, however, the general tendency is to improve the trade-off between fairness and accuracy, so these values are clustered around a singular axis. There is a clear trajectory that is guided by the multiple rounds of successive halving which, reduces the search space to only the more favourable regions.

One significant factor with this analysis is the limitation with the Grid Search. The Grid Search must be set to a span with an interval instead of a continuous variable. This means that the scope of the exhaustive search is only over all interval values. However, parameter combinations not lying on the interval might be explored with the Mitiband. Additionally, if an optimal region lies between these intervals, the Mitiband will guide the search towards this region while the Grid Search will simply be unaware of it. This appears to be the case on the upper end of the Mitiband algorithm results on the Adult dataset.

Both the German and COMPAS datasets have smaller search spaces, and smaller interval steps as well. This appears to allow the Grid Search to analyse the search space more effectively. However, Mitiband still finds a nearly equal trade-off for both the German and COMPAS dataset at the high end. Random search is also more effective on these two datasets. This is due to the smaller search space as well and with tighter bounds on the results. To thoroughly investigate the Grid Search approach would require an infeasible budget and runtime.

5.7 Summary

In this chapter, we presented the Mitiband mechanism, which uses bandit based hyperparameter tuning to optimise the fairness and accuracy trade-off of a mitigation mechanism. We first discussed our motivation, to achieve a better fairness accuracy trade-off given set thresholds for certain metrics. We discussed the background work from which this method was derived. Then we described how this method operates. Following this, we discussed the experimental processes we undertook to validate the efficacy of our mechanism.

We found that Mitiband can find better fairness accuracy trade-offs when the original fairness differential is large. In terms of runtime, the Mitiband algorithm takes only marginally longer than Random Search and can find better trade-offs in the search space. On datasets with lower original fairness differential the added benefit of the Mitiband algorithm is less perceptible since Random Search is more likely to select a close to optimal solution. One gap was the inability to use Grid Search to analyse the entire search space extensively. However, when dealing with very small datasets, the Grid Search was able to cover sufficient space and appears to have acceptable runtimes, thus, might still be preferred in these cases.

6 Conclusion

This chapter summarises our work and discusses the limitations and promising future research directions. Section 6.1 highlights major achievements from this research. Section 6.2 informs of the primary limitations of our work. Section 6.3 discusses promising directions for future research.

6.1 Achievements

The following list highlights the achievements of our research by chapter:

Chapter 3

- We conducted numerous experiments using novel measures to understand the effects of de-identification on fairness.
- We utilised these experimental techniques and novel bias measures to develop a bias-assessment framework that assesses the original level of bias and accuracy, and can assess the change in fairness and accuracy introduced by the de-identification process.

Chapter 4

- We conducted experiments to understand better the effects of de-identification on bias mitigation.

- We employed these experimental techniques to enhance the bias assessment framework, and to isolate and assess the change in fairness and accuracy due to the bias mitigation process, on the original and de-identified.

Chapter 5

- We developed a bandit-based hyper-parameter tuning mechanism that can assess the mitigation mechanism and search through the hyper-parameters of that mechanism to efficiently find an optimised fairness-accuracy trade-off on de-identified data with efficient runtime.

6.2 Limitations

Here we outline the primary limitations of this research. We discuss limitations related to research conducted in Chapters 3, 4, and 5. These limitations relate to users knowledge and the nature of mitigation procedures.

6.2.1 User Domain Knowledge

The bias-assessment framework runs an automated process that can assess the level of bias in the original, de-identified, and bias-mitigated data. However, it is necessary to know the data types and the attributes that correlate to the target in order to optimise this process. The framework requires the users to have a strong understanding of the data as well as statistical processes. Feature selection is a particularly challenging area to automate, and may require the user to perform preliminary investigations to determine the input attributes to the bias assessment framework.

6.2.2 Runtime

Hyper-parameter optimisation is a computationally expensive process, and the possible need to run this on numerous scenarios for different protected classes or target attributes results in potentially a time consuming process. Notably, in business cases, where the notion of fairness is not mandated but is voluntary, this could prove as a disincentive to employ such methodologies.

Another aspect of the processing time is the comparative advantage it presents. Given the slight advantage with Grid Search, in scenarios with small size of data and fast overall processing time, users might prefer an exhaustive Grid search. However, even with the Adult dataset, a medium-size dataset, Mitiband band shows clear advantages over both Grid Search and Random Search.

6.2.3 Data Manipulation

As mentioned in Chapter 2, bias mitigation to improve fairness runs counter to accuracy improvement. Since these are contradictory aims, the optimisation approach lends itself well to finding a trade-off between mitigation and fairness and the accuracy lost. However, if mitigation were improving the model at no cost, the mitigated model could simply be considered an improved model. Thus, mitigation will inevitably have a cost to it. This cost is in relation to the accuracy on the observed data.

When users seek to acquire data published by an organisation, the user may prefer to receive the data unprocessed, or processed as little as possible. Certain privacy restrictions may mandate that the data must be de-identified but users may prefer that the bias-mitigation is not performed in order to reduce the perceived alteration to the data.

6.3 Future Directions

The main focus of future research involves broader implementation. De-identification and mitigation procedures are less grounded in statistical theory as many machine learning processes. Building this foundation through research and real-world use would benefit this research and the adoption of these mechanisms. The following avenues present particularly promising future directions.

6.3.1 Mitigation Method Expansion

The implementation of this framework utilises the re-weight [49] mitigation method. This method was implemented for the straightforward nature of this technique which enables precise testing and comparisons. Further research into implementing other mitigation methods including those that transform the data into distributions rather than transformed datasets, would help determine the differential benefits achieved through these varying mitigation mechanisms.

As discussed in Section 4, mitigation can be performed at all stages of the data processing pipeline. This framework was implemented to be deployed at the pre-processing stage. However, this framework could be implemented with minor adaptation to optimise on in-processing and post-processing mitigation methods. This method employs classification for the assessment of the outcomes based on the metrics chosen. Additionally, since classification is already performed as part of the optimisation, simply adding classification hyper-parameters would allow for optimisation over the hyper-parameters of both the mitigation method and the classification approach. Furthermore, hyper-parameter tuning is an established method for determining the parameters of a neural network. Thus, the optimisation approach could be implemented to determine the best parameters to optimise fairness aware neural networks.

6.3.2 Field Implementation

Given the real-world implications of this research, implementation of these frameworks in different real-world use cases could allow for testing on a much greater scale and with new de-identification and bias mitigation techniques as they arise. This would allow for refinement of the processes and allow for greater acceptance of the bias mitigation process.

6.3.3 Method Standardisation

Currently, there are no universally known frameworks that can perform bias assessment and mitigation. Users must piece together solutions to their task. If fairness and privacy continue to become a greater concern, employment of these mechanisms in real-world use could improve the understanding of the complex interaction between these de-identification and mitigation methods and could show particular methods or mechanisms to be more effective in certain scenarios. With this information, standards of practice could be developed with known privacy and fairness outcomes, enabling the optimisation process to take place in a reduced search space.

6.3.4 Budget Optimisation

When employing Mitiband we utilised the classification threshold chosen based on the mitigation mechanism. Another direction could be to use a small portion of the search budget to sample and analyse results to optimise the classification threshold at each round however the effects this would have need to be explored further.

A
Appendix

A.1 Reproduceability

This appendix covers three key areas: the datasets, the terms used and the framework's code. First, we outline the datasets and where thy are accessed. Second, we provide a detailed index of all variables used. Third, we discuss the code and the necessary packages, and where the code is available.

A.1.1 Datasets

For all of our testing we used five datasets.

The Adult dataset, described in Chapter 2, is a set of record from the 1996 US census. This is a multivariate dataset with 48842 records and 14 attributes. This dataset is available at .

The German dataset is a described in Chapter 2 is a record of loan requests with approved or not approved. This is a multi variate dataset with 1000 records and 20 attributes. This dataset is available at .

The COMPAS dataset is described in Chapter 2. This is a record of a parole decision support system and a two year follow up record as to whether the individual re-offended. There are 7214 records and 56 attributes. This dataset is the 'compas-scores-two-years-violent.csv' file available at .

The MEPS dataset is described in Chapter 2. This is a set of patient records form both household (demographic) information to insurance (health status) information. This is a composite dataset. The 2015 file contains data from rounds 3,4,5 of panel 19 (2014) and rounds 1,2,3 of panel 20 (2015). The 2016 file contains data from rounds 3,4,5 of panel 20 (2015) and rounds 1,2,3 of panel 21 (2016). There are 15830 records with 139 attributes. The files are ‘h181.csv’ and ‘h192.csv’ available for download at .

The Synthetic data generator is a custom data generator that selectively samples a dataset. We input a dataset and select the protected attribute and the Target attribute and the level of imbalance. If both of these are binary attributes we would have four data classes and the and size of the smallest class determines how large the sample can be. We implemented this on the adult dataset producing a dataset of 8000 records. However, given knowledge of the protected attribute and the target attribute this could be used on any dataset. This is the ‘data_generator’ Jupyter notebook available at: .

A.1.2 Variable Index

This section covers the terms and variables used throughout this thesis. The variables are divide by the chapter they are first defined in:

Chapter 3

- D = dataset
- D^* = processed dataset
- $\hat{D}$ = De-identified data
- $\bar{D}$ = processed De-identified data
- C = protected class
- C_0 = unprivileged
- C_1 = privileged
- X = other variables
- Y = Target
- Y^+ = positive outcome
- Y^- = negative outcome
- $bias$ = bias depending on metric eg. disparate impact
- M = model

- M^* = model from de-identified data
- P = predictions
- P^* = predictions from de-identified data model
- ρ = percent of positive prediction
- β = percent positive true
- δ = ratio of positive outcomes lost
- Δ = difference between positive privileged and unprivileged class in terms of, ratio of positive outcomes lost

Chapter 4

- D^{*g} = original processed mitigated data
- $\hat{D}_i$ = various de-identification levels processed
- $\hat{D}_i^g$ = various levels of de-identification with mitigation applied

Chapter 5

- W_{PP} = weight for class which is privileged with positive outcomes
- W_{PN} = weight for class which is privileged with negative outcomes
- W_{UP} = weight for class which is unprivileged with positive outcomes
- W_{UN} = weight for class which is unprivileged with negative outcomes
- γ = constant
- ϵ = integration constant
- μ = direction of α change for trade off metric
- n = number of configurations
- B = total budget
- η = ratio of pruned configurations / budget increase per configurations
- R = maximum number of resources to a single configurations
- s a bracket in which successive halving is performed

- s_{max} = number of brackets for whole process
- Q = set of assessment metrics
- q = assessment metric
- q_a = accuracy metric
- q_f = fairness metric
- α = trade-off parameter
- o = objective measure
- O = set of objective metric for acceptable configurations
- v = scoring value
- V = set of values for different values for all acceptable configurations from each round
- h = hyper parameter set
- H = set of all hyper-parameter configurations
- k = number of configurations for next round
- T = set of configurations selected for next round
- λ = current configuration in T

A.1.3 Code

All of the code was written in Python using version 3. We preformed the all the experiments in Jupyter notebooks version 1.2.6. We used a python three virtual environment with numerous packages. The core packages we employed were:

- scikit-learn: version 0.22.1
- numpy: version 1.18.1
- pandas: version 1.1.3
- matplotlib: version 3.1.3
- AIF360: version 0.3.0

We performed all the experiments using an HP Omen laptop model year 2016. We ran Linux Ubuntu 20.04. The specifications were: a 64-bit, Intel Core i7-6700HQ CPU @ 2.60GHZ, there were 16GB of ram and partition of 230GB for storage. The base files and the Juyter notebooks are available at Each notebook is used for one or more tests. If multiple tests are performed variables all variable options are available and can be selected for different tests.

Bibliography

[1] Nabil R Adam and John C Wortmann. A Comparative Methods Study for Statistical Databases. *ACM Comp. Surveys*, 21(4):906–918, 1989.

[2] Julius A Adebayo et al. *FairML: ToolBox for diagnosing bias in predictive modeling.* PhD thesis, Massachusetts Institute of Technology, 2016.

[3] Charu C Aggarwal and S Yu Philip. A general survey of privacy-preserving data mining models and algorithms. In *Privacy-preserving data mining*, pages 11–52. Springer, Boston, 2008.

[4] Charu C. Aggarwal and Philip S. Yu. A framework for condensation-based anonymization of string data. *Data Mining and Knowledge Discovery*, 16(3):251–275, 2008.

[5] Dakshi Agrawal and Charu C. Aggarwal. On the design and quantification of privacy preserving data mining algorithms. In *Proceedings of the Twentieth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, PODS '01, page 247–255, New York, NY, USA, 2001. Association for Computing Machinery.

[6] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. *ProPublica, May*, 23:2016, 2016.

[7] Mike Atallah, Elisa Bertino, Ahmed Elmagarmid, Mohamed Ibrahim, and Vassilios Verykios. Disclosure limitation of sensitive rules. In *Proceedings 1999 Workshop on Knowledge and Data Engineering Exchange (KDEX'99)(Cat. No. PR00453)*, pages 45–52, Chicago, 1999. IEEE, KDEX'99.

[8] Jean-Yves Audibert, Sébastien Bubeck, and Rémi Munos. Best arm identification in multi-armed bandits. In *COLT*, pages 41–53, 2010.

[9] Brendan Avent, Javier Gonzalez, Tom Diethe, Andrei Paleyes, and Borja Balle. Automatic discovery of privacy-utility pareto fronts. *arXiv preprint arXiv:1905.10862*, 2019.

[10] Niels Bantilan. Themis-ml: A fairness-aware machine learning interface for end-to-end discrimination discovery and mitigation. *Journal of Technology in Human Services*, 36(1):15–30, 2018.

[11] Solon Barocas and Andrew D. Selbst. Big data's disparate impact. *California Law Review*, 104:671, 2016.

[12] Rachel KE Bellamy, Kuntal Dey, Michael Hind, Samuel C Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, A Mojsilović, et al. Ai fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM Journal of Research and Development*, 63(4/5):4–1, 2019.

[13] Rachel KE Bellamy, Kuntal Dey, Michael Hind, Samuel C Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, A Mojsilović, et al. Ai fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. Technical Report 4/5, IBM, 2019.

[14] Philippe Besse, Eustasio del Barrio, Paula Gordaliza, Jean-Michel Loubes, and Laurent Risser. A survey of bias in machine learning through the prism of statistical parity for the adult data set, 2020.

[15] Toon Calders and Sicco Verwer. Three naive bayes approaches for discrimination-free classification. *Data Mining and Knowledge Discovery*, 21(2):277–292, 2010.

[16] Flavio Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R Varshney. Optimized pre-processing for discrimination prevention. In *Advances in Neural Information Processing Systems*, pages 3992–4001, Boston, 2017. Mit Press.

[17] Andrew Chester, Yun Sing Koh, Jörg Wicker, Quan Sun, and Junjae Lee. Balancing utility and fairness against privacy in medical data. In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1226–1233. IEEE, 2020.

[18] Kevin A. Clarke. The phantom menace: Omitted variable bias in econometric research. *Conflict Management and Peace Science*, 22(4):341–352, 2005.

[19] Various Contributors. Catalogue of bias, 2017.

[20] André F Cruz, Pedro Saleiro, Catarina Belém, Carlos Soares, and Pedro Bizarro. A bandit-based algorithm for fairness-aware hyperparameter optimization. *arXiv preprint arXiv:2010.03665*, 2020.

[21] Rachel Cummings, Dhamma Kimpara, Varun Gupta, and Jamie Morgenstern. On the compatibility of privacy and fairness. *ACM UMAP 2019 Adjunct - Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*, 0(FairUMAP):309–315, 2019.

[22] R. G.P.J. de Jong, A. M. Burden, S. de Kort, M. P.P. van Herk-Sukel, P. A.J. Vissers, P. K.C. Janssen, H. R. Haak, A. A.M. Masclee, F. de Vries, and M. L.G. Janssen-Heijnen. Impact of detection bias on the risk of gastrointestinal cancer and its subsites in type 2 diabetes mellitus. *European Journal of Cancer*, 79:61–71, 2017.

[23] Kalyanmoy Deb. Multi-objective optimization. In *Search methodologies*, pages 403–449. Springer, 2014.

[24] Thomas G Dietterich and Eun Bae Kong. Machine learning bias, statistical bias, and statistical variance of decision tree algorithms. Technical report, Department of Computer Science, Oregon State University, 1995.

[25] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017.

[26] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3876 LNCS:265–284, 2006.

[27] Michael D Ekstrand, Rezvan Joshaghani, and Hoda Mehrpouyan. Privacy for All: Ensuring Fair and Equitable Privacy Protections. *Machine Learning Research*, 81:1–13, 2018.

[28] Eyal Even-Dar, Shie Mannor, Yishay Mansour, and Sridhar Mahadevan. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of machine learning research*, 7(6), 2006.

[29] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 259–268, New York, NY, USA, 2015. Association for Computing Machinery.

[30] Sam Fletcher and Md Zahidul Islam. Measuring Information Quality for Privacy Preserving Data Mining. *International Journal of Computer Theory and Engineering*, 7(1):21–28, 2014.

[31] Sorelle A Friedler, Carlos Scheidegger, Suresh Venkatasubramanian, Sonam Choudhary, Evan P Hamilton, and Derek Roth. A comparative study of fairness-enhancing interventions in machine learning. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 329–338, New York, NY, USA, 2019. Association for Computing Machinery.

[32] W Fuller. Masking procedures for microdata disclosure. *Journal of Official Statistics*, 9(2):383–406, 1993.

[33] Benjamin CM Fung, Ke Wang, Rui Chen, and Philip S Yu. Privacy-preserving data publishing: A survey of recent developments. *ACM Computing Surveys (Csur)*, 42(4):1–53, 2010.

[34] Quan Geng, Wei Ding, Ruiqi Guo, and Sanjiv Kumar. Tight analysis of privacy and utility tradeoff in approximate differential privacy. In *International Conference on Artificial Intelligence and Statistics*, pages 89–99, 2020.

[35] Guo Haixiang, Li Yijing, Jennifer Shang, Gu Mingyun, and Huang Yuanyue. Learning from class-imbalanced data : Review of methods and applications. *Expert Systems With Applications*, 73:220–239, 2017.

[36] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*, pages 3315–3323, Red Hook, New York, 2016. Curran Associates, Inc.

[37] Charles R. Harris, K. Jarrod Millman, Stéfan J van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585:357–362, 2020.

[38] Kathryn L Hassell. Population estimates of sickle cell disease in the us. *American journal of preventive medicine*, 38(4):S512–S521, 2010.

[39] Susan E. Herz. Don't Test, Do Sell: Legal Implications of Inclusion and Exclusion of Women in Clinical Drug Trials. *Epilepsia*, 38(s4):S42–S49, 1997.

[40] Naoise Holohan, Stefano Braghin, Pól Mac Aonghusa, and Killian Levacher. Diffprivlib: the ibm differential privacy library. *arXiv preprint arXiv:1907.02444*, 2019.

[41] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007.

[42] Isabelle Hupont and Carles Fernández. Demogpairs: Quantifying the impact of demographic imbalance in deep face recognition. In *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*, pages 1–7, Lille, France, 2019. IEEE, IEEE.

[43] Vasileios Iosifidis and Eirini Ntoutsi. Fabboo-online fairness-aware learning under class imbalance. In *Discovery Science*, 2020.

[44] V. S. Iyengar. Transforming Data to Satisfy Privacy Constraints. In *KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 279–288, New York, NY, USA, 2002. ACM.

[45] Kevin Jamieson, Matthew Malloy, Robert Nowak, and Sébastien Bubeck. lil'ucb: An optimal exploration algorithm for multi-armed bandits. In *Conference on Learning Theory*, pages 423–439. PMLR, 2014.

[46] Kevin Jamieson and Ameet Talwalkar. Non-stochastic best arm identification and hyperparameter optimization. In *Artificial Intelligence and Statistics*, pages 240–248. PMLR, 2016.

[47] Nathalie Japkowicz and Sha ju Stephen. The Class Imbalan e Problem: A Systemati Study. *Drugs and Therapy Perspectives*, 12(7):10, 1998.

[48] Shivaram Kalyanakrishnan, Ambuj Tewari, Peter Auer, and Peter Stone. Pac subset selection in stochastic multi-armed bandits. In *ICML*, volume 12, pages 655–662, 2012.

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

[50] Faisal Kamiran, Asim Karim, and Xiangliang Zhang. Decision theory for discrimination-aware classification. In *2012 IEEE 12th International Conference on Data Mining*, pages 924–929, Brussels, Belgium, 2012. IEEE, ICDM'12.

[51] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 35–50, Berlin, Heidelberg, 2012. Springer, Springer.

[52] Hillol Kargupta, Souptik Datta, Qi Wang, and Krishnamoorthy Sivakumar. On the Privacy Preserving Properties of Random Data Perturbation Techniques. In *Third IEEE International Conference on Data Mining*, pages 99–106, Melbourne, FL, 2003. IEEE.

[53] Zohar Karnin, Tomer Koren, and Oren Somekh. Almost optimal exploration in multi-armed bandits. In *International Conference on Machine Learning*, pages 1238–1246. PMLR, 2013.

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

[55] Nitin Kohli and Paul Laskowski. Epsilon voting: Mechanism design for parameter selection in differential privacy. In *2018 IEEE Symposium on Privacy-Aware Computing (PAC)*, pages 19–30. IEEE, 2018.

[56] Emmanouil Krasanakis. Adaptive Sensitive Reweighting to Mitigate Bias in Fairness-aware Classification. In *WWW '18: Proceedings of the 2018 World Wide Web Conference*, volume 2, pages 853–862, Lyon, 2018. International World Wide Web Conferences Steering Committee.

[57] Andreas Krause and Eric Horvitz. A utility-theoretic approach to privacy and personalization. In *Proceedings of the National Conference on Artificial Intelligence*, volume 2, pages 1181–1188, 01 2008.

[58] Joffrey L. Leevy, Taghi M. Khoshgoftaar, Richard A. Bauder, and Naeem Seliya. A survey on addressing high-class imbalance in big data. *Journal of Big Data*, 5(1):42, 2018.

[59] Kristen LeFevre, David J. DeWitt, and Raghu Ramakrishnan. Mondrian multidimensional K-anonymity. *Proceedings - International Conference on Data Engineering*, 2006:25, 2006.

[60] Lisha Li, Kevin Jamieson, Giulia DeSalvo, Afshin Rostamizadeh, and Ameet Talwalkar. Hyperband: A novel bandit-based approach to hyperparameter optimization. *The Journal of Machine Learning Research*, 18(1):6765–6816, 2017.

[61] Ninghui Li. t -Closeness : Privacy Beyond k -Anonymity and -Diversity. In *IEEE 23rd International Conference on Data Engineering*, number 2, pages 106—-116, Istanbul, Turkey, 2007. IEEE.

[62] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data*, 1(1):3–es, 2007.

[63] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *arXiv*, 2019.

[64] Ricardo Mendes and Joao P. Vilela. Privacy-Preserving Data Mining: Methods, Metrics, and Applications. *IEEE Access*, 5:10562–10582, 2017.

[65] Jason W Osborne. Data Cleaning Basics : Best Practices in Dealing with Extreme Scores. *Newborn and Infant Nursing Reviews*, 10(1):37–43, 2010.

[66] Delroy L. Paulhus. *Measurement and Control of Response Bias.* Academic Press, Inc., Washington, DC, third revised and enlarged edition edition, 1991.

[67] 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 E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[68] Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q Weinberger. On fairness and calibration. In *Advances in Neural Information Processing Systems*, pages 5680–5689, Red Hook, New York, 2017. Curran Associates, Inc.

[69] David Pujol, Ryan McKenna, Satya Kuppam, Michael Hay, Ashwin Machanavajjhala, and Gerome Miklau. Fair decision making using privacy-protected data. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 189–199, 2020.

[70] Andrew Rundle, Yun Wang, Sudha Sadasivan, Dhananjay A Chitale, Nilesh S Gupta, Deliang Tang, and Benjamin A Rybicki. Larger men have larger prostates: detection bias in epidemiologic studies of obesity and prostate cancer risk. *The Prostate*, 77(9):949–954, 2017.

[71] Pedro Saleiro, Benedict Kuester, Abby Stevens, Ari Anisfeld, Loren Hinkson, Jesse London, and Rayid Ghani. Aequitas: A bias and fairness audit toolkit, 2018.

[72] Andrzej Skowron and Cecylia Rauszer. The discernibility matrices and functions in information systems. In *Intelligent decision support*, pages 331–362. Springer, Dordrecht, 1992.

[73] Peter M. Steiner and Yongnam Kim. The Mechanics of Omitted Variable Bias: Bias Amplification and Cancellation of Offsetting Biases. *Journal of Causal Inference*, 4(2):20160009, 2016.

[74] Latanya Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. *International Journal of Uncertainty, Fuzziness and Knowlege-Based Systems*, 10(5):571–588, 2002.

[75] Latanya Sweeney. k-Anonymity: A Model for Protecting Privacy. *International Journal on Uncertainty,Fuzziness and Knowledge-based Systems*, 10(5):1–14, 2002.

[76] Yufei Tao, Hekang Chen, Xiaokui Xiao, Shuigeng Zhou, and Donghui Zhang. Angel: Enhancing the utility of generalization for privacy preserving publication. *IEEE transactions on knowledge and data engineering*, 21(7):1073–1087, 2009.

[77] Florian Tramer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, Jean-Pierre Hubaux, Mathias Humbert, Ari Juels, and Huang Lin. Fairtest: Discovering unwarranted associations in data-driven applications. In *2017 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 401–416, Paris, France, 2017. IEEE, EuroS&P.

[78] Ms Vigneswari, N Komal Kumar, G V Bharath Kumar, M Vamsi Krishna, Andhra Pradesh, and Andhra Pradesh. Performance Comparison of Privacy Preserving Perturbation algorithms in Association Rule Mining. In *Proceedings of the International Conference for Phoenixes on Emerging Current Trends in Engineering and Management (PECTEAM 2018)*, volume 142, pages 1–6, Amsterdam Netherlands, 2018. Atlantis Press.

[79] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, CJ Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake Vand erPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1. 0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

[80] Morgan Wadams and Tanya Park. Qualitative Research in Correctional Settings: Researcher Bias, Western Ideological Influences, and Social Justice. *Journal of forensic nursing*, 14(2):72–79, 2018.

[81] Christina Wadsworth, Francesca Vera, and Chris Piech. Achieving Fairness through Adversarial Learning: an Application to Recidivism Prediction. *FAT/ML*, July, 2018.

[82] Robin M. Weinick, Samuel H. Zuvekas, and Joel W. Cohen. Racial and ethnic differences in access to and use of health care services, 1977 to 1996. *Medical Care Research and Review*, 57(SUPPL. 1):36–54, 2000.

[83] Wes McKinney. Data Structures for Statistical Computing in Python. In Stéfan van der Walt and Jarrod Millman, editors, *Proceedings of the 9th Python in Science Conference*, pages 56 – 61, 2010.

[84] Xiaokui Xiao and Yufei Tao. Anatomy: Simple and effective privacy preservation. In *Proceedings of the 32nd international conference on Very large data bases*, pages 139–150, Seoul Korea, 2006. VLDB Endowment, ACM.

[85] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *International Conference on Machine Learning*, pages 325–333, Atlanta, Georgia, 2013. PMLR.

[86] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating Unwanted Biases with Adversarial Learning. In *AIES 2018 - Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 335–340, New York, NY, USA, 2018. Association for Computing Machinery.

[87] Jinhua Zhang, Jian Zhuang, Haifeng Du, et al. Self-organizing genetic algorithm based tuning of pid controllers. *Information Sciences*, 179(7):1007–1018, 2009.