Assessing fairness/bias in binary classification machine learning models on consumers

A dissertation submitted in partial fulfilment of the requirements for the MSc in Data Analytics

By Benedict Faria

Department of Computer Science and Information Systems
Birkbeck, University of London

September 2019

Supervisor: Roman Kontchakov

Word count of core report: 12,064

This report is substantially the result of my own work except where explicitly indicated in the text. I give my permission for it to be submitted to the TURNITIN Plagiarism Detection Service. I have read and understood the sections on plagiarism in the Programme Handbook and the College website. The report may be freely copied and distributed provided the source is explicitly acknowledged.

Acknowledgements

I would like to thank my supervisor Roman Kontchakov for his help and support. A huge thanks to Ernest Chow and Gareth Jones from Experian for their patience and real-world feedback on all stages of the project. Finally, thank you Frances for allowing me the space to do this.

Abstract

Measuring and mitigating bias in machine learning classification algorithms is relatively a new field in data science. There are numerous examples of bias in classification algorithms. There are no current examples that have been successfully challenged under discrimination law. Nevertheless, it is becoming imperative that this bias is addressed and mitigated against. Left untreated, bias in classification datasets can expose businesses to legal risk, and limit opportunity.

Datasets used to train classification models are the main source of bias. They often comprise data and feature selections that reflect historic social and economic disparities. Classifiers work by finding statistical relationships and patterns between attributes in the training dataset. Bias emerges when a sensitive attribute such as gender or race exhibits an overwhelming influence of the classifier's outcome, usually unintended. Mitigation techniques attempt to minimise this influence without compromising model performance.

This assessment reviewed the various approaches in mitigating bias in datasets. Bias was measured in terms of achieving fairness for individuals and groups. It used selected metrics and mitigation algorithms from the AIF360 open-source toolkit to measure and remove bias from three public datasets. Baseline measures of bias were compared with values from the post-mitigation process to assess the effectiveness of each mitigation algorithm. The results were discussed in terms of maximising bias mitigation and classifier performance. This report also discussed the business impacts of debiasing classification algorithms.

Table of Contents

List of Tables

List of Figures

1. Introduction

In law, bias or discrimination refers to unfair treatment of individuals because of their membership of a certain group. The bias could be based on attributes like ethnicity, gender, age, language, employment, or income level. It could result in adverse outcomes for certain individuals in education admissions, housing allocation, employment or credit scores. An early outcome of big data analytics was the widespread use of machine learning classification algorithms in consumer focussed applications. A 2014 US Government [1] report found that "big data analytics has the potential to eclipse longstanding civil rights protections in how personal information is used in housing, credit, employment, health, education, and the marketplace". It acknowledges that whilst unintended, biased outcomes from machine learning models stem from their underlying biased training data, which in turn reflects historical or institutional bias prevalent in society.

Bias in machine learning classifiers is now recognised as an issue that needs addressing. This is a new field of study in Data Science, with a growing body of literature [2] and evolving definitions and taxonomies of bias[3][4]. However proposals to outlaw this bias [5] are now forcing businesses to reassess their training data and subsequent algorithms upon which they rely on to make informed decisions.

This report assesses current approaches to measuring and mitigating bias in classification machine learning models. The assessment uses three datasets to measure the bias, implements mitigation algorithms, and compares the effectiveness of each algorithm. All bias metrics and algorithms are selected from the open-source AI Fairness 360 (AIF360) toolkit [6].

1.1 Assessment Objectives

This assessment set out to achieve the following objectives

- Objective 1: Measure data set and classification model bias using selected metrics from the AIF360 toolkit.
- Objective 2: Apply selected bias-mitigation algorithms to the data sets.
- Objective 3: Measure the effect of each bias-mitigation algorithm on model performance
- Objective 4: Assess the business implications of bias-mitigation algorithms.

Preliminary work for this assessment included testing for and mitigating bias in the Titanic training dataset [7], chosen for it's overwhelming evidence of bias. The dataset contains the passenger list of the Titanic, and records who survived and who didn't. The dataset's data objects refer to each passenger. This dataset is useful in establishing key terms that are used in this report. The **Sensitive attribute** refers to the dataset attribute that bias is measured for. The Sensitive attribute for this project is binary, containing values for **privileged** and **unprivileged groups**. The **label** refers to the outcome of the machine learning classification for each data object. The label normally comprises two classes, and contain values pertaining to a **favourable** class or **unfavourable** class for that data object. In the Titanic dataset's case, the Sensitive attribute is the Sex attribute, with females being the privileged group and males being the unprivileged group, as an overwhelming number of males perished. The label is the Survived column which indicates whether the passenger survived or perished. A passenger who survived has a favourable label, and the ones who didn't have an unfavourable label.

As in the Titanic example above, the all datasets used in this report have a binary Sensitive attribute and a two class Label.

1.2 Structure of the report

Chapter 2 discusses sources of bias, their measurement and mitigation approaches. Chapter 3 provides an overview and scope of the assessment. It discusses the datasets assessed for bias, the bias metrics used, mitigation algorithms deployed. Chapter 4 describes the implementation approach, feature engineering, initial data discovery and success criteria for the selected mitigation algorithms from the AIF360 toolkit. Chapter 5 discusses the results of the assessment and whether each mitigation algorithm met its success criteria. Chapter 6 briefly discusses the business impacts of bias mitigation. Chapter 7 offers a conclusion and suggestions for further development.

2. Background / Literature review

In the context of this report, bias in classification machine learning models is described as a phenomenon whereby the classifier makes predictions that favour individuals or groups of individuals over others. The quality of a classifier depends almost entirely on the quality of the training data used to train it. Hence training data is seen as the main source of bias in classification outcomes. As referred to in the Proposal, bias is introduced to training datasets in several ways, including under- or over-representation of individuals in the dataset, inaccurate labelling and inadequate feature engineering.

Bias is measured in terms of achieving fairness in classification model outcomes. The rest of this section discusses the fairness metrics and bias-mitigation algorithms used for the assessment.

2.1 Fairness Metrics - an approach to measuring bias

Fairness metrics are used to measure the scale of bias in a dataset or a classification's predicted labels. Bias mitigation algorithms seek to achieve fairness for unprivileged individuals and groups of individuals by improving the fairness metrics. Fairness metrics fall into two categories:

- **Group Fairness:** A classifier is said to satisfy Group Fairness if the proportion of members in a protected group receiving favourable classification labels is identical to the proportion in the population as a whole. This is achieved by using fairness regularisers, e.g weights, by modifying the training data, or by re-labelling the training data. The transformed dataset is then used to train a classification for fairer outcomes.
- **Individual Fairness:** A classifier is said to satisfy Individual Fairness if the classification outcomes are the same for similar individuals. This is referred to in the Proposal as "A mapping f : of a Construct Space (CS) to a Decision Space (DS) is said to be fair if the people that were close in the CS are also close in the DS", where CS is an idealistic representation of all the attributes we would like the classifier to have access to, and DS comprises the output of the classifier.

This assessment used fairness metrics defined in the paper Fairness Definitions Explained [8]. The paper is a collation of definitions and metrics of fairness. The most common metrics are implemented in the AIF360 toolkit. Fairness metrics are defined in terms of statistical and similarity-based measures. Statistical measures focus exclusively on the Sensitive attribute and labels, ignoring all other attributes. This approach is suitable for measuring group fairness, which uses label counts to measure fairness between privileged and unprivileged groups. However, satisfying group fairness doesn't always satisfy individual fairness [9]. For example, consider a credit card default training dataset

containing favourable and unfavourable outcomes for males and females. It may contain an equal number of favourable outcomes for males and females, thus satisfying group fairness. However, the males may have been given favourable outcomes at random, whilst the females with the most savings were given favourable outcomes. Thus, the classifier appears fair with respect to statistical parity, despite the discrepancy based on the gender. Similarity measures compensate for this inadequacy in statistical measures, by considering similarity between all attributes to achieve individual fairness. This assessment used the following fairness metrics defined in the Fairness Definitions paper and implemented in the AIF360 toolkit:

For group fairness:

- **Statistical Parity:** Statistical parity is achieved when the probability of achieving a favourable outcome is the same for all members of privileged and unprivileged groups.
- **Disparate Impact:** Disparate impact follows the 80% [10] rule, and is achieved when the ratio between the probabilities of privileged and unprivileged groups attaining a favourable outcome is not less than 80% or some other legal threshold.

For individual fairness:

- **Consistency:** Individual fairness is achieved when individuals from privileged and unprivileged groups with the same non-sensitive attributes are given the same classification outcome.
- **Equalised odds:** Individual fairness is achieved when the probability of an individual with an actual favourable label to be correctly assigned a favourable label, is the same as the probability of an individual with an actual unfavourable label to be correctly assigned an unfavourable label by the classifier.

These metrics are formulated below in the context of the relevant bias-mitigation algorithms that use the measures as fairness constraints.

2.2 Bias Mitigation Algorithms

The aim of any bias mitigation is to either produce a transformed and debiased training dataset that can be used to train a classifier, or tweak a classifier to perform in a non-discriminatory way. In their work on Conscientious Classification, D'Alessandro et al [11] suggest three bias mitigation approaches:

- Pre-processing Algorithms: these algorithms transform the training data to obfuscate any discriminatory patterns. They introduce new weights for each data object, produce fair intermediate representations of the training dataset, or modify the training data labels. However, the classification algorithms need to be fine-tuned for a particular dataset or sensitive attribute, and would require re-training for a new data set or a different set of protected attributes.
- In-Processing Algorithms: these algorithms tweak the classifier into becoming bias-aware using adversarial methods. They produce a fair classification model that can be used to make predictions on a production version of new data. The model would also require re-training for a new data set or a different set of protected attributes.
- Post-processing Algorithms: these algorithms don't require training data transformation or in-classifier manipulation. They simply modify the post classification labels to achieve fairness. They are not limited to a specific classifier. However, post-processing algorithms are the most intrusive as they alter the predictions after the classification.

The figure 2.1.1 below is modified from the Conscientious Classification paper and shows the where the bias mitigation algorithms are used at various points in a machine learning pipeline.

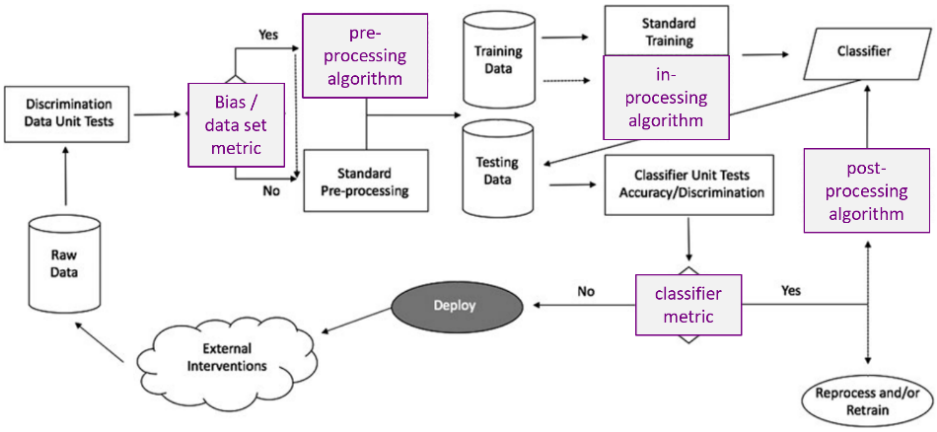


Figure 2.1.1 Bias Mitigation algorithms in machine learning

The following algorithms are a selection from the AIF360 toolkit and are assessed in this report. For ease, the following common notations will be used to describe the implementation of the fairness constraints in each of the selected bias-mitigation algorithms from the AIF 360 toolkit:

- X denotes the entire data set of individuals, where $x \in X$ represents a data object. x also refers to a vector for each data object. Vector components comprise the entire set of an individual's attributes.
- S denotes the sensitive variable. S=1 indicates that an individual x belongs to the privileged group and S=0 indicates that the individual x belongs to the unprivileged group.
- A denotes all non-sensitive variables so that $A \cap S = 0$ and $A \cup S = X$.
- N represents the number of samples in the training dataset.
- Z represents an intermediate data representation between the training data and the data set with the predicted labels.
- K represents a set of intermediate mappings k between X and Z
- Y denotes the 'ground truth' label in the training data, $\hat{Y}$ is the predicted label from the classification model
- Y=1, Y=0, $\hat{Y}$=1, $\hat{Y}$=0 are the binary values of the Y and $\hat{Y}$ respectively. Y=1 is the favourable label or the positive class, and Y=0 denotes the unfavourable label or the negative class.

2.2.1 Pre-Processing Algorithms

Learning Fair Representations (LFR).

This algorithm seeks to create an intermediate and fair representation of the training dataset [12]. It extends an approach suggested from 'Fairness through awareness' [9], which discuss a probabilistic mapping of the data objects to an intermediate and debiased representation of the data that satisfies both group and individual fairness. 'Fairness through awareness' also suggests the creation of an impartial regulators between data owners and vendors, who will define the probabilistic mapping. It suggests that vendors will then use their optimised classification algorithms against this debiased data representation for fairer outcomes. However, defining what is a fair distance function i.e. the distance metric to determine similarity between individuals requires domain expertise and may not be available or feasible. This approach is also a mapping exercise, and not a machine learning problem. Additionally, the approach is dataset specific and the resultant debiased trained model cannot be readily be applied to other datasets. These drawbacks are addressed in LFR.

LFR extends the approach taken in Fairness Through Awareness by approaching fairness as an optimisation machine learning exercise. It learns a distance function between data objects to achieve

individual fairness, and it establishes a set of intermediate representations or mapping prototypes of the data to achieve group fairness. The optimisation effort is directed at retaining the optimal amount of the original data in the intermediate data representation, whilst masking all attributes that can link the classification result to the sensitive attribute. LFR argues that the latter is a fundamental step in achieving fairness in classification. Any vendor's optimised classification model thus becomes fair by simply being applied to the fair representation of the original data. The vendor's classifier will also be unable to determine the Sensitive value from the representation. One of the properties of LFR is that if statistical parity (group fairness) is achieved, individual fairness is also increased.

This algorithm modifies the original training dataset and label assignments to meet the fairness constraints. It is implemented by learning three sets of mapping prototypes. These also form the three optimisation or minimising objectives of the algorithm:

- The mapping from the training data X to Z satisfies group fairness. The objective is to minimise the difference between probability mappings of X -> Z for all $x \in X$. In other words, the difference in probability mappings for privileged and unprivileged groups should be near or equal to 0 i.e $P(Z=k \mid S=1) - P(Z=k \mid S=0) \cong 0$ -----(a)
- The mapping of X -> Z retains the optimal amount of original individual data and omits information about membership of the sensitive group. The objective is to minimise the amount of information lost from the mapping. The equation (a) above implies that the difference in probabilities between the training data and intermediate representation of the data is small, e.g. $P(S = 1)|(Z = k) - P(S =1)$ and $P(S = 0)|(Z = k) - P(S =0)$ are both small. Therefore, the difference between the mutual information between Z and S is also small, accomplishing the objective of masking attributes that link Y to S.
- The mapping from X -> Z -> Y is as close to *f*, the classification function. The objective is to minimise the difference between each mapping prototype's prediction for $\hat{Y}$ and actual Y.

Two sets of parameters are learned by this algorithm:

- the Euclidean distance between data objects for each mapping prototype.
- the weights that govern the mapping from each prototype of X to Y. Individual weights are used for each feature dimension of vector x, depending on whether x belongs to the privileged group or not.

The algorithm is constrained by two fairness measures:

- Statistical parity for group, defined as:
  Probability of $(Y = 1 \mid S = 0)$ = Probability of $(Y = 1 \mid S = 1)$
- Consistency for individual fairness, defined as a measure of how similar individuals are treated in the classification model. It compares a model's classification prediction of a given data item x to its k-nearest neighbours, kNN(x).

Disparate Impact Remover

In US law, *disparate impact* occurs when a classification has different outcomes for privileged and unprivileged groups whilst appearing to be neutral. The legal doctrine emanates from Griggs v. Duke Power in 1971 [13] which made hiring based on ethnicity illegal, however unintended. This is different from disparate treatment, which refers to direct and illegal discrimination. This algorithm focusses on removing this unintended bias. The paper that underpins this algorithm [14] uses the 80% rule. Like LFR, this bias mitigation algorithm produces a 'repaired' training dataset so that:

- As much of the original training dataset information is retained so that classification is still possible with minimal loss of model accuracy. It does this by strongly preserving attribute rank.
- The Sensitive attribute cannot be predicted from the non-Sensitive attributes and classification labels.

To meet the Disparate Impact fairness constraints, this algorithm only modifies the original training data and not label assignments. Given a training dataset D = (S,A,Y) where S is the Sensitive variable (e.g. race, gender or age), A represents the rest of the non-sensitive variables, and Y is the binary classification label (e.g. good or bad credit), and using the 80% rule, the training dataset D exhibits Disparate Impact if:

$$\frac{\text{Probability of } (Y = 1 \mid S = 0)}{\text{Probability of } (Y = 1 \mid S = 1)} \leq 0.80$$

To measure Disparate Impact, consider the following confusion matrix of the classification Y, with S as the Sensitive variable:

Outcome	S=0	S=1
Y = 1	a	b
Y = 0	c	d

The 80% rule is quantified as: $\frac{c/(a+c)}{d/(b+d)} \geq 0.80$
The Sensitivity or True Positive Rate = $b \;/\; (b+d)$, i.e. the conditional probability of Y=1, given S=1
The Specificity or True Negative Rate = $a \;/\; (a+c)$ i.e. the conditional probability of Y=0, given S=0

The likelihood function is the probability of the predicted label $\hat{Y}$, given prior knowledge about the ground truth value of the label Y. The Likelihood Ratio for the favourable label, e.g. LR_1 for Y = 1 is:
LR_1 (Y,S) = Sensitivity / (1 – Specificity) = $\frac{d/(b+d)}{c/(a+c)}$
The training dataset D is said to have Disparate Impact (DI) if LR_+ (Y,S) $\geq$ 1/0.8.
Disparate Impact (DI) is measured as: DI = 1 / LR_+ (Y,S).

A value of DI equal or close to 1 indicates that the classifier satisfies the Disparate Impact measure. Disparate Impact Remover uses Disparate Impact as its fairness constraint.

Reweighing

The paper behind this pre-processing algorithm [15] defines a method to append weights to each data object to achieve statistical parity for the whole dataset. The weight for a data object will be the expected probability of its Sensitive attribute value and label class assuming independence of each, divided by its observed probability. Therefore for a Sensitive variable S and label Y with binary values, four weights will be calculated, one for each of the four combinations of (S=0, Y=0), (S=0, Y=1), (S=1, Y=0) and (S=1, Y=1) . The weights are determined by first calculating the probabilities each of the four combinations assuming that S and Y are independent, i.e. with full statistical parity and no discrimination. For example, for a data object with (S=0, Y=1) the expected probability is written as:

$$P_{Expected}(S = 0 \text{ AND } Y=1) = \frac{P(S = 0)}{N} \times \frac{P(Y = 1)}{N}$$

Then the observed probabilities are written as:

$$P_{Observed}(S = 0 \text{ AND } Y=1) = \frac{P(S = 0 \text{ AND } Y=1)}{N}$$

The weight for all data objects x with (S=0, Y=1) are calculated thus:

$$W_{(S=0,y=1)} = \frac{P_{Expected}(S = 0 \text{ AND } Y=1)}{P_{Observed}(S = 0 \text{ AND } Y=1)}$$

In this case, if $P_{Expected}$ is greater than $P_{Observed}$, it indicates a bias towards the unfavourable label class, e.g. Y=0. Similarly the weights for the other three combinations are obtained.

This results in higher weights given to the objects with S=0 and Y=1 (i.e. unprivileged individuals with a favourable label), whilst lower weights are given to objects with S=0 and Y=0, (i.e. unprivileged individuals with an unfavourable label). Similarly, lower weights are given to objects with S=1 and Y=1 (privileged individuals with a favourable label) and higher weights given to objects with S=1 and Y=0 (privileged individuals with an unfavourable label). The paper behind this asserts that for all classifiers, there will always be a trade-off between model accuracy and lower discrimination. This is discussed in the Business Impacts section later.

The Reweighing algorithm uses the Statistical Parity as its fairness constraint, and is measured in the labelled training dataset as: $Discrimination_{(S=0)} = P(\hat{Y} \mid S=1) = P(\hat{Y} \mid S=0)$, or the probability of the predicted label $\hat{Y}$ is the same for all values of the sensitive variable S.

$Discrimination_{(S=0)}$ is the difference in probability of between privileged and unprivileged groups to obtain favourable label outcomes. A discrimination value not equal to 0 indicates some bias with respect to the unprivileged groups gaining a favourable outcome from the classification.

The Reweighing algorithm does not relabel the data objects or modify the original training data. Weights for each data object are appended to the dataset and the classifier is trained on this weighted data. The output is a trained classifier that will have minimal correlation between the label Y and the sensitive attribute S. This method can be used with any classification method based on frequency counts.

2.2.2 In-Processing Algorithms

Adversarial Debiasing

Adversarial Learning [16] seeks to maximise a classifier's ability to predict the label Y, whilst minimising an 'adversary' from being able to predict the sensitive variable S from the predicted labels. The algorithm exploits a neural network's (NN) ability to learn through repeated iterations of applying weights to inner layers to achieve the desired outcome. Specifically, this algorithm uses Generative Adversarial Networks (GAN) to achieve this. GANs specialise in predicting attributes of a training dataset given the ground truth label Y. Adversarial Debiasing uses two NNs. One defines a 'predictor' NN model that predicts the label Y. These predictions $\hat{Y}$ are used as input to the second 'adversarial' NN, called the discriminator, denoted by $\overline{NN}$ which attempts to predict the sensitive variable S. The role of NN is to deceive $\overline{NN}$ by obfuscating information about S. By deciding whether to make the ground truth labels Y or predicted labels $\hat{Y}$ available to $\overline{NN}$, this algorithm attempts to remove bias by achieving the following fairness constraints:

- Statistical Parity is achieved when $P(\hat{Y} \mid S=1) = P(\hat{Y} \mid S=0)$, or the probability of the predicted label $\hat{Y}$ is the same for all values of the sensitive variable S.
- Equalised odds is achieved if the predicted label $\hat{Y}$ and the sensitive variable S=0 or S=1 are conditional independent, given the ground truth label Y. In other words S and $\hat{Y}$ are conditionally independent given Y if, given prior knowledge of the training data label Y, the

predicted label $\hat{Y}$ provides no information on the likelihood of S=0 or S=1. Conversely S=0 or S=1 provides no information on the likelihood of the value of $\hat{Y}$.

- Equality of Opportunity is an extension of Statistical Parity for discrete values of $\hat{Y}$, where, given a particular value of the ground truth label y, the probability of the predicted label $\hat{Y}$ is the same for all values of the sensitive variable S.

The algorithm does not modify the training dataset. It produces a debiased classification model that is dataset specific.

The algorithm works by attempting to meet each of the above constraints individually. The NN predicts $\hat{Y}$as accurately as possible. For Demographic Parity and Equality of Opportunity, the adversarial NN, denoted by $\overline{NN}$, is given access to $\hat{Y}$ and will attempt to predict S. The weights from $\overline{NN}$ will be incorporated into the weights used by NN specifically to minimise the information about S transmitted through $\hat{Y}$. For achieving equality of odds, $\overline{NN}$ has access to Y and $\hat{Y}$, thus forcing the predictor NN to limit the information about S contained in $\hat{Y}$.

Adversarial debiasing can be used for cases where the Sensitive and Label variables are discrete or continuous. This is useful for cases with non-binary Sensitive variables. This approach can be used for any weight-based classifier.

2.2.3 Post-Processing Algorithms

Reject Optimisation Classifier (ROC)

This algorithm is based on the notion that biased classification decisions are made close to the decision boundary because of decision maker's bias [17]. It proposes two solutions. The first, Reject Option based Classification (ROC) focusses on the low confidence region of a probabilistic classifiers and re-classifies labelled outcomes to reduce discrimination. The second solution, called Discrimination-Aware Ensemble (DAE), focusses on the disagreement region of a classifier to relabel outcomes. This assessment focusses on the ROC post-processing classification as this is bias-mitigation technique is part of the AIF 360 toolkit.

A classifier uses Bayesian posterior probabilities to allocate predicted label to each data object. A posterior probability is the revised or updated probability of an event occurring after taking into consideration new information, eg the ground truth label Y. A posterior probability over a certain threshold, usually 0.5, results in a favourable class label prediction, e.g Y=1.

ROC focuses on the region near the decision boundaries, where posterior probabilities (pp) are close to the threshold for predicting whether the label Y=1 or Y=0. Assume pp = P(Y=1 | X) is the posterior probability for data object x produced by a classifier. When pp is close to 0 or 1, the label prediction is made with a high degree of certainty. However, when pp is closer to 0.5, then predicting whether Y=0 or Y=1 is less certain. The Reject Option is defined so that data objects close to this decision boundary, i.e with max[pp, 1− pp] ≤ θ (where 0.5 < θ < 1) are not assigned labels and are considered as 'rejected'. θ is a loss function, later referred to as an inverse measure of bias.

ROC refers to this as the critical region. The rejected data objects this region are considered to be influenced by bias. To reduce the bias, the rejected data objects are relabelled as follows:
If the data object is in an unprivileged group (S=0), then it is labelled as Y=1. Otherwise it is labelled as Y=0, e.g. privileged group members get an unfavourable label. The instances outside the critical region are classified according to the standard decision rule, i.e., if pp is > threshold, Y=1, otherwise Y=0.

There is an accuracy loss when a data object with a predicted Y=1 is relabelled as Y=0 based on membership of a privileged or unprivileged group. The trade-off between accuracy loss and discrimination can be adjusted by adjusting the loss function θ. Therefore, bias decreases as θ increases, eg as more labels are re-classified. ROC asserts that for any level of bias θ, the classification accuracy loss will be minimal because only instances with close to the decision boundary may be reclassified.

ROC solutions can be used on any classifier. The classifiers don't need to be pre-trained on unbiased data as they are made bias-aware at decision time.

2.3 AI Fairness 360 toolkit

This assessment used IBM's AI Fairness 360 Open Source Toolkit [6]. The Python-based toolkit provides metrics and bias-mitigation algorithms to measure, report and mitigate bias in datasets and corresponding classification models. It is an ongoing collection and implementation of bias metrics and algorithmic research in this field, and currently features over 70 fairness metrics and 10 bias mitigation algorithms. The toolkit provides class definitions and method implementations for datasets, metrics and bias mitigation algorithms for pre-, in- and post-processing implementation.

Dataset classes include base classes for Structured Datasets and Binary Label datasets. These define methods for all aspects of dataset manipulation and reporting, including definitions for the Sensitive attribute, privileged and unprivileged group attributes, favourable and unfavourable label values, probability scores and tuple weights.

2.4 Dealing with multiple sensitive attributes in datasets

This project assessed datasets with binary values for the Sensitive variable, e.g. male / female. For multiple values of the sensitive variable (e.g. ethnicity = Caucasian / Asian / Black / Slav etc), the approach would be to consolidate all the unprivileged group values into one (e.g. non-Caucasian) group, and compare the classification treatment against the privileged group (e.g. Caucasian). Alternatively, a pairwise comparison can be made using the privileged group and each category of the unprivileged group (e.g. Caucasian / Asian, Caucasian / Black etc).

3. Assessment – Overview and Scope

3.1 Overview

The assessment of bias mitigation algorithms was made using three datasets. Each dataset was cleaned and feature engineered where appropriate. The datasets were then tested for bias using four metrics from the AIF360 toolkit to establish baseline measures for bias and classification model accuracy. The metrics comprise measures for ascertaining group and individual fairness. The bias for each dataset was then mitigated using five algorithms from the AIF 360 toolkit. These comprised three pre-processing, one in-processing and one post-processing algorithm. The datasets were then tested for bias again, and the results compared with the baseline measures. Pre-processing algorithms returned a transformed training dataset with which to train a subsequent classifier for fairer predictions. In-processing algorithms returned a bias-aware classification model for that dataset. Post-

processing algorithms took the dataset with predicted labels and returned a dataset with modified labels. Results of the assessment's findings are addressed in the Evaluation section of this assessment.

3.2 Scope of the assessment

3.2.1 Datasets

The three datasets were used for this assessment. The report originally intended to use the CDRC's British Population Survey and CAMEO Analysis Postcode Directory data sets for bias and subsequent mitigation. However, these datasets didn't include credit worthiness labels. Generating these labels manually by using gender, income, age or postcode as a Sensitive attribute for credit worthiness required considerable feature engineering and domain knowledge. As this fell outside the scope of this report, the Taiwan dataset was used instead. During the assessment, the Taiwan dataset showed little bias. Hence the German and Adult datasets were also used to measure for and transformed to remove their bias. This is addressed later in the Evaluation section.

3.2.1.1 Taiwan dataset

This dataset [18] classifies people into whether they defaulted on their credit card payment. Using this data, a credit card issuer could predict more accurately who will most likely default on their credit card payments. These predictions could influence their decisions on whom to extend credit to. Attributes include credit data, history of payments, and outstanding debt of credit card clients in Taiwan from April 2005 to September 2005. The dataset has 30,000 entries with 24 attributes, and a binary label that indicates whether the individual defaulted on their payment for October 2005 or not.

Using Gender as the Sensitive variable and its value of 'female' as the unprivileged group, the assessment found only slight evidence of bias. To test this further, an assessment of the same dataset using Marriage as the sensitive variable and its value of 'Singles' as the unprivileged group was made. These datasets are referred to in this report as Taiwan-Gender and Taiwan-Marriage.

3.2.1.2 German Dataset

This dataset [19] classifies people described by a set of attributes as good or bad credit risks. As in the Taiwan dataset, a credit card issuer could predict who is a good or bad credit risk and use this prediction to influence who they extend credit to. Each entry represents an individual who has been granted credit by a bank. It has 1000 entries with 20 attributes, and a binary label that indicates a good or bad credit risk.

3.2.1.3 Adult Dataset

This dataset [20] classifies individuals into whether they are high earners (earn over $50K) or not. This data was extracted from the US 1994 Census bureau database. A bank or credit card issuer could use this data to determine whether an individual would be a good customer or not based on their predicted income. It has 48,842 entries with 14 attributes and a binary label indicating high earnings or not.

3.2.2 Classification Models

This assessment focussed on bias-mitigation in binary classification models, as these were the most prevalent use case in the literature. Therefore, two classifiers from the sci-kit library, a Logistic

Regression and Random Forest classifier were used to assess the bias-mitigation performance of the five selected algorithms from the AIF 360 toolkit. As this assessment's focus was on bias-mitigation and not model performance, the classifiers were selected for their simplicity, and to provide baseline and post-mitigation performance readings. They are also ubiquitous in classification problems.

3.2.3 Metrics used for the assessment

The following measures for fairness were used to measure baseline and post-mitigation bias:

For measuring group fairness:

- **Statistical Parity:** Statistical parity is achieved when the probability of achieving a favourable outcome is the same for all members of privileged and unprivileged groups.
- **Disparate Impact:** Disparate impact follows the 80% [10] rule, and is achieved when the ratio between the probabilities of privileged and unprivileged groups attaining a favourable outcome is less than 80% or some other legal threshold.

For measuring individual fairness:

- **Consistency:** Individual fairness is achieved when individuals from privileged and unprivileged groups with the same non-sensitive attributes are given the same classification outcome.
- **Equalised odds:** Individual fairness is achieved when the probability of an individual with an actual favourable outcome to be correctly assigned a favourable outcome, is the same as the probability of an individual with an actual unfavourable outcome to be correctly assigned an unfavourable outcome by the classifier.

In addition, two more metrics were used to test the effectiveness of the bias-mitigation algorithms with pre-mitigation and post-mitigation comparisons:

- **Sensitive attribute test:** Most bias-mitigation algorithms selected for this assessment attempted to obfuscate information about the Sensitive attribute in the transformed dataset. Therefore, this test used the two classifiers to predict the Sensitive attribute, i.e. used the Sensitive attribute as the label. Classification model performance for both classifiers was recorded before and after bias-mitigation and results compared.

  The motivation behind this test was to check for 'redundant encoding'. Classifiers work by finding statistical relationships between attributes. Where a strong correlation exists between the Sensitive attribute and non-Sensitive attributes, the classification on the training dataset will favour one group over another. For example, an individual's University, or postcode - whilst innocuous, can be a proxy for the privileged group of the Sensitive attribute. A classifier will exploit the high correlation between these 'redundant encodings' and the Sensitive attribute, and use them as proxy Sensitive attributes.

- **Model accuracy:** the performance of each classifier to predict the label was tested before and after bias-mitigation using the balanced accuracy measure. Balanced accuracy is calculated as the sum of ratios of each correctly predicted class to the total number that class label. Whereas accuracy is calculated as the ratio of all correctly predicted labels to the total number of data objects. Balanced accuracy is a better measure for datasets with imbalanced class labels.

3.2.4 Bias-mitigation algorithms and their success criteria

The assessment used the following bias mitigation algorithms from the AIF360 toolkit. They are described here in terms of their success criteria:

a) **Learning Fair Representations (LFR):** this pre-processing algorithm focussed on delivering group and individual fairness, whilst obfuscating information about the Sensitive variable and maintaining model performance. The success criteria was:
   - An improvement in group and individual fairness
   - Sensitive attribute test - a decrease or no change in the model performance
   - Minimal reduction in model performance

b) **Disparate Impact Remover:** this pre-processing algorithm transformed attributes of the training dataset to meet group fairness constraints. The success criteria was:
   - An improvement in group fairness
   - Sensitive attribute test - a decrease or no change in the model performance
   - Minimal reduction in model performance

c) **Reweighing:** this pre-processing algorithm calculated four weights, one for each combination of Sensitive attribute and label value, and appends these to each data object respectively, to achieve statistical parity for the whole dataset. The success criteria was:
   - An improvement in group fairness
   - Sensitive attribute test - a decrease or no change in the model performance
   - Minimal reduction in model performance

d) **Adversarial Debiasing:** this in-processing algorithm used an adversarial technique and neural networks to produce a debiased classifier that delivers on both group and individual fairness. The success criteria was:
   - An improvement in group and individual fairness
   - Sensitive attribute test - a decrease or no change in the model performance
   - Minimal reduction in model performance

e) **Reject Option Classifier:** this post-processing algorithm met its fairness constraints by modifying the predicted labels returned from a classification model. The success criteria was:
   - An improvement in group fairness
   - Sensitive attribute test - a decrease or no change in the model performance
   - Minimal reduction in model performance

4. Implementation

4.1 Implementation approach

This assessment was implemented using Python scripts in Jupyter Notebooks. The implementation followed the task sequence below, for each of the bias-mitigation algorithm:

1. Obtain baseline measures for group and individual fairness, model accuracy and the Sensitive attribute test. This was done by running the logistic regression and random forest classifiers against the training data, and by using the AIF360 binary labelled dataset fairness measures.
2. Run the bias mitigation algorithm from the AIF360 toolkit against the training datasets.

3. For pre-processing algorithms, train both classifiers again, using the transformed datasets. For the in-processing algorithm, use the 'tweaked' model on the test dataset.
4. As in step 1 above, obtain debiased measures for group and individual fairness, model accuracy and the Sensitive attribute test from the transformed datasets or tweaked classifier (for the in-processing algorithm), using the AIF360 binary labelled dataset fairness measures.

All code and results used for this assessment can be accessed from the github repository [21]. There are 3 categories of code: The bias-mitigation code (e.g. Adult-PreProc-LFR.jpnb), data-prep code (e.g. Adult-Data-Prep.jpnb), and data-discovery code (e.g. Adult-Data-Discovery.jpnb)

4.2 Using the AIF360 toolkit

The AIF360 toolkit is implemented in Python. The toolkit provided class definitions and method implementations for datasets, metrics and bias mitigation algorithms for pre-processing, in-processing and post-processing implementation.

Dataset classes included classes for Structured datasets and Binary Label datasets. These defined methods for all aspects of dataset manipulation and reporting. Dataset manipulation methods included Binary Label Dataset to pandas dataframe conversion, and a split method to obtain training and test datasets. A mixture of the AIF360 split and sci-kit's split-test-train functions was used where appropriate. Both delivered the same results in terms of the magnitude of the split.

AIF360 reporting methods included declarations for the Sensitive attribute, privileged and unprivileged group attributes, favourable and unfavourable label values, probability scores and tuple weights. The Binary Label Dataset class was the expected input to all bias-mitigation algorithms.

The Metrics class included implementations for returning a Structured Dataset's statistical measures. These included various measures from the confusion matrix, including model accuracy, Statistical Parity, Equality of Odds, Disparate Impact and Consistency.

4.3 Feature Engineering

The assessment retained all attributes of all datasets where possible, so as not to lose any 'signal' of bias. The motivation for this retention was to test a bias-mitigation algorithm's ability to obfuscate the Sensitive attribute and all its correlated features in the resultant debiased dataset. The Sensitive attribute test discussed earlier, compares a classifier's accuracy in predicting the Sensitive variable before and after debiasing the datasets.

The AIF360 toolkit required a numerical representation of the data. Therefore, all non-numeric categorical attributes were converted to distinct numeric representations using the pandas factorize function. One hot encoding using the pandas get_dummies function was also assessed for numeric conversion, and was discounted due to the large number of additional columns it generated, which in turn obscured results from tests like feature importance before and after debiasing the dataset. This assessment found that some AIF360 toolkit examples dropped some non-numeric attributes from the Adult dataset before bias-mitigation. A query into this attribute omission with the AIF360 team did not receive a satisfactory response.

All datasets were scaled to ensure all attributes had a similar scale. Of the various scaling functions available in the scikit-learn library, the minmax scaler was chosen for its simplicity and ability to preserve the shape of the original distribution. It divides each attribute value by the range (difference between minimum and maximum) of that attribute domain.

All datasets were modified to give their Sensitive attribute S and classification label Y identical values for privileged and unprivileged groups, and favourable and unfavourable labels. These were:

For the Sensitive attribute:

S = 0: Unprivileged group, e.g. females or single individuals

S = 1: Privileged group, e.g. males or married individuals

For the Labels:

Y = 0: Unfavourable label, e.g. bad credit risk

Y = 1: Favourable label, e.g. good credit risk

4.4 Data Discovery

Each dataset was examined to discover the distribution of the Sensitive attribute values with respect to the binary label values, and subsequently tested for bias.

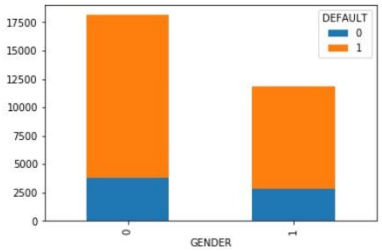
4.4.1 Taiwan Gender dataset

The Sensitive attribute was GENDER. The privileged group comprised males (1), and the unprivileged group comprised females (0). The label was DEFAULT, a 0 value indicated a default on payment and was the unfavourable label.

The Taiwan dataset generally exhibited low levels of bias against females. Attempts were made to tease out any bias 'signal' by:

- Creating new attributes to represent the ratio of payments made to outstanding debt for each of the 6 months, and for an aggregation for the 6 months.
- Creating a new 2 new attributes to present the average amount owed and paid over the 6 months.

None of these attempts improved the bias signal. Figure 4.4.1 below shows the DEFAULT label distribution for females and males:



DEFAULT	0	1
GENDER		
0	3763	14349
1	2873	9015

Where:
GENDER = 0 indicates females
GENDER = 1 indicates males
DEFAULT= 0 indicates a default on payment
DEFAULT= 1 indicates no default on payment

Figure 4.4.1 – Taiwan-Gender - DEFAULT label distribution for females and males

Further analysis revealed the following ratios in Table 4.1.1 which indicate that the distribution of males and females roughly matches the distribution of DEFAULT between the sexes in the dataset.

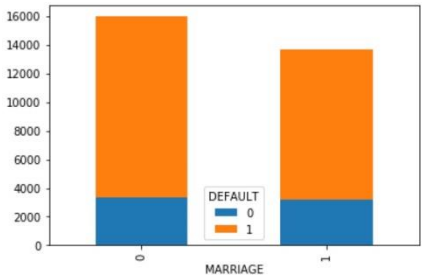
Number of Males	11,888 or 39.63%
Number of Females	18,112 or 60.37%
Ratio of Females to Males	1.523553
Number of Defaulted Males	9,015 or 38.58%
Number of Defaulted Females	14,349 or 61.42%
Ratio of Defaulted Females to Defaulted Males	1.309781

Table 4.1.1 - Taiwan-Gender - distribution of DEFAULT between females and males.

4.4.2 Taiwan-Marriage

The Sensitive attribute was MARRIAGE. The privileged group comprised Marrieds (1), the unprivileged group comprised Singles (0). The label was DEFAULT, a 0 value indicated a default, and was the unfavourable label.

The Taiwan-Marriage dataset's bias signal also did not improve with the addition of ratio or average columns for payments due and made. Figure 4.4.2 below shows the DEFAULT distribution for married and single individuals:



DEFAULT	0	1
MARRIAGE		
0	3341	12623
1	3206	10453

Where:
MARRIAGE = 0 indicates Singles
MARRIAGE = 1 indicates Marrieds
DEFAULT = 0 indicates a default on payment
DEFAULT = 1 indicates no default on payment

Figure 4.4.2 – Taiwan-Marriage - DEFAULT label distribution for Marrieds and Singles

Further analysis revealed the ratios in Table 4.4.2 which indicate that the distribution of Singles and Marrieds roughly matches the distribution of default between the marriage status in the dataset.

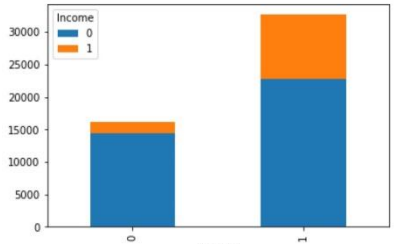
Number of Singles	15,964 or 53.9%
Number of Marrieds	13,659 or 46.1%
Ratio of Singles to Marrieds	1.168753
Number of Defaulted Singles	3,341 or 51%
Number of Defaulted Marrieds	3,206 or 49%
Ratio of Defaulted Singles to Marrieds	1.042109

Table 4.4.2 - Taiwan-Marriage - distribution of DEFAULT between marrieds and singles.

4.4.3 Adult Dataset

The Sensitive attribute was Gender. The privileged group comprised males (1), and the unprivileged group comprised females (0). The label was Income, a 0 value indicated an income less than $50K and was the unfavourable label.

Attribute Fnlwgt was dropped as it wasn't relevant to this assessment. It represents a data sampling weight to ensure the attribute distribution reflects the actual attribute distribution in the respective US states. Figure 4.4.3 below shows the Income distribution for males and females in the dataset:



Income	0	1
Gender		
0	14423	1769
1	22732	9918

Where:
Gender = 0 indicates females
Gender = 1 indicates males
Income = 0 indicates a low income
Income = 1 indicates a high income

Figure 4.4.3 - Adult - Income distribution between males and females.

Table 4.4.3 below provides some indication of the bias in the dataset, e.g females make up 33.2% or the total sample but represent 38.8% of low earners.

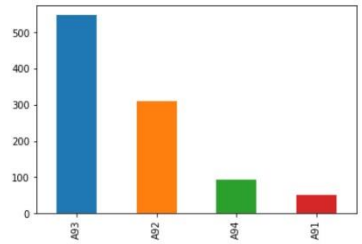
Number of Males	32,650 or 66.8%
Number of Females	16,192 or 33.2%
Ratio of Males to Females	2.016427
Number of Low-Income Males	22,732 or 61.2%
Number of Low-Income Females	14,423 or 38.8%
Ratio of Low Income Males to Females	0.687651

Table 4.4.3 - Adult - distribution of Income between males and females.

4.4.4 German Dataset

The Sensitive attribute was Gender. The privileged group comprised males (1), and the unprivileged group comprised females (0). The label was CreditStatus, a 0 value indicated a bad CreditStatus, and was the unfavourable label.

Data discovery revealed an imbalance. Figure 4.4.4.1 below shows attribute 9, comprising 5 codes for an applicant's marital status and gender. Three codes represented males, and two indicated females. There were no single females in the dataset, suggesting an imbalance in representation between the privileged/unprivileged groups. This attribute was engineered to contain only male and female values.

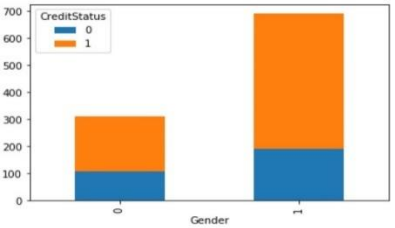


Where:
A91 : male: divorced/separated
A92 : female: divorced/separated/married
A93 : male: single
A94 : male: married/widowed
A95 : female: single

There are no single females (A95) in the dataset.

Figure 4.4.4.1 - German - Gender & Marital Status distribution between males and females.

Figure 4.4.4.2 below shows the CreditStatus distribution for males and females in the dataset:



CreditStatus	0	1
Gender		
0	109	201
1	191	499

Where:
Gender = 0 indicates females
Gender = 1 indicates males
CreditStatus = 0 indicates a Bad CreditStatus
CreditStatus = 1 indicates a Good CreditStatus

Figure 4.4.4.2 - Gender and CreditStatus distribution between males and females.

Table 4.4.4.1 below provides some indication of the bias in the dataset, e.g females make up % or the total sample but represent % of bad CreditStatus.

Number of Males	690 or 69%
Number of Females	310 or 31%
Ratio of Males to Females	2.225806
Number of Male Bad CreditStatus	191 or 63.6%
Number of Females Bad CreditStatus	109 or 36.4%
Ratio of Males to Females with Bad CreditStatus	1.752293

Table 4.4.4.1 - German - distribution of males and females and CreditStatus after the Gender attribute was consolidated.

5. Evaluation and Discussion

The evaluation approach used the success criteria to test the bias-mitigation capabilities of each mitigation algorithm. Red values in the tables below indicate an unexpected lower or higher result. Results are discussed in the Discussion section below. Table 5.1 shows the desired outcomes for each of the fairness metrics:

Statistical Parity	0	After mitigation, this value should be nearer to 0.
Disparate Impact	1	After mitigation, this value should be nearer to 1
Consistency	1	After mitigation, this value should be nearer to 1.
Equality of odds	0	After mitigation, this value should be nearer to 0

Table 5.1 - Desired outcomes for Fairness Metrics used.

5.1 Pre-processing algorithms

5.1.1 Results for Learning Fair Representations

Improvement in group fairness: Table 5.1.1 below shows the before and after bias mitigation measures for Statistical Parity and Disparate Impact.

	Taiwan-Gender	Taiwan-Marriage	Adult	German
Statistical Parity (Before)	0.0298339	0.02647544	0.1929391	0.0956693
Statistical Parity (After)	0.0092356	0.00154118	0.0085174	0.0011153
Disparate Impact (Before)	1.0391294	1.03474102	0.3620832	0.8648391
Disparate Impact (After)	1.0093853	0.99841252	0.1198747	1.0021945

Table 5.1.1.1 Pre-processing - LFR - Statistical Parity and Disparate Impact

Improvement in individual fairness: Table 5.1.1.2 below shows the before and after bias mitigation metrics for Consistency and Equality of Odds

	Taiwan-Gender	Taiwan-Marriage	Adult	German
Consistency (Before)	0.7854190	0.78197338	0.8326312	0.6968571
Consistency (After)	0.9996666	0.99951775	0.9997287	0.9720000
Equality of Odds (Before)	0.0303846	0.03795768	0.1724093	0.1475312
Equality of Odds (After)	0.0059917	0.00794666	0.0028328	0.0107687

Table 5.1.1.2 Pre-processing - LFR - Consistency and Equality od Odds.

Sensitive attribute test: Results found an expected decrease or no change in the model performance: Table 5.5.1.3 below shows the before and after bias mitigation Logistic Regression and Random Forest Classification model performance when attempting to predict the Sensitive class from the biased and transformed data:

Logistic Regression - Sensitive attribute model performance	**Taiwan-Gender**	**Taiwan-Marriage**	**Adult**	**German**
Balanced Accuracy (Before)	0.5373862	0.69811529	0.6952920	0.5851715
Balanced Accuracy (After)	0.4857890	0.50641664	0.4790547	0.5000000
Random Forest - Sensitive attribute model performance				
Balanced Accuracy (Before)	0.5205672	0.73264158	0.7569811	0.5634191
Balanced Accuracy (After)	0.4983500	0.52825630	0.4362819	0.4901960

Table 5.1.1.3 Pre-processing - LFR – Sensitive attribute test.

Minimal reduction in model performance: Table 5.5.1.4 below shows the before and after bias mitigation Logistic Regression and Random Forest Classification model performance.

Logistic Regression	Taiwan-Gender	Taiwan-Marriage	Adult	German
Balanced Accuracy (Before)	0.6432656	0.63359499	0.6914758	0.6155555
Balanced Accuracy (After)	0.5055595	0.50606092	0.5004230	0.6111111
Random Forest				
Balanced Accuracy (Before)	0.5963679	0.60121466	0.6781526	0.5711111
Balanced Accuracy (After)	0.5044285	0.50454117	0.5002115	0.5755555

Table 5.1.1.4 Pre-processing - LFR - Classification model performance.

5.1.2 Results for Disparate Impact Remover

Improvement in group fairness: Table 5.1.2.1 below shows the before and after bias mitigation measures for Statistical Parity and Disparate Impact.

	Taiwan-Gender	Taiwan-Marriage	Adult	German
Statistical Parity (Before)	0.0332706	0.02685611	0.1929391	0.0919999
Statistical Parity (After)	0.0666004	0.01790784	0.0864186	0.0976190
Disparate Impact (Before)	1.0437451	1.03520733	0.3620832	0.8722222
Disparate Impact (After)	1.1722253	0.96055666	0.7838038	0.8270042

Table 5.1.2.1 Pre-processing - Disparate Impact Remover - Statistical Parity and Disparate Impact

Improvement in individual fairness: Table 5.1.2.2 below shows the before and after bias mitigation metrics for Consistency and Equality of Odds:

	Taiwan-Gender	Taiwan-Marriage	Adult	German
Consistency (Before)	0.7844583	0.78298591	0.8326312	0.7055000
Consistency (After)	0.9187666	0.91858227	0.9120073	0.7599999
Equality of Odds (Before)	0.0416668	0.03533058	0.1724093	0.1266116
Equality of Odds (After)	0.0549178	0.01813129	0.0407425	0.1130705

Table 5.1.2.2 Pre-processing - Disparate Impact - Consistency and Equality od Odds.

Sensitive attribute test: Table 5.1.2.3 below shows the before and after bias mitigation Logistic Regression and Random Forest Classification model performance when attempting to predict the Sensitive class from the biased and transformed data:

Logistic Regression - Sensitive attribute model performance	Taiwan-Gender	Taiwan-Marriage	Adult	German
Balanced Accuracy (Before)	0.5391934	0.70077261	0.8003889	0.7100000
Balanced Accuracy (After)	0.5390558	0.70175749	0.7480806	0.6071428
Random Forest - Sensitive attribute model performance				
Balanced Accuracy (Before)	0.5121004	0.73806391	0.7569811	0.5595238
Balanced Accuracy (After)	0.5004226	0.73762533	0.6635963	0.5750000

Table 5.1.2.3 Pre-processing - Disparate Impact – Sensitive attribute test.

Minimal reduction in model performance: Table 5.1.2.4 below shows the before and after bias mitigation Logistic Regression and Random Forest Classification model performance.

Logistic Regression	Taiwan-Gender	Taiwan-Marriage	Adult	German
Balanced Accuracy (Before)	0.6481056	0.64200441	0.6914758	0.5742523
Balanced Accuracy (After)	0.6196062	0.62180424	0.7330637	0.6842510

Random Forest				
Balanced Accuracy (Before)	0.5845750	0.60384957	0.6781526	0.5592350
Balanced Accuracy (After)	0.6037297	0.60191430	0.7092362	0.5395969

Table 5.1.2.4 Pre-processing – Disparate Impact - Classification model performance.

5.1.3 Results for Reweighing

Improvement in group fairness: Table 5.1.3.1 below shows the before and after bias mitigation measures for Statistical Parity and Disparate Impact.

	Taiwan-Gender	Taiwan-Marriage	Adult	German
Statistical Parity (Before)	0.0354055	0.02495861	0.1906723	0.0636488
Statistical Parity (After)	2.220e-16	0.00000000	8.326e-17	2.220e-16
Disparate Impact (Before)	1.0467924	1.03252337	0.3685749	0.9130025
Disparate Impact (After)	1.0000000	1.00000000	1.0000000	0.9999999

Table 5.1.3.1 Pre-processing - Reweighing - Statistical Parity and Disparate Impact

Improvement in individual fairness: Table 5.1.3.2 below shows the before and after bias mitigation metrics for Consistency and Equality of Odds:

	Taiwan-Gender	Taiwan-Marriage	Adult	German
Consistency (Before)	0.7836833	0.7831125	0.8323445	0.7037500
Consistency (After)	0.7836833	0.7831125	0.8323445	0.7037500
Equality of Odds (Before)	0.0111595	0.04160280	0.1716719	0.0999312
Equality of Odds (After)	0.0252449	0.00874342	0.0170516	0.0249312

Table 5.1.3.2 Pre-processing - Reweighing - Consistency and Equality od Odds.

Sensitive attribute test: Table 5.1.3.3 below shows the before and after bias mitigation Logistic Regression and Random Forest Classification model performance when attempting to predict the Sensitive class from the biased and transformed data:

Logistic Regression - Sensitive attribute model performance	**Taiwan-Gender**	**Taiwan-Marriage**	**Adult**	**German**
Balanced Accuracy (Before)	0.5379510	0.73714819	0.7019187	0.6149162
Balanced Accuracy (After)	0.5379510	0.73714819	0.7019187	0.6149162
Random Forest - Sensitive attribute model performance				
Balanced Accuracy (Before)	0.5251818	0.73518748	0.7566385	0.5706494
Balanced Accuracy (After)	0.5251818	0.73518748	0.7566385	0.5706494

Table 5.1.3.3 - Pre-processing - Reweighing – Sensitive attribute test.

Minimal reduction in model performance: Table 5.1.3.4 below shows the before and after bias mitigation Logistic Regression and Random Forest Classification model performance.

Logistic Regression	**Taiwan-Gender**	**Taiwan-Marriage**	**Adult**	**German**
Balanced Accuracy (Before)	0.5941323	0.64975414	0.6861426	0.5888372
Balanced Accuracy (After)	0.5948680	0.65477351	0.6616281	0.5960836
Random Forest				
Balanced Accuracy (Before)	0.5982413	0.58712446	0.6664366	0.5575838
Balanced Accuracy (After)	0.6004866	0.58749374	0.6469660	0.5427038

Table 5.1.3.4 Pre-processing – Reweighing - Classification model performance

5.2 In-processing algorithm

5.2.1 Results for Adversarial Debiasing

Improvement in group fairness: Table 5.2.1.1 below shows the before and after bias mitigation measures for Statistical Parity and Disparate Impact.

	Taiwan-Gender	Taiwan-Marriage	Adult	German
Statistical Parity (Before)	0.0310865	0.02335915	0.1942230	0.0690158
Statistical Parity (After)	0.0308645	0.01319203	0.0343408	0.5254791
Disparate Impact (Before)	1.0409955	1.03057494	0.3616879	0.9028143
Disparate Impact (After)	0.9662387	1.01510716	0.7974075	0.4485465

Table 5.2.1.1 In-processing - Adversarial Debiasing - Statistical Parity and Disparate Impact

Improvement in individual fairness: Table 5.2.1.2 below shows the before and after bias mitigation metrics for Consistency and Equality of Odds

	Taiwan-Gender	Taiwan-Marriage	Adult	German
Consistency (Before)	0.7812333	0.78331505	0.8316945	0.694750
Consistency (After)	0.9738083	0.97370242	0.9591533	0.890250
Equality of Odds (Before)	0.0247457	0.01254009	0.0904799	0.0567813
Equality of Odds (After)	0.0162113	0.00782035	0.0904670	0.5857770

Table 5.2.1.2 In-processing - Adversarial Debiasing - Consistency and Equality od Odds.

Sensitive attribute test: Table 5.2.1.3 below shows the before and after bias mitigation Logistic Regression and Random Forest Classification model performance when attempting to predict the Sensitive class from the biased and transformed data:

Logistic Regression - Sensitive attribute model performance	**Taiwan-Gender**	**Taiwan-Marriage**	**Adult**	**German**
Balanced Accuracy (Before)	0.5303988	0.72781386	0.6893517	0.6431743
Balanced Accuracy (After)	0.5556714	0.72913436	0.6898084	0.7669471
Random Forest - Sensitive attribute model performance				
Balanced Accuracy (Before)	0.5112115	0.72774844	0.7554315	0.5250116
Balanced Accuracy (After)	0.5089027	0.72852605	0.7525725	0.7516362

Table 5.2.1.3 - In-processing - Adversarial Debiasing – Sensitive attribute test.

Minimal reduction in model performance: Table 5.2.1.4 below shows the before and after bias mitigation Logistic Regression and Random Forest Classification model performance.

Logistic Regression	**Taiwan-Gender**	**Taiwan-Marriage**	**Adult**	**German**
Balanced Accuracy (Before)	0.6470951	0.60022919	0.6818849	0.6331096
Balanced Accuracy (After)	0.8460778	0.91581243	0.7480738	0.8762183
Random Forest				
Balanced Accuracy (Before)	0.6002443	0.58665843	0.6725080	0.5484932
Balanced Accuracy (After)	0.7485771	0.78915013	0.7403673	0.7995451

Table 5.2.1.4 In-processing – Adversarial Debiasing - Classification model performance

5.3 Post-processing algorithms

5.3.1 Results for Reject Option Classifier

Improvement in group fairness: Table 5.3.1.1 below shows the before and after bias mitigation measures for Statistical Parity and Disparate Impact, using Logistic Regression (Log) and Random Forest Classifiers (Rfc).

	Taiwan-Gender	Taiwan-Marriage	Adult	German
Statistical Parity (Before)	0.0519715	0.03613819	0.2003601	0.0476190
Statistical Parity Log (After)	0.0399190	0.07523354	0.0350745	0.0060000
Statistical Parity Rfc (After)	0.0167206	0.01838759	0.0370729	0.1468253
Disparate Impact (Before)	1.0691643	1.04752642	0.3605318	1.0714285
Disparate Impact – Log (After)	1.0504995	1.10111443	0.6281866	1.0129870
Disparate Impact – Rfc (After)	1.0232405	1.02622463	0.5079544	1.3775510

Table 5.3.1.1 Post-processing - Reject Option Classification - Statistical Parity and Disparate Impact

Improvement in individual fairness: Table 5.3.1.2 below shows the before and after bias mitigation metrics for Consistency and Equality of Odds

	Taiwan-Gender	Taiwan-Marriage	Adult	German
Consistency (Before)	0.7388666	0.73857577	0.8245240	0.6640000
Consistency Log (After)	0.7840000	0.76361795	0.9578710	0.6140000
Consistency Rfc (After)	0.7476000	0.74114074	0.9882906	0.6300000
Equality of Odds (Before)	0.059800	0.0513000	0.327600	0.165800
Equality of Odds (After)	0.008900	0.0365000	0.030200	0.016700

Table 5.3.1.2 Post-processing - Reject Option Classification - Consistency and Equality od Odds.

Sensitive attribute test: Table 5.3.1.3 below shows the before and after bias mitigation Logistic Regression and Random Forest Classification model performance when attempting to predict the Sensitive class from the biased and transformed data:

Logistic Regression - Sensitive attribute model performance	**Taiwan-Gender**	**Taiwan-Marriage**	**Adult**	**German**
Balanced Accuracy (Before)	0.5008655	0.64191053	0.6957168	0.6416927
Balanced Accuracy (After)	0.5008655	0.64191053	0.6957168	0.6416927
Random Forest - Sensitive attribute model performance				
Balanced Accuracy (Before)	0.5102468	0.73508547	0.7531003	0.5805642
Balanced Accuracy (After)	0.5102468	0.73508547	0.7531003	0.5805642

Table 5.3.1.3 Post-processing - Reject Option Classification – Sensitive attribute test.

Minimal reduction in model performance: Table 5.3.1.4 below shows the before and after bias mitigation Logistic Regression and Random Forest Classification model performance.

Logistic Regression	**Taiwan-Gender**	**Taiwan-Marriage**	**Adult**	**German**
Balanced Accuracy (Before)	0.6007241	0.60223492	0.6783533	0.6068437
Balanced Accuracy (After)	0.6889000	0.69090000	0.6824996	0.6314000
Random Forest				
Balanced Accuracy (Before)	0.6180156	0.61884401	0.6709555	0.6553000
Balanced Accuracy (After)	0.7264000	0.71930000	0.6709555	0.6948000

Table 5.3.1.4 Post-processing – Reject Option Classification- Classification model performance

5.4 Discussion

The evaluation first reviewed how well the datasets responded to various bias-mitigation algorithms to achieve individual and group bias for each dataset. This part of the evaluation focussed on the Statistical Parity, Disparate Impact, Consistency and Equality of Odd metrics. Next, the algorithm effect on model performance was reviewed, followed by an assessment of the model performance on the Sensitive attribute test.

5.4.1 Dataset response to bias-mitigation

The findings show that at the outset, the Taiwan-gender and Taiwan-marriage dataset exhibited consistently low levels of bias for both group and individual fairness metrics. The assessment recorded baseline measures of 0.02 for Statistical Parity, 1.03 for Disparate Impact, 0.78 for Consistency and below 0.05 for Equality of Odds. Attempts to mitigate this bias using all bias-mitigation algorithms resulted in trace or no improvements, with a few exceptions discussed in the algorithm section below. These findings reflect the evenness of ratios of females:males, and singles:marrieds represented in

the favourable and unfavourable outcomes as noted during the data discovery phase. This finding also prompted the inclusion of the Adult and German datasets in the scope of this assessment to find examples of bias mitigation.

The Adult dataset exhibited bias, with its baseline measures of 0.19 for Statistical parity, 0.36 for Disparate impact, 0.83 for Consistency and below 0.17 for Equality of Odds. The results showed that the dataset responded well to most bias mitigation algorithms, with the exception of LFR where Disparate Impact actually increased by 0.25. In all other cases, the group and individual bias was reduced or unchanged. The Adult dataset responded best to the mitigation algorithms in all cases.

The German dataset showed some bias, with baseline measures of 0.09 for Statistical parity, 0.9 for Disparate impact, 0.7 for Consistency and below 0.14 for Equality of Odds. Its response to bias mitigation was patchy, and notably bad against Adversarial debiasing, where Disparate Impact fell from .7 to 0.44, when it was expected to be nearer to 1. Statistical Parity also rose from 0.09 to 0.52, and Equality of odds rose from 0.14 to 0.58. This could be explained by the imbalances in the datasets as noted in the data discovery.

Where a bias-mitigation algorithm changes the training data objects, the effect of this change can be observed in the feature importance of the attributes before and after the bias-mitigation. Figure 5.4.1.1 and 5.4.1.2 below are examples of changed feature importance using LFR on the Taiwan-gender dataset. It shows how the Sensitive attribute GENDER has reduced feature importance after debiasing. All feature importance changes for pre-processing algorithms can be found in the individual pre-processing bias-mitigation Jupyter notebooks in the github repository [21].

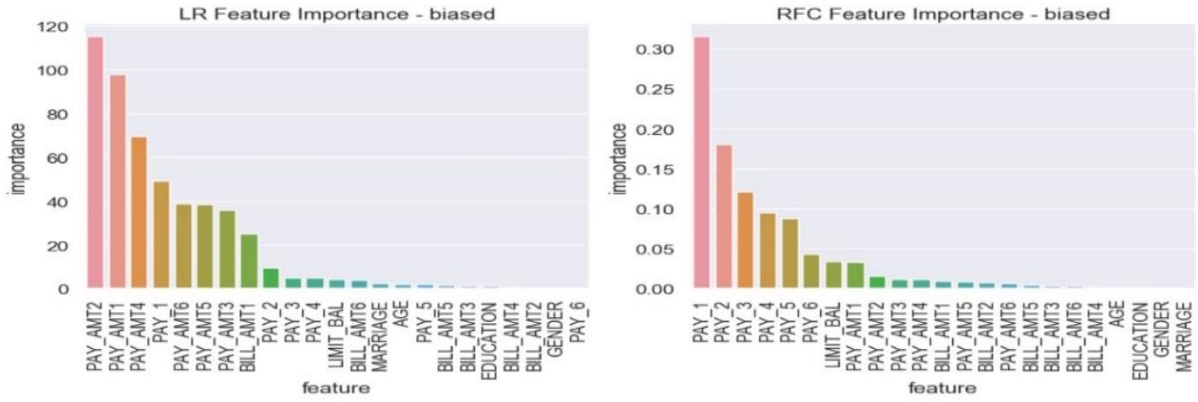


Figure 5.4.1.1 Taiwan-Gender - Feature Importance - Logistic Regression & Random Forest classifiers before mitigation using LFR

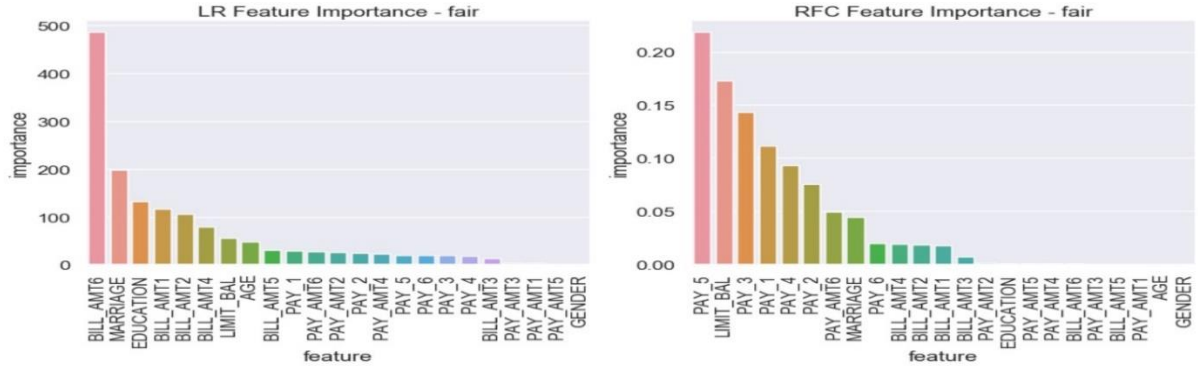


Figure 5.4.1.2 Taiwan-Gender - Feature Importance - Logistic Regression & Random Forest classifiers after mitigation using LFR

5.4.2 Bias-mitigation algorithm performance

Learning Fair Representation

This algorithm, with its intermediate fair representation of the datasets had the best test results in terms of meeting its success criteria. It showed excellent improvements from baseline measures for Consistency, Statistical Parity and Equality of Odds. Whilst Disparate Impact improved for the Taiwan and Adult datasets, it dropped for the German Dataset. This assessment was unable to explain this within the allowed timeframe.

In terms of obfuscating the Sensitive attribute, LFR showed little or slight improvement in decreasing a classifier's ability to predict the Sensitive attribute. This may be explained by the lack of bias in the Taiwan and German datasets. The obfuscation was most successful with the Adult dataset, where both classifier's balanced accuracy fell by 0.1.

However, LFR was also aggressive in modifying the original label values, the figures in table 5.4.2.1 below show. The impact of this change needs to be assessed in future research.

	Taiwan Gender	**Taiwan Marriage**	**Adult**	**German**
Before Label=0	4611	4636	29750	344
After Label = 0	218	622	38805	225
Before Label=1	16389	16100	9322	356
After Label = 1	20782	20114	268	475

Table 5.4.2.1 Label value counts - before and after bias mitigation using LFR.

Disparate Impact Remover

This algorithm surprisingly did not show improvements in mitigating Disparate Impact (DI) across all datasets, except for the German dataset. This assessment was unable to confirm whether this was a result of a low baseline DI, low bias, or imbalance in the data. It showed good results for individual fairness with all but the Adult dataset. It showed Consistency values near 1, and values for Equality of Odds nearer to 0 for the Taiwan-Marriage and Adult datasets.

In terms of obfuscating the Sensitive attribute, DI remover showed an improvement in decreasing a classifier's ability to predict the Sensitive attribute for the Taiwan-gender and Adult datasets. For the Taiwan-marriage and German dataset, accuracy increased by 0.001 and 0.02 respectively. This may be explained by the lack of bias in the Taiwan and imbalance in the German datasets. Different baseline values for the logistic regression and random forest classifiers makes it difficult to determine whether these are significant shortcomings of the algorithm.

Reweighing

This algorithm achieved good results for group fairness. Disparate Impact and Statistical Parity values were driven close to 1 and 0 respectively, for all datasets. It fared less well on individual fairness with small improvements in Equality of Odds nearer to 0, and no improvements in Consistency. This can be explained by the blunt approach of appending one of four weights to each data object, depending on its Sensitive attribute / label value combination. This approach is based on the label and Sensitive attribute count, whilst other non-Sensitive attributes that have to be considered for individual fairness, are ignored.

In terms of obfuscating the Sensitive attribute, Reweighing showed no improvement in decreasing a classifier's ability to predict it. This can be explained by the algorithm's use of weights for each object. This ignores the non-Sensitive attributes and their influence on the label outcome.

Adversarial Debiasing

This algorithm achieved good results for both Group and Individual fairness, with all measures showing a movement towards 0 or 1, depending on the measure. Bias for all datasets was reduced, except the German dataset, where bias was seen to have increased markedly. Again, we can explain this by the imbalance in the German dataset. Adversarial Debiasing also uses Generative Adversarial Networks (GAN) for its classification and mitigation algorithms. These require substantially more data to train the models than that available in the training datasets used for this assessment.

Adversarial Debiasing aims to completely obfuscate the Sensitive attribute via the adversarial neural network. In this assessment, the algorithm did not deliver on this objective, showing marginal increases or drops in predictive performance for the Sensitive attribute. Due to the lack of observational tools in AIF360, this assessment was unable to explain this neural network performance.

Adversarial Debiasing is aggressive in changing label values, as shown in the Table 5.4.2.2 below:

		Taiwan Gender	Taiwan Marriage	Adult	German
Training	Before Label=0	5352	5295	29713	249
	After Label = 0	2505	2836	32897	168
	Before Label=1	18648	18403	9360	551
	After Label = 1	21495	20862	6176	632

Table 5.4.2.2 Label value counts - before and after bias mitigation using Adversarial Debiasing.

Reject Option Classifier (ROC)

ROC showed good performance on individual fairness for the Adult dataset, with the Consistency value raised from 0.82 to 0.98, i.e. closer to 1, and Equality of Odds lowered from 0.3 to 0.02. Results for group fairness were also good for the Adult dataset, with Statistical Parity reducing from 0.2 to 0.03, and DI increasing from 0.3 to over 0.5. The results for bias mitigation on the other datasets showed marginal improvements, except for DI for Taiwan-marriage, which increased from 1.04 to 1.1.for a logistic regression classifier.

As in the case of the Reweighing algorithm, ROC focuses on the predicted outcomes and ignores the non-Sensitive attributes. Therefore, the Sensitive test performance was unchanged, showing that this algorithm did not obfuscate the Sensitive attribute at all.

ROC changed the predicted labels to meet its fairness constraints, as seen in table 5.4.2.3 below for the logistic regression (LogReg) and random forest classifiers (RFC) used in this assessment:

		Taiwan Gender	Taiwan Marriage	Adult	German
LogReg	Before Label=0	663	656	3679	32
	After Label = 0	565	646	4481	46
	Before Label=1	2337	2307	1206	68
	After Label = 1	2435	2317	404	54
RFC	Before Label=0	663	656	3679	32
	After Label = 0	815	858	4577	43
	Before Label=1	2337	2307	1206	68
	After Label = 1	2185	2105	308	57

Table 5.4.2.3 Label value counts - before and after bias mitigation using Reject Option Classification.

5.4.3 Classifier Performance on transformed datasets

As referred to in the literature, classification model accuracy usually decreases after bias-mitigation [22]. However, considering that the biased model accuracy may be higher due to some discrimination in label outcomes, then the lower classification accuracy may actually be close to the true model accuracy, and hence desirable [14]. This assessment found marginal falls or no changes in classification model accuracies across all bias-mitigation algorithms with the exception for Adversarial debiasing.

ROC was consistent in its approach of improving model accuracy for logistic regression and showed only a marginal loss for the random forest classifier.

After Adversarial Debiasing, accuracies increased markedly, across all datasets and for both, logistic regression and random forest classifiers. This can be explained as an outcome of the adversarial approach, implemented over neural networks. By increasing the complexity and predictive power of each neural network (predictor and adversary), e.g. by increasing the number of inner layers, the adversarial approach can be fine-tuned to meet both fairness and model accuracy.

A core objective of discrimination-aware classifier is for high accuracy of future predictions, and low discrimination of new data. In this respect, the LFR and Adversarial Debiasing approaches were found to have the best overall performance in achieving fairness consistently across all datasets and minimising classifier accuracy loss in the process.

5.5 Potential AIF360 toolkit improvements

The choice of classifier e.g. logistic regression, random forest or other is important, as each gives varying balanced accuracy figures for both, label prediction and the Sensitive attribute test. It would be useful to have a tool to provide the before and after-mitigation falls or gains in classifier accuracy in percentage terms and establish thresholds for acceptable falls in accuracy.

Changing the values of the labels either in the training dataset or after classification is intrusive and could have legal ramifications. The AIF360 toolkit could include tools to measure the impact of label changes or allow thresholds to be changed in the mitigation algorithms to limit the number of label changes. This will form part of the bias-accuracy trade-off discussed in the next section.

6. Business Impact and the Bias / Discrimination trade-off

As this report found, debiasing training datasets and machine learning classifiers can result in lower classifier accuracy [22]. To minimise commercial risk, businesses seek to debias their classifiers with minimal loss of model accuracy. However, they must find an acceptable trade-off between optimising model accuracy and minimising bias.

A machine learning classifier's accuracy is determined from its resultant confusion table. These compare how well a classifier predicted the labels against the ground truth labels from the training dataset. The predicted labels reflect the classifier's probabilistic confidence of that prediction. If the probability is above a certain threshold, say 0.5, a favourable label is assigned, otherwise the data object gets an unfavourable label. The thresholds are expressed in terms of a model's Sensitivity (or True Positive Rate, TPR) and Specificity (or True Negative Rate, TNR). TPR is the classifier's ability to correctly predict a favourable label value for training data object that actually had a favourable label.

TNR is the same ability for unfavourable labels. The balanced accuracy of a classifier can be determined from its sensitivity and specificity, e.g.

$$\text{Balanced Accuracy} = \frac{\text{Sensitivity} + \text{Specificity}}{2}$$

Discrimination v/s accuracy charts were used to compare a classifier's loss in accuracy after debiasing. This assessment attempted to produce charts between the various measures of discrimination v/s accuracy for all thresholds. However, a comparison and interpretation of these results in terms of business outcomes was not trivial and hence not followed up. Figure 6.1 below shows the discrimination v/s accuracy charts for the pre-processing LFR bias mitigation algorithm for Taiwan-gender, using a random forest classifier (RFC) before and after debiasing (e.g. making 'fair').

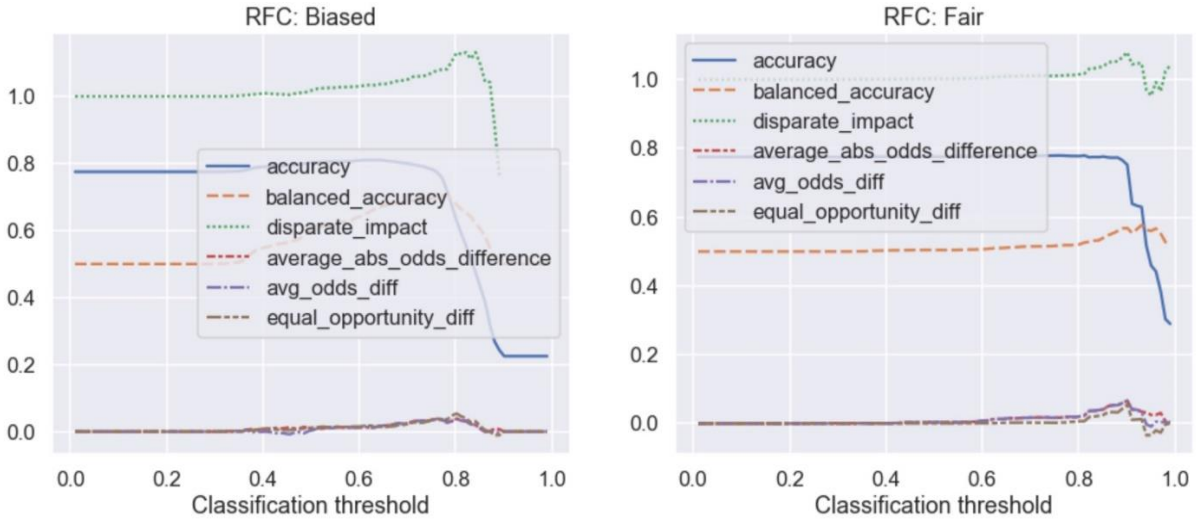


Figure 6.1 Discrimination v/s accuracy before and after using LFR bias mitigation algorithm using a random forest classifier.

All discrimination v/s accuracy charts for the pre-processing algorithms can be found in the individual pre-processing bias-mitigation Jupyter notebooks in the github repository [21].

Before or after debiasing, there is a trade-off between sensitivity and specificity, again based on business need. Setting a high threshold means that the classifier produces fewer favourable labels, e.g. offers lower true positives, but risks missing cases that actually should have been given favourable labels, e.g higher false negatives. These have business implications, and any change in thresholds as a result of debiasing need to be assessed in terms of business impacts. One way to measure the trade-off is to assign a cost-per-TPR and cost-per-TNR, so that business can enumerate their risk, or cost to the business, by varying the thresholds to reflect this risk. However, determining such unit costs may not be trivial for most businesses.

ROC curves [24] are used to show sensitivity/specificity trade-offs for the whole range of thresholds, from 0 -> 1, for binary classifiers. Computing the Area under the Curve (AUC) summarizes the trade-off in a single value. Figure 6.2 below shows the ROC curve and AUC for Taiwan-gender for the logistic regression (LR) and random forest classifiers (RFC) before and after bias mitigation using LFR.

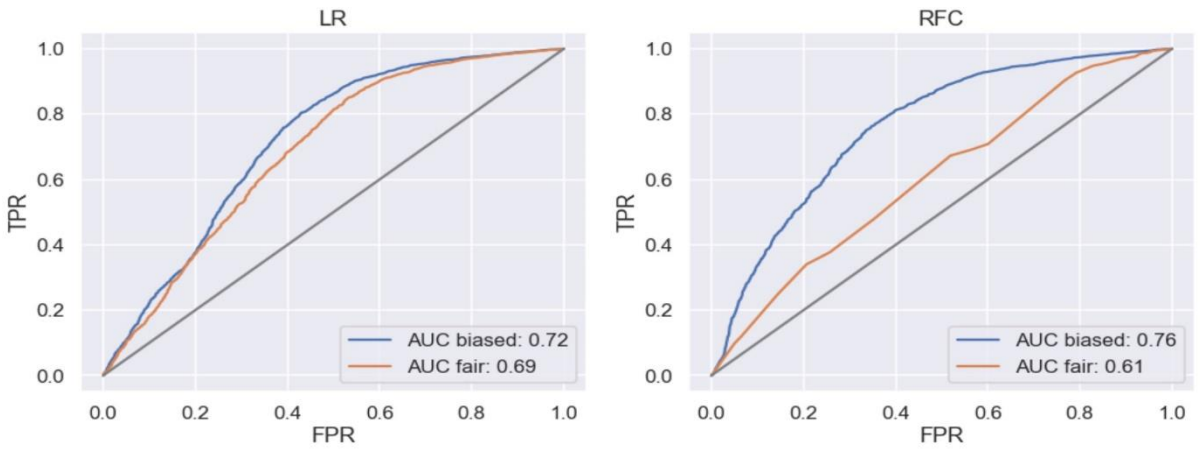


Figure 6.2: ROC curve and AUC for Taiwan-gender for the logistic regression (LR) and random forest classifiers (RFC) before and after bias mitigation using LFR.

ROC curves for before and after bias mitigation can be found in the scripts in the github repository [21].

It may not be possible to make a fair comparison of classifier accuracy before and after bias-mitigation if they have different thresholds. The 'relation between accuracy and fairness in binary classification' paper [23] discusses this problem and suggests the use of normalising accuracy techniques. Again, implementing this aspect of the assessment was not trivial and is not included in this report.

7. Conclusion & further work

Bias in machine learning classification problems is a new field of data science. Bias measures and mitigation approaches continue to evolve. Despite being trained on datasets that may reflect historic discrimination, biased outcomes from classifiers are generally unintended, and hitherto unchallenged in law. Nevertheless, businesses need to take start remedial measures ahead of any new laws that will make these measures statutory. When they do, business need to consider the impact bias-mitigation will have on their model accuracy, and in turn on their exposure to related business losses.

Current measures for bias include those for group and individual fairness. Achieving group fairness doesn't always guarantee individual fairness. Group fairness can be exploited by unscrupulous users to maintain bias against individuals. Any bias-mitigation algorithm deployed needs to be able to address both sets of measures.

LFR and Adversarial Debiasing mitigation algorithms met most of their success criteria compared to the other algorithms. LFR's mapping of the original training data objects to an intermediate unbiased representation can be generalised for any classification problem. The use of neural networks for Adversarial debiasing implies the need for very large training datasets.

The report uses the word debias instead of unbiased, because training datasets will always reflect some form of bias from another or several other attributes. Whilst these may not always be sensitive attributes, business will have to consider the business and legal implications of their influence on predicting the labels, and whether this implies a state of continuous debiasing.

7.1 Meeting the Report Objectives

This report met all its objectives:

- Objective 1: Measure data set and classification model bias using selected metrics from the AIF360 toolkit.
  The report measured group and individual bias for three datasets using four bias metrics.
- Objective 2: Apply selected bias-mitigation algorithms to the data sets.
  The report implemented five bias-mitigation algorithms
- Objective 3: Measure the effect of each bias-mitigation algorithm on model performance
  Results from dataset responses to bias-mitigation, bias measures, and mitigation algorithms were all captured and assessed.
- Objective 4: Assess the business implications of each bias-mitigation algorithm.
  The report discussed the bias / accuracy trade-off in terms of business risk.

7.2 Future work

There is a need for a commercial or open-source bias mitigation tool to automate the findings in this report. Given a biased training dataset, with a defined sensitive variable, privileged and unprivileged groups, and favourable and unfavourable labels, the tool would debias the dataset using available bias-mitigation algorithms, and return the various measures of discrimination, and an optimal debiased model accuracy / discrimination trade-off threshold.

More work is needed to assess the level of change in the data objects and the number of label changes for each algorithm. This would measure the 'intrusion' of each algorithm on the original training data, and couch these in terms of business impact. Changes in training data or predicted labels will be reflected in the resultant confusion matrix, which can be used to gauge the business cost for various levels of training data change.

The notion of an impartial Regulator model proposed in Fairness Through Awareness merits further investigation. An impartial regulator could oversee the creation of the fair intermediate training data representation whilst vendors could train their specialised classifiers against the debiased dataset, knowing that they had no exposure to legal challenge on any biased outcomes. Vendors would not have to change their classifiers every time the model was retrained to produce a new intermediate data representation.

8. References

[1] Big Data: Seizing Opportunities, Preserving Values
https://obamawhitehouse.archives.gov/sites/default/files/docs/big_data_privacy_report_may_1_2014.pdf

[2] 'Scattered literature' Assessing and Addressing Algorithmic Bias – But Before We Get There. Jean Garcia-Gathright et al 2018.
https://arxiv.org/pdf/1809.03332.pdf

[3] Algorithmic Bias in Autonomous Systems. David Danks et al 2017
https://www.ijcai.org/proceedings/2017/0654.pdf

[4] Data and Algorithmic Bias in the Web. Ricardo Baeza-Yates 2016
http://www.websci16.org/sites/websci16/files/keynotes/keynote_baeza-yates.pdf

[5] https://www.govtrack.us/congress/bills/116/hr1756/text/ih 2019
H.R. 1756: Preventing Credit Score Discrimination in Auto Insurance Act
https://www.govtrack.us/congress/bills/116/hr3875/text 2019
H.R. 3875: To prohibit Federal funding from being used for the purchase or use of facial recognition technology, and for other purposes.

[6] AI Fairness 360 Open Source Toolkit
https://aif360.mybluemix.net/

[7] Titanic dataset
https://www.kaggle.com/c/titanic/data

[8] Fairness Definitions Explained. Sahil Verma et al 2018
http://fairware.cs.umass.edu/papers/Verma.pdf

[9] Fairness Through Awareness, Dwork et al, 2011
https://arxiv.org/abs/1104.3913

[10] The 80% rule – a measure of Disparate Impact.
https://en.wikipedia.org/wiki/Disparate_impact#The_80.25_rule

[11] Conscientious Classification: A Data Scientist's Guide to Discrimination-Aware Classification. D'Alessandro et al 2017. https://www.liebertpub.com/doi/pdf/10.1089/big.2016.0048

[12] Learning Fair Representations, Zemel et al 2013.
https://www.cs.toronto.edu/~toni/Papers/icml-final.pdf

[13] Griggs v. Duke Power and the Disparate Impact Theory of Race Discrimination
http://www.law.unc.edu/documents/civilrights/dorosinnarapapergriggs.pdf

[14] Certifying and removing disparate impact. Feldman et al 2015
https://arxiv.org/abs/1412.3756

[15] Data pre-processing techniques for classification without discrimination. Faisal Kamiran et al. 2011
https://core.ac.uk/download/pdf/81728147.pdf

[16] Mitigating Unwanted Biases with Adversarial Learning. Brian Hu Zhang et al. 2018
https://arxiv.org/pdf/1801.07593.pdf

[17] Decision Theory for Discrimination-Aware Classification. Faisal Kamiran et al. 2012
https://mine.kaust.edu.sa/Documents/papers/ICDM_2012.pdf

[18] Taiwan dataset
https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients

[19] German Dataset
http://archive.ics.uci.edu/ml/datasets/Statlog+%28German+Credit+Data%29

[20] Adult Dataset
https://archive.ics.uci.edu/ml/datasets/adult

[21] Github repository with all code and results used for this assessment
https://github.com/benfaria/MSc-Project

[22] Discrimination Aware Decision Tree Learning. Kamiran et al, 2010
https://www.win.tue.nl/~mpechen/publications/pubs/KamiranICDM2010.pdf

[23] On the relation between accuracy and fairness in binary classification. Indre˙ Zliobait. 2015
https://arxiv.org/pdf/1505.05723.pdf

[24] Receiver operating characteristic (ROC) curve
https://en.wikipedia.org/wiki/Receiver_operating_characteristic

9. Code

All code relevant to this project can be found at https://github.com/benfaria/MSc-Project

There are 3 categories of code: The bias-mitigation code (e.g. Adult-PreProc-LFR.jpnb), data-prep code (e.g. Adult-Data-Prep.jpnb), and data-discovery code (e.g. Adult-Data-Discovery.jpnb)