Master Thesis Econometrics and Management Science

Specialization: Business Analytics and Quantitative Marketing

On the sensitivity of fair classification to deviations in data distributions

Abstract

As the use of machine learning within organisations increases, more attention goes to the associated ethical considerations. In recent years, many bias mitigation methods have been designed. However, not much focus has been attributed to the robustness of these methods. In this paper, we examine the sensitivity of several fair classification methods to deviations in the data distributions. We performed sensitivity analyses by creating different levels of bias in the training and test sets of the Taiwan Default data. Furthermore, we examined the performance of the models for Dutch census data of two different time periods with associated different amounts of bias. We find that the considered bias mitigation methods are tuned to the amount of bias contained in the data and are unable to adjust their predictions to deviating levels of bias between train and test sets. Furthermore, massaging or reweighing combined with the ensemble method XGBoost as classifier is found as the best option when taking into account sensitivity to data deviations, ability to reduce unfairness and predictive performance.

Author:
A.E. Vegter 580948

Supervisor: dr. M.H. Akyuz
Second assessor: dr. A. Alfons

August 4, 2021

Contents

1 Introduction

The use of machine learning in our society, and artificial intelligence (AI) in general, is becoming more prevalent. Over the last decade, the number of AI related papers has experienced exponential growth (D. Zhang et al., 2021). This phenomenon is not only visible in the academic world. More and more organisations, both in the public and private sector, are undergoing digital transformations and are deciding to make use of models, that use some form of AI, to base decisions on. Organisations that adopt AI in their business processes often report a revenue increase and, to a lesser extent, they report cost decreases (McKinsey, 2020). However, with the dramatic increase of AI, the corresponding ethical concerns have been underexposed. Some questions one may raise are: To what level do we want models to make decisions for us? Can we give these models the ethical principles we, as humans, have? Can a model exhibit discrimination and how can we restrict the level of discrimination?

Due to the black-box nature of many AI models, it is hard for engineers and users of these models to, firstly, understand and, more importantly, explain the behaviour and decisions made by the model. This problem raises concerns for both transparency and possible biases, causing (unintended) discrimination (Bostrom & Yudkowsky, 2014). In the context of ethics, we consider bias to be the systematic advantage privileged groups receive in contrast to the systematic disadvantage unprivileged groups receive (Bellamy et al., 2019). An illustration of this problem is the recruitment algorithm Amazon had been using, which turned out to be biased against women (Dastin, 2018). This bias was caused by a bias in the historical data used for training the model, which reflected the male dominance in the technical sector. Even after removing the protected attribute gender from the model, the bias did not completely disappear as some attributes highly correlated with gender were still present in the data. This example highlights the importance of integrating ethics into the stages of e.g. data collection, model development and model governance.

Over the last five years, the importance of applying ethical principles within AI has received more attention. In the AI Index Report 2021, D. Zhang et al. (2021) observe that the number of AI papers mentioning ethics-related keywords in the title is increasing, although still at a rather low level. These papers have contributed to the development of various ethical guidelines, which contain principles developers should comply with. Furthermore, they provide frameworks which assist developers in using the right tools to achieve fairness in their models (Bellamy et al., 2019; Saleiro et al., 2018). Hagendorff (2020) has made an extensive evaluation of the major guidelines within the field of AI ethics. However, he notes that the extent to which these ethical principles and values are implemented in the development and application of AI is rather low. One of the reasons ethics is not well integrated into the development of machine learning models is that it lacks a reinforcement mechanism. So far, no binding legal framework to adopt ethical guidelines in AI exists and organisations may only implement self-government. This causes the adoption of ethical guidelines to remain voluntary (Mokander & Floridi, 2021). Moreover, economic incentives play a role in the decision of organisations whether to implement ethical principles. Often, a trade-off exists between fairness and model performance, which is often correlated with earnings (Menon & Williamson, 2018). Lastly, domain-specific educational resources and tools are required (Holstein et al., 2019). However, being unable to integrate ethical considerations into the business processes can still cause damage to the brand by, for example, negative news headlines, of which the Amazon recruitment algorithm is a good illustration. Therefore, it is important for organisations to focus on implementing ethical principles, despite the lack of legal obligations.

As previously mentioned, tools have been developed to achieve fairness in AI and more specifically machine learning, which include various fairness evaluation metrics and bias mitigating algorithms. Hagendorff (2020) states that a stronger focus on technological details of the various methods and technologies regarding fairness in machine learning is needed. In recent years, much research has been conducted to determine which fairness metrics and bias mitigating algorithms are appropriate in conjunction with which modeling technique and for which context of use. However, not much focus has been attributed to the robustness of these models, for which it is desired that predictions and model performance do not change significantly for slight modifications in the input data or erroneous data. It will be important for the model designer to understand whether some bias mitigating techniques are more resilient to data changes than others, and if so, to what extent. This is relevant for both the modelling decisions as well as the expertise in determining which alerts should be set in the data monitoring phase. As a result, more knowledge on robustness may be important for the actual integration and acceptance of fairness in machine learning within business processes and, therefore, will be the focus of this research. This leads us to the following research question:

How sensitive are 'fair' classification models to deviations in the data?

By performing sensitivity analyses, we aim to answer the following sub-questions and subsequently our research question:

(1) *Are 'fair' models able to maintain their achieved level of fairness and predictive performance regarding changes in the data distributions?*

(2) *Which model combinations, in terms of type of classifier and applied bias mitigating technique, are least sensitive to data deviations?.*

In Section 2, we will elaborate on relevant definitions regarding discrimination, fairness and bias used in the context of machine learning. In Section 3, we will give an overview of existing literature related to our research and give an indication how our research will contribute to current academic knowledge. Furthermore, in this research, we will focus on a credit data set, which is used to predict the probability of default of customers on their credit debt in the next month. This data set, which is commonly known as the Taiwan Default data set, has been previously used in research regarding fairness. It has been shown that the data exhibits a bias towards male (Berk et al., 2017). Moreover, we will consider two Dutch census data sets, which are obtained in two different time periods and thereby contain different levels of discrimination. In the fairness literature, this data set has been used to predict whether an individual has a high level profession or not. Kamiran et al. (2010) have shown that the data sets exhibit a bias towards female, however the amount of bias is varying between the two time periods. These data sets will be elaborated upon in Section 4. To investigate the sensitivity of fair models, we will compare the performance between base classifiers and these classifiers combined with a bias mitigation algorithm and monitor how modifications in the Taiwan Default data affect the performance measures of the bias mitigation algorithm. These modifications will be done by means of resampling and relabelling in order to vary the level of unfair bias and class imbalance in the train data or the test data. We will perform this sensitivity analysis for several combinations of classifiers and bias mitigating algorithms in order to see how the sensitivity results may differ between different model configurations. Additionally, as the two Dutch census data sets contain different levels of discrimination, we are able to measure the sensitivity of our models on non-manipulated data and, thereby, validate our results. These methods are more extensively discussed in Section 5. In Section 6, we discuss the results. The main findings are summarized in Section 7 and, thereby, we provide answers to the two sub-questions and our research question. We found that the considered bias mitigation methods are tuned to the amount of bias contained in

the data and are unable to adjust their predictions to deviating levels of bias between the train and test sets. Massaging or reweighing combined with the ensemble method XGBoost is found as the best option when taking into account sensitivity to data deviations, the ability to reduce unfairness and predictive performance. Lastly, we discuss some limitations to our research and give directions for future research in Section 8.

2 Definitions

We will elaborate on definitions of often used terms throughout this paper regarding discrimination, fairness and bias in the context of machine learning. The aim of this section is to achieve consistency of terms used throughout this paper and to align this research with previous research.

2.1 Discrimination

In most parts of the world discrimination based on several characteristics such as race, gender, age etc. is prohibited. These characteristics are often referred to as protected attributes. What personal characteristics are exactly considered as protected attributes is determined by local law and regulations, but countries agree upon most of them. We can distinguish two sources of discrimination: direct discrimination and indirect discrimination (L. Zhang et al., 2016). We will use the following definitions:

Direct discrimination This form of discrimination occurs when individuals are treated less favourable explicitly based on the protected attribute(s). This type is also referred to as disparate treatment.

Indirect discrimination This form of discrimination arises when the treatment is based on solely non-protected attributes. However, some of these attributes are correlated with the protected attribute causing the outcome to still be unfavourable. This type is also referred to as disparate impact.

2.2 Fairness

Verma & Rubin (2018) have discussed the most prominent fairness definitions in the setting of classification and applied each one of them to a single case study, demonstrating that different definitions can and will have different outcomes regarding fairness. Therefore, no agreement has been established on which definitions are most appropriate and will depend on the notion of fairness one wants to adopt. As many definitions exist, we will only highlight the ones interesting for this research and refer to Verma & Rubin (2018) for the other, and more elaborate definitions. Fairness definitions can be subdivided into group fairness definitions and individual fairness definitions. The former ensures different groups are treated equally and the latter ensures similar individuals receive similar predictions. In this research, we will only consider group fairness.

Throughout this paper, we will consider the four group fairness definitions stated below. The first definition is more appropriate when one does not trust the labels of the data due to structural biases present in the data. This metric only incorporates predicted values and therefore can be used to close the possible gap between demographic groups. The last three definitions also incorporate the true value of the observation and thereby focus more on whether the correctness of the predictions is equal for different demographic groups. These metrics therefore account for possible differences in underlying abilities.

(1) Statistical parity The probability of being assigned to the positive class should be equal for both sub-groups. This fairness definition is also known as demographic parity.
(2) Equal opportunity The probability of correctly being assigned to the positive class should be equal for both sub-groups. Mathematically, this will be equal to the probability of incorrectly being assigned to the negative class. This fairness definition is also known as true positive rate parity.
(3) Equal mis-opportunity The probability of incorrectly being assigned to the positive class should be equal for both sub-groups. This fairness definition is also known as false positive rate parity.
(4) Equalized odds The probability of correctly being assigned to the positive class and the probability of incorrectly being assigned to the positive class should be equal for both sub-groups. The equalized odds criterion is equal to the equal opportunity and equal mis-opportunity criteria together. This fairness definition is also known as positive rate parity.

Mathematically, it has been shown that it is impossible to satisfy different fairness criteria simultaneously, except for extreme constrained cases (Kleinberg et al., 2016). Furthermore, enforcing group fairness often causes the model to suffer from individual bias as in order to ensure different demographic groups to be treated equally, similar individuals in the different groups have to be treated unequally (Maity et al., 2021). Hence, one should not blindly rely on one fairness criterion, but use these criteria with caution and attention to other measures.

2.3 Bias

Following the terminology of Hinnefeld et al. (2018), we define two types of bias data can contain in the context of fairness: 'sample bias' and 'label bias'. The distinction between these two types of bias is emphasized by the causal origin of the bias.

Sample bias This type of bias arises when specific sub-groups are sampled more often than other sub-groups, which causes an incorrect representation of the actual population. The existence of sample bias can have several reasons, e.g. wrong data collection or historical human biases present in the data.
Label bias This type of bias occurs when there is a causal link between certain sub-groups and the class label assigned to individuals of these sub-groups, which is not justified by ground truth.

3 Literature review

We will discuss the existing literature related to our research and indicate how our research will contribute to the current academic knowledge. Firstly, we discuss the application of fairness metrics and bias mitigating algorithms to similar use cases. Several of these metrics and algorithms exist and each situation or problem requires a different set of such tools. Therefore, to limit the scope of this literature review, we will only focus on research which meets the following requirements: (1) it is a classification problem (2) data exhibits class imbalance. Secondly, we examine how previous research has measured the sensitivity of similar methods.

3.1 Application of fairness metrics and bias mitigation algorithms

To promote a deeper understanding of fairness metrics and bias mitigation techniques, Bellamy et al. (2019) have created an open-source toolkit. This toolkit includes over 71 bias detection metrics and 9 bias mitigation algorithms. Methods to mitigate bias can be used in different stages

of the modelling process. We can distinguish three different types of algorithms: pre-processing algorithms, in-processing algorithms, and post-processing algorithms. Pre-processing algorithms are designed to reduce bias in the data by changing the training data. In-processing algorithms are aimed to reduce the bias in the classifier itself, as the classifier is taking fairness directly into account. Post-processing algorithms adjust the predictions of the model in order to reduce bias and, thereby, do not modify the underlying classifier and data. Suitability of the different metrics and algorithms is context-dependent. Therefore, there is no general consensus on which metrics and algorithms perform best. In order help data scientist navigate through all existing fairness metrics relevant for each use case, Saleiro et al. (2018) designed the Aequitas Fairness Tree. The tree is designed from the perspective of the decision maker and it is assumed that the decision maker has decided upon some policy options, such as whether the interventions based on the predictions will be assistive or punitive.

Kozodoi et al. (2021) specifically focus on the applicability of statistical fairness metrics and bias mitigating techniques in the context of profit-oriented credit scoring. They argue that the so-called separation criterion, which is equivalent to equalized odds, is the most suitable fairness criterion in the credit scoring context as this criterion accounts for the asymmetry in misclassification costs the customer as well as the financial institution face. Furthermore, they note that the choice of the bias mitigating technique depends on the feasibility of implementation and the preferences of the decision-maker regarding the profit-fairness trade-off. Post-processing methods, such as reject option classification, are easiest to implement, but come at a higher cost for improving fairness. In-processor methods, such as adversarial debiasing, on the other hand perform best in the profit-fairness trade-off. However, these methods require the deployment of a new algorithm. Focusing on the banking sector as well, Crupi et al. (2021) propose a general road map for fairness in machine learning and the implementation of a toolkit, `BeFair`, with the purpose of identifying and mitigating bias. The different stages of the road map include: regulatory aspects, data set assessment, choice of the fairness metrics, bias mitigation, and comparison and evaluation. The toolkit `BeFair` can be used to compare different models in order to identify the best strategy, given a chosen performance metric and fairness metric. Crupi et al. (2021) applied the framework and toolkit to a credit lending use case. Comparing different pre-processors, massaging the data set - which changes the labels of some observations - yielded best results. The in-processing techniques, adversarial debiasing and reductions, were able to reduce the statistical parity and equalized odds while maintaining the same performance as the baseline models. Applying pre-processing techniques to, among other data sets, the Dutch census data, Kamiran & Calders (2012) show that massaging and reweighing perform very well in lowering the discrimination at the cost of only a minor loss in accuracy.

Ravichandran et al. (2020) note that most bias mitigating methods are restricted to specific model families such as logistic regression or support vector machine models. However, other machine learning algorithms, such as XGBoost, have properties to be more scalable, transparent, robust and yield better performance. To combine these favourable properties with fairness, they propose a fairness variant of XGBoost that exploits the advantages while also reaching the level of fairness of the currently existing bias mitigating techniques. To compare the performance of their model with three in-processing bias mitigating techniques (prejudice remover, fair adversarial gradient tree boosting and adversarial debiasing), they use several common benchmark data sets, including the Taiwan Default data set. While using the disparate impact as fairness metric in combination with accuracy to measure model performance, their method outperforms on all but one data set. Y. Zhang & Zhou (2019) review statistical methods for imbalanced data treatment and bias mitigation. In their study, they focus on the impact of imbalanced data, bias metrics and the removal of biases. They

only consider the LightGBM algorithm, which is a gradient boosting framework that uses tree-based learning algorithms and therefore, the selection of modelling techniques together with parameter tuning is out of their scope. To deal with imbalance, they apply an over-sampling technique and furthermore consider the pre-processing method reweighing. Y. Zhang & Zhou (2019) apply their methodology to the Taiwan Default data set and consider statistical parity, equal opportunity and disparate impact as fairness metrics. In their study, the effect of balancing data is much higher than the effect of mitigating bias.

3.2 Measuring sensitivity

In order to investigate the ability of different fairness metrics to detect the two aforementioned types of bias, Hinnefeld et al. (2018) manipulate their data by adding artificial causal bias to the data. For the case of label bias, this is done by introducing different label thresholds regarding the protected attribute. In the case of sample bias the advantaged sub-group is sampled to have higher scores, while the disadvantaged sub-group is uniformly sampled. Their results show that metric sensitivity is dependent on the level of imbalance in the data and the bias type, emphasizing the importance of considering the causal origin of the bias in the data when selecting a fairness metric. However, they do notice that it is often not known a priori which type of bias the data contains. Fogliato et al. (2020) propose a sensitivity analysis framework for statistically evaluating risk assessment instruments according to several common fairness metrics. Their method shows how these fairness properties change with the level of bias present in the data. These results are used to determine the level of bias sufficient to contradict conclusions made on the fairness of the model. In contrast to this paper, they do not apply bias mitigating techniques and therefore focus on the robustness of the fairness metrics itself rather than methods to mitigate the bias.

Rukat et al. (2020) state that in order to measure the impact of data quality issues on the performance of a machine learning model in general, one can use manipulated copies of the original data to predict the performance of classifiers. Each copy then resembles common errors and data quality issues. The model performance and quantified output distribution will be obtained by applying the original classifier to the manipulated data sets. They suggest to use these outputs as input to a regression model, that learns to predict the model performance. This allows to set an alarm if the predicted performance falls below a specified threshold. In this research no bias mitigation algorithm is considered. However, these methods can be adjusted to be suitable to our context. Kamiran & Calders (2012) shortly touched upon the choice of the base classifier for the pre-processing technique massaging. They conducted controlled experiments for the k-nearest neighbour classifier, as the stability of this classifier can be influenced by the parameter k. They observe that if minimized discrimination is the main objective, an unstable classifier, i.e. one that is more sensitive to noise, is the better option. However, if one is also concerned with high accuracy, a stable classifier will be more suitable. Hence, to our knowledge, so far no research has been focused on measuring the sensitivity of bias mitigation algorithms to deviating data. However, we can make use of the set-up and results of the aforementioned researches.

4 Data description

In this study, we will use two different sources of data. Firstly, we will run sensitivity analyses on the Taiwan Default data. We will create several additional data sets with different levels of artificial bias, by means of sampling and relabelling the original data. However, as these sensitivity analyses are based on manipulations to the data, we will validate our findings with Dutch census data. This

data consists of two data sets, which are obtained in two different points in time and thereby reflect different levels of discrimination. Therefore, we do not need to perform any manipulations to the Dutch census data. In this case, the classifier can be trained on one data set and tested on the other to measure how sensitive the classifier is to deviations in the data.

4.1 Taiwan Default data

The Taiwan Default data set is publicly available in the UCI ML Repository (Yeh & Lien, 2009). The data is collected in October 2005 from an important bank in Taiwan and the targets were credit card holders of the bank. The data contains information on default payments, demographic factors, credit data, history of payment and bill statements of credit card clients from April 2005 to September 2005. The exact attributes and their variable type are given in Table 1. This data set has been previously used in research regarding algorithmic fairness (Berk et al., 2017; Grari et al., 2020; Lipton et al., 2018; Y. Zhang & Zhou, 2019). It has already been shown that the data exhibits a degree of bias towards males. As the goal of this research is not to show whether this actually the case, but to perform a sensitivity analysis of fair classifiers, we can make use of these previously obtained results.

Var nr	Variable	Variable type
1	Default payment	Binary
2	Amount of given credit	Continuous
3	Gender	Binary
4	Education	Nominal
5	Marital status	Nominal
6	Age	Continuous
7-12	Repayment status for months April - September	Nominal
13-18	Amount of bill statements for months April - September	Continuous
19-24	Amount paid for months April - September	Continuous

Table 1: Variables contained in Taiwan Default data set

The goal of this data is to predict the probability of default on payments of customers, where the target variable is given by *default payment*. The data set contains 30,000 observations of which 22.12% are defaults. We consider *gender* to be the protected attribute. In Table 2 we have displayed a cross tab of the target variable *default payment* and the protected attribute *gender*. We observe that females are represented in 60.4% of the cases while males constitute only 39.6% of the observations. Furthermore, for females the percentage of defaults equals 20.7%, while this percentage is higher for males, namely 24.2%.

		Gender		
		Female	Male	All
Default payment	No default	14,349	9,015	23,364
	Default	3,763	2,873	6,636
	All	18,112	11,888	30,000

Table 2: Cross tab of target variable and protected attribute

As previously mentioned, we will perform sensitivity analyses by manipulating both test data sets and train data sets. These manipulations are meant to introduce artificial bias into the data and will be obtained by resampling and relabelling the data. These manipulations will be elaborated upon in Section 5.3.

4.2 Dutch Census data

As the manipulated Taiwan Default data sets are not able to fully reflect real data, we will test our findings by means of the Dutch Virtual Census data (Minnesota Population Center, 2020). This data contains two data sets, one from 2001 and one from 2011. The data sets, released by Statistics Netherlands (CBS), contain information on personal characteristics, education level and profession. The Dutch census data is collected in such a way that it is representative of the total Dutch population. We have matched the coding of the variables of the two data sets to be completely aligned and the variables used in our analyses are displayed in Table 3 along with their variable type.

Var nr	Variable	Variable type
1	High profession	Binary
2	Gender	Binary
3	Age	Ordinal
4	Household size	Ordinal
5	Place of residence one year prior to census	Binary
6	Country of citizenship	Nominal
7	Country of birth	Nominal
8	Education	Nominal
9	Industry	Nominal
10	Marital status	Nominal

Table 3: Variables contained in Census data set

The 2001 data has been previously used by Kamiran et al. (2010) to predict whether an individual has a high level profession or not. They showed that the data exhibits bias towards females. After filtering out observations with missing values and under aged persons, the 2001 data contains 147,210 observations of which 19.4% have a high level profession and the 2011 data contains 293,430 observations of which 22.0% have a high level profession. Hence, the target variable is given by *high profession* and furthermore we consider *gender* to be the protected attribute. Comparing Table 4 with Table 5, we observe that the bias towards females seems to be decreased over time. In 2001, 13.5% of the women had a high profession compared to 26.2% of the men. These numbers have become more fair in 2011, as at that time 18.5% of the women had a high profession and 25.5% of men, although still a significant difference is present. We will train our models on the less discriminatory 2011 data and test it on the more discriminatory 2001 data. This is more elaborately described in Section 5.4.

		Gender		
		Female	Male	All
High profession	Yes	9,857	18,758	28,615
	No	65,680	52,915	118,595
	All	75,537	71,673	147,210

Table 4: Cross tab of target variable and protected attribute for 2001 data

		Gender		
		Female	Male	All
High profession	Yes	27,408	37,033	64,441
	No	120,650	108,339	228,989
	All	148,058	145,372	293,430

Table 5: Cross tab of target variable and protected attribute for 2011 data

5 Methodology

In this section, we discuss the fairness metrics used throughout this research to assess the level of fairness of the different models. Secondly, we make an outline of the different bias mitigation techniques applied. Then, we discuss how the sensitivity analyses will be performed for the different manipulated test and train sets of the Taiwan Default data. Lastly, we describe how we are going to validate our results by means of the Dutch census data.

5.1 Fairness metrics

To scope this research, we will focus on two fairness metrics. As there is no consensus on which metrics are most appropriate in which use case, we want to briefly clarify our choices. We will base our decision on both Aequitas' fairness tree (Saleiro et al., 2018) and previous research. We make the assumption that the predictions of probability of default will be used to assess whether customers of the bank should be granted credit and that the predictions of having a prestigious occupation will be used to assess whether an individual should be offered a prestigious job. Then, we argue that the predictions in both cases are used to intervene with a rather large part of the population. As these interventions can both be interpreted as helpful to individuals (assistive) and hurtful to individuals (punitive), we would like to consider both the true positive parity and false positive parity. Hence, we will focus on the equalized odds criterion, which is comprised of the criteria for true positive parity and false positive parity. In previous literature, equalized odds is often used to assess the fairness of credit scoring models (Crupi et al., 2021; Kozodoi et al., 2021; Y. Zhang & Zhou, 2019) and of models predicting a prestigious occupation (Pessach & Shmueli, 2020; Xu et al., 2020). Let us denote Y to be the true value of the target variable and let $\hat{Y}$ be the predicted value. In the binary classification case this implies that $Y \in \{0, 1\}$ and $\hat{Y} \in \{0, 1\}$. Furthermore, we want to denote the protected attribute *gender*, by G, where $G = 1$ is used for the protected group and $G = 0$ for the unprotected group. We measure how well the equalized odds

criterion is satisfied by means of the average odds difference (AOD):

$$\text{AOD} = \frac{1}{2} \cdot \underbrace{P[\hat{Y} = 1|G = 1, Y = 1] - P[\hat{Y} = 1|G = 0, Y = 1]}_{\text{true positive parity difference}} + \frac{1}{2} \cdot \underbrace{P[\hat{Y} = 1|G = 1, Y = 0] - P[\hat{Y} = 1|G = 0, Y = 0]}_{\text{false positive parity difference}}.$$

However, a flaw of the equalized odds criterion is it may not help to close the gap between demographic groups if structural biases are present in the data. Therefore, we also consider statistical parity difference. This metric, also known as demographic parity difference, is well-known and commonly used in the broader fairness literature as well as specifically for the context of our two use cases (Crupi et al., 2021; Kamiran et al., 2010; Pessach & Shmueli, 2020; Y. Zhang & Zhou, 2019). Furthermore, this metric is embedded in practical use as it is associated with anti-discriminatory regulatory laws. When replacing the difference with a ratio we obtain disparate impact. This ratio is used in the so-called four-fifths or 80% rule. This rule, established by the U.S. Equal Employment Opportunity Commission, states that the selection rate for the protected group should be at least 80% of the selection rate of the unprotected group (Zafar et al., 2017). Statistical parity equalizes outcomes across different demographic groups. This causes each group to be represented proportional to their representation in the overall population. However, it should be noted that this criterion does not take into account possible differences in underlying abilities. The statistical parity difference (SPD) is measured by:

$$\text{SPD} = P[\hat{Y} = 1|G = 1] - P[\hat{Y} = 1|G = 0].$$

Concluding, in this research we will focus on the following two fairness metrics:

1. Average odds difference
2. Statistical parity difference.

5.2 Bias mitigation algorithms

In this research we will only consider pre-processing and in-processing techniques. Pre-processing techniques are all meant to, in some way, change the train data which is being used as input to the predicting model in order to remove discrimination. Firstly, for the pre-processing phase we have chosen two methods based on previous literature: massaging and reweighing. Massaging and reweighing differ in the way they change the data (Kamiran & Calders, 2012). The former changes the labels of some observations while the latter changes the weights given to each observation. Both methods are aimed at achieving statistical parity. After the pre-processing step, a regular classifier is trained on the cleaned data. Crupi et al. (2021) showed that massaging yielded best results in their credit lending use case. In the prestigious job use case, Kamiran & Calders (2012) stated that massaging is slightly performing better than reweighing. However, Kozodoi et al. (2021); Y. Zhang & Zhou (2019) apply reweighing and show good improvements regarding fairness scores. We will examine the two mentioned pre-processing algorithms combined with four different well-known classifiers, namely logistic regression, random forest, decision tree and XGBoost.

In-processing methods replace an already deployed scoring model with a new algorithm. In this research, we will consider adversarial debiasing. This technique has been proven to perform very well in the performance-fairness trade-off (Crupi et al., 2021; Kozodoi et al., 2021). Adversarial debiasing constitutes of simultaneously training two competing neural networks (B. H. Zhang et al., 2018). We will follow the notation previously used and complement the notation by denoting X to be the set of predictors. The first network, the predictor, is trained to accomplish the task of predicting Y given X. The predictor tries to minimize its own loss function and at the same time, in order to maximize the loss function of the adversary, aims to hold back any additional information on the protect attribute in its output. The second network, the adversary, takes the output layer of the first network as input with the goal of predicting the protected attribute G. The adversary is only interested in minimizing its own loss function. The result of training these networks simultaneously is ensuring fairness. Concluding, we will investigate the following 'fair' classifiers:

1. Reweighing and logistic regression
2. Reweighing and random forest
3. Reweighing and decision tree
4. Reweighing and XGBoost
5. Massaging and logistic regression
6. Massaging and random forest
7. Massaging and decision tree
8. Massaging and XGBoost
9. Adversarial debiasing.

5.3 Sensitivity analyses

For the different aforementioned model combinations, we will perform sensitivity analyses to see how deviations in the data - by creating artificial bias - influence the performance to mitigate bias and at the same time make correct predictions. These sensitivity analyses will be performed on the Taiwan Default data and we will run both analyses for manipulated test sets as well as manipulated train sets. Because of these manipulations to the original data, we are able to observe how the models perform despite deviations in the data.

We have illustrated the steps to be taken in the sensitivity analyses for the manipulation of test sets in the pseudo-code below. This pseudo-code is written down generally for the three bias mitigation techniques we consider.

Algorithm 1: Sensitivity analysis for a 'fair' classifier for manipulated test sets

Input: Original data set $\mathcal{D}$

1 Split data $\mathcal{D}$ into 5 folds $\mathcal{F}_i$, $i = 1, \ldots, 5$;
2 **foreach** $\mathcal{F}_i$ *(*$i = 1, \ldots, 5$*)* **do**
3 Use fold $\mathcal{F}_i$ as test data: $\mathcal{D}_{test} = \mathcal{F}_i$;
4 Use remaining folds as training data: $\mathcal{D}_{train} = \mathcal{F}_j, j \neq i$;
5 Using 5-fold cross-validation, perform hyperparameter tuning on $\mathcal{D}_{train}$ of (1) only the base classifier and (2) the base classifier combined with a bias mitigation algorithm;
6 Obtain the best models $\mathcal{M}_{best}$ for (1) and (2) based on the averaged performance scores;
7 Manipulate $\mathcal{D}_{test}$ in 4 different ways to obtain several manipulated test sets $\mathcal{T}_k$, $k = 2, ..., 5$ and the original test set $\mathcal{T}_1$;
8 **foreach** $\mathcal{T}_k$ *(*$k = 1, ..., 5$*)* **do**
9 Test $\mathcal{M}_{best}$ on $\mathcal{T}_k$;
10 Obtain performance and fairness scores of (1) and (2) for manipulation k;
11 **end**
12 **end**
13 Average the performance and fairness scores for each manipulation k over the 5 folds;
14 **return** *Averaged performance and fairness scores of (1) and (2) for each k*

To obtain stable sensitivity results, we perform a nested cross-validation. The outer loop as well as the inner loop are 5-fold cross validations. This pseudo-code will differ between the pre-processing algorithms and in-processing algorithms in the inner loop (lines 5-6). In this step the train data $\mathcal{D}_{train}$ will again be split five times into train and validation sets. Firstly, we will only train the base classifier (without any bias mitigation). Secondly, we will train the base classifier in combination with a bias mitigation algorithm. For the pre-processing algorithm, we will apply the pre-processing technique on the train data. The classifier will be trained on the transformed train data, after which we test the models on the validation data sets in order to determine the best model settings. This process differs a bit for the in-processing algorithm. We will apply the in-processing technique directly on the train data, after which we test the models on the validation data to obtain the best model. We will determine which model configuration is best by means of performance scores. For the pre-processing techniques, we have chosen to optimize the model configurations over recall[1] rather than accuracy[2], as we are dealing with imbalanced data. The implementation of the in-processing technique we use, is set up in such a way that this algorithm optimizes for accuracy. We explicitly do not optimize for one of the fairness metrics, as we want to mimic the business setting as much as possible and their main concern will usually still be the predictive performance, despite possible concerns of unfairness.

After model training, we test the model on unseen data. However, unlike usual we do not only feed the original test data to the model, but also several manipulated test sets. These manipulated test set groups are created by means of sampling. For each test group, we sample one specific subgroup of the population relatively more, which changes the level of unfair bias in that test group. We refer to this type of bias as sample bias, as discussed in Section 2.3. We measure the level of unfair bias by the difference in ratio of default and non-default between the protected group and the unprotected group. At the same time, we also take into account that the resulting differences in class imbalance, the total percentage of defaults, between the test groups are well distributed. The resulting grouping of manipulated test sets is displayed in Table 6, where test group C reflects

[1] Recall $= \frac{\text{True positives}}{\text{True positives+False negatives}}$
[2] Accuracy $= \frac{\text{True positives+True negatives}}{\text{Total}}$

the original data distribution. Then, we calculate the performance and fairness scores for the model with and without bias mitigation. By comparing these scores we can measure the effect of bias mitigation for different test data distributions.

		Unfair bias		
		Smaller	Original	Larger
Class imbalance	Smaller	A	-	E
	Original	-	C	-
	Larger	B	-	D

Table 6: Dimensions of manipulations resulting in new test data sets

Moreover, we do not only perform manipulations to the test data, but to the training data as well. Again, we have illustrated the steps to be taken in the sensitivity analysis for the manipulation of training sets in the pseudo-code below. This pseudo-code differs only slightly from the one for manipulations of the test sets, and we will therefore only elaborate on the differences.

Algorithm 2: Sensitivity analysis for a 'fair' classifier for manipulated train sets

Input: Original data set $\mathcal{D}$
Split data $\mathcal{D}$ into 5 folds $\mathcal{F}_i$, $i = 1, \ldots, 5$;
foreach $\mathcal{F}_i$ *(*$i = 1, \ldots, 5$*)* **do**
  Use fold $\mathcal{F}_i$ as test data: $\mathcal{D}_{test} = \mathcal{F}_i$;
  Use remaining folds as training data: $\mathcal{D}_{train} = \mathcal{F}_j, j \neq i$;
  Manipulate $\mathcal{D}_{train}$ in 4 different ways to obtain several manipulated training sets $\mathcal{T}_k$, $k = 2, ..., 5$ and the original training set $\mathcal{T}_1$;
  foreach $\mathcal{T}_k$ *(*$k = 1, ..., 5$*)* **do**
    Using 5-fold cross-validation, perform hyperparameter tuning on $\mathcal{T}_k$ of (1) only the base classifier and (2) the base classifier combined with a bias mitigation algorithm;
    Obtain the best models $\mathcal{M}_{best}$ for (1) and (2) based on the averaged performance scores;
    Test $\mathcal{M}_{best}$ on $\mathcal{D}_{test}$;
    Obtain performance and fairness scores of (1) and (2) for manipulation k;
  end
end
Average the performance and fairness scores for each manipulation k over the 5 folds;
return *Averaged performance and fairness scores of (1) and (2) for each k*

Compared to the case of the different test sets, we now obtain several manipulated training sets, before training the classifiers. These manipulations will be done in two different ways: sampling and relabelling. The method of sampling will be similar to the case of manipulating test sets by sampling a subgroup of the population relatively more and thereby creating artificial sample bias in the data. However, as we are now considering train sets, we are also able to perform manipulations by means of relabelling. Now, for specific subgroups of the data, we relabel some observations. So, observations with *default* are relabelled to *no default* and vice versa. This type of bias is referred to as label bias, as discussed in Section 2.3. Again, both of these manipulations are meant to change the level of unfair bias for that specific train group while at the same time making sure the differences in class imbalance are well distributed among the train groups. The resulting grouping of manipulated training sets is displayed in Table 7. The other parts of the sensitivity analysis are similar to the case of manipulating test data.

		Unfair bias		
		Smaller	Original	Larger
Class imbalance	Smaller	A and G	-	E and I
	Original	-	C and H	-
	Larger	B and F	-	D and J

Table 7: Dimensions of manipulations resulting in new training data sets (groups A-E are the result of sampling and groups F-J are the result of relabelling)

5.4 Census data

In order to see whether the results obtained from the sensitivity analyses on the manipulated data sets also hold for "real" data, we consider the Dutch census data sets of 2001 and 2011. The level of discrimination towards females regarding having a prestigious job has naturally decreased over the years. We will train the different models on the less discriminatory data set of 2011 and make predictions for the more discriminatory data of 2001. Furthermore, for sake of comparison we will also test the trained models on the 2011 data itself.

As we do not perform any manipulations to the data sets, it is hard to know for sure which kind of bias the data might exhibit. However, one can argue that the bias contained in the data could be due to label bias caused by historical human biases in which women are considered to be less suitable for higher functions.

6 Results

This section elaborates on the results[3] of the sensitivity analyses for the Taiwan Default data and the modelling results for the census data. Firstly, we will discuss the results of sensitivity analyses for the differentiation in test sets of the Taiwan Default data. Secondly, we discuss the results of the sensitivity analyses regarding differentiation in training sets of the Taiwan Default data. Lastly, we describe the obtained results for the two Dutch census data sets.

For all the different models that have been trained it holds that no variable selection has been done, hence all variables in Tables 1 and 3 are used. Furthermore, the nominal variables have been converted into dummy variables and specifically for the Taiwan Default data, the data has been scaled in order for the features to have the same ranges. Furthermore, for each classifier we have performed hyperparameter tuning by running a random search on a grid of hyperparameter values. The grid used for each classifier can be found in Appendix, Section 9.1. Regarding the bias mitigation algorithms, we have implemented the reweighing and adversarial debiasing algorithms from the `aif360 sklearn`[4] library and the massaging algorithm from the `themis-ml`[5] library.

We want to compare the effect that the bias mitigation methods have on the level of unfairness for the different levels of bias contained in the data. To calculate this effect on the statistical parity difference (SPD) and average odds difference (AOD), which can be interpreted as the percentage of

[3] Used code can be found on: `https://github.com/AukjeE/MasterThesis`

[4] `https://github.com/Trusted-AI/AIF360/tree/master/aif360/sklearn`

[5] `https://github.com/cosmicBboy/themis-ml`

unfairness that is eliminated, we use the following formulas:

$$\text{Effect}_{SPD} = \frac{|SPD_{\text{no bias mitigation}}| - |SPD_{\text{bias mitigation}}|}{|SPD_{\text{no bias mitigation}}|}$$

$$\text{Effect}_{AOD} = \frac{|AOD_{\text{no bias mitigation}}| - |AOD_{\text{bias mitigation}}|}{|AOD_{\text{no bias mitigation}}|}.$$

In cases where unfairness increases after bias mitigation instead of decreases, the above formulas are not very suitable and therefore the effect is calculated slightly different to prevent dividing by a value close to zero and keep the results interpretable. These exceptional cases are marked with an asterisk (*) in the tables and will make use of the following formulas for the effect on SPD and AOD:

$$\text{Effect}^*_{SPD} = \frac{|SPD_{\text{no bias mitigation}}| - |SPD_{\text{bias mitigation}}|}{|SPD_{\text{bias mitigation}}|}$$

$$\text{Effect}^*_{AOD} = \frac{|AOD_{\text{no bias mitigation}}| - |AOD_{\text{bias mitigation}}|}{|AOD_{\text{bias mitigation}}|}.$$

Furthermore, in the boxplots, we display the effect of bias mitigation on the unfairness in a different way. In this case, we calculate the effect as the difference in SPD and AOD scores without and with bias mitigation:

$$\text{Difference}_{SPD} = SPD_{\text{no bias mitigation}} - SPD_{\text{bias mitigation}}$$
$$\text{Difference}_{AOD} = AOD_{\text{no bias mitigation}} - AOD_{\text{bias mitigation}}.$$

6.1 Differentiation in test sets Taiwan Default data

For generating the different test sets, we have sampled specific subgroups relatively more to create "artificial" sample bias. In Table 8, the resulting test sets of these different manipulations are reported, together with their characteristics regarding the unfair bias and class imbalance. The level of unfair bias is represented by the difference in default ratio for males and females in the test group.

	Subgroup sampled relatively more	Difference in default ratio male and female	Class imbalance
Test group A	*Male with no default*	-3.1%	20.0%
Test group B	*Female with default*	-0.5%	24.1%
Test group C	*Original distribution*	3.5%	21.9%
Test group D	*Male with default*	7.1%	23.6%
Test group E	*Female with no default*	9.9%	16.9%

Note: Reported numbers are the averages over five folds

Table 8: Generated test sets and their characteristics

These different test sets are used for measuring the sensitivity of both the pre-processing methods reweighing and massaging, as well as for the in-processing method adversarial debiasing. We measure the sensitivity of these methods by means of changes in the aforementioned fairness metrics SPD and AOD. Furthermore, to measure the predictive performance of the model, we report the recall and accuracy scores. However, as we deal with imbalanced data, we mainly focus on the recall scores and report accuracy scores for completeness and the ability to compare with other papers. As no correlation has been observed between the level of class imbalance and the sensitivity of the different methods, we will not discuss this further.

6.1.1 Reweighing

We have combined the pre-processing technique reweighing with four different classifiers, namely logistic regression, decision tree, random forest and XGBoost. The results are presented in Tables 9 and 10.

In Table 9, the SPD scores of the different classifiers are presented without reweighing as well as with reweighing. Furthermore, we have reported the effect of reweighing by calculating the percentage of unfairness that is eliminated. As can be seen from this table, for each classifier the SPD without reweighing increases with the amount of sample bias in the test data. The size of effect reweighing has on the SPD varies between the classifiers and test groups. Random forest barely reduces any bias present in the data and also decision tree is not very effective. Although for logistic regression and XGBoost reweighing has a better ability to reduce unfairness in their predictions, this ability is not adaptive to the level of bias contained in the different test groups as the effect reduces with the increasing bias. The variability in effect is slightly higher for XGBoost. Inspecting Table 10, we observe that the AOD is not clearly correlated with the amount of sample bias in the test data. Similar to SPD, the size of effect differs between classifiers and test groups. Again, reweighing has the least ability to reduce unfairness when combined with random forest and decision tree. The variability of the effect of reweighing for logistic regression and XGBoost is similarly and does not seem to be affected by the level of bias in the different test groups. As the differences between performance scores without and with reweighing for each classifier are negligible, we have only presented recall and accuracy scores after reweighing in Table 11. A clear relation can be seen between the SPD and AOD after reweighing and the recall scores of the classifier, this phenomenon is often referred to as the fairness-performance trade-off. However, it is notable that this trade-off is only observed between classifiers and not between test groups.

	No reweighing				Reweighing				Effect			
	LR	RF	DT	XGB	LR	RF	DT	XGB	LR	RF	DT	XGB
Test A	1.3%	0.6%	1.8%	1.0%	-0.2%	0.6%	1.3%	0.0%	85%	0%	28%	100%
Test B	2.0%	1.4%	2.3%	1.6%	0.3%	1.3%	1.7%	0.5%	85%	7%	26%	69%
Test C	3.1%	2.5%	3.2%	2.7%	1.4%	2.4%	2.4%	1.8%	55%	4%	25%	33%
Test D	4.1%	3.7%	3.8%	4.0%	2.5%	3.6%	3.2%	2.9%	39%	3%	16%	28%
Test E	5.2%	4.9%	4.7%	5.1%	3.8%	4.9%	3.9%	4.1%	27%	0%	17%	20%
Average	3.1%	2.6%	3.2%	2.9%	1.6%	2.6%	2.5%	1.9%	58%	3%	22%	50%

Note: Reported numbers of Test A - Test E are the averages over five folds

Table 9: SPD scores without and with reweighing and the effect for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB)

	No reweighing				Reweighing				Effect			
	LR	RF	DT	XGB	LR	RF	DT	XGB	LR	RF	DT	XGB
Test A	3.1%	1.9%	2.0%	2.1%	0.5%	1.7%	1.9%	0.7%	84%	11%	5%	67%
Test B	2.8%	1.7%	2.1%	1.7%	0.1%	1.3%	1.7%	0.3%	96%	24%	19%	82%
Test C	2.8%	1.7%	2.0%	2.0%	0.2%	1.3%	1.6%	0.6%	93%	24%	20%	70%
Test D	2.7%	1.5%	2.1%	1.9%	0.2%	1.2%	1.5%	0.5%	93%	20%	29%	74%
Test E	3.6%	2.4%	2.7%	2.6%	1.2%	2.1%	2.1%	1.2%	67%	13%	22%	54%
Average	3.0%	1.8%	2.2%	2.1%	0.4%	1.5%	1.8%	0.7%	86%	18%	19%	69%

Note: Reported numbers of Test A - Test E are the averages over five folds

Table 10: AOD scores without and with reweighing and the effect for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB)

	Recall				Accuracy			
	LR	RF	DT	XGB	LR	RF	DT	XGB
Test A	0.335	0.372	0.411	0.360	0.832	0.828	0.734	0.830
Test B	0.335	0.374	0.417	0.361	0.808	0.807	0.720	0.808
Test C	0.332	0.374	0.414	0.360	0.820	0.818	0.728	0.819
Test D	0.333	0.377	0.419	0.362	0.810	0.810	0.722	0.810
Test E	0.334	0.372	0.412	0.358	0.853	0.848	0.749	0.850
Average	0.334	0.374	0.415	0.360	0.824	0.822	0.731	0.823

Note: Reported numbers of Test A - Test E are the averages over five folds

Table 11: Performance scores after reweighing for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB)

To dive deeper into sensitivity of the different models, we consider the boxplots in Figure 1 and Figure 2. In these boxplots the effect of reweighing is displayed as the difference in percentage points of SPD and AOD before and after reweighing. These boxplots show the variability of the effect for SPD and AOD within test groups for the four different classifiers. Within these test groups the level of sample bias is constant. However, the observations correspond to different folds of the test data. In Figure 1 we observe that for SPD, logistic regression is the most stable in terms of deviation within test groups. In contrast, the performance of decision tree with respect to SPD performs very volatile as it has a high deviation within the different test groups. Both XGBoost and random forest behave mediocre with respect to this deviation, however random forest barely reduces any bias in the data. The behaviour of these different classifiers can be explained by their design, as random forest and XGBoost combine several decision trees, which makes them more robust in general. Looking at Figure 2, similar conclusions can be drawn with respect to AOD, however some minor differences regarding these conclusions can be observed. Decision tree behaves even more volatile in reducing bias and random forest now seems to have a small effect on decreasing the bias, when considering AOD as fairness metric.

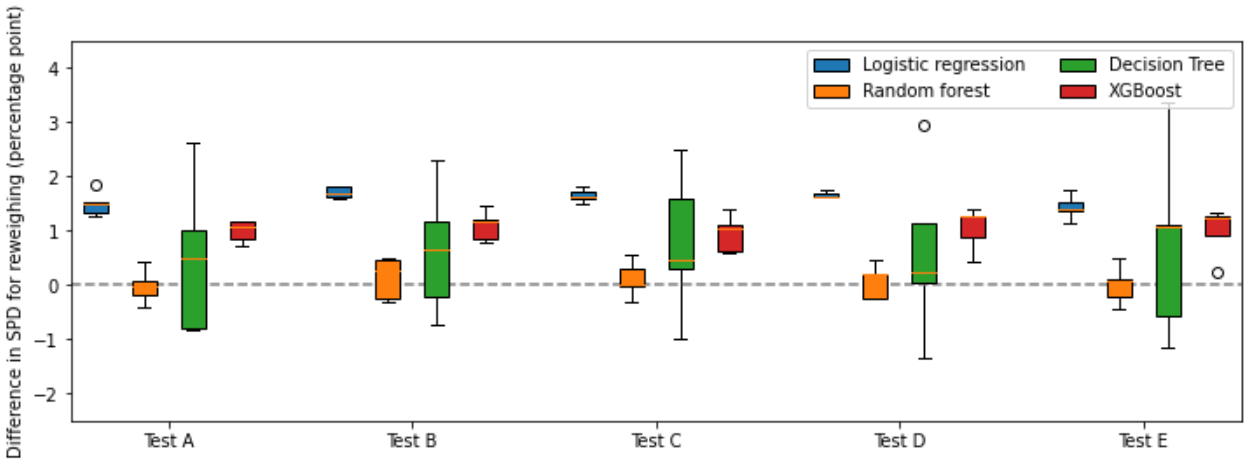

Figure 1: Effect of reweighing on SPD for different classifiers and test groups

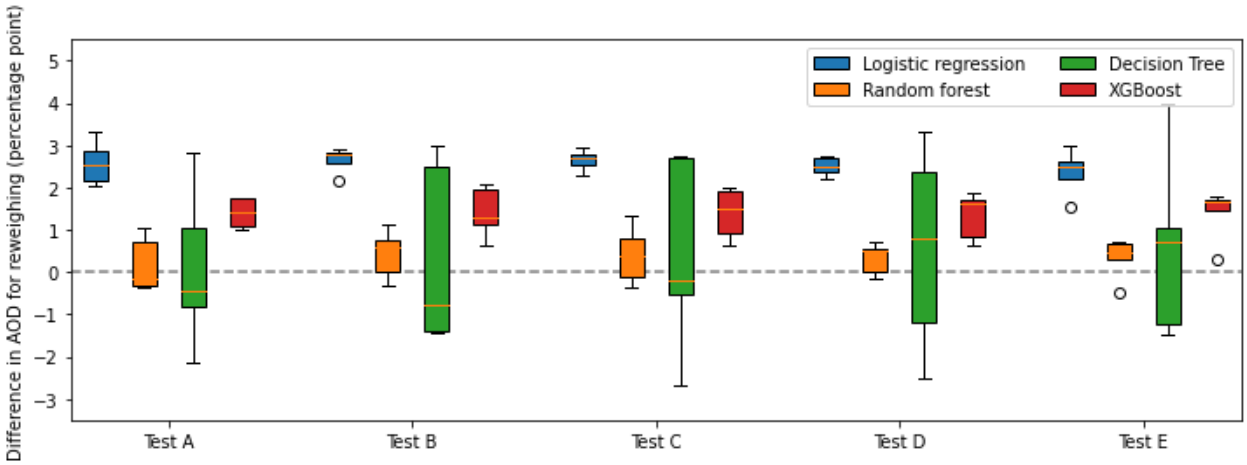

Figure 2: Effect of reweighing on AOD for different classifiers and test groups

6.1.2 Massaging

Similar to reweighing, we have combined massaging with the following four classifiers: logistic regression, decision tree, random forest and XGBoost. In Tables 12 and 13, we have presented the SPD and AOD scores without massaging and with massaging as well as the effect for the four classifiers.

Inspecting Table 12, we observe that, similar to reweighing, the SPD scores are correlated with the level of sample bias in the test groups. We observe that except for test group A, the ability of massaging to remove all bias in the predictions decreases with the amount of bias in the test data. As for test group A, massaging is working "too well" for all classifiers, the predictions contain a reversed bias and therefore the effect of massaging on the fairness is rather low. Comparing the effect of massaging on SPD for the different classifiers, decision tree is most effective as well as least differentiating between test groups. In Table 13, we note that for decision tree and XGBoost, massaging overcompensates the level of unfair bias when considering AOD as fairness metric. For reducing the AOD, massaging is most effective when combined with logistic regression. This method

is least variable in the effect as well. From Table 14, we observe that the predictive performance of massaging in combination with the different classifiers is almost equal to reweighing. To a lesser extent than in the case of reweighing, a fairness-performance trade-off exists.

	No massaging				Massaging				Effect			
	LR	RF	DT	XGB	LR	RF	DT	XGB	LR	RF	DT	XGB
Test A	1.3%	0.6%	1.8%	1.0%	-0.5%	-0.5%	-0.8%	-0.7%	62%	17%	56%	30%
Test B	2.0%	1.4%	2.3%	1.6%	0.1%	0.3%	-0.5%	-0.2%	95%	79%	78%	88%
Test C	3.1%	2.5%	3.2%	2.7%	1.1%	1.5%	0.5%	0.8%	65%	40%	84%	70%
Test D	4.1%	3.7%	3.8%	4.0%	2.2%	2.6%	1.3%	2.0%	46%	30%	66%	50%
Test E	5.2%	4.9%	4.7%	5.1%	3.4%	3.9%	2.1%	3.3%	35%	20%	55%	57%
Average	3.1%	2.6%	3.2%	2.9%	1.3%	1.6%	0.5%	0.8%	60%	37%	68%	59%

Note: Reported numbers of Test A - Test E are the averages over five folds

Table 12: SPD scores without and with massaging and the effect for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB)

	No massaging				Massaging				Effect			
	LR	RF	DT	XGB	LR	RF	DT	XGB	LR	RF	DT	XGB
Test A	3.1%	1.9%	2.0%	2.1%	0.1%	0.2%	-0.6%	-0.2%	97%	89%	70%	90%
Test B	2.8%	1.7%	2.1%	1.7%	-0.1%	0.1%	-0.9%	-0.7%	96%	94%	57%	59%
Test C	2.8%	1.7%	2.0%	2.0%	-0.2%	0.4%	-0.6%	-0.9%	93%	76%	70%	55%
Test D	2.7%	1.5%	2.1%	1.9%	-0.2%	0.0%	-0.8%	-0.9%	93%	100%	62%	53%
Test E	3.6%	2.4%	2.7%	2.6%	0.6%	0.9%	0.3%	0.2%	83%	63%	89%	92%
Average	3.0%	1.8%	2.2%	2.1%	0.0%	0.3%	-0.5%	-0.5%	92%	85%	70%	70%

Note: Reported numbers of Test A - Test E are the averages over five folds

Table 13: AOD scores without and with massaging and the effect for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB)

	Recall				Accuracy			
	LR	RF	DT	XGB	LR	RF	DT	XGB
Test A	0.338	0.378	0.414	0.368	0.831	0.826	0.737	0.821
Test B	0.339	0.379	0.423	0.369	0.807	0.805	0.724	0.799
Test C	0.336	0.377	0.417	0.365	0.819	0.815	0.729	0.809
Test D	0.337	0.380	0.422	0.365	0.809	0.807	0.725	0.800
Test E	0.338	0.376	0.413	0.368	0.851	0.844	0.750	0.840
Average	0.338	0.378	0.418	0.376	0.823	0.819	0.733	0.814

Note: Reported numbers of Test A - Test E are the averages over five folds

Table 14: Performance scores after massaging for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB)

We have plotted the variability in effect of massaging on the AOD and SPD between the different folds for each test group and classifier in Figure 3 and Figure 4. This effect is measured by the difference in percentage points of AOD and SPD scores without and with massaging. We compare the deviation of each classifier within the test groups by looking at the sizes of the boxes and

whiskers in the box plot. It can be noted that of the four investigated classifiers, decision tree has the most unstable effect within test groups for both SPD and AOD. On the other hand, random forest is showing least deviation in effect for SPD and AOD within test groups. As can be concluded, the degree of variability within test groups for each classifier is similar for the two different fairness metrics SPD and AOD.

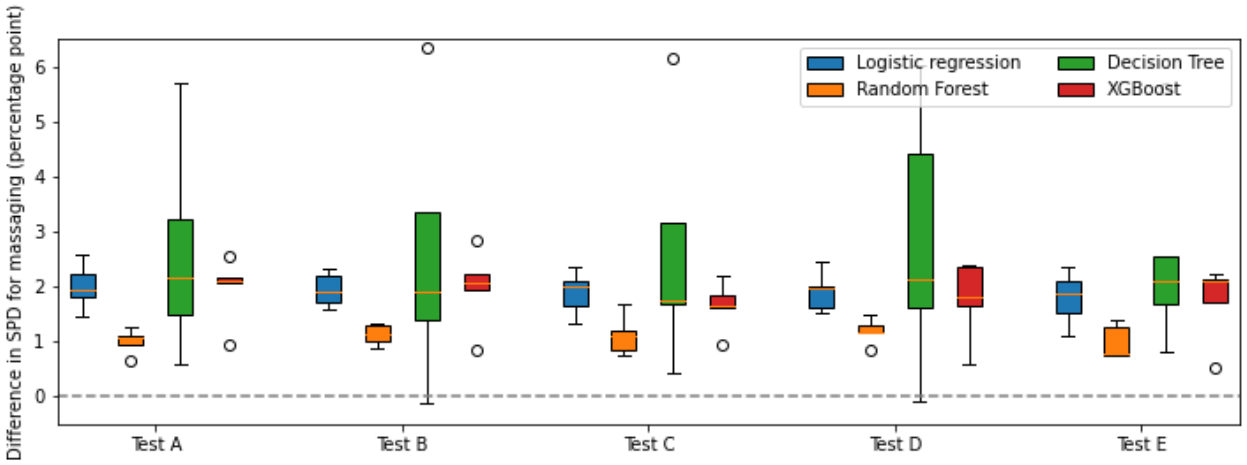

Figure 3: Effect of massaging on SPD for different classifiers and test groups

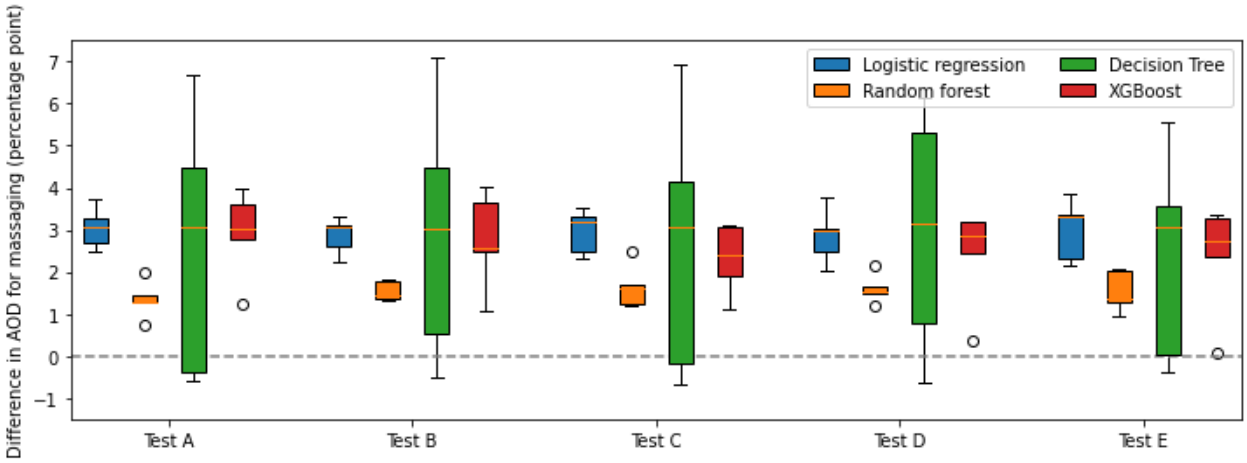

Figure 4: Effect of massaging on AOD for different classifiers and test groups

6.1.3 Adversarial debiasing

We now consider the in-processing technique adversarial debiasing, which simultaneously trains two competing neural networks; a classifier and an adversary. The results of the sensitivity analysis can be found in Table 15. We observe that, again, the SPD scores are clearly correlated with the sample bias in the test data, however this correlation cannot be found for AOD. Adversarial debiasing reduces the amount of bias in the predictions, however judging the AOD scores with adversarial debiasing now the predictions contain a certain bias towards female instead. We notice that the predictive performance of the model, for which we focus on the recall scores, is lower than most of the classifiers combined with massaging or reweighing. Moreover, when inspecting the boxplots in

Figure 5 we notice that the effect of adversarial debiasing on the SPD and AOD scores differs quite a bit within test groups.

	No adversarial debiasing		Adversarial debiasing		Effect		Predictive performance	
	SPD	AOD	SPD	AOD	SPD	AOD	Recall	Accuracy
Test A	0.8%	2.3%	-0.9%	-0.4%	-11%*	83%	0.350	0.829
Test B	1.6%	2.1%	-0.3%	-0.6%	81%	71%	0.355	0.806
Test C	2.5%	1.8%	0.7%	-0.8%	72%	56%	0.348	0.816
Test D	4.0%	2.2%	2.0%	-0.4%	50%	82%	0.349	0.808
Test E	5.0%	2.9%	3.3%	0.3%	34%	90%	0.350	0.847
Average	2.8%	2.3%	1.0%	-0.4%	45%	76%	0.350	0.821

Note: Reported numbers of Test A - Test E are the averages over five folds

Table 15: SPD and AOD scores without and with adversarial debiasing, the effect of adversarial debiasing and the recall and accuracy scores after adversarial debiasing

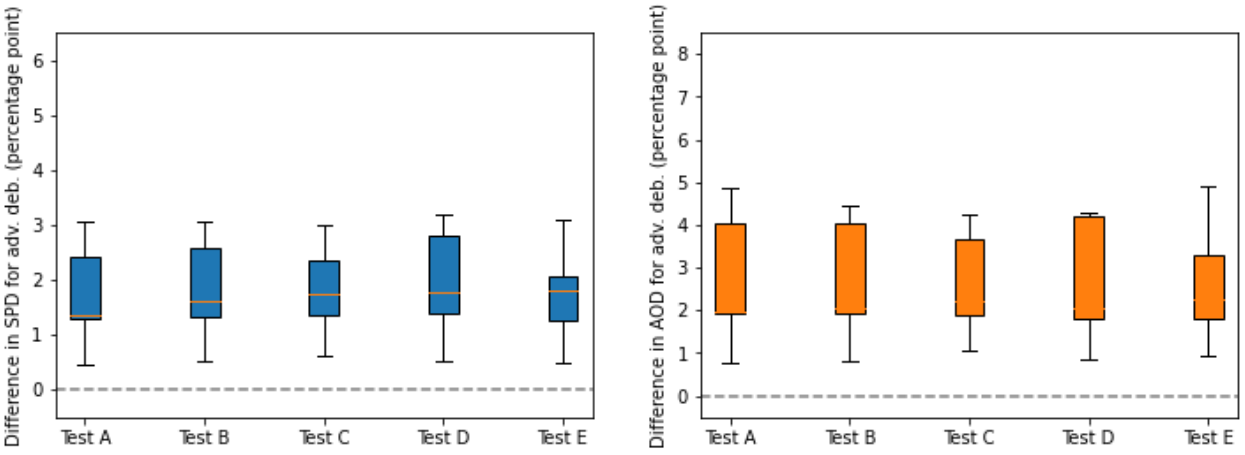

Figure 5: Effect of adversarial debiasing on SPD (left) and AOD (right) for different test groups

6.1.4 Comparison between bias mitigation algorithms

We will now compare the results of the different bias mitigation techniques. Firstly, for all three bias mitigation techniques we observe that SPD is able to measure the difference in bias introduced into the test set by our manipulations, which cannot be concluded for AOD. This can be explained by the nature of the bias we created and the nature of how the fairness metrics calculate their scores. We have only introduced sample bias into the test sets, which lets the ratio of defaults between gender differ, as is clearly measured by SPD. However, we do not change any of the labels. As AOD checks for equality between genders in the correctness of the predictions, this metric is not directly affected by our introduced sample bias. Therefore, in the next section we will also present the results of the sensitivity analyses when manipulating the training sets. In this case we have been able to change both the level of sample bias as well as the level of label bias. The second notion we want to make is, as the level of sample bias increases for the test groups, the percentage of SPD reduction for all bias mitigation techniques decreases. The bias mitigation techniques are tuned on the level of bias in the training set and apparently do not adopt well to other amounts of

bias. Thirdly, comparing the sensitivity of the pre-processing techniques in combination with the four different classifiers, we do not observe a clear distinction between reweighing and massaging. Moreover, we see that overall logistic regression and XGBoost exhibit the least deviation to different data sets. On the other hand, the effect of pre-processing techniques in combination with decision tree is the most volatile. Lastly, adversarial debiasing is less sensitive to deviations in data than the pre-processing techniques with decision tree as classifier. However, the effect is clearly deviating more than the pre-processing techniques in combination with the other three classifiers.

Concluding, when taking into account both the sensitivity and ability in bias reduction of the different methods and classifiers, massaging in combination with the classifier logistic regression is performing best with respect to fairness. However, as a fairness-performance trade-off exists, other methods reach a higher predictive performance at the expense of fairness. It depends on the (business) objective which method is more desirable. If one wants to reach a higher predictive performance and accepts a slightly lower level of fairness, the best option will be to use XGBoost in combination with reweighing or massaging - depending on whether one is more interested in equalizing SPD or AOD - as this method is still quite stable with respect to deviations in the data.

6.2 Differentiation in training sets Taiwan Default data

For generating different training set groups, we have performed manipulations on two axes: sampling and relabelling. The former one is similar to the manipulations done on the test sets and the corresponding training sets contain artificial "sample bias". The latter one is used to introduce artificial "label bias" into the training data. In this case, specific subgroups of the data have been relabelled. Thus, observations with *default* are relabelled to *no default* and vice versa. In Table 16 and Table 17, the resulting training groups together with their characteristics regarding the unfair bias and class imbalance are reported for the case of sample bias and label bias, respectively. The several training groups are used to train the different models and the non-manipulated test data is used to measure their sensitivity.

	Subgroup sampled relatively more	Difference in default ratio male and female	Class imbalance
Training group A	*Male with no default*	-2.7%	19.9%
Training group B	*Female with default*	-0.2%	24.2%
Training group C	*Original distribution*	3.3%	22.0%
Training group D	*Male with default*	6.9%	23.7%
Training group E	*Female with no default*	10.2%	17.1%

Note: Reported numbers are the averages over five folds

Table 16: Generated training sets which contain sample bias and their characteristics

	Part of subgroup relabelled	Difference in default ratio male and female	Class imbalance
Training group F	*Female with no default*	-1.9%	25.5%
Training group G	*Male with default*	0.7%	21.0%
Training group H	*No relabelling*	3.6%	22.1%
Training group I	*Female with default*	6.1%	20.6%
Training group J	*Male with no default*	8.9%	24.2%

Note: Reported numbers are the averages over five folds

Table 17: Generated training sets which contain label bias and their characteristics

Again, we have considered the pre-processing techniques reweighing and massaging and the in-processing technique adversarial debiasing as bias mitigating methods. The sensitivity is measured by considering the changes in SPD and AOD with and without applying a bias mitigation technique. The predictive performance of the models is represented by recall and accuracy scores, however we mainly focus on the recall scores. As the results can be interpreted comparably to the previous section on differentiation in test data, we will only report the averages over the training groups corresponding to the artificial sample bias (train groups A-E) and artificial label bias (train groups F-J) as well as the boxplots. We want to refer the interested reader to Appendix, Section 9.2 for the detailed results per training group and the predictive performance scores.

Firstly, as the level of bias contained in the train groups increases (both for sample bias and label bias) the SPD and AOD scores without bias mitigation increase. This means that as a model is trained on more biased data, the predictions will also contain more bias, as expected. Secondly, the effect of applying a bias mitigation technique increases, on average, with the increased amount of bias contained in the training data, as can be observed from Figures 6-10. Similarly reasoning as to the case of manipulating test data, we note that the bias mitigation methods are tuned on the level of bias in the training set and reduce this amount of bias in the predictions, regardless of the data distribution of the test set. Moreover, we want to note that cases in which the training data exhibits a lower level of bias or even bias towards the non-protected group, the bias mitigation algorithms are unable to detect this and the predictions contain a higher level of bias with bias mitigation than without, which is of course not desirable.

		No reweighing				Reweighing				Effect			
		LR	RF	DT	XGB	LR	RF	DT	XGB	LR	RF	DT	XGB
SPD	*Train A-E*	3.2%	2.8%	3.5%	2.9%	1.4%	2.3%	2.3%	1.9%	39%	3%	16%	17%
	Train F-J	3.2%	2.6%	2.9%	3.3%	1.4%	2.3%	2.3%	2.1%	37%	9%	15%	26%
AOD	*Train A-E*	3.1%	1.9%	2.7%	2.0%	0.2%	1.1%	1.2%	0.5%	67%	43%	58%	46%
	Train F-J	3.0%	1.7%	2.0%	2.6%	0.1%	1.1%	1.0%	0.9%	59%	10%	11%	5%

Note: Reported numbers are the averages over the train groups

Table 18: Average SPD and AOD scores without and with reweighing and the average effect of reweighing for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB) over the different training groups

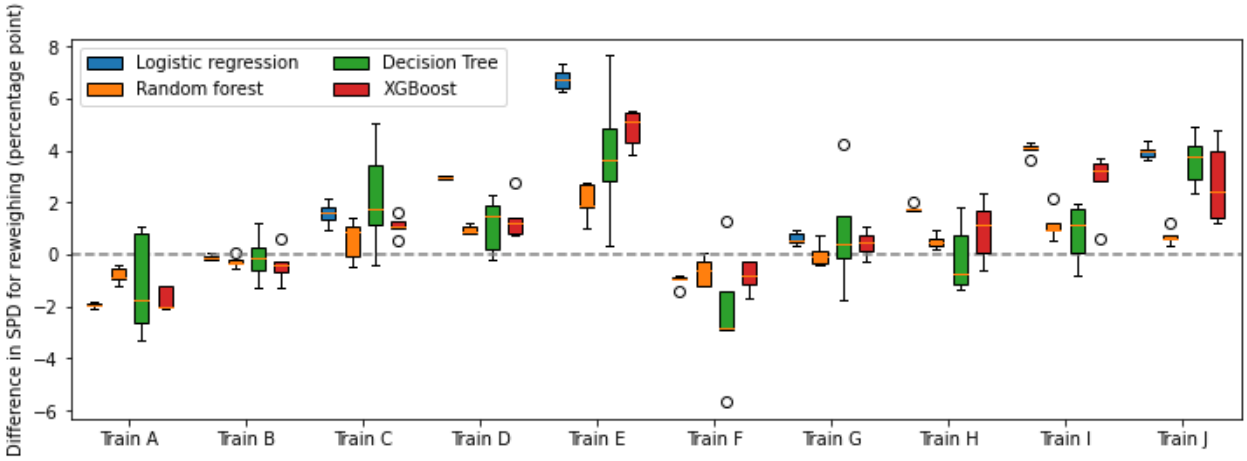

Figure 6: Effect of reweighing on SPD for different classifiers and training groups

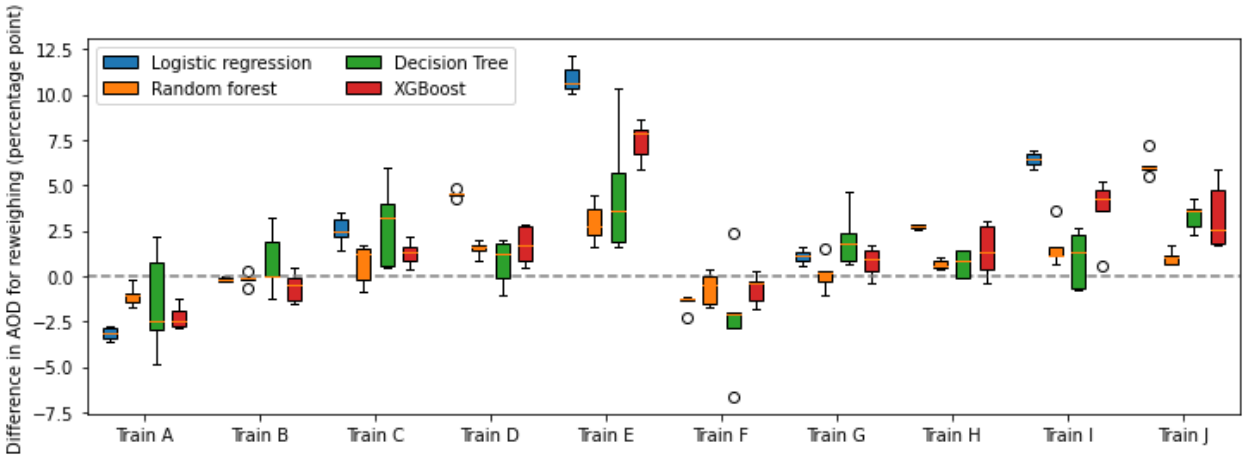

Figure 7: Effect of reweighing on AOD for different classifiers and training groups

Inspecting Table 18, we observe that reweighing is able to, on average, reach the same level of fairness for the training groups with sample bias as well as with label bias. The fairness is highest, for both SPD and AOD, when reweighing is combined with logistic regression. However, if one is not concerned with reaching the highest fairness level, but also needs to take predictive performance into account, using XGBoost as classifier will be a better option. This combination will reach a higher predictive performance, while also obtaining a better fairness level than random forest and decision tree and moreover is quite stable to deviations within train groups, when judging Figures 6 and 7.

		No massaging				Massaging				Effect			
		LR	RF	DT	XGB	LR	RF	DT	XGB	LR	RF	DT	XGB
SPD	*Train A-E*	3.2%	2.8%	3.5%	2.9%	1.1%	1.0%	0.8%	0.5%	38%	27%	26%	40%
	Train F-J	3.2%	2.6%	2.9%	3.3%	1.0%	1.0%	0.5%	0.6%	42%	44%	45%	62%
AOD	*Train A-E*	3.1%	1.9%	2.7%	2.0%	-0.2%	-0.5%	-0.3%	-1.2%	67%	45%	47%	42%
	Train F-J	3.0%	1.7%	2.0%	2.6%	-0.5%	-0.4%	-0.9%	-1.1%	61%	11%	37%	0%

Note: Reported numbers are the averages over the train groups

Table 19: Average SPD and AOD scores without and with massaging and the average effect of massaging for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB) over the different training groups

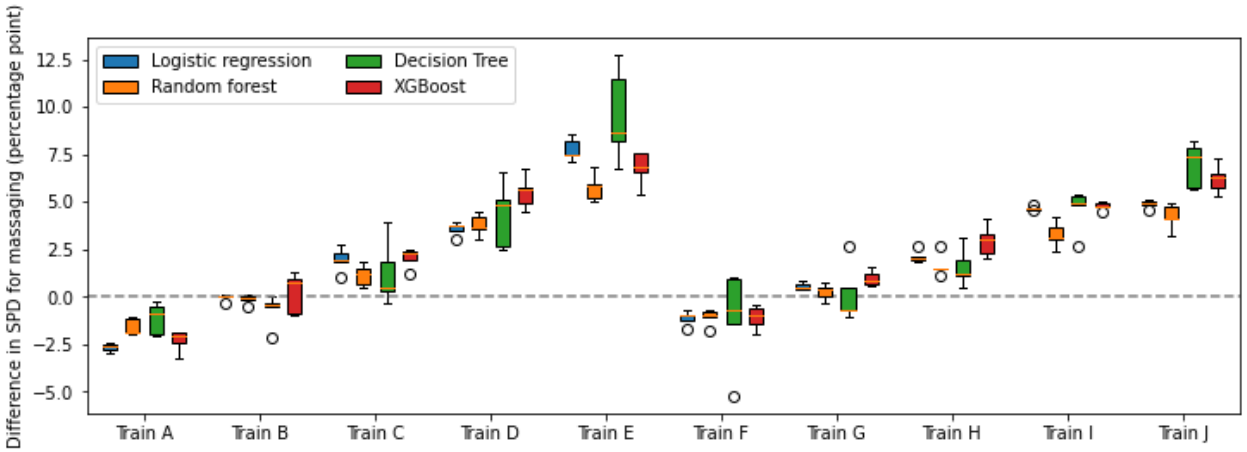

Figure 8: Effect of massaging on SPD for different classifiers and training groups

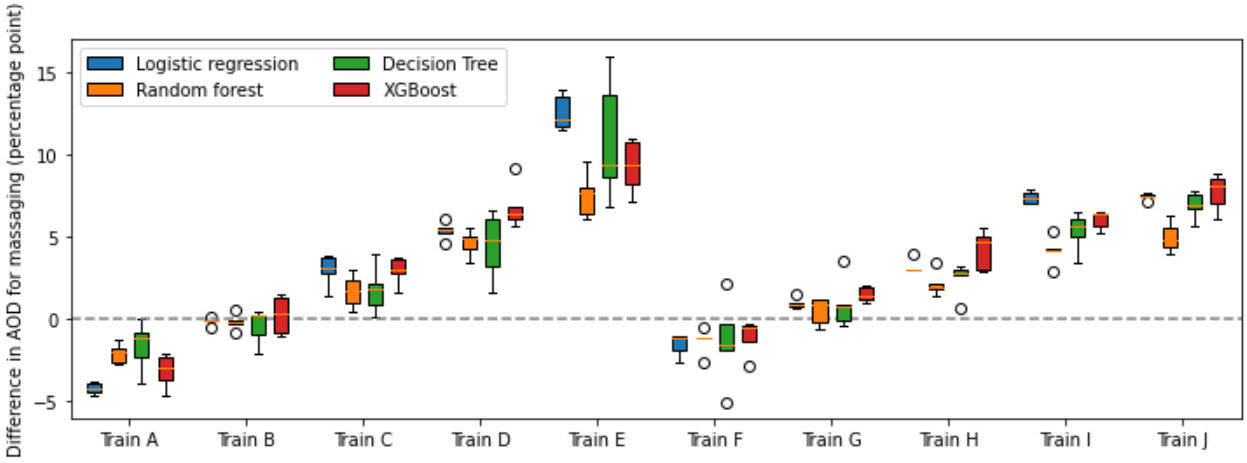

Figure 9: Effect of massaging on AOD for different classifiers and training groups

Looking at Table 19, we note that compared to reweighing, massaging has a bigger effect on the level of fairness. However, when judging the AOD scores, massaging is overcompensating for the

bias present in the data and thereby causing the predictions to be biased towards females. This is observed for the training groups that exhibit label bias and, to a lesser extent, for the training groups that exhibit sample bias. Judging Table 20, the same observation for overcompensating, when basing on AOD, can be made for adversarial debiasing. The performance of adversarial debiasing, on the axes of bias mitigation and predictive performance, is comparable to massaging with logistic regression. However, the latter one is less sensitive to data deviations when we compare Figures 8 and 9 with Figure 10.

		No adversarial debiasing	Adversarial debiasing	Effect
SPD	*Train A-E*	2.8%	1.1%	60%
	Train F-J	3.1%	0.9%	19%
AOD	*Train A-E*	2.3%	-0.2%	24%
	Train F-J	2.7%	-0.6%	19%

Note: Reported numbers are the averages over the train groups

Table 20: Average SPD and AOD scores without and with adversarial debiasing and the average effect of adversarial debiasing over the different training groups

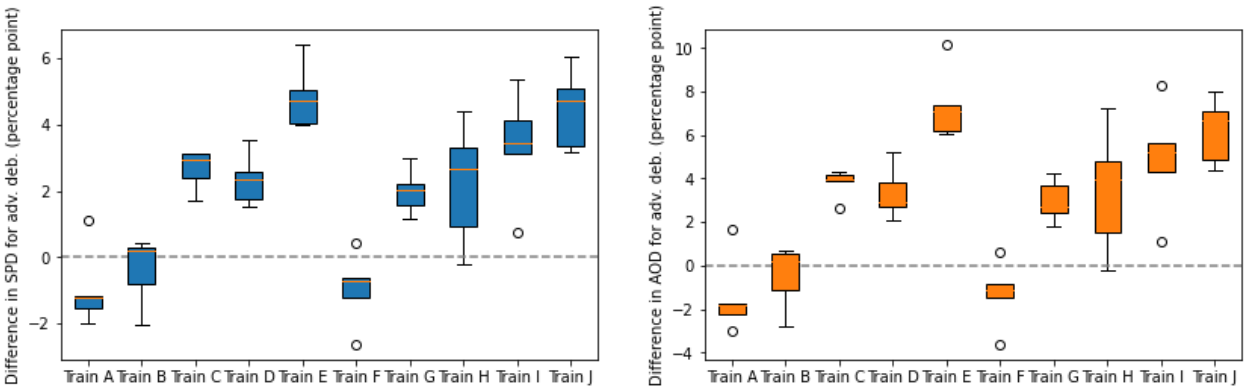

Figure 10: Effect of adversarial debiasing on SPD (left) and AOD (right) for different training groups

6.3 Census data

For validating the results obtained in the previous sections, we make use of the Dutch census data of 2001 and 2011. We have trained the models on the less discriminatory data of 2011 and tested these models on both the more discriminatory data of 2001 as well as the 2011 data itself. We measure the effect of the bias mitigation techniques reweighing, massaging, and adversarial debiasing by means of changes in the fairness metrics SPD and AOD. These calculated effects can then be interpreted as the percentage of unfairness eliminated from the predictions by applying bias mitigation methods. Furthermore, we report the performance metrics recall and accuracy. We want to note that, compared to the results of the Taiwan Default data, the scores without bias mitigation are primarily negative instead of positive as the desired outcome of the dependent variable is now 1, which was 0 in the case of the Taiwan Default data. However, the results can be interpreted in the same way. As the standard threshold of the probabilities generated by the model, which is used to

determine whether an observation should be given a 0 or 1, did not result in good predictive results, we based the threshold on the highest F1-score, which is the harmonic mean between precision and recall.

In Table 21, the SPD and AOD scores of the different classifiers without reweighing and with reweighing are presented along with the effect of reweighing, calculated as the percentage of unfairness eliminated. Furthermore, we reported the recall and accuracy scores in Table 22. Firstly, we note that, based on the recall scores, the predictive performance of the models for the 2001 test data is, on average, lower than for the 2011 test set. As expected, although the 2001 and 2011 have the exact same features and corresponding coding, things have changed over ten years and the model trained on 2011 data is less suitable for making predictions regarding the 2001 data. Furthermore, we observe that reweighing removes quite a part of the unfairness from the 2011 predictions, when looking at the SPD scores of the different classifiers. However, judging from the AOD scores the unfairness has increased, which could be due to reweighing aiming to achieve statistical parity. The average odds difference for the 2011 predictions without any bias mitigation was already quite low and, while trying to achieve statistical parity using reweighing, this caused the average odds difference to increase. However, we do not see this ability of the reweighing algorithm to remove unfairness extrapolate to the predictions of the 2001 data. Looking at Figure 11, which contains boxplots of the differences in SPD and AOD between no reweighing and reweighing, we observe that the volatility for the 2011 predictions is quite low. Nonetheless, for the 2001 predictions, which are based on a model trained with different data, the volatility increases. We note that decision tree has the highest deviance, as was also observed during the sensitivity analyses of the Taiwan Default data. Also, XGBoost is deviating quite a bit.

		No reweighing				Reweighing				Effect			
		LR	RF	DT	XGB	LR	RF	DT	XGB	LR	RF	DT	XGB
SPD	2001	-10.6%	-0.4%	-0.1%	-4.7%	-6.6%	-0.4%	-0.1%	-4.8%	37%	-12%*	-11%*	-2%*
	2011	-2.6%	-5.8%	-5.5%	-4.7%	-2.2%	-3.4%	-2.9%	-2.1%	15%	42%	47%	55%
AOD	2001	0.1%	10.5%	11.6%	7.0%	5.4%	10.5%	10.5%	7.1%	-98%*	0%	9%	-2%*
	2011	3.6%	1.2%	1.6%	2.6%	4.1%	4.1%	4.7%	4.2%	-11%*	-70%*	-66%*	-38%*

Note: Reported numbers are the averages over five folds

Table 21: SPD and AOD scores without and with reweighing and the effect of reweighing for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB) and for the 2001 and 2011 test sets

		No reweighing				Reweighing			
		LR	RF	DT	XGB	LR	RF	DT	XGB
Recall	2001	0.646	0.570	0.429	0.590	0.597	0.568	0.423	0.594
	2011	0.603	0.735	0.737	0.743	0.599	0.727	0.729	0.593
Accuracy	2001	0.819	0.815	0.817	0.835	0.865	0.816	0.799	0.841
	2011	0.842	0.854	0.854	0.861	0.843	0.856	0.855	0.845

Note: Reported numbers are the averages over five folds

Table 22: Recall and accuracy scores without and with reweighing for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB) and for the 2001 and 2011 test sets

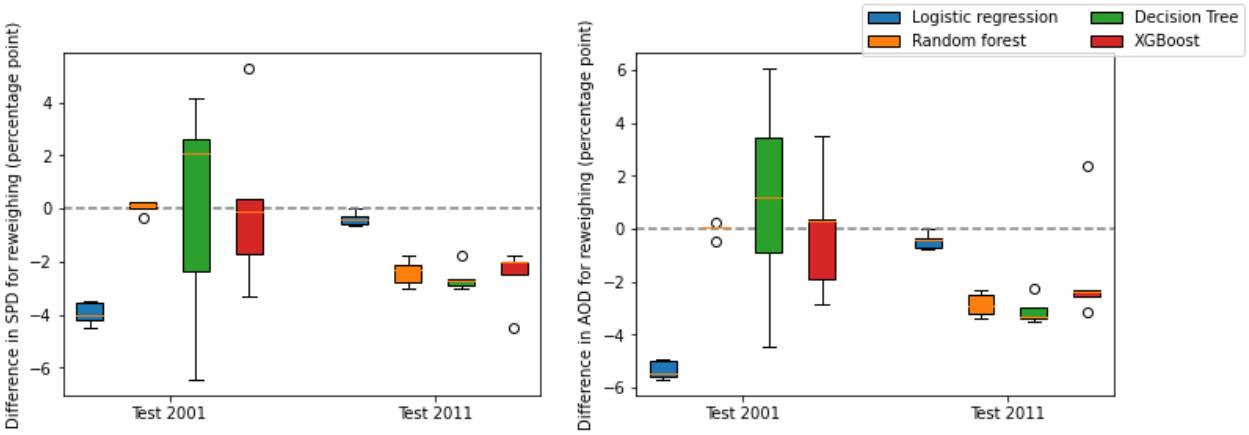

Figure 11: Effect of reweighing on SPD (left) and AOD (right)

We have displayed the SPD and AOD scores without and with massaging in combination with the four different classifiers in Table 23, along with the calculated effect of massaging. Moreover, the recall and accuracy scores have been reported in Table 24. Again, we observe substantially lower recall scores for the 2001 predictions. Compared to reweighing, massaging has a smaller effect of reducing SPD in the predictions of 2011. On the other hand, this also causes the increase in AOD, on average, to be relatively smaller. Inspecting Figure 12, the volatility of the different classifiers is very low for the 2011 predictions and at the same time is quite high for the 2001 predictions, except for logistic regression. For the AOD scores, random forest has the highest volatility. However, for the SPD scores we see that, again, decision tree has the highest deviance, and, to a lesser extent, random forest and XGBoost exhibit some deviance.

		No massaging				Massaging				Effect			
		LR	RF	DT	XGB	LR	RF	DT	XGB	LR	RF	DT	XGB
SPD	2001	-10.2%	-0.8%	-2.1%	-4.6%	-5.5%	-4.8%	-4.9%	-5.4%	47%	-83%*	-58%*	-14%*
	2011	-3.3%	-6.0%	-5.0%	-4.8%	-2.2%	-4.9%	-4.4%	-3.2%	33%	17%	13%	32%
AOD	2001	0.4%	4.5%	6.8%	7.3%	6.8%	6.9%	6.8%	6.7%	-95%*	-35%*	0%	8%
	2011	4.5%	1.0%	2.2%	2.5%	5.8%	2.2%	3.0%	4.3%	-22%*	-57%*	-27%*	-43%*

Note: Reported numbers are the averages over five folds

Table 23: SPD and AOD scores without and with massaging and the effect of massaging for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB) and for the 2001 and 2011 test sets

		No massaging				Massaging			
		LR	RF	DT	XGB	LR	RF	DT	XGB
Recall	2001	0.639	0.308	0.381	0.563	0.575	0.566	0.571	0.615
	2011	0.754	0.732	0.731	0.741	0.751	0.737	0.738	0.742
Accuracy	2001	0.821	0.807	0.805	0.841	0.867	0.853	0.830	0.855
	2011	0.858	0.853	0.852	0.861	0.858	0.856	0.853	0.861

Note: Reported numbers are the averages over five folds

Table 24: Recall and accuracy scores without and with massaging for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB) and for the 2001 and 2011 test sets

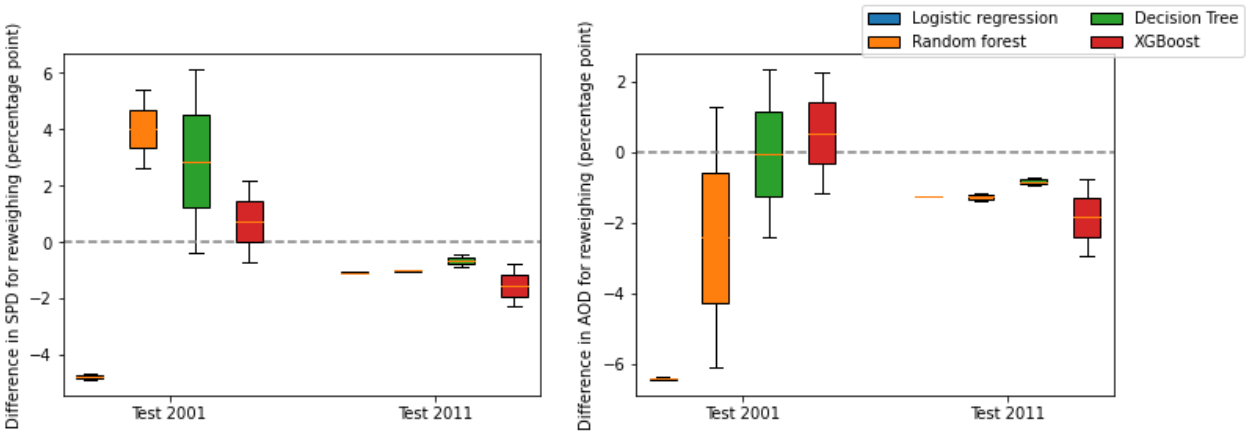

Figure 12: Effect of massaging on SPD (left) and AOD (right)

In Table 25, we have reported the SPD, AOD, recall and accuracy scores of the predictions for the 2001 and 2011 test data with and without adversarial debiasing along with the calculated effect adversarial debiasing has for SPD and AOD. Furthermore, in Figure 13 we have displayed boxplots of the differences in SPD and AOD for the two different test sets. For the 2011 predictions and to a lesser extent for the 2001 predictions, adversarial debiasing is useful in removing the SPD. However, at the same time, causes an increase in the AOD, as was also observed for massaging and reweighing. Judging Figure 13, the volatility of adversarial debiasing is again very small for the 2011 predictions and very large for the 2001 predictions. It seems that the adversarial debiasing algorithm is quite sensitive to the differences in data distributions of the 2001 and 2011 test data.

	No adversarial debiasing				Adversarial debiasing				Effect	
	SPD	AOD	Recall	Accuracy	SPD	AOD	Recall	Accuracy	SPD	AOD
2001	-10.3%	1.6%	0.779	0.597	-6.4%	3.0%	0.671	0.695	38%	-48%*
2011	-3.8%	3.6%	0.742	0.732	-1.5%	6.4%	0.863	0.863	60%	-44%*

Note: Reported numbers are the averages over five folds

Table 25: SPD, AOD, recall and accuracy scores without and with adversarial debiasing and the effect of adversarial debiasing on SPD and AOD for the 2001 and 2011 test sets

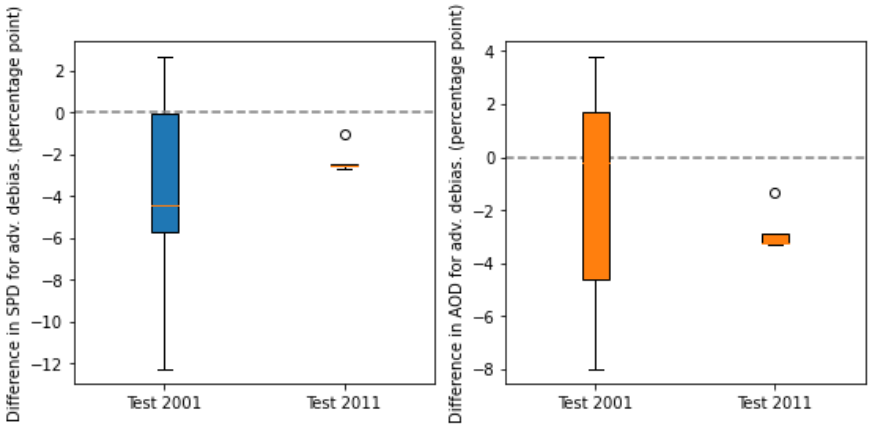

Figure 13: Effect of adversarial debiasing on SPD (left) and AOD (right)

7 Conclusion

This research examined the sensitivity of several fair classification methods to deviations in the data distributions. We investigated this sensitivity in two ways: (1) we performed sensitivity analyses by creating different levels of artificial bias in the training and test sets of the Taiwan Default data and (2) we examined the performance of the models for Dutch census data of two different periods in time with associated different amounts of bias. In this paper, we focused on the two pre-processing methods reweighing and massaging, and the in-processing method adversarial debiasing. Furthermore, we have combined the pre-processing methods with the well-known classifiers logistic regression, random forest, decision tree and XGBoost. By observing how the fairness metrics change when applying a bias mitigation, we were able to calculate its effect on the fairness of the predictions and, thereby, how this deviates for different data distributions. The fairness metrics we considered in this research are statistical parity difference (SPD), which is focused at the possible gap between demographic groups and average odds difference (AOD), which is focused at the probability of correct predictions for different demographic groups.

The modifications to the Taiwan Default data, in order to create artificial bias, have been done in two axes: resampling and relabelling. We only performed resampling to the test data. For the training data, both resampling and relabelling methods have been applied. These manipulations resulted in different test sets, for which we made predictions based on models trained on the original data, as well as different training sets, which resulted in different trained models which we tested using the same original test data. We first consider the case of differentiation in test set data. Overall, no clear distinction has been observed in the sensitivity between reweighing and massaging. Of the four classifiers, logistic regression and XGBoost show least deviation for different data sets in their fairness results. On the contrary, decision tree has been shown to be very volatile. These observations can be explained by the design of the classifiers and are in line with the findings of Kamiran & Calders (2012). Comparing adversarial debiasing to the two pre-processing techniques, we observed that this method is less sensitive than massaging or reweighing combined with decision tree, but more sensitive than when combined with the other classifiers. If one opts for achieving the highest fairness, massaging in combination with logistic regression is performing best. However if one is also concerned with the predictive performance and as a performance-fairness trade-off exists, XGBoost with reweighing or massaging could be the better option, depending on the busi-

ness objective. These model combinations are still performing well in terms of fairness and have a higher predictive performance, while at the same time they are also quite stable. This observation is in line with Kozodoi et al. (2021), who found that achieving perfect fairness is very costly, but reducing the bias to a reasonable extent is possible, while still achieving a relatively high predictive performance.

For the case of differentiation in training data sets, we were able to create both artificial sample bias as well as label bias. Overall, the highest fairness is achieved by combining reweighing with the logistic regression classifier. Again, if predictive performance is also an important (business) consideration, one could better pick the classifier XGBoost as this one obtains a higher predictive performance, still achieves a rather high level of fairness and has small deviations in its results. When judging the AOD scores, we note that massaging is overcompensating the level of bias contained in the data, which is observed a bit more for the training sets with label bias than the training sets with sample bias. This could the explained by the finding of Hinnefeld et al. (2018), who found that fairness metrics have a differing sensitivity to different causal origins of the bias. Moreover, this is also the case for adversarial debiasing. Furthermore, in cases where the training data exhibits a lower level of bias than the test data or even bias towards the non-protected group, the bias mitigation algorithms are unable to detect this and the unfairness in the predictions increases instead of decreasing, which is undesirable. Moreover, for both the case of differentiation in test sets and training sets, we see that the bias mitigation algorithms are tuned to a certain level of bias in the training set and are, on average, able to reduce this amount of fairness in the predictions, regardless of the data distribution of the test set.

Using the data of Dutch censuses in 2001 and 2011, we were able to see how our chosen models performed for deviations in the data distributions without the need to manipulate the data ourselves. We trained the different model combinations on the 2011 data and tested them on both the 2001 data, which contains more bias, and the 2011 data itself. The data sets only differ in their observations, the used features and coding of those features are exactly the same for both years. Firstly, we observed that the predictive performance for the 2001 data decreased compared to the 2011 case. Apparently, the 2011 data is not very suitable to make predictions for the 2001 case. Moreover, we want to note that reweighing and massaging are designed to decrease the SPD. However, while aiming for statistical parity, this caused the AOD to increase for this use case, which is undesirable. This is in line with the findings of Kleinberg et al. (2016), which state that it is almost never possible to satisfy different fairness criteria simultaneously and that a trade-off between the fairness metrics is present. Comparing the pre-processing techniques, reweighing performs better when considering SPD as the fairness metric, while massaging performs better when considering AOD. As massaging has, on average, a smaller volatility in its results, this technique seems to be performing better than reweighing. Then, basing our judgement on the ability to reduce unfairness, the associated predictive performance, and its volatility, we consider XGBoost as the most appropriate classifier when combined with massaging. Lastly, adversarial debiasing is very sensitive to differences in data distributions which causes a high volatility in its results.

Combining these obtained results, we are able to answer our sub-questions and thereby the research question. The 'fair' classification methods are tuned on the level of bias contained in the training set and do not adopt well to other amounts of bias. Hence, when changes in the data distributions occur, the 'fair' models are unable to maintain their achieved level of fairness. Furthermore, we observe that the predictive performance suffers from the deviations. In terms of sensitivity, we note that no model combination is the winner on all facets of interest. However, XGBoost in com-

bination with massaging or reweighing seems to be the best option, of course depending on the use case and business objective. Concluding, we have shown that 'fair' classification is quite sensitive to deviations in the data and is, on average, unable to deal with different levels of bias between train en test data. These findings can be used in further research, on which we will elaborate in Section 8.

8 Discussion

In this section, we discuss several limitations of our research and propose suggestions for future research. Firstly, the scope of this research was bounded to binary classification problems with only one protected attribute. However, the total fairness literature also includes multi-class classification or regression and several protected attributes. The set-up of this research could be extended to those areas as well. Secondly, we chose to only consider two pre-processing bias mitigation techniques and one in-processing technique. Of course, the same analyses could also be performed for other bias mitigation techniques, in order to get a broader and completer picture of the sensitivity of the different methods. Although we made our statements as general as possible, our conclusions are based on only two use cases and it might be possible that these results do not (completely) extrapolate to other use cases.

We want to note that the analyses had rather long run times, usually several hours per model. This is due to the fact that we performed extensive hyperparameter tuning, applied cross-validation and needed to test several training or test sets for each model combination we considered. Therefore, we decided not to apply any feature selection in order to keep the run times within reasonable ranges. However, applying feature selection would have approximated the real-life situation even better. Furthermore, during hyperparameter tuning, we decided for the pre-processing techniques to optimize the classifiers for recall and at the same time, the first network of adversarial debiasing is aimed at maximizing accuracy. Hence, this causes the results to not be optimized for fairness. It is possible that the sensitivity results for the fairness metrics would change if the models are optimized for these fairness metrics. However, we do feel that our approach is more general and therefore also more applicable to the real-life situation than for such models, which are inherently fair.

The bias mitigating algorithms we considered are all aimed at achieving statistical parity, and as was also observed in the results, it can thereby happen that the average odds difference increases after applying a bias mitigation algorithm. Depending on what is the desired way of defining fairness, it is important to know that these algorithms may not help in achieving the fairness one aims at. Therefore, one should not focus on solely one fairness metric, but use several criteria to obtain a completer picture of the model's fairness. Moreover, in order to scope this research we only considered group fairness criteria. These criteria ensure that different groups are treated equally. However, this could cause the model to suffer from individual bias. Therefore, it could be possible that, when also considering individual fairness metrics, we would have obtained different outcomes with respect to the best model combinations.

In addition, we did not write the code of the bias mitigation algorithms ourselves, but implemented already existing packages from specific libraries. Overall these functions worked properly. However, in exceptional cases it seemed that they broke down and are not (yet) able to cope with this, for which we had to adjust the code. Therefore, it could be very beneficial for the integration

of these bias mitigation algorithms into the development of machine learning models to improve the code of these packages. Furthermore, the current algorithms are quite static in the sense that they are tuned to the level of bias contained in the training sets and do not adopt well to changing amounts of bias in the test sets. It could be beneficial in cases where data distributions change to have less static algorithms, which are in some way able to detect the changed distribution of the data and adjust their predictions accordingly. We would like to highlight these suggestions as directions for further research.

References

Bellamy, R., Dey, K., Hind, M., Hoffman, S., Houde, S., Kannan, K., ... Zhang, Y. (2019). AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM Journal of Research and Development*, *63*(4/5), 4-1.

Berk, R., Heidari, H., Jabbari, S., Joseph, M., Kearns, M., Morgenstern, J., ... Roth, A. (2017). A convex framework for fair regression. *arXiv preprint arXiv:1706.02409*.

Bostrom, N., & Yudkowsky, E. (2014). The ethics of Artificial Intelligence. *The Cambridge handbook of Artificial Intelligence*, *1*, 316–334.

Crupi, R., Del Gamba, G., Greco, G., Naseer, A., Regoli, D., & Gonzalez, B. (2021). BeFair: Addressing fairness in the banking sector. *arXiv preprint arXiv:2102.02137*.

Dastin, J. (2018). *Amazon scraps secret ai recruiting tool that showed bias against women.* Thomson Reuters. Retrieved from `https://www.reuters.com/article/us-amazon-com-jobs-automation-insight-idUSKCN1MK08G`

Fogliato, R., Chouldechova, A., & G'Sell, M. (2020). Fairness evaluation in presence of biased noisy labels. In *International Conference on Artificial Intelligence and Statistics* (pp. 2325–2336).

Grari, V., Ruf, B., Lamprier, S., & Detyniecki, M. (2020). Achieving fairness with decision trees: An adversarial approach. *Data Science and Engineering*, *5*, 99–110.

Hagendorff, T. (2020). The ethics of AI ethics: An evaluation of guidelines. *Minds and Machines*, *30*(1), 99–120.

Hinnefeld, J. H., Cooman, P., Mammo, N., & Deese, R. (2018). Evaluating fairness metrics in the presence of dataset bias. *arXiv preprint arXiv:1809.09245*.

Holstein, K., Wortman Vaughan, J., Daumé III, H., Dudik, M., & Wallach, H. (2019). Improving fairness in machine learning systems: What do industry practitioners need? In *Proceedings of the 2019 CHI conference on human factors in computing systems* (pp. 1–16).

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

Kamiran, F., Calders, T., & Pechenizkiy, M. (2010). Discrimination aware decision tree learning. In *2010 IEEE International Conference on Data Mining* (pp. 869–874).

Kleinberg, J., Mullainathan, S., & Raghavan, M. (2016). Inherent trade-offs in the fair determination of risk scores. *arXiv preprint arXiv:1609.05807*.

Kozodoi, N., Jacob, J., & Lessmann, S. (2021). Fairness in credit scoring: Assessment, implementation and profit implications. *arXiv preprint arXiv:2103.01907*.

Lipton, Z. C., Chouldechova, A., & McAuley, J. (2018). Does mitigating ML's impact disparity require treatment disparity? In *Proceedings of the 32nd International Conference on Neural Information Processing Systems* (pp. 8136–8146).

Maity, S., Xue, S., Yurochkin, M., & Sun, Y. (2021). Statistical inference for individual fairness. *arXiv preprint arXiv:2103.16714*.

McKinsey. (2020). *Global survey: The state of AI in 2020* (Tech. Rep.). Retrieved from https://www.mckinsey.com/business-functions/mckinsey-analytics/our-insights/global-survey-the-state-of-ai-in-2020

Menon, A. K., & Williamson, R. C. (2018). The cost of fairness in binary classification. In *Conference on Fairness, Accountability and Transparency* (pp. 107–118).

Minnesota Population Center. (2020). Integrated public use microdata series, International: Version 7.3 [dataset]. *Minneapolis, MN: IPUMS*. doi: 10.18128/D020.V7.3

Mokander, J., & Floridi, L. (2021, 02). Ethics-based auditing to develop trustworthy AI. *Minds and Machines*. doi: 10.1007/s11023-021-09557-8

Pessach, D., & Shmueli, E. (2020). Algorithmic fairness. *arXiv preprint arXiv:2001.09784*.

Ravichandran, S., Khurana, D., Venkatesh, B., & Edakunni, N. U. (2020). FairXGBoost: Fairness-aware classification in XGBoost. *arXiv preprint arXiv:2009.01442*.

Rukat, T., Lange, D., Schelter, S., & Biessmann, F. (2020). Towards automated data quality management for machine learning. In *ML Ops workshop at the Conference on ML and Systems (MLSys).*

Saleiro, P., Kuester, B., Hinkson, L., London, J., Stevens, A., Anisfeld, A., ... Ghani, R. (2018). Aequitas: A bias and fairness audit toolkit. *arXiv preprint arXiv:1811.05577*.

Verma, S., & Rubin, J. (2018). Fairness definitions explained. In *2018 IEEE/ACM International Workshop on Software Fairness (Fairware)* (pp. 1–7).

Xu, R., Cui, P., Kuang, K., Li, B., Zhou, L., Shen, Z., & Cui, W. (2020). Algorithmic decision making with conditional fairness. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp. 2125–2135).

Yeh, I., & Lien, C. (2009). The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert Systems with Applications*, *36*(2), 2473–2480.

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

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

Zhang, D., Mishra, S., Brynjolfsson, E., Etchemendy, J., Ganguli, D., Grosz, B., ... Perrault, R. (2021). *The AI Index 2021 Annual Report.*

Zhang, L., Wu, Y., & Wu, X. (2016). A causal framework for discovering and removing direct and indirect discrimination. *arXiv preprint arXiv:1611.07509*.

Zhang, Y., & Zhou, L. (2019). Fairness assessment for Artificial Intelligence in financial industry. *arXiv preprint arXiv:1912.07211*.

9 Appendix

9.1 Hyperparameter tuning

Tuned hyperparameter	*Grid values*
Logistic regression	
penalty	[none, l2]
C	[0.01, 0.2, 1, 10, 100]
Random forest	
n estimators	[100, 144, 188, 233, 277, 322, 366, 411, 455, 500]
max depth	[10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110]
min samples split	[2, 5, 10]
min samples leaf	[1, 2, 4]
XGBoost	
learning rate	[0.05, 0.10, 0.15, 0.20, 0.25, 0.30]
max depth	[3, 4, 5, 6, 8, 10, 12, 15]
min child weight	[1, 3, 5, 7]
gamma	[0, 0.1, 0.2, 0.3, 0.4]
colsample bytree	[0.3, 0.4, 0.5, 0.7]
Decision tree	
max depth	[10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110]
max features	[1, 3, 5, 7, 10]
min samples leaf	[1, 2, 4]
criterion	[gini, entropy]

Table 26: Grid values for random search on hyperparameters of classifiers

9.2 Differentiation in training sets Taiwan Default data

9.2.1 Reweighing

	No reweighing				Reweighing				Effect			
	LR	RF	DT	XGB	LR	RF	DT	XGB	LR	RF	DT	XGB
Train A	-1.0%	0.5%	0.4%	-0.7%	1.0%	1.3%	1.6%	1.1%	-3%*	-61%*	-74%*	-37%*
Train B	1.7%	2.2%	1.4%	1.4%	1.8%	2.5%	1.5%	1.8%	-6%*	-11%*	-7%*	-22%*
Train C	3.2%	2.7%	2.5%	3.0%	1.6%	2.1%	0.3%	1.9%	50%	21%	89%	36%
Train D	4.7%	3.9%	4.2%	5.0%	1.7%	2.9%	3.1%	3.6%	63%	25%	27%	28%
Train E	7.5%	4.8%	8.7%	5.9%	0.7%	2.7%	4.8%	1.0%	90%	43%	44%	83%
Average	3.2%	2.8%	3.5%	2.9%	1.4%	2.3%	2.3%	1.9%	39%	3%	16%	17%
Train F	0.4%	1.9%	-0.4%	1.2%	1.4%	2.6%	1.9%	2.1%	-73%*	-25%*	-79%*	-40%*
Train G	0.9%	1.1%	0.9%	1.1%	0.3%	1.1%	0.1%	0.7%	64%	1%	93%	39%
Train H	3.3%	2.6%	2.2%	3.3%	1.5%	2.1%	2.4%	2.4%	54%	20%	-6%*	28%
Train I	5.3%	3.7%	4.3%	5.3%	1.3%	2.6%	3.5%	2.5%	76%	31%	19%	53%
Train J	6.2%	3.8%	7.5%	5.6%	2.2%	3.0%	3.9%	2.9%	64%	19%	48%	49%
Average	3.2%	2.6%	2.9%	3.3%	1.4%	2.3%	2.3%	2.1%	37%	9%	15%	26%

Note: Reported numbers of Train A - Train E and Train F - Train J are the averages over five folds

Table 27: SPD scores without and with reweighing and the effect for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB)

	No reweighing				Reweighing				Effect			
	LR	RF	DT	XGB	LR	RF	DT	XGB	LR	RF	DT	XGB
Train A	-3.5%	-1.3%	-1.0%	-2.9%	-0.3%	-0.2%	0.4%	-0.7%	91%	82%	59%	77%
Train B	0.5%	1.0%	0.7%	-0.2%	0.7%	1.2%	0.0%	0.4%	-24%*	-15%*	95%	-39%*
Train C	2.9%	1.7%	2.0%	2.1%	0.4%	1.0%	-0.8%	0.8%	87%	41%	61%	61%
Train D	5.1%	3.2%	3.4%	4.5%	0.5%	1.7%	2.6%	2.8%	90%	47%	22%	38%
Train E	10.3%	4.9%	8.5%	6.8%	-0.5%	1.9%	3.9%	-0.6%	95%	61%	54%	91%
Average	3.1%	1.9%	2.7%	2.0%	0.2%	1.1%	1.2%	0.5%	67%	43%	58%	46%
Train F	-1.5%	0.7%	-1.3%	0.0%	0.0%	1.4%	1.0%	0.7%	100%	-49%*	24%	-98%*
Train G	-0.4%	-0.4%	-0.1%	-0.5%	-1.5%	-0.5%	-2.1%	-1.3%	-74%*	-24%*	-97%*	-63%*
Train H	3.0%	1.5%	1.6%	2.6%	0.3%	0.8%	0.9%	1.2%	90%	45%	45%	54%
Train I	6.6%	3.5%	3.9%	5.5%	0.2%	1.9%	3.0%	1.9%	97%	46%	24%	66%
Train J	7.4%	3.1%	5.6%	5.2%	1.3%	2.0%	2.3%	1.8%	83%	33%	58%	64%
Average	3.0%	1.7%	2.0%	2.6%	0.1%	1.1%	1.0%	0.9%	59%	10%	11%	5%

Note: Reported numbers of Train A - Train E and Train F - Train J are the averages over five folds

Table 28: AOD scores without and with reweighing and the effect for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB)

	Recall					Recall			
	LR	RF	DT	XGB		LR	RF	DT	XGB
Train A	0.288	0.335	0.376	0.329	*Train F*	0.353	0.387	0.416	0.384
Train B	0.357	0.395	0.428	0.386	*Train G*	0.286	0.347	0.379	0.332
Train C	0.331	0.365	0.373	0.355	*Train H*	0.333	0.368	0.404	0.366
Train D	0.351	0.388	0.422	0.384	*Train I*	0.277	0.336	0.379	0.332
Train E	0.232	0.306	0.368	0.295	*Train J*	0.347	0.390	0.420	0.379
Average	0.312	0.358	0.394	0.350	*Average*	0.319	0.365	0.400	0.358

Note: Reported numbers of Train A - Train E and Train F - Train J are the averages over five folds

Table 29: Recall scores after reweighing for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB)

	Accuracy					Accuracy			
	LR	RF	DT	XGB		LR	RF	DT	XGB
Test A	0.815	0.818	0.742	0.817	*Test F*	0.819	0.817	0.712	0.812
Test B	0.819	0.815	0.721	0.806	*Test G*	0.814	0.816	0.730	0.817
Test C	0.819	0.818	0.733	0.819	*Test H*	0.819	0.817	0.732	0.810
Test D	0.819	0.817	0.720	0.809	*Test I*	0.814	0.815	0.726	0.816
Test E	0.810	0.814	0.739	0.816	*Test J*	0.819	0.815	0.711	0.811
Average	0.817	0.817	0.731	0.813	*Average*	0.817	0.816	0.722	0.813

Note: Reported numbers of Train A - Train E and Train F - Train J are the averages over five folds

Table 30: Accuracy scores after reweighing for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB)

9.2.2 Massaging

	No massaging				Massaging				Effect			
	LR	RF	DT	XGB	LR	RF	DT	XGB	LR	RF	DT	XGB
Train A	-1.0%	0.5%	0.4%	-0.7%	1.7%	2.1%	1.6%	1.6%	-44%*	-76%*	-74%*	-60%*
Train B	1.7%	2.2%	1.4%	1.4%	1.7%	2.3%	2.1%	1.2%	-1%*	-6%*	-33%*	15%
Train C	3.2%	2.7%	2.5%	3.0%	1.3%	1.6%	1.2%	0.9%	61%	41%	50%	69%
Train D	4.7%	3.9%	4.2%	5.0%	1.1%	0.1%	-0.1%	-0.5%	76%	97%	98%	89%
Train E	7.5%	4.8%	8.7%	5.9%	-0.3%	-1.0%	-0.8%	-0.9%	96%	79%	90%	85%
Average	3.2%	2.8%	3.5%	2.9%	1.1%	1.0%	0.8%	0.5%	38%	27%	26%	40%
Train F	0.4%	1.9%	-0.4%	1.2%	1.5%	3.0%	0.7%	2.3%	-75%*	-36%*	-42%*	-47%*
Train G	0.9%	1.1%	0.9%	1.1%	0.4%	0.9%	0.8%	0.1%	56%	21%	14%	87%
Train H	3.3%	2.6%	2.2%	3.3%	1.2%	0.9%	0.7%	0.4%	64%	63%	69%	88%
Train I	5.3%	3.7%	4.3%	5.3%	0.7%	0.5%	-0.3%	0.5%	87%	87%	93%	90%
Train J	6.2%	3.8%	7.5%	5.6%	1.3%	-0.5%	0.5%	-0.6%	79%	88%	93%	90%
Average	3.2%	2.6%	2.9%	3.3%	1.0%	1.0%	0.5%	0.6%	42%	44%	45%	62%

Note: Reported numbers of Train A - Train E and Train F - Train J are the averages over five folds

Table 31: SPD scores without and with massaging and the effect for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB)

	No massaging				Massaging				Effect			
	LR	RF	DT	XGB	LR	RF	DT	XGB	LR	RF	DT	XGB
Train A	-3.5%	-1.3%	-1.0%	-2.9%	0.7%	0.8%	0.6%	0.2%	79%	40%	43%	94%
Train B	0.5%	1.0%	0.7%	-0.2%	0.6%	1.2%	1.1%	-0.5%	-16%*	-13%*	-35%*	-51%*
Train C	2.9%	1.7%	2.0%	2.1%	0.0%	0.0%	0.2%	-0.9%	99%	98%	89%	56%
Train D	5.1%	3.2%	3.4%	4.5%	-0.3%	-1.5%	-1.1%	-2.3%	94%	55%	69%	49%
Train E	10.3%	4.9%	8.5%	6.8%	-2.2%	-2.7%	-2.4%	-2.5%	78%	44%	71%	63%
Average	3.1%	1.9%	2.7%	2.0%	-0.2%	-0.5%	-0.3%	-1.2%	67%	45%	47%	42%
Train F	-1.5%	0.7%	-1.3%	0.0%	0.1%	2.0%	0.0%	1.1%	95%	-65%*	97%	-99%*
Train G	-0.4%	-0.4%	-0.1%	-0.5%	-1.4%	-0.9%	-1.0%	-2.0%	-72%*	-58%*	-94%*	-76%*
Train H	3.0%	1.5%	1.6%	2.6%	-0.2%	-0.7%	-0.9%	-1.7%	94%	57%	42%	35%
Train I	6.6%	3.5%	3.9%	5.5%	-0.8%	-0.6%	-1.4%	-0.6%	88%	82%	64%	90%
Train J	7.4%	3.1%	5.6%	5.2%	-0.1%	-1.9%	-1.3%	-2.5%	99%	37%	77%	51%
Average	3.0%	1.7%	2.0%	2.6%	-0.5%	-0.4%	-0.9%	-1.1%	61%	11%	37%	0%

Note: Reported numbers of Train A - Train E and Train F - Train J are the averages over five folds

Table 32: AOD scores without and with massaging and the effect for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB)

	Recall					Recall			
	LR	RF	DT	XGB		LR	RF	DT	XGB
Train A	0.309	0.342	0.357	0.344	*Train F*	0.360	0.402	0.414	0.393
Train B	0.358	0.393	0.433	0.384	*Train G*	0.292	0.354	0.379	0.339
Train C	0.333	0.368	0.357	0.364	*Train H*	0.336	0.378	0.403	0.381
Train D	0.356	0.396	0.422	0.393	*Train I*	0.299	0.341	0.379	0.340
Train E	0.259	0.298	0.349	0.302	*Train J*	0.355	0.399	0.419	0.397
Average	0.323	0.359	0.384	0.357	*Average*	0.328	0.375	0.399	0.370

Note: Reported numbers of Train A - Train E and Train F - Train J are the averages over five folds

Table 33: Recall scores after massaging for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB)

	Accuracy					Accuracy			
	LR	RF	DT	XGB		LR	RF	DT	XGB
Test A	0.816	0.816	0.746	0.816	*Test F*	0.819	0.816	0.718	0.809
Test B	0.820	0.815	0.723	0.809	*Test G*	0.815	0.816	0.727	0.815
Test C	0.817	0.816	0.736	0.812	*Test H*	0.818	0.814	0.727	0.806
Test D	0.817	0.808	0.722	0.802	*Test I*	0.815	0.809	0.727	0.807
Test E	0.810	0.802	0.744	0.801	*Test J*	0.816	0.805	0.714	0.798
Average	0.816	0.811	0.734	0.808	*Average*	0.817	0.812	0.723	0.807

Note: Reported numbers of Train A - Train E and Train F - Train J are the averages over five folds

Table 34: Accuracy scores after massaging for logistic regression (LR), random forest (RF), decision tree (DT) and XGBoost (XGB)

9.2.3 Adversarial debiasing

	No adversarial debiasing		Adversarial debiasing		Effect		Predictive performance	
	SPD	AOD	SPD	AOD	SPD	AOD	Recall	Accuracy
Train A	-1.0%	-3.2%	0.0%	-1.8%	100%	45%	0.322	0.816
Train B	1.3%	0.1%	1.7%	0.6%	-23%*	-79%*	0.374	0.816
Train C	2.6%	1.8%	0.0%	-2.0%	99%	-8%*	0.344	0.817
Train D	4.4%	4.3%	2.0%	1.0%	54%	77%	0.368	0.817
Train E	6.7%	8.6%	1.9%	1.2%	72%	86%	0.274	0.814
Average	2.8%	2.3%	1.1%	-0.2%	60%	24%	0.336	0.816
Train F	1.7%	0.6%	2.6%	1.9%	-37%*	-69%*	0.377	0.818
Train G	-0.4%	-2.5%	-2.4%	-5.4%	-84%*	-54%*	0.333	0.813
Train H	3.0%	2.6%	0.8%	-0.9%	75%	67%	0.340	0.816
Train I	6.1%	7.3%	2.8%	2.4%	55%	67%	0.320	0.816
Train J	5.1%	5.3%	0.7%	-0.9%	87%	83%	0.367	0.817
Average	3.1%	2.7%	0.9%	-0.6%	19%	19%	0.347	0.816

Note: Reported numbers of Train A - Train E and Train F - Train J are the averages over five folds

Table 35: SPD, AOD scores without and with adversarial debiasing, the effect of adversarial debiasing and the recall and accuracy scores after adversarial debiasing