DTU Compute

MSc Eng Thesis
January 2022

Fairness-Oriented Interpretability of Predictive Algorithms

Caroline Amalie Fuglsang-Damgaard (s164175)
Elisabeth Zinck (s164204)

Supervisor: Aasa Feragen

Abstract

Background: Predictive algorithms can have large impacts on people's lives as they are used for making increasingly high stakes decisions in society. It is therefore of paramount importance that algorithms produce fair predictions for individuals regardless of gender, sex, race, and other sensitive characteristics. Consequently, fair machine learning is an active and quite new field of research with many definitions of fairness. However, it is often mathematically impossible to satisfy all these fairness definitions simultaneously. Optimizing for one criterion can exacerbate unfairness with respect to another, and there is hence a need for performing comprehensive fairness analysis taking more than one criterion into account.
Objective: The aim of the project is to design and construct a toolkit for analyzing fairness of predictive algorithms enabling model developers to perform broad and nuanced assessments of fairness considering several fairness criteria.
Methods: The toolkit, named *BiasBalancer*, is implemented in Python and is aimed at developers of binary predictive algorithms. BiasBalancer is designed in three levels and introduces a new summary fairness metric, a comprehensive overview of a wide variety of existing fairness criteria, and methods for further in-depth fairness analysis. BiasBalancer is showcased on various smaller examples, including canonical fairness datasets, and through a medical case study of the CheXpert dataset.
Results: The analyses of canonical fairness datasets support results from previous fairness analyses while showing the ease of use of BiasBalancer. In the fairness analysis of the CheXpert dataset, the fairness criteria yield conflicting results from the analysis considering patients' race. One criterion suggests discrimination of Whites while another suggests discrimination of Blacks.
Conclusion: Our results show how different types of unfairness interact and highlight the need for more comprehensive fairness analyses taking several criteria into account. We show how the toolkit BiasBalancer facilitates such fairness analyses.

List of Abbreviations

AUC:	Area under the curve
CI:	Confidence interval
CNN:	Convolutional neural network
FDR:	False discovery rate
FNR:	False negative rate
FOR:	False omission rate
FPR:	False positive rate
FFNN:	Feed forward neural network
NPV:	Negative predictive value
PPV:	Predicted positive value
ROC:	Receiver operating characteristic
TNR:	True negative rate
TPR:	True positive rate
WMQ:	Weighted misclassification quotient
WMR:	Weighted misclassification rate

Acknowledgements

We would like to extend our sincere thanks to our supervisor Aasa Feragen for her guidance, insightful suggestions, and never-failing enthusiasm about the project. Special thanks to Eike Petersen for reading through the thesis and giving valuable suggestions. We would also like to thank the HPC support at DTU and Søren Hartmann for patient technical assistance. Finally, we are very grateful for the unconditional support from our boyfriends, family, and friends. It has been invaluable to have you all by our side while writing this thesis.

Contents

Chapter 1

Introduction

1.1 General Introduction

Advances in machine learning technologies during the last decades have enabled model developers to build increasingly sophisticated models and algorithms. These models increasingly become part of our daily lives: We use Google Translate when we're missing a word in a foreign language, facial recognition to unlock our smartphones, voice recognition to interact with intelligent home speakers, and we see recommendations tailored to our profile when shopping online. Predictive algorithms are also increasingly influencing more high-stake decisions affecting our lives, such as hiring [Bogen and Rieke, 2018], and who should be granted bail after committing a crime [Angwin et al., 2016]. Within the medical field, artificial intelligence can now predict diseases such as skin cancer and pneumonia from medical images on par or even better than experts within the field [Esteva et al., 2017, Rajpurkar et al., 2017]. Such predictive algorithms are expected to increasingly aid health professionals when making decisions concerning our health [Bohr and Memarzadeh, 2020].

When important decisions are made about our lives, we want to be treated fairly and equally no matter our sensitive attributes such as race, gender, sex, or sexual orientation. The use of predictive models could improve transparency and fairness in decision-making because decisions made using models can be monitored and checked to a greater extent than human-made decisions. However, predictive models heavily depend on real-world datasets, and these datasets are rarely representative of the population and commonly contain the biases present in our society.

During recent years many examples of discriminative algorithms have been detected, and some examples have even found their way into the public debate. In [Angwin et al., 2016], ProPublica described how the COMPAS algorithm, used for predicting the risk of criminals recidivating, was biased against Black offenders. It has been shown that commercially available facial recognition systems had lower accuracies for women and non-White faces compared to White men [Klare et al., 2012]. Twitter admitted that their image cropping system had a preference for showing White individuals and cropping individuals of other races out of the image [Chowdhury, 2021]. And an algorithm, widely used at U.S. hospitals to choose which patients are given extra care, was revealed to require that Black patients had to be sicker to be assigned the same score as

White patients [Obermeyer et al., 2019].

It is evident that the problems with biased algorithms will increase as we become more reliant on algorithms – if decisive action is not taken to mitigate the bias. In the past years, fairness in machine learning has grown to a large field of research, and many steps, are taken simultaneously to ensure or encourage fair machine learning. In April 2021, the European Commission proposed a legal framework aiming to regulate artificial intelligence to "minimize risks and discriminatory outcomes" [Directorate-General for Communications Networks Content and Technology, 2021]. The ongoing public debate and courses in fairness and ethics in machine learning at universities create hope that the new generation of data and computer scientists are more aware of these challenges. However, the current literature on fairness in machine learning is overwhelming and fast-changing while legislation is lagging behind, and best practice is still under discussion. Hence, tools are needed to aid future model developers in assessing models with respect to fairness.

1.2 Related Work

Many toolkits exist for the purpose of making fairness in machine learning and AI more accessible and applied by the people involved in deploying predictive models for decision-making. One of these toolkits is Aequitas [Saleiro et al., 2019]. Aequitas can be accessed through a web tool, python, or the command line. The purpose of the toolkit is to look at disparities between subgroups, and it uses the predictions, true labels, and sensitive group attributes to audit the predictions from a model. Aequitas stands out with great usability, and it is not overwhelming to navigate. However, the disparity analyses are fairly simple, and the fairness analyses are only based on the binary predictions and not the scores. Another toolkit is Fairlearn, which also enables the user to get information about absolute differences or disparities across subgroups [Bird et al., 2020]. Fairlearn also includes some mitigation algorithms, which can be used by data scientists to alleviate potential unfairness found in their models. Thus, it has more features than Aequitas, but it is also less user-friendly as it includes notebooks with usage examples that are less thorough and accessible than those showcasing Aequitas. The toolkit AI Fairness 360 (AIF360) is perhaps the most comprehensive open-source fairness toolkit currently existing. AIF360 was originally made by IBM, and it aims to provide a common framework for research within fairness metrics and mitigation techniques. It includes more than 70 metrics and ten mitigation algorithms, which can be overwhelming to navigate, but to aid users, they have created an interactive experience to get started [Bellamy et al., 2019]. The authors of [Johnson et al., 2020] have created Fairkit-learn, which combines AIF360 with the classic python machine learning package scikit-learn. Fairkit-learn focuses on incorporating model selection with considerations of fairness and accuracies, enabling the user to make informed choices about the trade-offs given chosen metrics. FairKit-learn includes a wrapper such that the toolkit can be extended to handle models not originating from scikit-learn. Since neural networks have become a standard modeling approach, it is easy to imagine a potential user's need to extend the toolkit such that it also considers, e.g., a PyTorch model. A common feature of the above-mentioned toolkits is that the user typically specifies a privileged group working as the reference to which the other groups are compared. In AIF360, this is described as a group which historically has been put at an advantage in a given situation, and this choice will therefore be task specific.

1.3 Thesis Aim and Motivation

What constitutes a fair prediction algorithm depends on the context, and it may often be the case that several different fairness criteria are relevant in a given setting. However, some of the most central fairness criteria in machine learning have been shown to exclude one another under only mild assumptions [Barocas et al., 2019, Pleiss et al., 2017]. This means that *the* fair model, which is fair with respect to all relevant criteria, often doesn't exist. Therefore, creating fair models inevitably involves a trade-off between different notions of fairness. A common practice is to choose only one of the relevant fairness criteria in the given scenario and ensure absolute fairness with respect to this criterion. However, optimizing with respect to one fairness criterion will often come at a price of increasing unfairness with respect to other criteria. We believe that a model developer is only able to make an informed choice about the fairness trade-off in a predictive model after being presented to and thinking thoroughly about the consequences of the predictions according to a range of fairness definitions. This motivates the aim of the project:

> The thesis aims to design and construct a toolkit for analyzing fairness of predictive algorithms. The toolkit should facilitate a comprehensive and nuanced fairness analysis, emphasizing a broad assessment of several criteria and providing an easily interpretable overview of fairness.

Moreover, this project showcases the use of the toolkit on a wide range of example datasets, including an in-depth fairness analysis of a predictive algorithm in a medical setting.

The thesis is structured in 7 chapters, and you are currently reading the first. In chapter 2 we introduce the essential observational fairness criteria along with a presentation of how they relate to one another. The chapter also includes a brief introduction to the predictive models used in the project. Chapter 3 describes the example datasets and predictive models. The fairness toolkit, BiasBalancer, is presented in chapter 4, which also includes examples of the use of the toolkit. The medical case study is found in chapter 5, where we use BiasBalancer to analyze unfairness with respect to race and sex in a model predicting the medical condition cardiomegaly from radiographs. Finally, the results and toolkit are discussed and concluded in chapter 6 and 7.

Chapter 2

Theory

In this chapter, we present the theory relevant to this thesis. First, section 2.1 presents the so-called confusion matrix and the rates calculated based on it, such as the false positive rate. Then follows a section (section 2.2) on how these rates can be used to define fairness criteria and how those fairness criteria interact. Finally, the predictive models used in this thesis are explained in section 2.3.

2.1 Confusion matrix

Many fairness criteria evaluate the fairness of an algorithm based on how observations are classified or predicted relative to their actual or true class. These properties of the classifier are neatly summarized in a *confusion matrix*. In this thesis, we will only consider binary classification problems, and table 2.1 shows the layout of a confusion matrix for a binary classification problem. True positives (TP) and false positives (FP) are the number of true observations that were predicted to be true and false respectively, while true negatives (TN) and false negatives (FN) are the number of false observations that were predicted to be false and true respectively. The two left-most and the two top cells contain the marginal values calculated as the sum of each row or column, respectively. The number of predicted positives and predicted negatives are denoted by PP and PN, respectively, while the number of actual positives and actual negatives are denoted by P and N. Since we use N to denote the number of actual negatives, we will, contrary to more common notation, use n to denote the number of observations in total. We will use the notation introduced in table 2.1 throughout this report.

		Predicted	
		Positive (PP)	Negative (PN)
Actual	Positive (P)	True positive (TP)	False negative (FN)
	Negative (N)	False positive (FP)	True negative (TN)

Table 2.1: Confusion matrix for a binary classification problem.

From the values in the confusion matrix the following rates to describe the performance of a classifier can be defined:

- True positive rate: $TPR = \frac{TP}{TP+FN} = \frac{TP}{P}$
- False negative rate: $FNR = \frac{FN}{TP+FN} = \frac{FN}{P}$
- True negative rate: $TNR = \frac{TN}{TN+FP} = \frac{TN}{N}$
- False positive rate: $FPR = \frac{FP}{TN+FP} = \frac{FP}{N}$

The true positive rate is sometimes called the *recall* or *sensitivity*, and the true negative rate is also called the *specificity*. The trade-off between the true positive rate and false positive rate is easily visualized in a so-called *ROC curve* (Receiver operating characteristic curve). A ROC curve plots the true positive rate against the false positive rate when varying the threshold used for the decision.

In the four rates defined above, the denominator is the number of observations belonging to either the actual positive or actual negative class. Similarly, four other measures can be defined where the denominator is the number of observations predicted to be positive or negative:

- Positive predictive value: $PPV = \frac{TP}{TP+FP} = \frac{TP}{PP}$
- False discovery rate: $FDR = \frac{FP}{TP+FP} = \frac{FP}{PP}$
- Negative predictive value: $NPV = \frac{TN}{TN+FN} = \frac{TN}{PN}$
- False omission rate: $FOR = \frac{FN}{TN+FN} = \frac{FN}{PN}$

The positive predictive value is also called the *precision*.

All eight rates defined above come in pairs that share the denominator, and these pairs all sum to 1:

$$TPR + FNR = TNR + FPR = PPV + FDR = NPV + FOR = 1 \tag{2.1}$$

This is easily shown mathematically by adding the two fractions of a pair and simplifying by letting the denominator and numerator cancel out. These relations mean that, as an example, if the true positive rate increases, the false negative rate must decrease by the same amount.

2.2 Observational Fairness Criteria

Observational fairness criteria are a group of fairness criteria based upon properties of the joint distribution $p(X, A, Y, \hat{Y})$, where Y is the binary target variable, $\hat{Y}$ is the prediction or classification, A are some sensitive/protected attributes, and X are the remaining non-sensitive features. Thus, given observations from the joint distribution, it is possible to verify if a given predictive algorithm is fair according to the observational fairness criteria without needing knowledge about the context in which the algorithm was used [Barocas et al., 2019]. We will briefly touch upon other types of algorithmic fairness in section 2.2.6.

All of the observational fairness criteria presented in this section can be expressed via statistical relationships between the three random variables Y, $\hat{Y}$ and A, i.e. X will not be considered. Many of the presented observational criteria have several names and different relaxations. Some of these relaxations are also considered fairness criteria themselves. [Barocas et al., 2019] organize the criteria into three categories *independence*, *separation* and *sufficiency*. This thesis will use this terminology while also presenting some of the relaxations within each category alongside them. The criteria are presented in a setting with only two sensitive groups, but they can easily be generalized to handle a larger number of groups. Finally, existing synonyms of each criterion will be presented, aiding the reader to navigate the different terminology within the field.

2.2.1 Summary Table of Criteria

Table 2.2 summarizes the observational fairness criteria such that they are easily compared. The table works as a reference to what the different criteria enforce. Each criterion is presented with one or more references for further reading and understanding, and they will be elaborated in the coming sections. The table also includes a collection of other names for the criteria, which we have encountered in our process. These names are not necessarily mentioned in the references supplied in the table, but they make it possible to use the table if the user is already familiar with another name than the one we have chosen to use.

We summarize what each criterion enforces by using the group-wise false negative rate (FNR), false positive rate (FPR), false discovery rate (FDR), false omission rate (FOR) and the acceptance rate $\frac{PP}{n}$. The rates have been presented in section 2.1. Let FNR be the false negative rate based on all predictions, the false negative rate for a subgroup a, based on their sensitive attributes, is then defined as FNR_a. Similar notation is used for all rates.

Fairness Criterion	Enforces	Reference	Other Names
Independence	$\frac{PP_a}{n_a} = \frac{PP_b}{n_b}$	[Barocas et al., 2019, p. 46]	Statistical Parity, Demographic Parity, Group Fairness
Sufficiency	$FOR_a = FOR_b$ and $FDR_a = FDR_b$	[Barocas et al., 2019, p. 50]	Predictive Rate Parity, Conditional Use Accuracy Parity
Predictive Parity$^\bullet$	$FDR_a = FDR_b$	[Verma and Rubin, 2018]	Outcome Test
Separation	$FPR_a = FPR_b$ and $FNR_a = FNR_b$	[Hardt et al., 2016], [Barocas et al., 2019, p. 50]	Equalized Odds, Positive Rate Parity, Disparate Mistreatment, Conditional Procedure Accuracy Equality,
FPR-balance*	$FPR_a = FPR_b$	[Verma and Rubin, 2018]	Predictive Equality
Equal Opportunity*	$FNR_a = FNR_b$	[Hardt et al., 2016], [Barocas et al., 2019, p. 50]	FNR-balance

Table 2.2: Summary of Observational Fairness Criteria. $\star$ indicates a relaxation of separation and $\bullet$ indicates a relaxation of sufficiency. The subscript indicates subgroups based on protected attributes. Abbreviations: *PP*: Predicted positives, *FPR*: False positive rate, *FNR*: False negative rate, *FDR*: False discovery rate, *FOR*: False omission rate, n: number of observations.

2.2.2 Independence

Alternative names: *Demographic parity, equal acceptance rate, statistical parity, group fairness, benchmarking.*

The first observational criterion, independence, requires the predicted probability to be independent of the sensitive attributes, i.e., $A \perp \hat{Y}$. This means that the predicted probability of belonging to the positive class must be the same for all sensitive groups A or, equivalently, that the share of predicted positives is equal for all groups [Barocas et al., 2019]. Defining $P_a(\hat{Y} = c) := P(\hat{Y} = c \mid A = a)$, independence can be expressed as

$$P_a(\hat{Y} = 1) = P_b(\hat{Y} = 1). \tag{2.2}$$

For a binary classifier satisfying independence, the probability of belonging to the negative class will also be equal across groups because $P(\hat{Y} = 1) + P(\hat{Y} = 0) = 1$. It is worth noting that a perfect predictor, which always predicts $\hat{Y} = Y$, does not fulfill independence if Y and A are correlated. Also, independence does not impose any constraints on *which* observations from each sensitive group should be predicted to be in the positive class. Consequently, a classifier can be deemed fair in terms of independence even if the accuracy for one group is much lower than for another [Hardt et al., 2016].

2.2.3 Separation

Alternative names: *Equalized odds, disparate mistreatment, conditional procedure accuracy equality*

The second observational criterion, *separation*, allows for correlation between the sensitive attributes and the predictions, but only if the correlation is justified by the target variable [Hardt et al., 2016]. Separation requires the classification and the sensitive attributes to be independent conditioned on the target variable:

$$A \perp \hat{Y} \mid Y. \tag{2.3}$$

Written in terms of probabilities the classifier must satisfy

$$P_a\left(\hat{Y} = j \mid Y = i\right) = P_b\left(\hat{Y} = j \mid Y = i\right), \quad i \in \{0,1\}, j \in \{0,1\}, j \neq i. \tag{2.4}$$

This enforces the false negative rate and false positive rate to be equal across sensitive subgroups for each rate. Moreover, because the rates come in pairs (eq. (2.1)), it also yields equality of the true positive rate and true negative rate across subgroups.

Contrary to the independence criterion, a perfect classifier will always satisfy separation. Given the scores from a classifier, it is possible to visualize how difficult it would be to satisfy separation by plotting a ROC curve separately for each class. The classifier satisfies separation if the threshold is chosen to be where the ROC curves intersect because then true positive rate (TPR) and false positive rate (FPR) will be equal for all groups. However, the ROC curves for the sensitive groups are not guaranteed to intersect. In that case, separation can be achieved by using randomized thresholds [Hardt et al., 2016].

There exist two relaxations of separation, which are used as fairness criteria themselves. These are *false positive error rate balance* and *equal opportunity*.

False Positive Error Rate Balance

Alternative names: Predictive equality

As the name suggests, the false positive error rate balance (FPR-balance) only requires a balance in the false positive rate (or, equivalently, true negative rate) for sensitive subgroups [Verma and Rubin, 2018]. In terms of probability, it implies that

$$P_a(\hat{Y} = 1 \mid Y = 0) = P_b(\hat{Y} = 1 \mid Y = 0), \tag{2.5}$$

which means that it should be equally probable for observations truly belonging to the negative class to be misclassified as positive regardless of the state of the sensitive attributes A.

Equal opportunity

Alternative names: False Negative Error Rate Balance

Analogous to the false positive error rate balance, equal opportunity (or false negative error rate balance) only requires a balance in the false negative rate (or equivalently true positive rate) for groups with different values in their sensitive attributes [Hardt et al., 2016]. In terms of probability, it implies that

$$P_a(\hat{Y} = 0 \mid Y = 1) = P_b(\hat{Y} = 0 \mid Y = 1), \tag{2.6}$$

which means that it should be equally probable for observations truly belonging to the positive class to be misclassified as negative regardless of the state of the sensitive attributes A.

2.2.4 Sufficiency

Alternative names: *Predictive rate parity, conditional use accuracy equality.*

The third and last observational fairness criterion to be considered is *sufficiency*. Sufficiency requires the target, Y, and sensitive attributes, A, to be independent conditioned on the classification, $\hat{Y}$ [Barocas et al., 2019]. This can in shorthand notation be written as

$$Y \perp A \mid \hat{Y}, \tag{2.7}$$

or equivalently in terms of probabilities:

$$P_a(Y = j \mid \hat{Y} = i) = P_b(Y = j \mid \hat{Y} = i), \quad i \in \{0, 1\}, j \in 0, 1, j \neq i. \tag{2.8}$$

This enforces the false discovery rate and the false omission rate must be equal across the sensitive groups, i.e., $FDR_a = FDR_b$ and $FOR_a = FOR_b$.

Sufficiency can be achieved through calibration by group. In fact, calibration by groups implies sufficiency [Barocas et al., 2019]. A classifier with scores $R \in [0, 1]$ is calibrated by group if it satisfies

$$P(Y = 1 | R = r, A = a) = r \quad \forall a \in A, \forall r \in [0, 1]. \tag{2.9}$$

Hence a classifier, which is calibrated by group, will for all groups have scores, where $100 \cdot r\%$ of observations with predicted score r is expected to belong to the positive class. Equation (2.9) is a stronger condition than equation (2.8) and classifiers satisfying calibration by group will therefore also satisfy sufficiency.

Predictive Parity

Alternative name: *Outcome test.*

Predictive parity is a relaxation of sufficiency, where equality across subgroups is not required for the false omission rate [Verma and Rubin, 2018]. Thus, predictive parity only requires

$$P_a(Y = 0 \mid \hat{Y} = 1) = P_b(Y = 0 \mid \hat{Y} = 1), \tag{2.10}$$

which means there must be balance in false discovery rates (or, equivalently, in the positive predictive value).

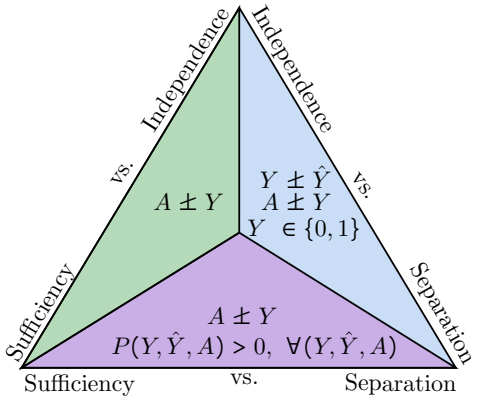


Figure 2.1: Each corner of the triangle represents one of three observational fairness criteria. The triangle connecting two corners contains the assumptions necessary to carry out a proof showing that both observational fairness criteria cannot be fulfilled at the same time.

2.2.5 Relationships Between Criteria

The observational fairness criteria presented so far are all characterized by requiring different properties of the joint distribution of the random variables A, Y and $\hat{Y}$. However, the properties required are non-trivial and adopting any of the criteria, (i.e., independence eq. (2.2), separation eq. (2.4), or sufficiency eq. (2.8)), as a hard constraint will constrain the joint distribution in a manner that only degenerate cases remain if one imposed an additional one as a hard constraint [Barocas et al., 2019, chapter 3].

For any pair of the three observational criteria, the triangle shown in Figure 2.1 depicts a scenario where only one of the two criteria can be satisfied. Each of the corners represents an observational criterion and the colored boxes each indicate a scenario where, given the assumptions inside, one cannot satisfy both of the criteria in the corners. Therefore, the algorithm owner is forced to choose a single corner, and thereby a single metric, if the fairness criterion is to be enforced using hard constraints.

This section will consider each combination of pairs of the criteria and prove why both cannot be fulfilled at the same time (proofs based on [Barocas et al., 2019]). We will moreover argue why the assumptions in each of the three scenarios are reasonable.

Independence and Sufficiency

The green scenario in figure 2.1 has only one assumption: The sensitive attributes, A, and the target variable, Y, are *not* independent. This assumption is often fulfilled as it is very common that the target variable is correlated with the sensitive attribute. In the case of a binary target, it simplifies to not having the same base rate of actual positives across groups. Given this assumption, it is not possible to obtain both independence and sufficiency.

Proof. If both sufficiency ($A \perp Y \mid \hat{Y}$) and independence ($A \perp \hat{Y}$) are satisfied, it implies that

$$A \perp \hat{Y} \;\wedge\; A \perp Y \mid \hat{Y} \quad \Rightarrow \quad A \perp Y, \tag{2.11}$$

and the contrapositive yields

$$A \not\perp Y \quad \Rightarrow \quad A \not\perp \hat{Y} \;\vee\; A \not\perp Y \mid \hat{Y}. \tag{2.12}$$

From the latter, it can be concluded that, given the assumption (the target variable is not independent of the sensitive attribute), it is not possible to satisfy both sufficiency and independence. □

Independence and Separation

The blue scenario in figure 2.1 includes three assumptions: $Y \not\perp \hat{Y}$, $A \not\perp Y$ and $Y \in \{0, 1\}$. In this thesis, we only consider binary classification, and we have previously argued why it is reasonable to assume that the target variable is correlated with the sensitive attributes. Moreover, it is reasonable to assume that any relevant classifier would yield a correlation between the predictions and the target.

Proof. To show that independence and separation cannot both be satisfied under the three assumptions, we need to show:

$$A \not\perp Y \;\wedge\; \hat{Y} \not\perp Y \quad \Rightarrow \quad A \not\perp Y \;\vee\; \hat{Y} \not\perp A \mid Y. \tag{2.13}$$

Taking the contrapositive of (2.13) gives the equivalent expression

$$A \perp Y \;\wedge\; \hat{Y} \perp A \mid Y \quad \Rightarrow \quad A \perp Y \;\vee\; \hat{Y} \perp Y, \tag{2.14}$$

which we choose to show instead. Using the law of total probability, we can write $P(\hat{Y} \mid A)$ as

$$P(\hat{Y} \mid A) = \sum_y P(\hat{Y} \mid A, Y = y) P(Y = y \mid A). \tag{2.15}$$

Applying the assumptions in the antecedent of eq. (2.14) to the equation above gives

$$P(\hat{Y}) = \sum_y P(\hat{Y} \mid Y = y) P(Y = y \mid A). \tag{2.16}$$

By using the law of total probability, we can also express $P(\hat{Y})$ in another way as

$$P(\hat{Y}) = \sum_y P(\hat{Y} \mid Y = y) P(Y = y). \tag{2.17}$$

Setting the two expressions for $P(\hat{Y})$ in equations (2.16) and (2.17) equal gives

$$\sum_y P(\hat{Y} \mid Y = y) P(Y = y \mid A) = \sum_y P(\hat{Y} \mid Y = y) P(Y = y). \tag{2.18}$$

When Y is binary, the above equation is only satisfied when either $P(Y \mid A) = P(Y)$, i.e. $Y \perp A$, or when $P(\hat{Y} \mid Y = 0) = P(\hat{Y} \mid Y = 1)$, i.e. $Y \perp \hat{Y}$. Hence, we have shown that when both independence and separation holds, then one of the assumptions does not hold (eq. (2.14)). This is the contrapositive of what we needed to prove (eq.(2.13)). □

Separation and Sufficiency

Finally, the purple triangle in figure 2.1 shows that when $A \not\perp\!\!\!\perp Y$ and the joint distribution $p(A, Y, \hat{Y})$ is always positive, sufficiency and separation cannot hold at the same time. This scenario is quite common because the assumption about the joint distribution is equivalent to having a strictly positive number of true positives, true negatives, false positives, and false negatives for all of the sensitive groups.

Proof. We would like to show that

$$A \not\perp\!\!\!\perp Y \quad \Rightarrow \quad A \not\perp\!\!\!\perp Y \mid \hat{Y} \ \vee \ A \not\perp\!\!\!\perp \hat{Y} \mid Y. \tag{2.19}$$

Using a property of conditional independence, which only holds when all events in the joint distribution have positive probability, [Wasserman, 2004, Theorem 17.2], we have that

$$A \perp\!\!\!\perp \hat{Y} \mid Y \ \wedge \ A \perp\!\!\!\perp Y \mid \hat{Y} \quad \Rightarrow \quad A \perp\!\!\!\perp (Y, \hat{Y}) \quad \Rightarrow \quad A \perp\!\!\!\perp \hat{Y} \ \wedge \ A \perp\!\!\!\perp Y. \tag{2.20}$$

Taking the contrapositive gives (2.19), which was what we needed to show. □

Quantifying Incompatibility of Criteria

The proofs given in the previous section show how the three criteria mutually exclude each other when enforced as hard constraints. It leads to the natural question of to what degree they can't co-exist. [Liu et al., 2019] show that unconstrained learning naturally leads to predictions satisfying the sufficiency criterion and that this results in larger violations of the independence and separation criteria.

The considered fairness criteria can be expressed in terms of expectations [Liu et al., 2019]. Consider the score function, $f(X)$, mapping the features in X to $[0, 1]$. Sufficiency (presented in section 2.2.4) requires equal probability of belonging to the true class given the predicted class across two sensitive groups a and b. In terms of the expectation, it can be written as:

$$\text{Sufficiency:} \qquad \mathbb{E}[Y \mid f(X)] = \mathbb{E}[Y \mid f(X), A], \tag{2.21}$$

i.e., the expectation of Y conditioned on the predicted score should not gain any additional information by also conditioning on the sensitive attribute.

The fairness criterion separation (presented in section 2.2.3) requires equal probability of being classified as a certain class given the actual class across two sensitive groups a and b. Using expectations, it can be reformulated as:

$$\text{Separation:} \qquad \mathbb{E}[f(X) \mid Y] = \mathbb{E}[f(X) \mid Y, A], \tag{2.22}$$

i.e., the expectation of the score function $f(X)$ conditioned by the target Y and sensitive attribute A, should not gain any information by the conditioning on A.

The fairness criterion independence (presented in section 2.2.2) requires equal probability of being classified as the positive class across two sensitive groups a and b. In terms of the expectation, independence can be written as:

$$\text{Independence:} \qquad \mathbb{E}[f(X)] = \mathbb{E}[f(X) \mid A], \tag{2.23}$$

i.e., conditioning on the sensitive attribute should not affect the expected value of the score function $f(X)$.

The expectation of the difference between the left hand side and right hand side in equations (2.21), (2.22), and (2.23) is used to quantify how much the three criteria are violated. These expectations define the sufficiency gap, separation gap, and independence gap denoted by $\mathbf{suf}_f(A)$, $\mathbf{sep}_f(A)$, and $\mathbf{ind}_f(A)$ respectively. Explicitly these can be formulated as

$$\mathbf{suf}_f(A) = \mathbb{E}[|\mathbb{E}[Y \mid f(X)] - \mathbb{E}[Y \mid f(X), A]|] \tag{2.24}$$
$$\mathbf{sep}_f(A) = \mathbb{E}_{Y,A}[|\mathbb{E}[f(X) \mid Y, A] - \mathbb{E}[f(X) \mid Y]|] \tag{2.25}$$
$$\mathbf{ind}_f(A) = \mathbb{E}_A[|\mathbb{E}[f(X) \mid A] - \mathbb{E}[f(X)]|]. \tag{2.26}$$

Using these formulations [Liu et al., 2019] present theorems specifying upper and lower bounds for the gaps under certain conditions. We will present the theorems and argue why their assumptions are fulfilled in our setting. We refer to [Liu et al., 2019] for a more rigorous explanation and proofs of theorems.

Given a loss function $\ell : [0,1] \times \{0,1\}$ and a score function f, the population risk $\mathcal{L}(f)$ is defined as

$$\mathcal{L}(f) = \mathbb{E}[\ell(f(X), Y)]. \tag{2.27}$$

In [Liu et al., 2019] they compare the score function $f(X)$ with the *calibrated Bayes score* f^B, which serves as a benchmark. The calibrated Bayes score is defined as

$$f^B(x, a) := \mathbb{E}[Y \mid X = x, A = a]. \tag{2.28}$$

The Bayes score satisfies sufficiency and is the perfect predictor if Y is deterministic given X and A. The population risk of the calibrated Bayes score is then defined as $\mathcal{L}^* := \mathcal{L}(f^B) = \mathbb{E}\left[\ell(f^B(X, A), Y)\right]$. The *excess risk* of the model, f, is defined as $\mathcal{L}(f) - \mathcal{L}^*$. The main theorem stated in [Liu et al., 2019] is

$$\mathbf{suf}_f(A) \leq 4\sqrt{\frac{\mathcal{L}(f) - \mathcal{L}^*}{\kappa}} \leq O\left(\sqrt{\mathcal{L}(f) - \mathcal{L}^*}\right), \tag{2.29}$$

which implies that the sufficiency gap is upper bounded by the excess risk. This means that a better model, i.e., when $\mathcal{L}(f)$ becomes more similar to $\mathcal{L}(f^B)$, will have a smaller sufficiency gap. The theorem relies on some assumptions about the loss function $\ell(f, Y)$, which also determines the value of κ. Furthermore, [Liu et al., 2019] proves that the logistic loss function

$$\ell(f, Y) = -(Y \log f + (1 - Y) \log(1 - f)) \tag{2.30}$$

satisfies the assumptions required for equation (2.29) to hold with $\kappa = 2/\log 2$. The logistic loss is also called the binary cross entropy loss and is often used to train binary classifiers in supervised

learning. The key takeaway is that the current framework of supervised learning using logistic loss favors the fairness criterion Sufficiency because better models will yield a smaller sufficiency gap.

In fact, [Liu et al., 2019] states that the current framework for supervised learning disfavors the other fairness criteria. They show that, under the same assumptions as in the previous theorem, the separation gap, $\mathbf{sep}_f$ is lower bounded by

$$\mathbf{sep}_{\hat{f}} \geq C_{f^B} \cdot Q_A - 2\sqrt{\frac{\mathcal{L}(f) - \mathcal{L}^*}{\kappa}}, \tag{2.31}$$

where C_{f^B} and Q_A are constants specific to the classification problem. Using $\overline{q} := P(Y = 1)$ and $q_A := P(Y = 1 \mid A)$, the constants can defined as

$$Q_A = \mathbb{E}[\mid \overline{q} - q_A \mid] \tag{2.32}$$

$$C_{f^B} = \frac{\mathbb{E}_{\mathcal{D}}[\mathrm{Var}[Y \mid X, A]]}{\mathrm{Var}[Y]}. \tag{2.33}$$

Thus, Q_A is the L_1-variation in base rates among sensitive groups, and C_{f^B} is the intrinsic noise level of the prediction problem. The intrinsic noise level is zero if the target Y is deterministic given X and A. Equation (2.31) therefore implies that a better model, i.e., with a smaller excess risk, increases the lower bound on the separation gap. Thus, in order to satisfy the separation criterion, it could be necessary to have a larger excess risk to compensate for differences in base rates across sensitive groups.

A similar result holds for the independence gap, $\mathbf{ind}_f(A)$, which is lower bounded by

$$\mathbf{ind}_{\hat{f}}(A) \geq Q_A - 2\sqrt{\frac{\mathcal{L}(f) - \mathcal{L}^*}{\kappa}}. \tag{2.34}$$

Thus, if the base rates of positive outcomes vary across sensitive groups, the excess risk needs to be larger in order to make the independence gap smaller and thereby come closer to satisfying the independence fairness criterion.

2.2.6 Types of Fairness in Machine Learning

The presented fairness criteria all belong to the category *Group Fairness* within fair machine learning research. These are also known as *Statistical Measures*, and they are based on parity between people belonging to different sensitive subgroups. The fairness criterion independence, which sometimes is referred to as simply "group fairness", only relies on the predicted target, while the remaining presented criteria depend on both the predicted and actual value of the target.

Another approach to fairness in machine learning is *Individual Fairness*, which has been created on the notion that people who are similar in task-relevant features should receive similar predictions. An example of this is the similarity-based fairness metric created by the authors of [Dwork et al., 2012]. In their article, they argue that the group fairness metric independence is insufficient by itself and that the two notions of fairness might be conflicting in cases where similar individuals are assigned different outcomes as a result of trying to satisfy the group fairness criterion. Even though the notion of similar people being treated similarly might sound appealing,

this method has a major drawback. Individual fairness heavily relies on a task-specific similarity metric, which expresses "the ground truth or at least, when the ground truth is unavailable, the metric may reflect the "best" available approximation as agreed upon by society" [Dwork et al., 2012, p. 1]. Such a metric is very hard to come by in practice, and in the setting of a fairness toolkit, it would require us to enable the user of our toolkit to choose such similarity measures. Thus, we have chosen not to consider individual fairness in this thesis.

Besides group fairness and individual fairness, there exist fairness definitions based on causal reasoning. Causal reasoning is a conceptual and technical framework for addressing questions about the effect of hypothetical actions or interventions [Barocas et al., 2019]. The fairness definitions are defined using a causal graph modelling the relationship between the features, sensitive attributes, and outcome. The causal fairness framework is a large area of research, and in order to limit the scope of this thesis, we have chosen not to work further with causality and fairness.

Lastly, we only consider purely algorithmic fairness in this thesis. We will therefore not dive into higher-level ethical fairness questions such as the question of whether algorithms should be used for decision-making at all.

2.3 Predictive Modelling

In this thesis, we will use three different predictive models to create example predictions to perform the fairness analyses on. The models are logistic regression, a feed-forward neural network, and DenseNet, which is a specific type of convolutional neural network. We will briefly describe each of the models in this section.

2.3.1 Logistic Regression

Logistic regression is a widely used classification technique based on a probabilistic model for classification. Logistic regression extends to multiple classes but in its simplest form it is often used for binary classification. This thesis only presents the binary logistic regression since only binary prediction problems are considered. Given a binary classification problem with a D-dimensional feature observation $x_i \in \mathbb{R}^D$ and a target $Y \in \{0, 1\}$, the logistic regression models the observation as Bernoulli distributed with probability p_i, where

$$p_i = P(Y = 1 \mid X = x_i). \tag{2.35}$$

Because p_i is constrained to be in the interval $[0, 1]$, it cannot be modelled directly with a linear model. Instead a link function F is chosen such that the probability p_i can be modelled by $p_i = F(\phi^T x_i)$. The link function used in logistic regression is the sigmoid function, which yields the expression

$$p_i = \sigma(\phi^T x_i) = \frac{1}{1 + \mathrm{e}^{-\phi^T x_i}}. \tag{2.36}$$

The choice of link function means that the logit of p_i can be modelled linearly with respect to x_i:

$$\sigma^{-1}(p_i) = \log \frac{p_i}{1 - p_i} = \phi^T x_i, \tag{2.37}$$

where the inverse of the sigmoid function is the logit function, i.e., $\sigma^{-1}(p_i) = \log \frac{p_i}{1-p_i}$.

The model parameters, ϕ, are estimated using maximum likelihood estimation. Let y_i be the value of the target for observation x_i then for n observations, the likelihood of the model can be written as

$$L(\phi) = \prod_{i=1}^{n} \sigma(\phi^T x_i)^{y_i} \cdot (1 - \sigma(\phi^T x_i))^{1-y_i}. \tag{2.38}$$

Taking the logarithm, the log likelihood becomes

$$\ell(\phi) = \sum_{i=1}^{n} \Big(y_i \log \sigma(\phi^T x_i) + (1 - y_i) \log\Big(1 - \sigma(\phi^T x_i)\Big)\Big). \tag{2.39}$$

The model is fitted by maximizing the log likelihood

$$\phi^* = \max_{\phi} \ell(\phi). \tag{2.40}$$

Due to the sigmoid function, the log likelihood is non-linear in ϕ, which means that we do not have a closed-form solution to the optimization problem, but need optimization methods for non-linear optimization problems [Hastie et al., 2009].

Logistic regression can be regularized using Lasso or Ridge regularization [Hastie et al., 2009, sec. 4.4.4]. The regularization is done by subtracting the norm of the weights from the log likelihood. The norm is scaled by a non-negative hyper-parameter, λ, called the regularization strength, which controls the degree of regularization. Lasso regularization uses the L1-norm, and it is often used to encourage sparsity in the weights. Ridge regularization uses the L2-norm and encourages shrinkage of the weights. Solving the Ridge regularized logistic regression consists of solving the optimization problem

$$\phi^*_{Ridge} = \max_{\phi} \ell(\phi) - \lambda ||\phi||_2^2. \tag{2.41}$$

Similarly the Lasso regularized version of the logistic regression is fitted by solving

$$\phi^*_{Lasso} = \max_{\phi} \ell(\phi) - \lambda ||\phi||_1. \tag{2.42}$$

2.3.2 Neural Networks

Neural networks have lately become a popular choice for modelling data where simple linear models are not flexible enough to capture the nature of the data. Thus, making them very relevant to consider in this thesis.

Feed Forward Neural Network

A feed forward neural network is composed of a number of layers, where the output of the previous layer is the input to the next layer.

Consider the first layer of the neural network. Let $[x_1, \ldots, x_D] \in \mathbb{R}^D$ be an observation from the feature matrix $X \in \mathbb{R}^{n \times D}$, and let the first layer of the neural network consist of M hidden units. The hidden units are also called the *neurons* of the neural network. To compute these, one must first compute the activation a_j and then pass it through a non-linear activation function. The activations of the first hidden layer are computed as linear combinations of the input

$$a_j^{(1)} = \sum_{i=1}^{D} w_{ji}^{(1)} x_i + w_{j0}^{(1)}, \quad j = 1 \ldots M \tag{2.43}$$

where $w_{ji}^{(1)}$ and $w_{j0}^{(1)}$ are said to be the weights and biases of the first layer. This gives a total of $D \cdot M + M$ weights in the first layer. The hidden units, z_j, are computed by passing the activations through a non-linear activation function $h(\cdot)$

$$z_j^{(1)} = h(a_j^{(1)}), \quad j = 1 \ldots M, \tag{2.44}$$

which gives the hidden units in the first hidden layer. Typical choices of activation functions are the hyperbolic tangent (tanh), the sigmoid (σ), and the Rectified Linear Unit (ReLU).

This process is repeated for each layer of the network using the hidden units of the previous layer as the input to the next layer. Finally, the output activation(s) are calculated as a linear combination of the units in the last hidden layer. The output activations are then passed through an activation function to obtain the final output of the neural network. The number of output units and the final activation function depend on the nature of the classification or regression task at hand. For binary classification, a single output unit and the sigmoid activation function is a common choice [Bishop, 2006].

Figure 2.2 shows an example of the architecture of a feed forward neural network with input $[x_1, \ldots, x_6]$, two hidden layers, and a single output activation. The hidden layers have four and three hidden units respectively. Neural networks are non-linear because of the non-linear activation functions used when computing the hidden units. The networks become linear if all activation functions are linear. Moreover, a feed forward neural network with no hidden layers, the sigmoid as output activation function, and a binary output is equivalent to the logistic regression presented in section 2.3.1.

A feed forward neural network is trained by minimizing a specified loss function. For binary classification when using the sigmoid as the final activation function, the binary cross entropy loss is typically used.

For n observations with labels $t_1, \ldots, t_N$ and scores $y_1, \ldots, y_N$, the loss is defined as

$$E(\boldsymbol{w}) = -\sum_{n=1}^{n} \left(t_n \ln(y_n(\boldsymbol{w})) + (1 - t_n) \ln(1 - y_n(\boldsymbol{w}))\right). \tag{2.45}$$

The objective of the training is to optimize the weights, $\boldsymbol{w}$, such that the loss function is minimized. This can be done using gradient descent, which iteratively takes a step in the direction of the negative gradient

$$\boldsymbol{w}^{\tau+1} = \boldsymbol{w}^{\tau} - \eta \nabla E(\boldsymbol{w}^{\tau}), \tag{2.46}$$

where η denotes the step size (also called the learning rate). The gradient is calculated using a method called *backpropagation*, which analytically calculates the gradient of the loss function

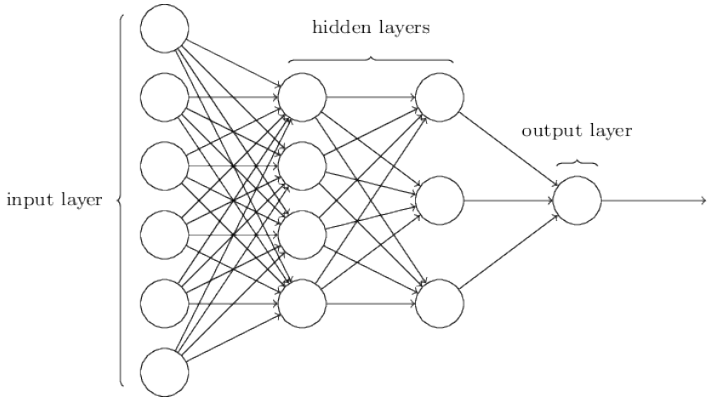


Figure 2.2: Example of an architecture of a simple 3-layer feed forward neural network. The circles in the input layer represents an input observation $[x_1, \ldots, x_6]$. The circles in the hidden layers represent hidden units. The first hidden layer thus consist of four hidden units i.e. $z_1^{(1)}, z_2^{(1)}, z_3^{(1)}$ and $z_4^{(1)}$. Finally there is an output layer with a single neuron. Hence, this neural network outputs a single output. Figure from [Nielsen, 2015].

with respect to the weights by sequentially applying the chain rule backward in the network starting from the last layer [Nielsen, 2015, Chap. 2]. A faster-converging updating algorithm, called *stochastic gradient descent* (SGD), is obtained by calculating the gradient over smaller batches of the dataset, such that steps are taken more frequently but with noisy gradients. Stochastic gradient descent has several advantages over ordinary gradient descent. Not only is it computationally cheaper to compute gradients over smaller batches, but the stochastic element of the optimizer also makes it less prone to getting stuck in local minima making it more robust [Bishop, 2006]. The optimizer *ADAM* further improves on stochastic gradient descent by using momentum and adapting the step size while training [Kingma and Ba, 2015]. Momentum is a method of using an exponentially moving average of the calculated gradients for the update step, which accelerates the descent by dampening fluctuations of the trajectory. The step size is adapted such that the step size decreases when an optimum is close.

The complexity of neural networks increases as the architecture becomes deeper and the number of layers and units increase. However, with increasing complexity, comes the risk of overfitting. There are several regularization methods for preventing overfitting, and a commonly used method is *dropout*. During training with dropout, each weight is randomly set to zero with some probability p. This reduces co-adaptation of neurons, which happens when some neurons can only detect features depending on the output of other specific neurons. Thus, adding dropout reduces the likelihood of overfitting [Hinton et al., 2012].

Another approach to reduce overfitting is *weight decay*, which is an L2 regularization of the model weights [Nielsen, 2015, Chap. 3]. The regularization is created by adding a scaled sum of squares of the model weights such that

$$E(\boldsymbol{w})_{WD} = E(\boldsymbol{w}) + \frac{\lambda}{2n}||\boldsymbol{w}'||_2^2 \tag{2.47}$$

where $E(\boldsymbol{w})$ is the non-regularized loss function from equation (2.45), λ is the regularization parameter, and $\boldsymbol{w} = [\boldsymbol{w}' \ \boldsymbol{w_0}]$, where $\boldsymbol{w}'$ are the model weights and $\boldsymbol{w_0}$ are the model biases. The regularization parameter, λ, thus controls the trade-off between small weights, encouraged

by the regularization, and the minimization of the loss function. The partial derivatives of the regularized loss function w.r.t. the weights $\boldsymbol{w}'$ and biases $\boldsymbol{w}_0$ become

$$\frac{\partial}{\partial \boldsymbol{w}'} E(\boldsymbol{w})_{WD} = \frac{\partial}{\partial \boldsymbol{w}'} E(\boldsymbol{w}) + \frac{\lambda}{n} \boldsymbol{w}' \tag{2.48}$$

$$\frac{\partial}{\partial \boldsymbol{w}_0} E(\boldsymbol{w})_{WD} = \frac{\partial}{\partial \boldsymbol{w}_0} E(\boldsymbol{w}). \tag{2.49}$$

Thus, weight decay does not impact the partial derivatives of the biases, which means the biases can be updated using gradient descent following the same concept presented in equation (2.46). In order to handle the gradient w.r.t. to the weights, the last term, $\frac{\lambda}{n}\boldsymbol{w}'$, is simply added to the gradient of the non-regularized loss-function also calculated w.r.t. the weights. This means that the weights can be updated using gradient descent with a small modification:

$$\boldsymbol{w}'^{,\tau+1} = \left(1 - \frac{\eta\lambda}{n}\right) \boldsymbol{w}'^{,\tau} - \eta \frac{\partial}{\partial \boldsymbol{w}'} E(\boldsymbol{w}^{\tau}), \tag{2.50}$$

Thus, the current weights are rescaled by a factor of $\left(1 - \eta\frac{\lambda}{n}\right)$ before taking a step in the direction of the negative gradient of the non-regularized loss function. Weight decay can therefore also be used with the previously mentioned optimization algorithms, Stochastic Gradient Descent and ADAM.

Convolutional Neural Network

Feed forward neural networks (FFNN) presented above fall short when modelling images. Images are typically high dimensional (since the number of features equals the number of pixels), which requires a large number of weights in the input layer. Moreover, the spatial location of the pixels conveys information and images typically have *translation invariance*, meaning that images can be, e.g., shifted or rotated without changing the meaning of the motive [Nielsen, 2015, Chap. 6]. These properties are not modelled efficiently in the classical network architecture, and the convolutional neural network (CNN) addresses these challenges.

Analogously to the section presenting the feed forward neural network, the convolutional neural network will be presented with the first layer as starting point. Each pixel in the input image corresponds to an input neuron in the CNN. Unlike in the FFNN, the hidden units in a CNN are only connected to an $L \times M \times C$ grid of the input neurons, where C is the number of channels in the input image. The region of the input neurons accessible to a hidden neuron is called the local receptive field. The hidden unit $z^{(1)}_{j,k}$ in the first hidden layer, is calculated using the $L \times M \times C$ kernel, $\boldsymbol{w}$, in the following way:

$$z^{(1)}_{j,k} = h\left(b^{(1)} + \sum_{l=0}^{L-1}\sum_{m=0}^{M-1}\sum_{c=0}^{C-1} w^{(1)}_{l,m,c} \cdot x_{j+l,k+m,c}\right), \tag{2.51}$$

where $h(\cdot)$ is the activation function, b is the bias, $w_{l,m,c}$ is the $(l,m,c)^{th}$ entry in the kernel and $x_{j+l,k+m,c}$ is the $(j+l,k+m,c)^{th}$ entry in the input image. This calculation is performed repeatedly for different local receptive fields, which are created by moving the kernel across the input neurons. How far the kernel is moved each time is called the *stride*. A stride of 1 corresponds to moving the kernel a single pixel at a time when computing hidden neurons of the

first layer. A larger stride and/or a larger kernel will therefore yield fewer local receptive fields and thereby fewer hidden units.

The hidden units computed with a specific kernel collectively form a feature map where each unit shares its weights and biases with the other hidden units, i.e., the weights and bias seen in equation (2.51). Thus, by moving the kernel across the input neurons, the local receptive field identifies the same characteristics across all sections of the image. By performing the calculations in this way, only a single kernel ($M \times L \times C + 1$ weights in total) needs to be learned to create a feature map, which drastically reduces the required number of parameters compared to a FFNN. Typically, there will be a number of feature maps produced in each layer such that they can be trained to identify different characteristics based on the input neurons. This is done by using several kernels on the input neurons and letting each feature map become a channel in the hidden layer. The collection of these feature maps created by hidden units are called a *convolutional layer*. When computing the neurons of the next layer of the CNN, the produced feature maps from the previous layer are then used as the new input. [Nielsen, 2015, Chap. 6].

Besides the convolutional layers, CNNs typical includes *pooling* and *batch normalization*. The pooling layer performs an aggregation of the local receptive field in each feature map thus reducing the size of the feature map. Possible aggregation functions used in the pooling layer include taking the average, max or the $L2$ norm of the units in the local receptive field. By using pooling, only the presence of a certain feature in a region of the image is retained, while the exact location is left out. This helps to further reduce the number of parameters in the network [Nielsen, 2015, Chap. 6]. Batch normalization was proposed by [Ioffe and Szegedy, 2015] to make the task of training neural networks easier. Batch normalization normalizes the produced feature maps to have zero-mean and unit standard deviation, and instead makes the mean and standard deviation learnable parameters. Batch normalization speed up the training of the network because the distribution of the input does not change during training due to the updates of the weights.

Finally, the last stage of convolutional neural networks consists of flattening the last convolutional- or pooling layer and adding one or more fully connected layers to create an output that can be used for, e.g., classification. Figure 2.3 shows an example of the architecture of a simple convolutional neural network with a single input channel, a convolutional layer, a pooling layer, and a fully connected output layer.

DenseNet

DenseNet is a model with a specific architecture proposed by [Huang et al., 2018]. It combines the idea of densely connected feed forward neural networks and convolutional neural networks. The authors call their architecture a Dense Convolutional Network - i.e., DenseNet.

As described in the previous subsection, ordinary convolutional neural networks (CNN) create a number of feature maps in each layer of the network. The first feature maps are learned directly from the input pixels. The maps are then passed to the second layer, which creates new feature maps that are passed to the third layer, etc. The feature maps in layer ℓ are thus calculated as

$$\boldsymbol{x}_\ell = H_\ell(\boldsymbol{x}_{\ell-1}), \tag{2.52}$$

where $\boldsymbol{x}$ are the feature maps and $H_\ell(\cdot)$ is the non-linear transformation of the feature maps in

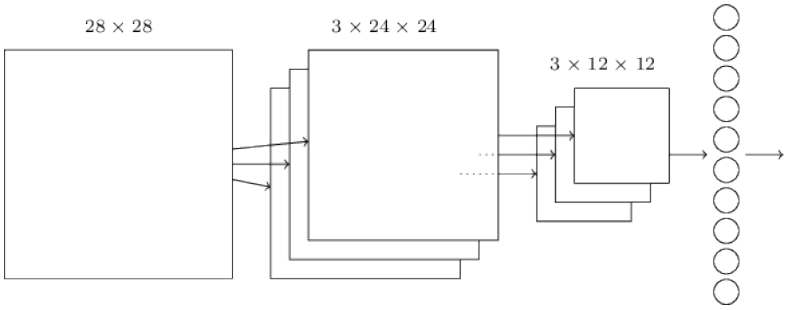


Figure 2.3: Example of a convolutional neural network with a single convolutional layer, a pooling layer, and a fully connected output layer. The input is an image consisting of 28×28 pixels and a single channel. Three feature maps, each with 24×24 hidden neurons, are created in the convolutional layer. This is followed by a pooling reducing the size of each feature map to 12×12. Finally, the fully connected layer has ten neurons, which could, e.g., correspond to the output of a classification task with ten classes. Figure from [Nielsen, 2015].

the ℓ^{th} layer[1]. The feature maps created in any subsequent layer of an ordinary CNN should thus learn new information as well as preserve needed information of the feature maps from the previous layer. Increasing the depth of the network increases the distance between input and output, which can hinder the flow of information and gradients through the network. The architecture of DenseNet alleviates this issue by creating a "collective knowledge" in the network using the concatenated feature maps learned in the previous layers as input. Thus, the feature maps in layer ℓ are calculated as

$$\boldsymbol{x}_\ell = H_\ell([\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_{\ell-1}]), \tag{2.53}$$

where $[\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_{\ell-1}]$ are the feature maps from the preceding layers concatenated along the channel dimension. In DenseNet, $H_\ell(\cdot)$ is a composite of three operations: batch normalization, the rectified linear unit (ReLU) function and a 3×3 convolution.

The DenseNet architecture is composed of a number of *dense blocks* in which each layer takes the feature maps from all preceding layers in the dense block as input. The number of feature maps produced by $H_\ell(\cdot)$ is called the growth rate. Figure 2.4 shows an example of a dense block with a growth rate of four. Between the dense blocks, transition layers with 1×1 convolution and a 2×2 average pooling are used to reduce the dimensions of the network and thereby the number of required parameters.

[1]The notation used in section is slightly different from the notation used in subsection 2.3.2. We have used this notation to preserve the notation used in the original paper proposing the architecture [Huang et al., 2018].

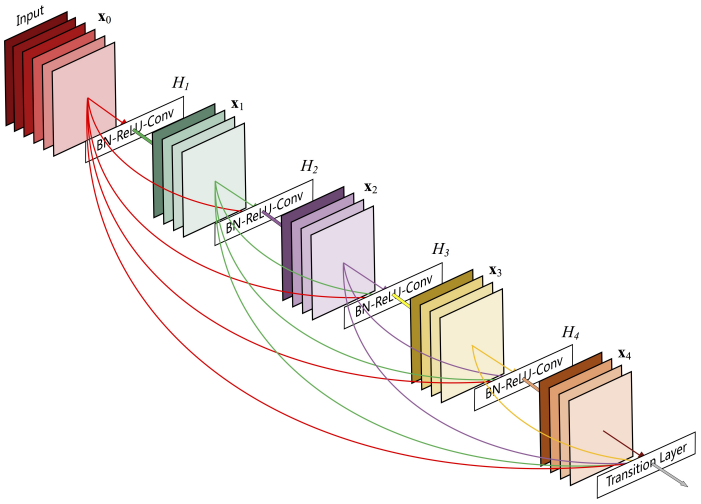


Figure 2.4: Layout of a dense block with a growth rate of four in DenseNet. Figure from [Huang et al., 2018]

The DenseNet network architecture was a big improvement on existing networks such as ResNet, and at the time of publication, DenseNet significantly improved the performance on the two CIFAR datasets, ImageNet, and SVHN [Huang et al., 2018].

Chapter 3

Materials and Methods

In this chapter, we will present the datasets that are used as examples for the fairness analyses and describe how they have been modelled. In section 3.1 the datasets are presented, including a description of the pre-processing process and the distribution of observations broken down by the sensitive group and target variable. In section 3.2 we present how the predictive models are optimized and trained on the example datasets.

3.1 Example Datasets

Four small datasets have been chosen as example datasets in this thesis. The datasets are used to showcase the use of the toolkit to do fairness analyses, which will be presented in chapter 4. Each of the datasets contains a binary classification problem, where the predictions made by an algorithm potentially have a large impact on an individual's life. The datasets come from two different domains: Recidivism and credit scoring, and they will be presented by category one at a time.

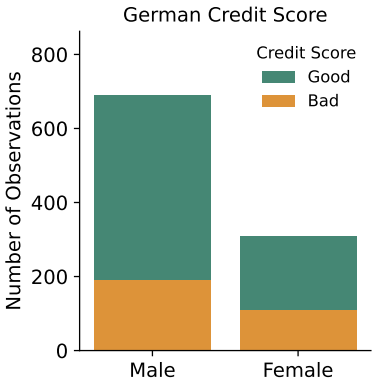
3.1.1 Credit Scoring

The first example is two credit score datasets from Germany and Taiwan. A credit score is a score used to approximate individuals' creditworthiness, and this score is used by banks when deciding whether to offer credit or a loan to an individual. The score is calculated based on the individual's financial situation [Brock, 2021]. It is important the scores are fair, since the predicted score influences an individual's loan opportunities.

German Credit Score Data

The first credit data set is the German Credit Data dataset [Dua and Graff, 2017], which is commonly used as an example in the fairness literature [Verma and Rubin, 2018, Bellamy et al.,

2019, Mehrabi et al., 2019, Kamiran and Calders, 2012]. The dataset is from 1994 and contains 1000 observations with 20 attributes each. The attributes include credit history, employment situation, personal status, and sex. The target variable is the credit score. The credit score is binary, and it can be either good (y = 0) or bad (y = 1). In the fairness analysis, we will analyze the fairness of the predictions from the models trained on the dataset with respect to sex. A total of 30% of the individuals had a bad credit score, and a slightly larger proportion of women (35%) than men (28%) had a bad credit score. Out of the 1000 individuals, 690 were male (figure 3.1).



(a)

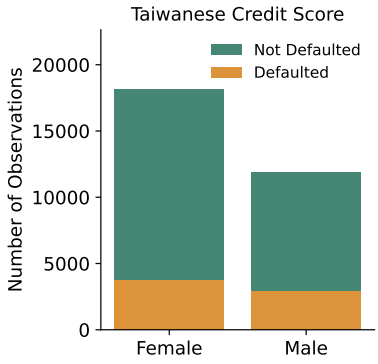
Sex	n	Bad Credit Score
Male	690	191 (28%)
Female	310	109 (35%)
Total	1000	300 (30%)

(b)

Figure 3.1: Number of observations and percentage of individuals receiving a bad credit score broken down by males and females. The figure (a) visualizes the numbers shown in the table (b).

Taiwanese Credit Score Data

The second credit score dataset is from a Taiwanese bank, and the dataset consists of 30,000 credit card holders from the bank [Yeh and Lien, 2009, Dua and Graff, 2017]. The dataset contains the following attributes: Gender, amount of credit given, education, marital status, age, history of past payment delays, history of bill statements, and history of payment amount. The three last attributes contain monthly history from April 2005 to September 2005 amounting to 18 attributes containing the history for the past six months for each of the three variables. The target is to predict whether a credit card holder will default the payment due the next month (October 2005). The dataset does not contain any missing values. Similar to the German credit score dataset, the Taiwanese dataset will be evaluated using sex as the sensitive group. Figure 3.2 shows that the rate of default is similar for men and women (24% and 21 % respectively), and 18,112 out of the 30,000 credit holders are female.



(a)

Sex	n	Defaulted
Female	18112	3763 (21%)
Male	11888	2873 (24%)
Total	30000	6636 (22%)

(b)

Figure 3.2: Number of credit holders and percentage of credit holders defaulting on their loan broken down by males and females. The figure (a) visualizes the numbers shown in the table (b).

3.1.2 Criminal Recidivism

Predictive algorithms can be used for risk-assessment, and one of the most widely known use cases is predicting the risk of offenders recidivating. There can be many reasons why recidivating is interesting to predict from a law enforcement perspective. We will consider two datasets to predict recidivism of offenders in Florida, USA (COMPAS dataset) and juvenile offenders in Catalonia, Spain.

COMPAS

COMPAS stands for *Correctional Offender Management Profiling for Alternative Sanctions* and is an algorithmic tool used across the United States to assess an offender's risk of recidivating [Northpointe, 2015, Angwin et al., 2016]. The tool aids the judges in determining whether or not an offender is too dangerous to be let out on bail. The tool was created and owned by *Northpointe Inc.*, who later have become *Equivant*. The algorithm allocates a risk score between 1 and 10 to each offender, categorizing them into low, medium or high risk in terms of recidivating within two years. Prior to this prediction, each offender is screened by answering more than 130 questions, and the answers are used to calculate the risk [Northpointe Inc., 2011, Northpointe, 2015].

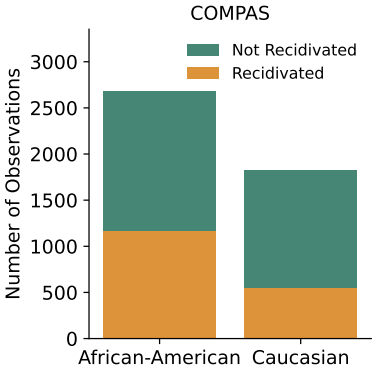
In 2016 ProPublica, which is a non-profit organization aiming to produce investigative journalism in the public's interest, published an article analyzing scores from the COMPAS algorithm [Angwin et al., 2016]. The ProPublica analysis concluded that the algorithm discriminated against African-Americans because African-Americans generally received higher risk scores and had a higher false positive rate than White offenders. The analysis sparked discussions regarding algorithmic fairness in general, and more specifically, about algorithmic amplification of discrimination of African-Americans in the American justice system. Because of this debate, the dataset has become a cornerstone in fairness-related research, and it is often used for benchmarking purposes.

ProPublica obtained scores by the COMPAS algorithm for defendants pretrail in Broward

County, Florida, from 2013 and 2014. They matched the COMPAS scores with criminal records in the county collected on April 1st, 2016. In this way, they could determine who had become recidivists by April 2016. ProPublica has gathered their processed data in a csv file called *compas-scores-two-years.csv*, which can be accessed on their github [Pro-publica, 2017]. This was created specifically to investigate recidivism within two years, and it is the one Propublica used in their analysis. ProPublica chose a time horizon of two years based on Equivant's Practitioner's Guide, which states that the general recidivism score should predict "a new misdemeanor or felony offense within two years of the COMPAS administration date" [Northpointe, 2015, p. 27]. The COMPAS scores lie in [1,10], which correspond to a risk classification ([1,4] = low, [5,7] = medium, [8,10] = high). In this analysis, medium and high risk classification is considered a positive prediction of recidivism because the Equivant Practitioner's Guide states that "scores in the medium and high range garner more interest from supervision agencies than low scores, as a low score would suggest there is little risk of general recidivism" [Northpointe, 2015, p. 27].

The original data file contains 7214 observations of COMPAS scores. The COMPAS scores were linked to the criminal case associated with the individual's COMPAS score. For some COMPAS scores, no criminal case for the individual was found within 30 days of the date of the individual's COMPAS score, and these observations were excluded from the dataset in the ProPublica analysis. We have excluded the same observations, which leaves 6172 observations in the dataset. ProPublica has been criticized for introducing a two-year sample cutoff only for non-recidivists, but not for recidivists [Barenstein, 2019]. We agree with this critique and therefore omit all individuals who recidivated after April 1st, 2014, as their non-recidivated counter-parts cannot be included because the follow-up time is less than 2-years.

The dataset contains an attribute describing the race of each person. Due to small group sizes, we excluded Hispanics (n = 448), Asians (n = 27), Native Americans (n = 9) and the category Other (n = 309). After this exclusion, 4511 observations remain. Figure 3.3 shows how observations are distributed across race and recidivism status. There are more African-Americans recorded in the dataset, and a higher proportion of African-Americans recidivate (44%) compared to Caucasians (30%).



(a)

Race	n	Recidivists
African-American	2682	1168 (44%)
Caucasian	1829	548 (30%)
Total	4511	1716 (38%)

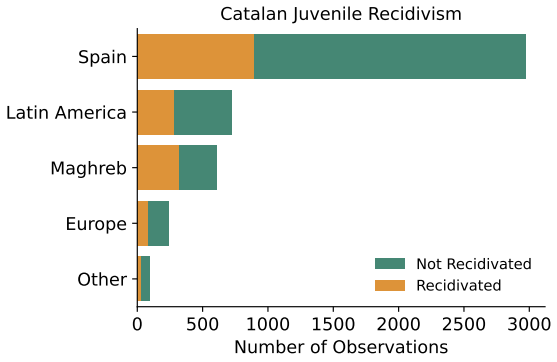
(b)

Figure 3.3: Number of observations and percentage of individuals recidivating broken down by race. The figure (a) visualizes the numbers shown in the table (b).

Catalan Juvenile Recidivism

The Government of Catalonia's Department of Justice has a *Centre for Legal Studies and Specialized Training* (CEJFE). They have published several datasets used for their research within the social and criminological research and training area of the CEJFE. One of the published datasets contains information about juvenile crime and recidivism in Catalonia. The data contains information about juveniles who completed an educational program, as a consequence of their criminal behavior, in 2010 in Catalonia. The dataset also includes information about whether each juvenile recidivated by 2013 and by 2015 [Centre d'Estudis Jurídics i Formació Especialitzada, 2016]. The dataset contains n = 4753 records and 132 attributes grouped into 11 categories [Centre d'Estudis Jurídics i Formació Especialitzada, 2015]. We will restrict our analysis to use attributes from three categories: Demographic variables, criminological variables, and variables regarding the assigned educational program. The remaining categories contained detailed information about the juveniles' recidivism or the different programs attended. These remaining categories were not included because the attributes about the programs contained many missing values, and we concluded that we did not require such detailed data about each program for our purpose. Data about each juvenile's recidivism status have been collected for the years 2013 and 2015, respectively. We have chosen to model whether or not the juveniles have recidivated after five years, i.e., in 2015, as the target variable.

The dataset has a high level of detail, and to avoid groups containing very few individuals, we omit the categorical attributes describing the committed crime in most detail, the region of the juvenile, and the attribute describing the exact country of origin of each juvenile. We also extract year and month from date attributes. All text in the dataset, e.g., names of attributes, are in Catalan, so we have translated attribute names and selected categorical variables to English. Table A.1 in the appendix lists the final processed attributes, including a short explanation and a summary of their domains. The juveniles' area of origin, which is grouped into the categories: Spain, Latin America, Maghreb, Europe, and Other, is used as the sensitive attribute for the analysis. Figure 3.4 shows that the groups are very unbalanced in size and that the recidivism rate differs by area of origin. The recidivism rate of juveniles from the Maghreb region is 52%, which is much larger than the overall recidivism rate in the dataset (34%).



(a)

Area of Origin	n	Recidivists
Spain	2972	893 (30%)
Latin America	727	279 (38%)
Maghreb	610	315 (52%)
Europe	244	83 (34%)
Other	99	24 (24%)
Total	4652	1594 (34%)

(b)

Figure 3.4: Number of observations and percentage of individuals recidivating broken down by origin. The figure (a) visualizes the numbers shown in the table (b).

3.2 Modelling Example Data Sets

In order to analyze the fairness of the predictive models, we first need to build the predictive models. The COMPAS dataset includes the output of the COMPAS risk classification, and this classification is used for the fairness analysis. For the remaining datasets, we construct two types of predictive models: A neural network and a logistic regression. We have chosen to use a logistic regression because it is simple to implement and use and, therefore, quite commonly used in the industry. The neural network is chosen because neural networks are receiving much attention in the industry and have become state of the art for many prediction tasks.

The datasets are split into training, validation, and test sets as is the standard for supervised learning. First, the data is split into two sets containing 20% and 80% of the data, respectively. The 20% split is the test set, while the 80% is further partitioned with another 80/20 split such that the smaller part is the validation set, and the larger is the training set. Each of the three subsets is standardized using *StandardScaler* from scikit-learn [Pedregosa et al., 2011] fitted on the training set. The German credit score dataset and the Catalan juvenile recidivism dataset are small, and to get a prediction of all observations, we make five different random splits of data inspired by 5-fold cross validation. Within each split, 20% are considered test data, while the remaining 80% is further partitioned into a training and validation using a new 80/20 split. Each of the five partitions is then standardized using the training set of that specific fold. By splitting the data in this manner we will fit a model to each of the splits' training data yielding five models. The sizes of all test, training, and validation splits can be seen in table A.2 in the appendix.

The logistic regression is fitted on each of the training sets using scikit-learn's implementation [Pedregosa et al., 2011]. We are using logistic regression with Ridge regularization and a regularization strength of 1. Out of the possible solvers available in the scikit-learn implementation, we use the liblinear solver. The chosen regularization type and strength are the scikit-learn default settings for logistic regression. We have chosen not to change these because an off-the-shelf model is adequate for the examples. Moreover, it is interesting to use these settings to create a predictive model for an unfairness analysis as it is likely that many other logistic regression models are fitted in a similar manner. The liblinear solver has been chosen as it often performs well on small datasets. Predictions are then made on the corresponding test set.

The neural network models are simple fully connected networks with dropout and ReLU activation between the layers and a sigmoid activation function after the last layer. The number of layers, the number of hidden units in each layer, the learning rate, and the amount of dropout are optimized using Optuna. Optuna is a hyper-parameter optimization software, which allows the user to dynamically construct the search space tailoring it to a specific task [Akiba et al., 2019]. We use Optuna to select the network architecture rather than specifying it ourselves because it minimizes the influence of our decisions on the network and thus the predictions. Each network architecture is optimized using 100 trials with a maximum of 100 epochs with the TPE-sampler and pruning, which terminates unpromising trials. Table 3.1 shows the hyper-parameter values that are being optimized over. We have chosen these sets of values to have a smaller variance in the network architectures and encourage cohesiveness across the models. This choice is also motivated by it being easier to interpret the difference between a network with 5 versus 10 hidden units in a layer compared to 5 versus 6 hidden units.

The networks are trained using binary cross entropy loss, and the validation loss is used to

tune the hyper-parameters. The default cut-off of 0.5 is used to make the binary predictions. The networks are implemented using PyTorch [Paszke et al., 2019] and trained using PyTorch Lightning. Implementations can be found in our Github repository [Fuglsang-Damgaard and Zinck, 2021].

Parameter	**Domain**
n_layers	{1, 2, 3}
n_hidden	{5, 10, 15, 20}
learning_rate	{1e-6, 1e-5, 1e-4, 1e-3, 1e-2, 1e-1}
p_dropout	{0.0, 0.1, 0.2, 0.3, 0.4, 0.5}

Table 3.1: The table shows the possible values for the hyper-parameters

Chapter 4

Fairness Toolkit

This chapter presents the toolkit BiasBalancer. Firstly, the toolkit itself will be presented and argued for in detail. This is followed by an end-to-end example of how BiasBalancer can be used, showcased in a Jupyter Notebook in which the canonical COMPAS dataset is analyzed. In the end, unfairness analyses using the toolkit will be carried out on the predictions by logistic regressions and feed-forward neural networks using the three example datasets presented in section 3.1. These analyses are presented together to showcase how BiasBalancer can be used in different scenarios.

4.1 Presentation of BiasBalancer

In this section, we present the fairness toolkit, *BiasBalancer*. The aim of BiasBalancer is to combine several fairness measures in order to gain a nuanced and comprehensive analysis of the fairness of a predictive algorithm. The source code and documentation of BiasBalancer can be found in the Github respository [Fuglsang-Damgaard and Zinck, 2021] and in the documentation page [Fuglsang-Damgaard and Zinck, 2022]. A screenshot from the documentation page is included in the appendix (figure A.3) to give the reader an impression of the documentation. BiasBalancer has a hierarchical structure with three levels, where each level assesses the fairness in ascending level of detail. This gives the user easier navigation through the fairness analysis of the algorithm in question. The levels can be summarized as:

- **Level 1:** A single number that aims to summarize the potential degree of unfairness present in the predictive algorithm.
- **Level 2:** An overview plot visualizing the potential sources of the unfairness measured in level 1.
- **Level 3:** A suite of methods that can be used to dive further into the sources of the unfairness in the predictions.

The following sections present each of the three levels.

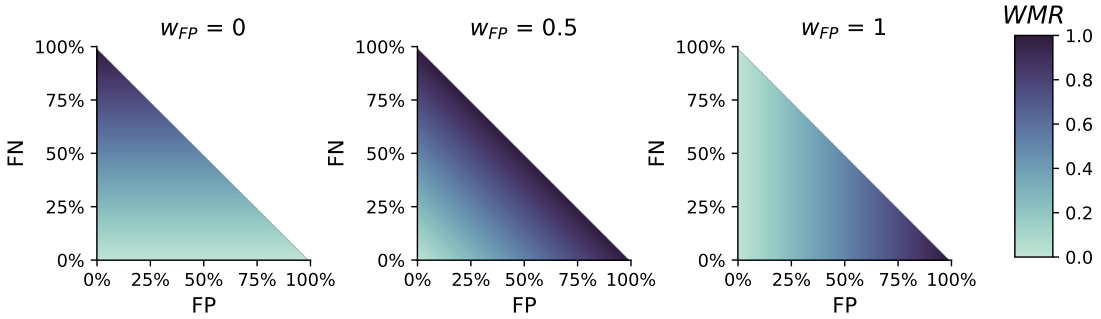


Figure 4.1: Weighted misclassification rate (WMR) for varying choices of the false positive weight, w_{FP}, depicted as a function of the percentage of false positives and false negatives out of a total of n classified observations.

4.1.1 Level 1: Single Unfairness Measure

The first level of BiasBalancer is a single metric measuring the degree of unfairness present in the predictions. The metric is a single number because it aims to allow for easy comparison between models and is a quick indicator of the level of unfairness.

To create this metric, we first construct the *weighted misclassification rate* (WMR). The weighted misclassification rate is a measure of how large the error of a classifier is, weighted by how severe a false positive is compared to a false negative. The weighted misclassification rate is defined as

$$WMR = c(w_{FP}) \cdot \frac{w_{FP}FP + (1 - w_{FP})FN}{n}, \tag{4.1}$$

where FP and FN are the number of false positives and false negatives respectively, n is the total number of observations, $w_{FP} \in [0, 1]$ is the false positive weight, and $c(w_{FP})$ is a normalization constant. The false positive weight indicates how unfavorable it is to receive a false positive compared to a false negative. The false positive weight is restricted to be within the interval $[0, 1]$, and a higher false positive weight indicates that false positives are considered more severe than false negatives. The normalization constant, $c(w_{FP})$, ensures that WMR is always in the interval $[0, 1]$. The constant is defined as

$$c(w_{FP}) = \min\left(\frac{1}{w_{FP}}, \frac{1}{1 - w_{FP}}\right). \tag{4.2}$$

A derivation of the normalization constant is found in appendix A.1.

Figure 4.1 shows the weighted misclassification rate as a function of the percentage of false negatives and false positives of the total number of observations for $w_{FP} \in \{0, 0.5, 1\}$. When $w_{FP} = 0.5$, the WMR increases as the percentage of either misclassification type increases. Given $w_{FP} = 0$, the WMR disregards the proportion of false positives and only increases as the percentage of false negatives increases. Vice versa for $w_{FP} = 1$. Thus, the WMR reflects the trade-off between the severity of false positives and false negatives by the choice of w_{FP}.

The weighted misclassification rate for a sensitive group a, WMR_a, can be calculated by replacing FP, FN and n with FP_a, FN_a and n_a, where, e.g., FP_a is the number of false positives in the

sensitive group a. By using the group wise WMR we can calculate the *weighted misclassification quotient* (WMQ) for each group. Let WMR_{min} be the minimum weighted misclassification rate across all groups: $WMR_{min} = \min_{a \in A} WMR_a$. Then the weighted misclassification quotient for group a is defined as

$$WMQ_a = \frac{WMR_a - WMR_{min}}{WMR_{min} + \varepsilon} \cdot 100\% \quad \text{for } a \in A, \tag{4.3}$$

where ϵ is a small number added to the denominator to make the weighted misclassification quotient well-defined for $WMR_{min} = 0$. The final metric, which is a single number measuring the overall unfairness of the model, is defined as the maximum WMQ across all sensitive groups:

$$\max_{a \in A} WMQ_a. \tag{4.4}$$

The maximum weighted misclassification quotient measures the maximum difference in weighted misclassification rate across the sensitive groups. We also introduce the fairness criterion weighted misclassification rate balance (WMR-balance), which requires the weighted misclassification rate to be equal in all sensitive groups. If the maximum weighted misclassification quotient is equal to zero, the predictions fulfill WMR-balance.

Motivation of Design

The weighted misclassification quotient is designed to satisfy a number of requirements to make it a useful measure of unfairness. In the following list, we will argue for the motivation behind each requirement and discuss how the chosen metric fulfills it.

Requirement 1: A seemingly fair model should have a metric close to 0, and the unfairness metric should increase as a model becomes more unfair

We have chosen to measure the degree of unfairness on a positive scale starting at zero because such a scale is easy to interpret and understand. A seemingly fair model should at least predict equally wrong across the sensitive groups, which would be reflected in similar weighted misclassification rates across groups giving small weighted misclassification quotients. However, if one group, a, sees more false negatives or false positives than the remaining groups, this would increase WMR_a and WMQ_a of group a, which would increase the overall model unfairness. The unfairness measure thus satisfies to increase as the algorithm becomes more unfair. It should be mentioned that the maximum weighted misclassification quotient is chosen to highlight the largest gap between groups. In this way, the weighted misclassification quotient serves as a red flag, highlighting the potentially largest source of unfairness. A weighted misclassification quotient of 0% should not be seen as a guarantee that the algorithm is fair. However, it does show that the predictions are similar in severeness across the specified sensitive groups.

Requirement 2: A perfect classifier (classifying $\hat{Y} = Y$ for all observations) should be fair according to the metric.

The supervised machine learning community strives to develop models that accurately predict an outcome based on a given target. It is crucial that the targets are chosen carefully. Given the chosen targets are sensible and fair, a perfect classifier is considered fair because every sensitive group gets the optimal prediction. The weighted misclassification rate (4.1) only considers the

false positives and false negatives of a classifier. It means that a perfect classifier would have a weighted misclassification rate of zero for all groups, thus implying a maximum weighted misclassification quotient of 0%. A perfect classifier is, therefore, deemed fair according to the metric.

Requirement 3: The metric should be customizable such that it can appropriately reflect the degree of severity of false positives and false negatives in the specific setting

In real-life cases, false positives and false negatives often yield consequences of different severity. Thus, it depends on the context whether a sensitive group is discriminated against when experiencing more false negatives or false positives compared to the other groups. The weighted misclassification rate accommodates this by including a weight w_{FP}, which enables the user to customize the metric such that it appropriately reflects the degree of severity of false positives and false negatives. If false positives and false negatives are equally bad, then $w_{FP} = 0.5$ and the weighted misclassification rate (4.1) becomes equivalent to the ordinary misclassification rate.

Requirement 4: The metric should be easily interpretable regardless of prior knowledge of fairness literature

This requirement is important as the first level of BiasBalancer aims to be a gateway to understanding the nuances of the fairness of a predictive model. Such a gateway should be simple, and using the percent-wise difference between groups is an easily understandable and interpretable way of comparing them. Furthermore, because the maximum weighted misclassification quotient simplifies to the largest percent-wise difference in misclassification rates when $w_{FP} = 0.5$, it is easily interpretable without any introduction to fairness literature. The weight is also intuitive to use as increasing w_{FP} emphasizes the importance of avoiding false positives while decreasing it emphasizes avoiding false negatives.

Requirement 5: The metric should allow for comparisons between different models.

It is common to compare the misclassification rate of different models to choose the better option for, e.g., a classification task. Analogous to that concept, the weighted misclassification quotient allows for the comparison of the largest gap in fairness across sensitive groups in a modeling situation using different models. Like the misclassification rate, the weighted misclassification quotient is independent of the number of observations in the dataset, which means that the metric is on the same scale for all datasets.

Comparison of *WMR* to other rates

Like the other presented observational fairness criteria, the weighted misclassification quotient is calculated based on rates determined by the group-wise confusion matrices. *WMQ* is based on *WMR*, separation is based on the *FPR* and *FNR*, sufficiency is based on *FOR* and *FDR*, while finally independence is based on the predicted positive rate, *PP/n*. Similarly to sufficiency and separation, the weighted misclassification rate only takes misclassified observations into account. A perfect classifier is, hence, always deemed to be fair. In this way, the weighted misclassification quotient is more similar to separation and sufficiency than the independence criterion, which typically will not deem a perfect classifier fair. *WMQ* only takes a single rate, *WMR*, into account, while separation and sufficiency are based on two rates each. Separation requires the misclassification rate to be equal across sensitive subgroups based on the true class

label. Similarly, sufficiency requires an equal misclassification rate across sensitive subgroups based on the predicted class label. As a consequence, the two rates in sufficiency and separation can yield different results. In order to get a unified answer about which group receives the most unfavorable predictions according to sufficiency or separation, it is necessary to weigh the potentially conflicting results in some way. The weighted misclassification rate is an attempt at exactly this, and the weighting of the results is done using the false positive weight w_{FP}. This weighing means that different treatments of subgroups can "cancel out" and result in an overall fair model.

Let's take an example with separation: Say groups A and B have the same number of observations ($n_A = n_B = 100$) and same underlying class distribution ($P_a = P_b = 50$). Imagine a scenario, where $FP_A = FN_B = 20$ and $FN_A = FP_B = 10$. Then the separation criterion would say that group A is treated unfairly with respect to FPR because $FPR_A = 2 \cdot FPR_B$, and similarly group B is treated unfairly with respect to FNR. However, the *WMR* -balance criterion would overall deem the model fair because both groups get a total of 30 unfavorable predictions - the unfairness experienced by the groups can be said to "cancel out". If $w_{FP} > 0.5$, indicating that false positives are more desirable than false negatives, the weighted misclassification rate would be higher for group A than for group B. The opposite would be true if $w_{FP} < 0.5$. A similar comparison is possible with sufficiency. A final difference between WMR and sufficiency and separation is that *WMR* -balance is not sensitive to the underlying distribution of actual positives or predicted positives, but only depends on the number of misclassifications and the total number of observations.

Hence, the weighted misclassification quotient is useful as a single metric attempting to measure the overall unfairness present in the model. It is, however, important to emphasize that a single metric *cannot* stand alone. Fairness analysis should include analysis of the predictions with respect to many different fairness metrics to get a nuanced picture. This is what the next level of BiasBalancer is intended for, and it is presented in the next section.

4.1.2 Level 2: Overview of Unfairness

The second level of BiasBalancer provides an overview of the potential sources of unfairness present in the predictive model. It creates the overview through a visualization, and this section will first introduce the quantities visualized and then present the visualization itself. As presented in section 2.2 many of the fairness criteria are defined as balances in rates, such as the false positive rate, across sensitive groups. The visualization, therefore, depicts absolute and relative rates for each sensitive group, as well as how they impact the different fairness criteria in an *Unfairness barometer*.

Absolute Rates

The rates chosen for the visualization are the false positive rate ($FPR = \frac{FP}{N}$), false negative rate ($FNR = \frac{FN}{P}$), false discovery rate ($FDR = \frac{FP}{PP}$) and false omission rate ($FOR = \frac{FN}{PN}$). We have chosen to show these rates in the visualization because they have either the number of false positives or false negatives in the numerator of their respective expression. By choosing these rates, we keep the focus on misclassifications, just as in the first level. Moreover, the remaining

four rates, (true positive rate, true negative rate, positive predictive value, and negative predictive value), can be derived from the four presented rates because they come in pairs summing to 1, see eq. (2.1). Along with the rates, an $\alpha = 0.05$ Wilson confidence interval is included to indicate whether differences in the rates across sensitive groups could be attributed to random fluctuations due to the size of the dataset. [Brown et al., 2001] analyzes the coverage probabilities of different types of confidence intervals for interval estimation in a binomial proportion. They show that the standard Wald confidence interval falls short, especially if the mean is near the boundaries of the distribution. Instead, their findings show that the Wilson confidence interval is a better choice, and we hence use this confidence interval.

Relative Rates

In level 1 of BiasBalancer, the difference between the weighted misclassification rates across the sensitive groups was quantified using the weighted misclassification quotient. The difference between the rates, *FPR*, *FNR*, *FDR*, and *FOR*, will be quantified in a similar manner using the *Relative rate*. For sensitive group $a \in A$ and rate $r \in \{FNR, FPR, FDR, FOR, WMR\}$, the relative rate, $RR_a(r)$, is calculated as

$$RR_a(r) = \frac{r_a - r_{min}}{r_{min} + \varepsilon} \cdot 100\% \tag{4.5}$$

$$r_{min} = \min_{a \in A} r_a, \tag{4.6}$$

where r_a is the value of rate r for sensitive group a. ϵ is a small number added in the denominator to ensure the relative rate is well-defined if $r_{min} = 0$. By construction, this means that the group with the smallest rate will have a relative rate (RR) of 0%. Notice that when $r = WMR$ then $RR_a(r)$ is equivalent to the weighted misclassification quotient (WMQ) defined in equation (4.3).

Unfairness Barometer

The fairness criteria considered in section 2.2 depend on balances in rates across sensitive groups. Table 4.1 connects the observational fairness criteria, and the weighted misclassification quotients, to those rates that need to be balanced across groups in order for the criteria to be satisfied in a strict sense. We present the *Unfairness Barometer* as a way to present an overview of how close the analyzed predictions are to satisfying the criteria. Using *unfairness* in the name of the barometer emphasizes that it is not possible to deem a model fair in all senses using the barometer, but that large values in the barometer should be used to reveal potentially unfair treatment of subgroups.

As a way of quantifying the degree to which predictions violate a fairness criterion, we introduce the *mean maximum relative rate* (MMRR). The MMRR quantifies the gap between the relevant rates of the best predicted and worst predicted groups based on the relative rates defined in eq. (4.5). The mean maximum relative rate (MMRR) for fairness criterion f is computed as

$$MMRR(f) = \frac{1}{|R_{balanced}(f)|} \sum_{r \in R_{balanced}(f)} \max_{a \in A} RR_a(r), \tag{4.7}$$

where $RR_a(r)$ is the relative rate of rate r for group a as seen in equation (4.5). $R_{balanced}(f)$ is the set of rates needed to be balanced in order to satisfy criteria f seen in table 4.1, and $|R_{balanced}(f)|$ is the number of rates the criterion is based on.

The sensitive group most unfavorably treated by the algorithm according to a fairness criterion is defined as the group with the highest relative rate on which the fairness criterion depends. For instance, the group with the highest relative false negative rate will be deemed the most disadvantaged sensitive group by the equal opportunity fairness criterion. The independence criterion is different from the other criteria because it does not depend on the number of misclassifications but instead on the fraction of predicted positives or equivalently the fraction of predicted negatives. In order for the mean maximum relative rate to correctly identify the most disadvantaged group for the independence criterion, the rate used depends on the false positive weight. Whenever the false positive rate is above 0.5, the mean maximum relative rate of the independence criterion is calculated based on the fraction of predicted positives, because a predicted positive is less desirable than a predicted negative. Vice versa, if $w_{FP} < 0.5$, the fraction of predicted negatives is used. If the false positive weight is equal to 0.5, indicating that positives and negatives are considered to be similar in terms of how desirable a prediction is, the independence criterion is not included in the unfairness barometer.

Equation (4.7) will simplify to $\max_{a \in A} RR_a(r)$ for fairness criteria only requiring balance in one type of rate. Whenever a fairness criterion depends on two rates, the mean of the maximum relative rate is used. The definition of the MMRR implies that $MMRR($*WMR*-balance$)$ is equivalent to the weighted misclassification quotient from equation (4.4).

The fairness criteria are originally formulated as hard constraints, such that a criterion is only fulfilled when there is no difference between the rates across sensitive groups. Using the mean maximum relative rate to quantify the degree to which a criterion is not satisfied leaves the question of when the violation of the criterion is small enough to not cause grave concerns. For this purpose, the Title VII of the Civil Rights Act of 1964 [U.S. Equal Employment Opportunity Commission, 1964] is used as an inspiration. According to [U.S. Equal Employment Opportunity Commission, 1979], the four fifths-rule has been adopted as a rule of thumb in order to avoid adverse impact. It is used as a heuristic to indicate whether or not there is a substantial difference in the rate of selection between groups based upon race, color, religion, sex, or national origin. Based on this rule, generally, values of the mean maximum relative rate above 20% should be investigated.

Fairness Criterion, f	**Rates to be Balanced**, $R_{balanced}(f)$
Separation	*FPR*, *FNR*
False Positive Error Rate Balance	*FPR*
Equal Opportunity	*FNR*
Sufficiency	*FDR*, *FOR*
Predictive Parity	*FDR*
Independence	PN/n or PP/n*
WMR.balance	*WMR*

Table 4.1: Observational fairness criteria listed with the rates needed to be balanced across groups for the criteria to be satisfied in a strict sense. $*PP/n$ is used when $w_{FP} > 0.5$ and PN/n is used when $w_{FP} < 0.5$. Abbreviations: *FPR* = false positive rate, *FNR* = false negative rate, *FDR* = false discovery rate, *FOR* = false omission rate, *WMR* = weighted misclassification rate.

The Visualization

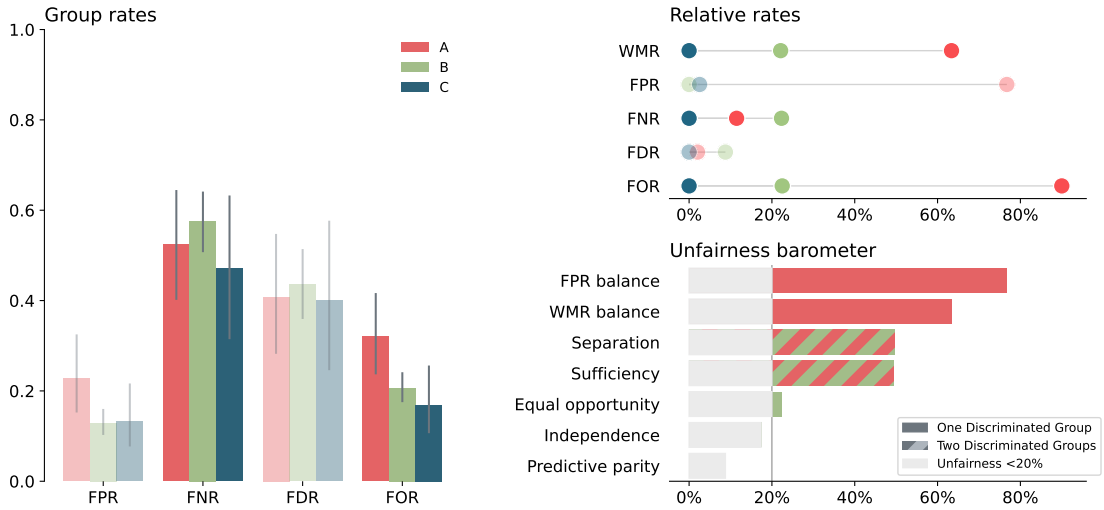


Figure 4.2: The figure shows an example of the unfairness overview visualization from the second level of BiasBalancer. It depicts the absolute and relative group rates, including how they impact the fairness criteria. The left subplot shows the false positive rate (FPR), false negative rate (FNR), false discovery rate (FDR), and false omission rate (FOR) of each sensitive group. The top right subplot shows the relative rate (RR), i.e., how many percent larger each group's rates are compared to the smallest value of each rate. The bottom right subplot shows the Unfairness Barometer, which depicts the mean maximum relative rate ($MMRR$) which measures the maximum gap between rates relevant to the given fairness criteria. The opacity is controlled by the weight w_{FP} and helps draw attention to the important rates. In this plot false positives had a small weight with $w_{FP} = 0.2$. Focus is drawn to values above 20% by shading the bars below this value. Values above 20% are colored according to the group receiving the most unfavorable predictions.

Now when the concepts have been introduced, we can present the visualization of these concepts. Figure 4.2 shows an example of BiasBalancer's second-level visualization. The prediction algorithm is only used for the purpose of presenting the visualization. The algorithm models a binary classification task with three sensitive groups: A, B, and C.

The left subplot is a barplot showing the value of the four rates for each sensitive group. The grey line on each bar indicates the 95% Wilson confidence interval of the rate. The opacity in the bars is controlled by the false positive weight. The opacity is used to draw attention to those rates that are most important with respect to either false positives or false negatives. If $w_{FP} = 0.5$, then every part of the plot will have the same opacity. Increasing the weight will highlight the rates based on false positives since false positives are then considered the most undesirable outcome. Decreasing the false positive weight will highlight the rates based on false negatives leaving the other rates less opaque. In the example seen in figure 4.2, the weight is $w_{FP} = 0.2$ yielding a stronger coloring of FNR and FOR because those rates are influenced by the number of false negatives.

The top right subplot is a plot showing the *relative rate* (RR) of the rates from the left subplot.

Moreover, it also includes the weighted misclassification quotient (WMQ). Since the weighted misclassification quotient is the relative weighted misclassification rate, the label in the visualization is WMR and not WMQ. Similar to the left subplot, the false positive weight controls the opacity of the rates. Because the WMR depends on both false positives and false negatives, the weighted misclassification quotient is kept at the same opacity for all values of w_{FP}.

Finally, the right bottom subplot shows the Unfairness Barometer depicting the *mean maximum relative rate* ($MMRR$) for each fairness criterion. The color of the bar is the color of the sensitive group, which is deemed to receive the most unfavorable predictions based on the criterion. Whenever the criteria sufficiency and separation point toward possible discrimination of two different groups, the bar is colored in both colors using stripes. All bars are gray below 20% to let the violations above 20% stand out. By adding the gray area to the visualization, we thus bring focus on substantial relative differences according to different fairness criteria between sensitive groups. Moreover, this helps to understand the magnitude of the values when comparing several second-level visualizations because the x-axis adapts to the data.

4.1.3 Level 3: Analyses of Unfairness Sources

The third level of BiasBalancer enables the user to dive further into the fairness of the predictive model and the potential sources of unfairness. It consists of a number of separate analyses, which contribute to a more nuanced understanding of the fairness of the model. An overview of the possible analyses is available in table 4.2. All analyses include a plot and an output table containing the data shown in the plot. This section will elaborate on the methods seen in the table. Moreover, it is possible to see all visualizations in figure 4.3. It is the intention that the user of BiasBalancer does not use all method but instead chooses which analyses are relevant based on the findings from the second level.

Method	When	What
Confusion Matrix	The dataset or a group contain few observations	Confusion matrix for each group
w_{FP} Influence	Unsure about how w_{FP} influences the result	WMQ for each sensitive group as a function of w_{FP}
ROC Curves	Separation, FPR-balance or equal opportunity is large in unfairness barometer	The ROC curve for each sensitive group
Calibration	Sufficiency or predictive parity is large in unfairness barometer	Calibration curve for each group.
Prediction Rates	Independence is large in unfairness barometer	Fraction of predicted positives across groups

Table 4.2: Overview of the available methods in level 3 of BiasBalancer. Abbreviations: w_{FP}: False positive weight, WMQ: Weighted misclassification quotient, ROC: Receiver operation characteristic, FPR: False positive rate.

Confusion Matrix

The confusion matrix method calculates the confusion matrix separately for each sensitive group. It shows how the distribution of predictions and targets vary by group. Each cell in the confusion matrix shows the fraction and number of observations belonging to the cell. It is optional to include the number of observations in each cell. An example of the output plot can be seen in subfigure 4.3a.

w_{FP} Influence

The chosen false positive weight, w_{FP}, influences the weighted misclassification rate *WMR* and weighted misclassification quotient *(WMQ)*. The chosen w_{FP} can have a significant impact on both the degree of measured unfairness and which group is deemed to be treated unfairly. The method calculates and illustrates *WMQ* for all values of w_{FP} (subfigure 4.3b). The method can also show the weighted misclassification rate instead of the misclassification quotient.

ROC Curves

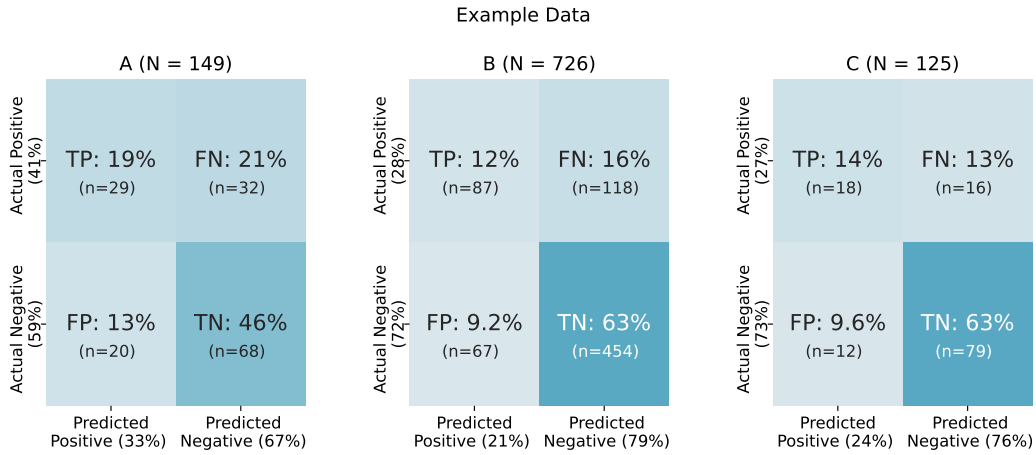
The fairness criterion *separation* is violated when the false positive and false negative rate differ between the sensitive groups. This analysis shows the ROC curves by sensitive group, which shows how the true positive and false negative rates change when changing the classification threshold. The points symbolize the classifier of the chosen threshold, τ, and the crosses indicate 95% Wilson confidence intervals of each group's true positive rate and false negative rate, respectively. An example of the output plot using a default threshold of $\tau = 0.5$ for all subgroups can be seen in subfigure 4.3c.

Calibration

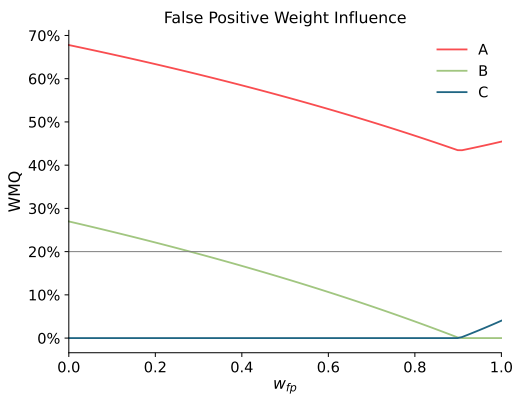
A prediction model does not satisfy *sufficiency* when there is unbalance in the false discovery rate or false omission rate between the sensitive groups. A way to fulfill sufficiency is to ensure that the classifier is calibrated by group. The calibration plot, which can be seen in subfigure 4.3d, and output data show how far the classifier is from being calibrated by group.

Prediction Rates

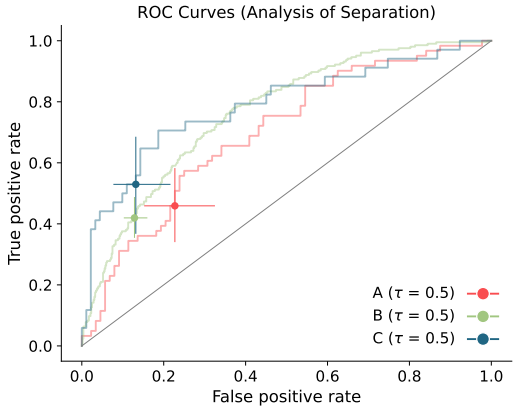
In order to satisfy *independence*, the fraction of predicted positives (or equivalently predicted negatives) must be the same for all sensitive groups. The analysis supplies this fraction along with a 95% Wilson confidence interval [Brown et al., 2001]. This illustrates how the independence criterion may be violated and whether the difference could be attributed to random fluctuations due to small group sizes. The analysis supplies the fraction of predictions of the most unfavorable outcome based on the supplied w_{FP}. An example of the output plot can be seen in subfigure 4.3e



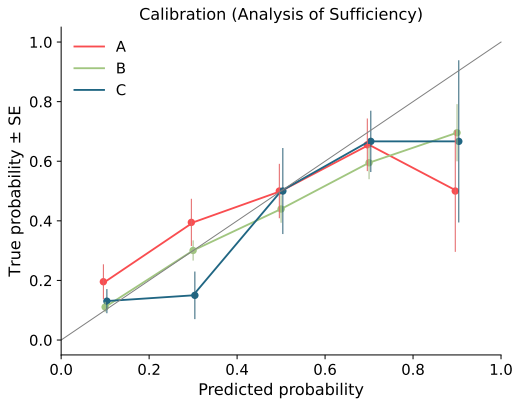
(a)



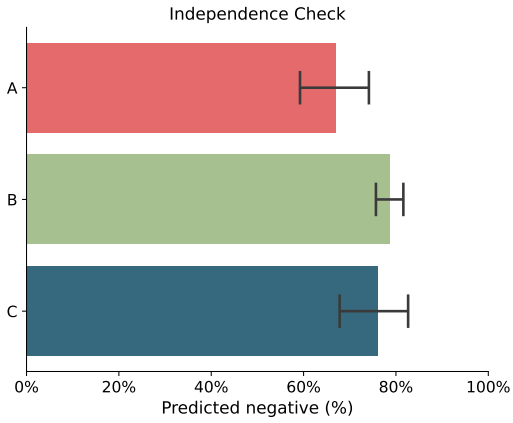
(b)



(c)



(d)



(e)

Figure 4.3: Visualizations created using the third level of BiasBalancer. Each plot enables the user to further analyze an unfairness source.

4.2 Fairness Analysis using BiasBalancer

In the following section, we demonstrate how BiasBalancer can be used to analyze unfairness on four different datasets and seven different models. In the first section (section 4.2.1) we will dive into the COMPAS dataset in a Jupyter Notebook showcasing the use of BiasBalancer. In section 4.2.2 the remaining datasets and their corresponding models are analyzed focusing on one level of BiasBalancer at a time.

4.2.1 COMPAS Example and Usage

The COMPAS dataset is analyzed using BiasBalancer in the tutorial notebook (found here in the GitHub repository). The fairness analysis is performed with respect to race, and the analysis supports the results found in the ProPublica analysis [Angwin et al., 2016]. We have chosen to analyze the COMPAS predictions in a Jupyter Notebook because the COMPAS dataset is widely known in the fairness community. The dataset, hence, serves as a good example case for showcasing how to use BiasBalancer to potential users of the toolkit.

4.2.2 Modelled Examples

We used BiasBalancer on a variety of different models and datasets (see section 3.1) in order to analyze how different types of unfairness are expressed in the analyses using BiasBalancer. In the analysis, we will present the results one level at a time to allow for comparisons between the datasets and models.

Level 1

Table 4.3 shows the number of test observations (n), the false positive weight (w_{FP}), maximum weighted misclassification quotient (Max WMQ), the group with most undesirable predictions and prediction accuracy for all models. The false positive weight is chosen to be w_{FP} = 0.9 for the recidivism risk examples because a positive prediction could result in a longer sentence or higher bail for the individual. The false positive weight is also chosen to be w_{FP} = 0.9 in the case of credit scoring. This is because a bad credit rating, which is encoded as Y = 1 in the datasets, could mean individuals did not get access to loans for which they were financially suited.

The maximum weighted misclassification quotient (Max WMQ) in table 4.3 summarizes the output of using the first level of BiasBalancer on each of the presented datasets. The measured unfairness differs greatly between the datasets, and prediction algorithms for the Catalan Recidivism dataset, COMPAS dataset, and the logistic regression model for the Taiwanese Credit dataset have maximum weighted misclassification quotients above the typical threshold of concern of 20%. The largest unfairness is present in the Catalan Juvenile Recidivism dataset, where the maximum WMQ of juveniles from the region Maghreb is 204.0% and 205.1% higher than the best-predicted group for the logistic regression and neural network, respectively. The prediction models based on the German Credit Scoring dataset see the least discrimination with maximum WMQ at 13.5% and 1.5%. There is no systematic pattern of whether a neural network or a

logistic regression results in more biased predictions. It is interesting how the two models of the Taiwanese Credit dataset can yield substantially different maximum weighted misclassification quotients even though the accuracies are less than one percentage point apart. The neural networks have substantially lower Max *WMQ* for the credit datasets.

Dataset	Model	n	w_{FP}	Max *WMQ*	Disfavored Group	Accuracy
German Credit	Logistic regression	1000	0.9	13.5%	Female	74.4%
German Credit	Neural network	1000	0.9	1.5%	Female	72.9%
COMPAS	Decile scores	4511	0.9	47.9%	African-American	66.0%
Catalan Recidivism	Logistic regression	4652	0.9	204.0%	Maghreb	73.2%
Catalan Recidivism	Neural network	4652	0.9	205.1%	Maghreb	72.9%
Taiwanese Credit	Logistic regression	6000	0.9	30.0%	Male	81.0%
Taiwanese Credit	Neural network	6000	0.9	10.0%	Male	81.9%

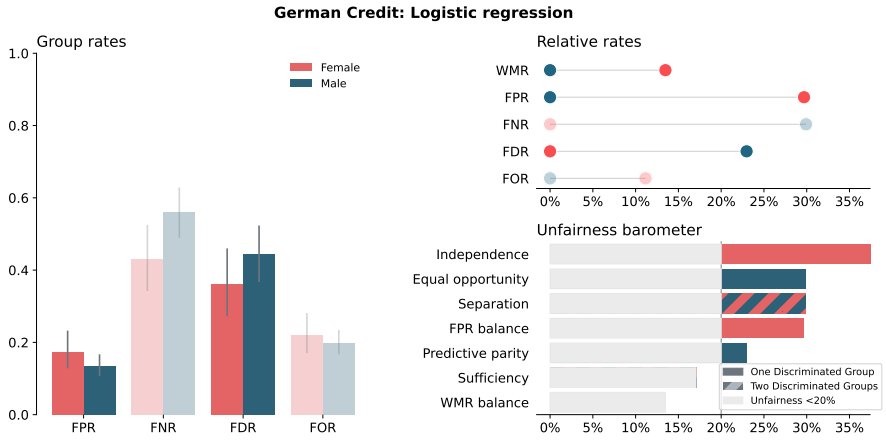
Table 4.3: The table shows the number of test observations (n), the chosen false positive weight (w_{FP}), maximum weighted misclassification ratio (Max *WMQ*), most disfavored group and accuracy for each combination of dataset and model type.

Level 2

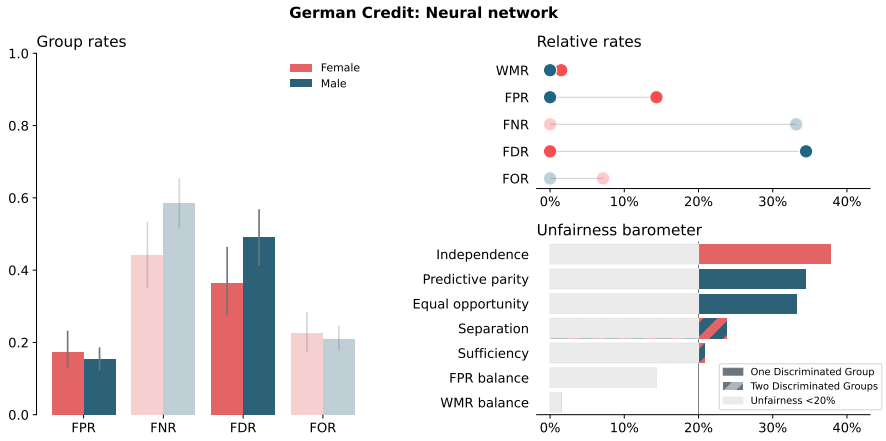
Figures 4.4, 4.5 and 4.6 show the plots generated by the second level for each of the datasets excluding the COMPAS data set, which was presented in the Jupyter Notebook. First, consider the German Credit Score dataset; in the unfairness analysis of the logistic regression predictions (Figure 4.4a) and neural network predictions (Figure 4.4b), both men and women see discrimination depending on the fairness criteria considered. The German Credit score dataset is small and imbalanced (see figure 3.1), which leads to wide 95% confidence intervals on the group rates across both classifiers. Even though the unfairness barometer contains values above the 20% threshold, the confidence intervals overlap across all the rates. The results should therefore be considered with great caution since small changes in the dataset or the model could significantly change the results of the unfairness analysis. However, the dataset is included for completeness since it is commonly used in the fairness literature.

The two classifiers predicting the creditworthiness in the Taiwanese credit score data set have very similar accuracies (table 4.3), but the unfairness barometers are very different for the two models. The barometer shows no unfairness of concern present in the predictions by the neural network (figure 4.5b). On the contrary, the unfairness barometer points towards discrimination of primarily men in the predictions by the logistic regression (Figure 4.5a). The visualizations also show that the false negative rate is high for both groups and both classifiers, which is likely be caused by the target class unbalance (see figure 3.2). The potential discrimination of men seen in the unfairness barometer of the logistic regression is largely due to the violation of FPR-balance. It is worth noting that the *FPR* is very small for both groups. This behavior shows that small absolute values of the group rates to the left in the visualization can result in large relative differences. The 95% confidence intervals are overlapping in almost all cases, but they barely overlap for the two *FPR* in the analysis of the predictions by the logistic regression. It indicates that the violation of the FPR-balance in the unfairness barometer (figure 4.5a) might need to be investigated further to understand the unfairness present in the predictions.

The unfairness analysis of the predictions by the logistic regression (figure 4.6a) and the neu-



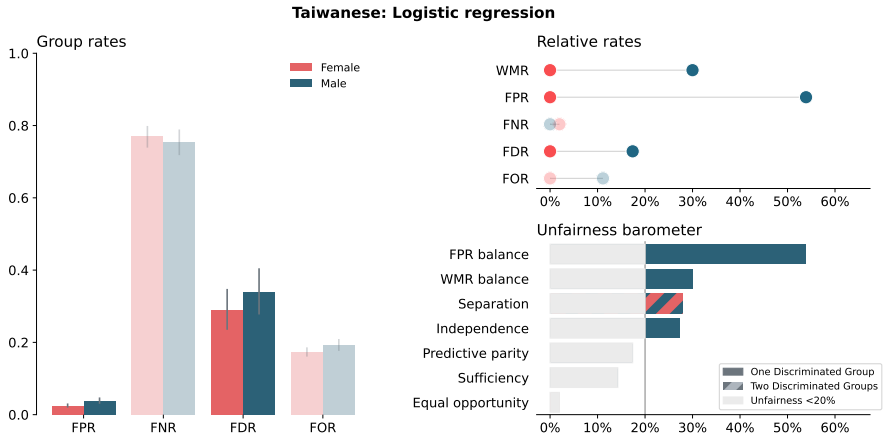
(a)



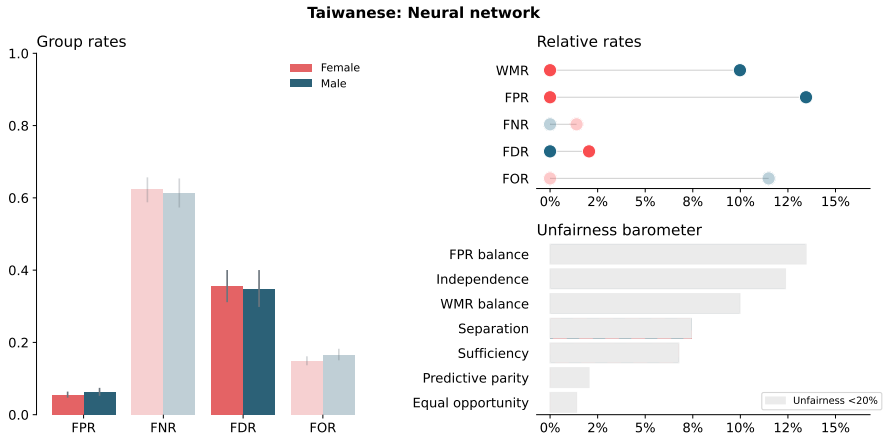
(b)

Figure 4.4: Layer 2 visualizations of models on German Credit Scoring data.

ral network (figure 4.6b) on the Catalan Juvenile Recidivism dataset reveals large amounts of unfairness for the juveniles originating from Maghreb. The unfairness barometers in the two plots point toward the same main source of unfairness. The juveniles originating from Maghreb have a much higher false positive rate, which results in large violations of the FPR-balance, WMR balance, and separation criteria. The independence criterion suggests Maghreb juveniles are discriminated against in both classifiers because more Maghreb juveniles receive positive predictions, i.e., are classified as likely recidivists. Generally, it is worth noting that the data set with many sensitive groups sees the largest amount of unfairness. With many subgroups, there are more possibilities of unfairness since the difference between the smallest and largest rate will increase due to increased variation. Some subgroups also see wide confidence intervals on their group rates, highlighting the difficulty with unequal group sizes.



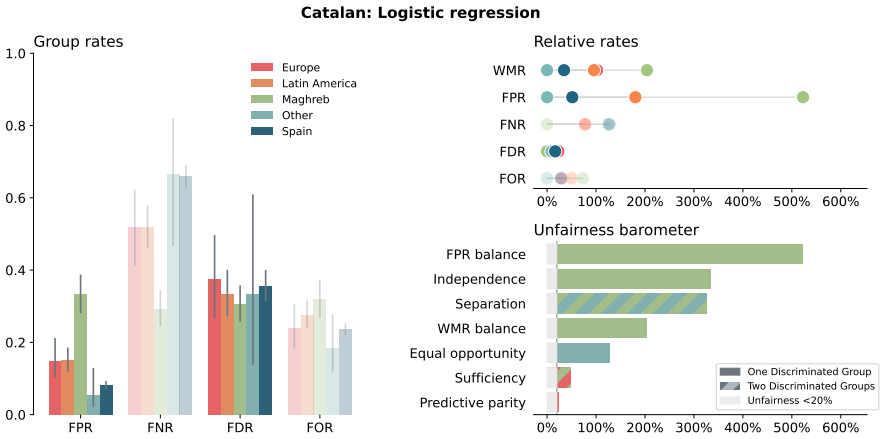
(a)



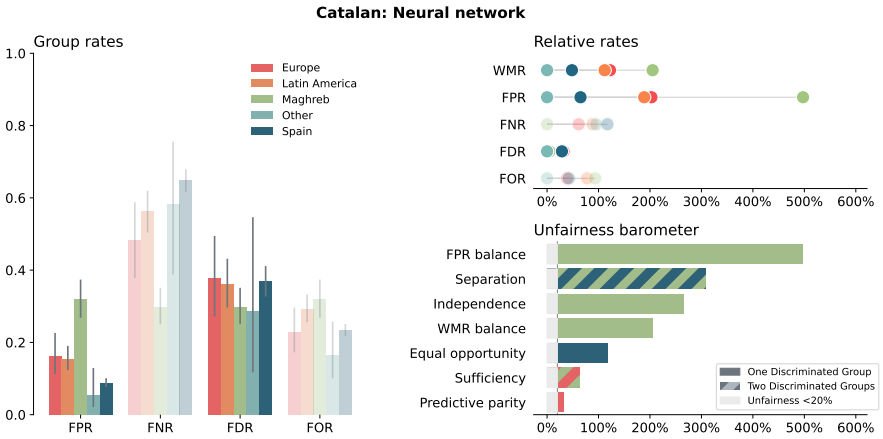
(b)

Figure 4.5: Layer 2 visualizations of models on Taiwanese Credit Scoring data.

The sufficiency criterion and predictive parity criterion, which is a relaxation of sufficiency, were among the three least violated criteria for all models except the German neural network. This fits well with the bounds presented in section 2.2.5 that showed that the supervised learning framework generally favors the sufficiency criterion.



(a)



(b)

Figure 4.6: Layer 2 visualizations of the analysis of the predictions by logistic regression (a) and neural network (b) on Catalan Juvenile Recidivism data.

Level 3

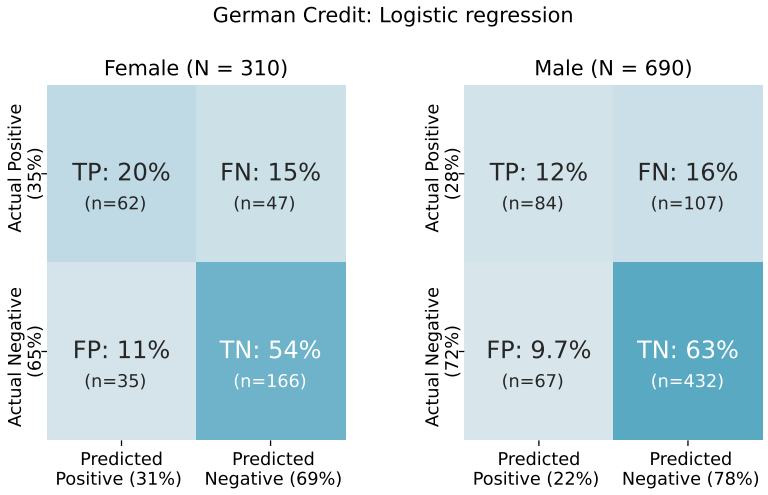
We will dive into a couple of relevant analyses from the third level of BiasBalancer for each dataset to highlight the most relevant findings.

The second-level unfairness analysis of the predictions for the German Credit dataset showed disfavoring against both men and women depending on the criterion considered. The dataset is small and imbalanced, which was seen in the large confidence intervals on the absolute rates. Figure 4.7 shows the confusion matrices based on the predictions by the two models. The logistic regression correctly classifies one woman more than the neural network, and the two confusion matrices for women are thus almost identical. The confusion matrices for the male subgroup are more different when focusing on the percentage. However, inspecting the specific numbers reveal that the neural network only misclassifies 14 males more than the logistic regression. This

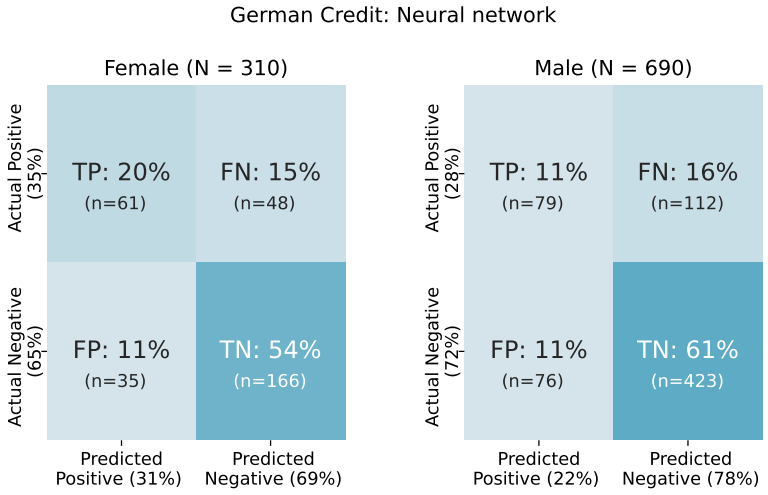
shows that small changes in the classification can result in quite large changes in the assessment of unfairness when the dataset is small and imbalanced.

Unbalanced false positive rates are the main source of unfairness in the Taiwanese Credit dataset logistic regression predictions. The predictions also suffer from large false negative rates (≈ 0.8). Figure 4.8 shows that the ROC curves for the two sensitive groups are identical at $TPR \approx 0.5$ and $FPR \approx 0.1$. This means that the FNR could be decreased from 0.8 to 0.5 while creating FPR-balance by just changing the thresholds. Moreover, by mitigating the FPR-balance, one would also mitigate separation since the separation criterion enforces both FPR and FNR-balance.

The dataset yielding the most unfair predictions is the Catalan juvenile recidivism dataset. The predictions for juveniles from Maghreb are deemed unfair according to the FPR-balance, independence, separation, and WMR-balance criteria. In figure 4.9c the ROC curve for Maghreb juveniles follows the other ROC curves closely, showing that a change in threshold for Maghreb juveniles could mitigate the violation of FPR-balance and separation. Figure 4.9b shows that the independence criterion is violated because the Maghreb juveniles receive significantly more predicted positives. However, the recidivism rate is also higher for Maghreb juveniles (see figure 3.4). The WMR-balance criterion also points to possible discrimination of Maghreb juveniles with $\max WMQ = 204.0\%$ for the logistic regression. This evaluation rests on the assumption that a false positive prediction is much more unfavorable than a false negative, which is parameterized with $w_{FP} = 0.9$. However, if a positive prediction of recidivism resulted in positive effects such as better access to guidance and education rather than longer jail times, the picture could be the opposite. Figure 4.9a shows that the choice of false positive weight is crucial because the discrimination of Maghreb juveniles quickly decreases if w_{FP} is lowered.



(a)



(b)

Figure 4.7: Confusion matrices for males and females in the German Credit Data with the logistic regression in (a) and neural network in (b).

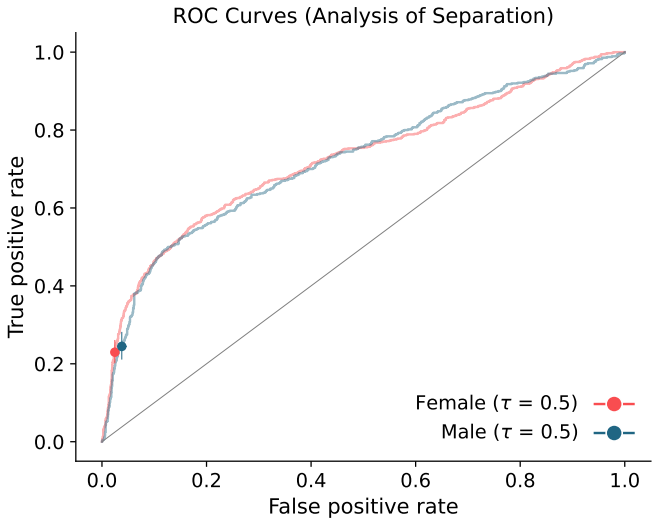
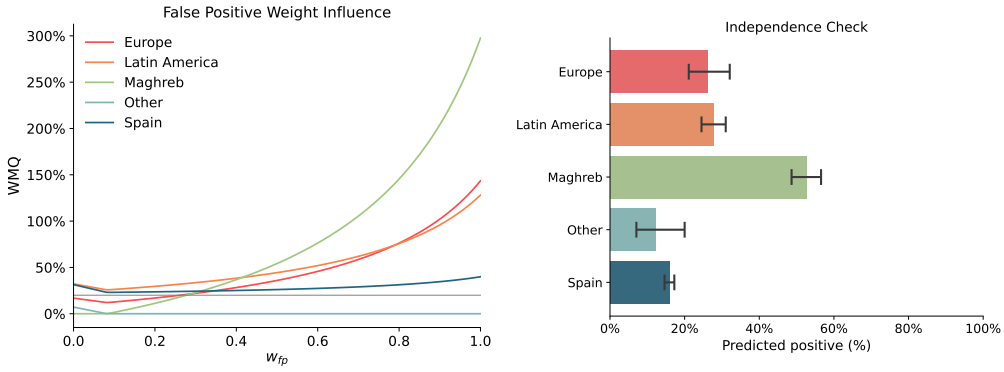
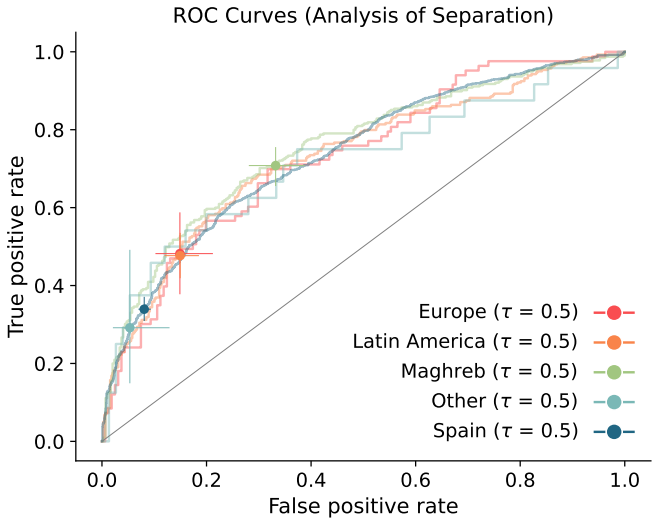


Figure 4.8: ROC curves by sensitive group of the logistic regression predictions on the Taiwanese Credit data. The points symbolize the classifier of the chosen threshold and the crosses indicate 95% Wilson confidence intervals of the rates.



(a)

(b)



(c)

Figure 4.9: Plots for fairness analyses from level three of BiasBalancer used on logistic regression predictions of Catalan juvenile recidivism. Subfigure (a) shows how the chosen false positive weight influences the *WMQ* , subfigure (b) shows the percent predicted positive for each sensitive group, and subfigure (c) shows the ROC curves by sensitive group. On the ROC curves, the points symbolize the classifier of the chosen threshold, and the crosses indicate 95% Wilson confidence intervals of the rates.

Chapter 5

Case Study: CheXpert

In the previous chapter, we saw several examples of how to use BiasBalancer on smaller datasets. In this section, we present a case study of the CheXpert dataset, which contains chest x-ray images. The aim of the case study is two-fold: We want to show how to use BiasBalancer to detect unfairness in the predictions from a predictive model fitted on a larger dataset, and we want to investigate more in-depth how a predictive model created to maximize performance with no fairness criteria in mind fares when inspecting its predictions in a fairness analysis.

Recent work by [Larrazabal et al., 2020] and [Banerjee et al., 2021] has explored the CheXpert dataset considering gender and race respectively. [Larrazabal et al., 2020] trained different types of convolutional neural networks for the classification of diseases in the CheXpert x-rays solely using training sets of either male or female patients. Based on the area under the ROC curve (AUC), they showed that regardless of the type of model or the gender of the patients present in the test set, the models were significantly better at predicting the presence of the diseases for the gender from which the model had seen training data. This difference was statistically significant for the majority of diseases. The work highlights the importance of diverse representation in data. [Banerjee et al., 2021] have recently shown that deep convolutional neural networks can predict a patient's race based on medical images alone, including the radiographs from the CheXpert dataset. They write that the classification of race based on x-ray images is not possible for trained radiologists, and the study shows that even when seriously compromising the quality of the image, the neural network is still able to detect the race of the patient. It causes concern because it suggests that the models can see patterns not visible to human experts. Thus it is possible that humans unknowingly could build models for disease classification that also exploit these patterns introducing a risk of different treatment based on a patient's race. The authors of [Glocker and Winzeck, 2021] further investigate if there is any indication that deep convolutional neural networks used for disease detection in the CheXpert dataset use sensitive information for classification. They try to answer what kind of information is used by the neural networks and also comment on the work done by [Banerjee et al., 2021] and argue that more extensive analyses are needed.

The findings in the cited articles above inspired us to use the CheXpert dataset in the case study. The dataset is of a decent size, and both the sex and race of the patients are relevant to consider when analyzing potential unfairness in the predictions of a model trained on the dataset.

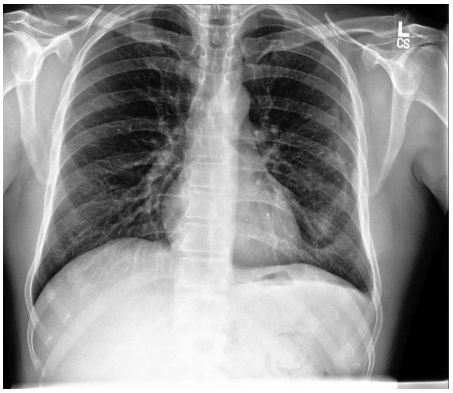


Figure 5.1: Example of a chest radiograph from [Irvin et al., 2019].

The CheXpert dataset is currently used for research in fair machine learning, which makes it interesting to use in order to make our own contribution to this debate. Moreover, many of the articles encouraged further investigation and caution when deploying models for medical image classification. We will act on this encouragement and thus, in this chapter, present the CheXpert dataset, create a model for classification and carry out a fairness analysis focusing on race and sex.

5.1 Dataset

The CheXpert dataset consists of 224,316 chest radiographs of 65,240 patients [Irvin et al., 2019]. The chest radiographs are from Stanford Hospital and were taken between October 2002 and July 2017. Each image comes with 14 labels corresponding to 14 diseases or conditions. The label indicates whether the disease or condition is present, unmentioned, uncertain, or not present. The labels were extracted from the free text radiology reports of the images using an automated, rule-based labeler. An example of a chest radiograph from the CheXpert dataset is seen in figure 5.1.

Since the focus of this thesis is unfairness in binary classification, we choose a single label to predict from the x-ray image. We have chosen *cardiomegaly* (enlargement of the heart) because it is well represented in the dataset with an overall prevalence of 12.24%. Moreover, cardiomegaly was one of the diseases showing differences in AUC for every train-test combination of the genders in the analysis by [Larrazabal et al., 2020] regardless of the chosen model. The 14 labels in the CheXpert dataset are encoded as either present(positive), not-present (negative) or uncertain. [Irvin et al., 2019] showed that the binary encoding *U-zeros*, where uncertain labels are mapped to not-present, resulted in the best model performance for cardiomegaly detection. [Larrazabal et al., 2020] also used the U-zeros uncertainty encoding in their study. Hence, we will use this uncertainty encoding as well.

The CheXpert dataset includes metadata containing the sex and age of the patients. As an addition to the CheXpert data set, the CheXpert Demo Data is published by Stanford University Center for Artificial Intelligence in Medical imaging [Stanford AIMI, 2021]. The Demo Data includes age, gender and the self-reported racial and ethnic identity of 65,401 patients. A total

of 301 patients from the CheXpert dataset were not present in the demographic data. In order to make a fairness analysis of the model predictions, we will use attributes from both the metadata of the CheXpert dataset and the CheXpert Demo Data. The two sources of demographic data are merged using the unique patient id. The patients not included in both datasets are omitted from the fairness analysis.

Patients in the CheXpert dataset can have gotten more than one radiograph taken in the 15-year period from 2002 to 2017, and their age reported in the metadata therefore differs across their recorded x-rays. On the contrary, only one age per patient is recorded in the CheXpert Demo Data. We have chosen to omit 17 observations where the absolute age difference between the age registrations in the datasets exceeds 15 years since this is the maximum possible age difference in the 2002-2017 period.

Please note that we only omit the above-mentioned observations when performing the fairness analysis and not when training the models. This is because the model does not use the metadata or the demographic data, and nothing suggests issues with the specific radiographs. However, discrepancies in the demographic data might impact the results of the fairness analysis.

The sex and the gender of the patients are available in the CheXpert metadata and the Demo data, respectively. We choose to make the fairness analysis based on the recorded sex and not gender because cardiomegaly is a physical condition, and we deem that the biological sex can have a larger influence on the risk of cardiomegaly than gender. There are only three observations where the two attributes do not coincide.

The categories in the *race* attribute are the patients' self-reported racial identities. They fall within 23 different categories, where some apply to only a few individuals. The *ethnicity* attribute contains six categories mainly divided into Hispanic, Non-Hispanic, and not reported. We choose to pre-process the race attribute following the approach in [Glocker and Winzeck, 2021], such that we aggregate the labels to the following four categories: White, Black, Asians, and Other or Unknown. In [Glocker and Winzeck, 2021] and [Banerjee et al., 2021], all individuals whose ethnicity is not Non-Hispanic are excluded. We choose *not* to exclude observations based on the ethnicity attribute since it would exclude almost 1/3 of all available observations. We deemed it most important not to exclude any observations because, unlike in [Banerjee et al., 2021], we do not need high-quality labels for predicting race but instead only for determining groups for the fairness analysis.

5.2 Modelling

We model the presence of cardiomegaly in frontal chest radiographs from the CheXpert dataset using the convolutional neural network DenseNet [Huang et al., 2018]. We use the DenseNet-121 architecture, available on Pytorch Hub (repository = 'pytorch/vision:v0.10.0'), which is pre-trained on ImageNet [Deng et al., 2009]. In order to use the network for binary classification (cardiomegaly versus no cardiomegaly), the final classification layer is replaced with a new classification layer with one output feature. Binary cross entropy is used as the loss function. A benchmark model is implemented following the method in the recent paper [Larrazabal et al., 2020]. The model is trained using ADAM with standard parameters $(\beta_1, \beta_2) = (0.9, 0.999)$ and with an initial learning rate of 0.001 [Kingma and Ba, 2015]. The learning rate is decreased by

Image Aug.	Dropout (p)	Weight decay (λ)	AUC_{val}	Loss_{val}	AUC_{train}	Loss_{train}
None	**0.2**	**0.000**	**0.854**	**0.273**	**0.876**	**0.255**
Simple	0.2	0.000	0.851	0.271	0.874	0.256
Extensive	0.0	0.000	0.851	0.272	0.870	0.260
Simple	*0.0*	*0.000*	*0.850*	*0.272*	*0.874*	*0.257*
Extensive	0.4	0.000	0.849	0.274	0.863	0.265
Extensive	0.2	0.000	0.848	0.273	0.859	0.269
None	0.0	0.000	0.842	0.282	0.869	0.261
Simple	0.4	0.000	0.835	0.283	0.840	0.282
None	0.4	0.000	0.833	0.286	0.816	0.297
None	0.2	0.001	0.830	0.292	0.809	0.300
None	0.0	0.001	0.821	0.291	0.841	0.282
Extensive	0.0	0.001	0.819	0.294	0.820	0.297
Simple	0.0	0.001	0.795	0.307	0.786	0.314
Extensive	0.2	0.001	0.784	0.330	0.755	0.328
Simple	0.2	0.001	0.772	0.322	0.746	0.332
None	0.0	0.010	0.766	0.326	0.742	0.336
Extensive	0.4	0.001	0.753	0.344	0.717	0.343
Simple	0.4	0.001	0.740	0.348	0.721	0.342
None	0.4	0.001	0.719	0.344	0.693	0.350
Simple	0.0	0.010	0.714	0.346	0.685	0.353
Extensive	0.0	0.010	0.685	0.353	0.656	0.360
Simple	0.4	0.010	0.581	0.370	0.567	0.372
Extensive	0.4	0.010	0.574	0.372	0.539	0.376
Simple	0.2	0.010	0.500	0.373	0.485	0.375
None	0.4	0.010	0.500	0.373	0.485	0.375
None	0.2	0.010	0.500	0.373	0.486	0.375
Extensive	0.2	0.010	0.500	0.373	0.485	0.375

Table 5.1: Overview of all models trained on the CheXpert dataset. For each model the type of image augmentation, the dropout probability p, and the amount of weight decay λ is shown. The area under the ROC curve (AUC) and binary cross entropy loss is listed for both the validation and train set.

a factor of 10 when an epoch did not improve the validation loss. The images are reduced to (224×224) pixels and converted into RGB images. The images are flipped horizontally with probability $p = 0.5$.

We train 27 additional models aiming to improve the benchmark model. The additional models introduce dropout, weight decay, more extensive image augmentation, and no image augmentation. We try the dropout values $p = \{0, 0.2, 0.4\}$ and weight decay $\lambda = \{0, 0.01, 0.001\}$. The extended image augmentation includes random horizontal flipping ($p = 0.25$), random affine transformation (degrees $\in [-15, 15]$ and scale $\in [0.9, 1, 1]$), random adjustment of sharpness (sharpness factor $= 2$), and random rotation (degrees $\in [-15, 15]$). The image augmentations are applied sequentially to the image, each with probability $p = 0.25$. All models are trained on

the same split of the data, where 20% of patients were set aside for testing, and the remaining 80% of the patients are split into a training (80%) and validation (20%) set. All images of the same patient are in the same split, ensuring no data leakage between the sets. We verify that the prevalence of cardiomegaly in the training, validation, and test set is similar (train: 12.38%, val: 12.32%, test: 11.70%). We choose the best model as the one with the largest area under the ROC curve (AUC) for the validation set. The test set is set aside and is not looked at until the best model is selected.

Table 5.1 shows the performance of all trained models sorted by decreasing validation AUC. The benchmark model is highlighted using *italics* and the best performing model is highlighted in **bold**. Based on the validation AUC ($\text{AUC}_{val} = 0.854$), the best model is chosen to be the model with dropout $p = 0.2$, no image augmentation, and no weight decay. The best model improves on the benchmark model only slightly, which had a validation AUC of $\text{AUC}_{val} = 0.85$. Generally, we see that the models without weight decay perform the best and using some image augmentation or some dropout seems to improve the model.

The chosen model is used for making the predictions analyzed in the fairness analysis. Since the dataset is unbalanced, and false negatives come with a higher cost than false positives, the threshold used for making the predictions was chosen as the threshold, which led to a false positive rate of 20% on the training set following the method used in [Glocker and Winzeck, 2021]. The threshold, τ, was calculated to be $\tau = 0.1002$. The predictions made on the test set using this threshold are analyzed with respect to fairness in the following section.

5.3 Fairness Analysis

In this section, the best performing DenseNet model is analyzed with respect to fairness for sensitive groups based on sex, race, and a combination of the two. Firstly, we will present an overview of the results, which is followed by three sections. In these sections, we analyze the underlying distribution of the true labels across sensitive groups and present the fairness analysis using BiasBalancer on each of the three possible sensitive attributes.

Table 5.2 shows the weighted misclassification rate (WMR), the weighted misclassification quotient (WMQ), the area under the ROC curve (AUC), and the accuracy computed on the test set for each sensitive subgroup in the three fairness analyses. The WMR has been computed with $w_{FP} = 0.1$, which puts a large emphasis on false negatives. This choice reflects that an undiscovered disease is the worst outcome when using predictive models for diagnosing. The smallest and largest value of WMQ, AUC, and accuracy are highlighted with *italic* and **bold**, respectively. We see that Black patients have the highest WMQ both when analyzing race alone and the combination of race and sex. In the analysis wrt. sex, women have the highest, although very low, WMQ. Generally, the group with the highest accuracy score also has the highest AUC. In two experiments, the best-predicted group also has the smallest WMQ. In the analyses investigating race and the combination of sex and race, the subgroup with the largest WMQ also has the lowest accuracy. However, the AUC for Blacks in the fairness analysis concerning race is the second highest.

Sensitive Group	Subgroup	n	WMR	WMQ %	AUC	Accuracy %
Sex	Female	15636	0.049	**2.3**	**0.857**	**80.2**
	Male	21917	0.048	*0.0*	*0.852*	*78.9*
Race	Asian	3857	0.044	*0.0*	**0.866**	**80.7**
	Black	1956	0.055	**24.1**	0.859	*75.3*
	Other	9763	0.048	9.0	0.853	79.2
	White	21977	0.048	7.4	*0.850*	79.7
Race and Sex	Asian_Female	1681	0.049	19.9	0.846	79.7
	Asian_Male	2176	0.041	*0.0*	**0.879**	**81.6**
	Black_Female	1069	0.054	30.9	0.877	75.8
	Black_Male	887	0.057	**39.7**	*0.823*	*74.7*
	Other_Female	3990	0.050	23.4	0.854	79.2
	Other_Male	5773	0.047	15.0	0.852	79.2
	White_Female	8896	0.047	15.1	0.850	81.2
	White_Male	13081	0.048	17.8	0.850	78.7

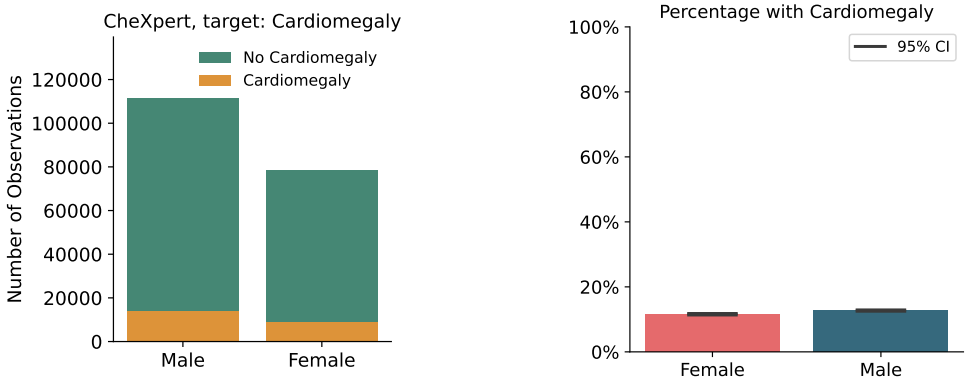
Table 5.2: Table summarizing the first level of the fairness analyses carried out with BiasBalancer using w_{FP} = 0.1. The fairness analyses investigates unfairness in subgroups based on sex, race, and a combination of the two. The bold font indicates the largest value of a measure within an analysis and italic font indicates the smallest.

5.3.1 Sensitive Group: Sex

The authors of [Larrazabal et al., 2020] showed how different proportions of male and female images in the training set of a neural network trained for multi-label prediction on the CheXpert dataset resulted in significantly different performances across the two recorded genders. Because differences across gender likely also mean differences across sex, [Larrazabal et al., 2020] inspired us to analyze the potential unfairness w.r.t. sex when training DenseNet using all data in a standard training-validation-test split scenario.

Figure 5.2c shows a table with the number of observations in the two sexes as well as the proportion of cardiomegaly cases, including a 95% Wilson confidence interval across the entire dataset. The numbers in the table are also visualized in the figures 5.2a and 5.2b. The dataset contains more male observations than female observations, and there is a higher percentage of men with cardiomegaly (12.71%, CI: [12.51%, 12.90%]) compared to women (11.58%, CI: [11.35%, 11.80%]) in the dataset. Note that these confidence intervals are calculated under the assumption of independent observations. However, there are several images from the same patient in the dataset, and these observations are hence dependent. The average number of images per patient is 2.96 (median = 1), while the maximum is 91. Due to this, the confidence intervals presented in this analysis of the CheXpert dataset are too narrow. However, we have decided to include the intervals during the analysis because they still accurately indicate when differences are *not* significant.

Table 5.2 showed the unfairness analysis from the first level of BiasBalancer. Females saw the largest *WMQ*, but only at 2.3%, and females had the highest accuracy and AUC ROC. This could indicate an overall better classifier for female patients, but a larger fraction of false negatives



(a) (b)

Sex	N	Has Cardiomegaly	CI
Male	111595	14181 (12.71%)	[12.51%, 12.90%]
Female	78564	9095 (11.58%)	[11.35%, 11.80%]
Total	190159	23276 (12.24%)	[12.09%, 12.39%]

(c)

Figure 5.2: Number of observations and percentage with presence of cardiomegaly broken down by sex. The figures (a) and (b) visualizes the numbers shown in the table (c).

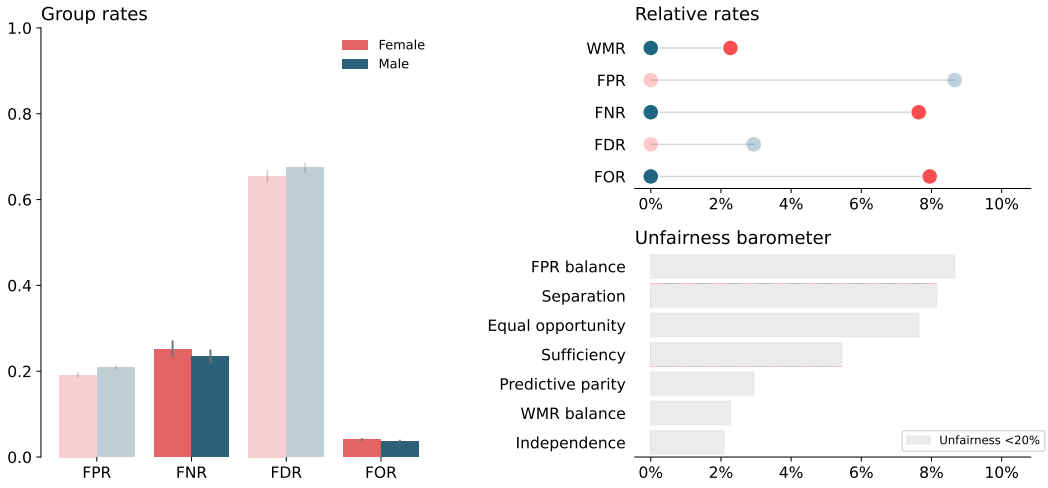


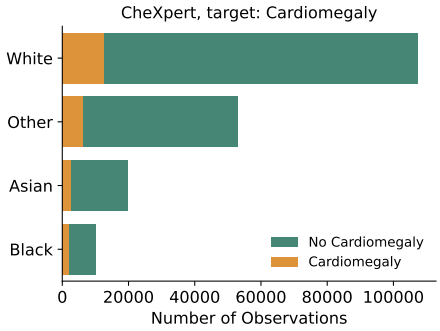
Figure 5.3: The second level visualization of the CheXpert dataset analyzing unfairness with respect to sex.

could be yielding the larger value of *WMQ*. Figure 5.3 shows the results from the second level of BiasBalancer. We see that both sexes have a false positive rate close to 0.2, which is the value used when deciding the threshold based on the training data. The difference between the sexes is small for all rates, and the confidence intervals overlap for all rates except for the false positive rate where the intervals are close to overlapping (Female: 0.190 [0.184, 0.197],

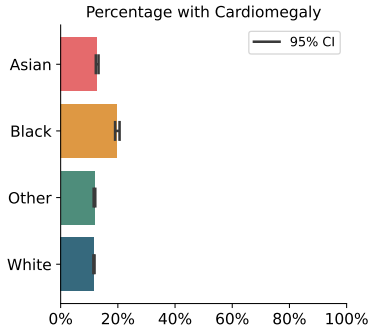
Male: 0.206 [0.200, 0.212]). The relative differences are all smaller than the 20% limit of concern. Based on the analysis, the predictive model does not appear to discriminate based on sex for any of the fairness criteria.

5.3.2 Sensitive Group: Race

The remarkable accuracy of predictive models trained for race prediction from medical x-rays raised the concern that models trained for disease detection might also use racial information in medical images [Banerjee et al., 2021]. Thus, we decided to investigate potential unfairness with respect to race in the trained DenseNet model for cardiomegaly detection. Figure 5.4 shows the number of observations for each race in the entire dataset and the proportion with cardiomegaly, including a 95% confidence interval. The dataset is imbalanced with respect to race, and there are more than 10 times more observations of White patients than Black patients. Moreover, a total of 19.81% of chest radiographs of Black patients are labeled with cardiomegaly compared to the other groups where Asians have the second most incidences (12.76%) followed by the images with race Other (11.82%) and White (11.64%). The only two groups with overlapping confidence intervals are Other and White.



(a)



(b)

Race	N	Has Cardiomegaly	CI
White	107487	12516 (11.64%)	[11.45%, 11.84%]
Other	52990	6264 (11.82%)	[11.55%, 12.10%]
Asian	19636	2506 (12.76%)	[12.30%, 13.24%]
Black	10046	1990 (19.81%)	[19.04%, 20.60%]
Total	190159	23276 (12.24%)	[12.09%, 12.39%]

(c)

Figure 5.4: Number of observations and percentage in the CheXpert dataset with presence of cardiomegaly broken down by race. The figures (a) and (b) visualizes the numbers shown in the table (c).

In the overview in table 5.2, we saw that Blacks see the highest *WMQ* (24.1%) and smallest accuracy, but not the smallest AUC. The high *WMQ* and large AUC indicate that Blacks are potentially seeing a large number of false negatives relative to their group size but have a good classifier based on the group's ROC curve. Figure 5.5 shows the results from the second level of

BiasBalancer. The left subfigure shows the rates are similar for Whites, Asians, and the group Other, while the rates for Blacks stand out with higher false positive rate and false omission rate and lower false negative rate and false discovery rate. Therefore, the predictions violate both the separation criterion (incl. the relaxations equal opportunity and FPR balance) and the sufficiency criterion. Both criteria suggest unfairness toward both Whites and Blacks.

Let's focus on the false negative rate and false omission rate, because a false negative in this medical setting is considered more severe than a false positive. A higher proportion of Blacks than Whites have cardiomegaly, which decreases the false negative rate for Blacks because the false negative rate has the number of true positives in the denominator. The false omission rate has predicted negatives in the denominator. This rate is higher for Blacks, most likely because a higher rate of actual positives should lead to a lower rate of predicted negatives relative to Whites. The pattern is very similar to the pattern seen in the COMPAS data (see figure A.1), where Blacks also had a higher rate of actual positives. However, the setting is different since a false positive was the most severe outcome in the COMPAS analysis because it could lead to bail being unrightfully denied offenders.

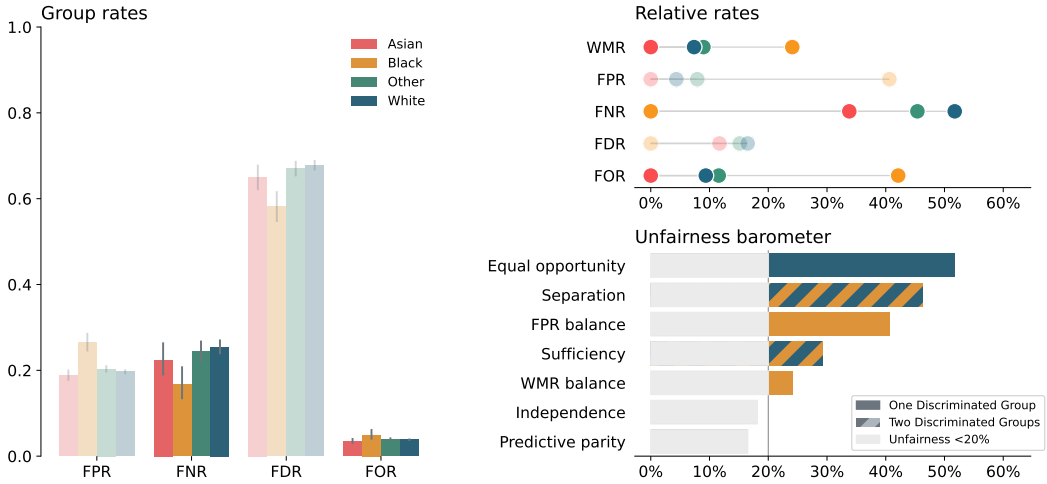


Figure 5.5: The figure shows the second level visualization of the CheXpert dataset with race as the sensitive attribute.

Because the unfairness barometer indicates unfairness above the usual 20% threshold, we will dive further into the analysis using the third level of BiasBalancer. The relative unfairness according to *WMR*-balance is not much larger than 20%, and the violation of this criterion may be caused by a larger amount of false negatives relative to group size for Black patients. The confusion matrices for each subgroup show that Blacks do indeed have a slightly higher percentage of false negatives at 3.1% compared to Asians (2.6%), Whites (2.8%) and Others (2.8%). The matrices are seen in the appendix as figure A.2. It should be noted that the 3.1% false negatives correspond to only 61 images from 52 Black patients.

According to the unfairness barometer, the biggest amount of relative unfairness is toward Whites using the criterion equal opportunity. This means that Whites with cardiomegaly have a smaller probability of being diagnosed correctly than Blacks. This can be further investigated using the group-specific ROC curves, which can also shed light on the *FPR*-balance and separation criteria.

The ROC curves are seen in figure 5.6 including points indicating the chosen classifier with the threshold $\tau = 0.1002$. Recall, that the points should align horizontally (equal TPR) for equal opportunity to be satisfied, align vertically (equal FPR) for FPR-balance to be satisfied, and lie in the intersection between the four ROC curves for separation to be satisfied (equal FPR *and* TPR). The predictions for White patients are unfair due to the lower true positive rate (corresponding to a higher false negative rate), and predictions for Black patients are unfair due to the higher false positive rate. The separation criterion thus points towards two different groups being unfairly treated. Since false negatives are deemed more serious than false positives, one could argue for putting more emphasis on the part of the separation criterion suggesting unfair predictions for White patients. Apart from the ROC curve for Asian patients, the curves are overall very similar. This means it is possible to find group-specific thresholds such that separation is closer to being satisfied. The disparity could be mitigated by increasing the threshold for Blacks, which would move the rates for Blacks closer to the rates of the remaining races. This mitigation possibility will be further discussed in section 5.4.

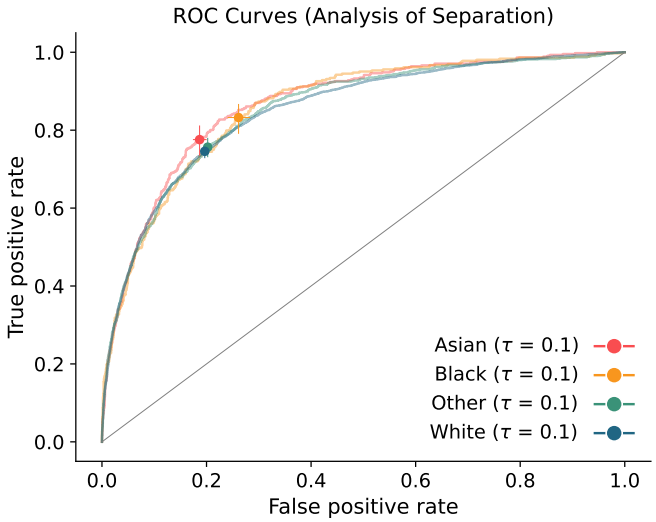


Figure 5.6: ROC Curves by each race. The point indicates the chosen DenseNet model corresponding to a threshold of $\tau = 0.1$, which yielded a FPR of 0.2 when evaluating on the entire training set.

The unfairness barometer of BiasBalancer also shows a violation of the sufficiency criterion, where both Whites and Blacks experience unfair treatment. Sufficiency is violated when the relative difference in FDR and FOR rates is too large across the sensitive subgroups. The relaxation, predictive parity, only depends on the false discovery rate, and it is not seriously violated in the unfairness barometer. The absolute rates show that Blacks do have a smaller false discovery rate (FDR), but because the FDR is very high for all subgroups, the relative difference seen in the upper right plot of figure 5.5 becomes smaller than 20%. Thus, it is the relative difference in false omission rates (FOR) that drives the violation of sufficiency. The false omission rate is generally low for all subgroups, but it is 40% larger for Blacks compared to Asians. Hence, a substantially higher fraction of the predicted negatives for Blacks are false negatives. Since false negatives are deemed more unfavorable than false positives, one could argue that overall the sufficiency criterion points toward unfairness towards Blacks.

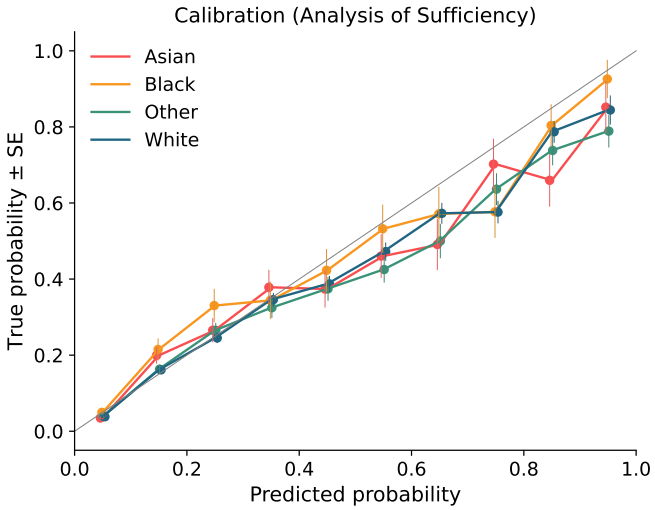


Figure 5.7: Calibration plot by race on the predictions of cardiomegaly using DenseNet. The vertical lines indicates the interval created by one standard error from the estimated true probability.

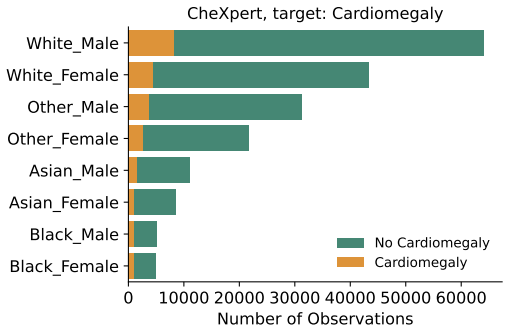
A calibration plot shows the influence of the threshold on the false omission rate and false discovery rate. Figure 5.7 shows that the calibration curves are somewhat similar for patients of all races. The curves lie close to the diagonal line, but generally, the true probability lies below the predicted probability for predicted probabilities larger than 0.4. It indicates that the model generally overpredicts the risk of cardiomegaly for all races. Since the threshold was set to $\tau = 0.1002$, the leftmost point of each line corresponds to the predicted negatives, while the remaining observations were predicted as positive. The higher false omission rate for blacks, and thereby the violation of sufficiency, is seen in the plot, when looking closely, by the leftmost point for Blacks being slightly higher than the others. Blacks have a larger true probability of having cardiomegaly compared to their predicted probability for predicted scores between 0.1 and 0.3. Consequently, increasing the threshold, which was previously discussed, could exacerbate the disparity in false omission rates for Blacks.

In summary, the conclusion of the unfairness analysis with respect to race depends on whether most weight is put on the sufficiency or separation criterion since the two criteria point in different directions. This rather complicated example shows how the different criteria can yield different results.

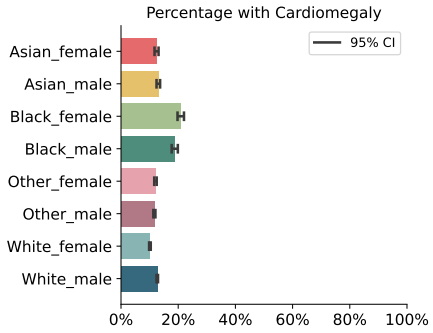
5.3.3 Sensitive Group: Race and Sex

The fairness analysis of sex showed no significant discrimination against any of the sexes, while the analysis of race showed that the result heavily depended on the fairness criterion used. Since both information on sex and race is available, we found it interesting to further divide the sensitive groups such that they include both sex and race. Figure 5.8 shows the number of observations in each category and the fraction of cardiomegaly cases. A larger fraction of men than women have cardiomegaly for all races except for Blacks, and the highest prevalence of all

groups is for Black women (20.87%). The confidence intervals of the fraction of cardiomegaly cases overlap for men and women of the same race except for Whites.



(a)



(b)

Race_sex	N	Has Cardiomegaly	CI
White_Male	64168	8134 (12.68%)	[12.42%, 12.94%]
White_Female	43320	4382 (10.12%)	[9.83%, 10.40%]
Other_Male	31203	3640 (11.67%)	[11.31%, 12.03%]
Other_Female	21787	2624 (12.04%)	[11.62%, 12.48%]
Asian_Male	11133	1451 (13.03%)	[12.42%, 13.67%]
Asian_Female	8503	1055 (12.41%)	[11.72%, 13.13%]
Black_Male	5093	956 (18.77%)	[17.72%, 19.87%]
Black_Female	4955	1034 (20.87%)	[19.76%, 22.02%]
Total	190162	23276 (12.24%)	[12.09%, 12.39%]

(c)

Figure 5.8: Number of observations and percentage with presence of cardiomegaly broken down by race and sex. The figures (a) and (b) visualizes the numbers shown in the table (c).

Figure 5.9 shows the level 2 visualization from BiasBalancer when considering both sex and race. Most noticeable is that Black men have a much higher false negative rate and false discovery rate compared to Black women. The rates for Black men are higher than the rates for Blacks overall, as seen in figure 5.5. This difference especially stands out because the prevalence for Black men and women was similar and with overlapping confidence intervals (figure 5.8). Due to this surprisingly large difference, the prevalence of cardiomegaly in the test set compared to the training set for Black men was investigated. The prevalence of cardiomegaly in the test set for Black men was 14.32% (CI: [12.17%, 16.78%]), which deviates significantly from the training data, where the prevalence was 18.95% (CI: [17.65%, 20.32%]). The big difference in prevalence in the training and test data is problematic since disparities seen in the unfairness analysis may largely be attributed to this distributional difference and not reflect actual disparities in the predictive model. Therefore, we will not dive further into the unfairness analysis based on the combination of race and sex. The underlying distribution of the prevalence in the test and training splits can be seen in table A.3 in the appendix. Note that this issue is not present for the analyses with respect to only race and only sex.

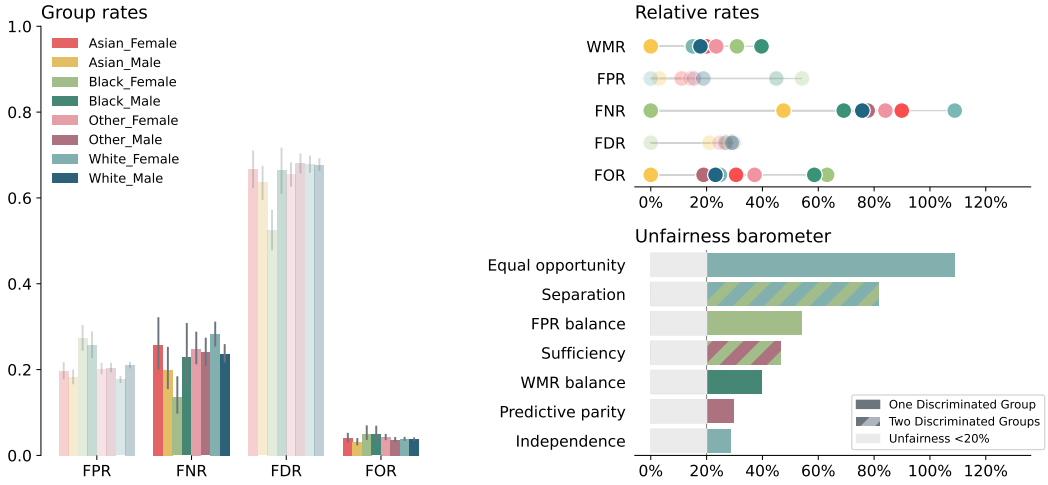


Figure 5.9: The second level visualization of the CheXpert dataset, where fairness is analyzed with BiasBalancer and sensitive groups based the sensitive attribute encoding both sex and race.

5.4 Discussion of Case Study

This section discusses the results of the CheXpert fairness analysis, the chosen threshold, and different mitigation strategies in the context of the CheXpert predictive model.

5.4.1 Interpretation of results

The fairness analysis investigating fairness across the sexes showed no unfairness of concern. The authors of [Larrazabal et al., 2020] showed that training on severely unbalanced data concerning sex seriously compromises the quality of the predictions for the poorly represented sex. Thus, representative data is crucial for the performance of a predictive algorithm and, ultimately, fairness. The CheXpert dataset is slightly imbalanced with 58.6% men and 41.4% women, but the imbalance does not seem to seriously affect the predictions for women. This suggests that there might be enough diversity in the dataset to train a good predictive model for both men and women.

The result of the fairness analysis investigating race is less straightforward to interpret. In the analysis, false negatives were considered the most unfavorable outcome for an individual. Given this setting, the fairness criteria based on the false negative rate and false omission rate are most concerning. Based on this, the separation and sufficiency criteria suggested disfavoring of Whites and Blacks, respectively. The proposed overall unfairness measure in this thesis, *WMR*-balance, showed discrimination towards Blacks, which reflects that this is the group with the highest proportion of false negatives. Moreover, the prediction accuracy was lowest for Blacks, which could be caused by the relatively small amount of Blacks in the dataset ($\approx$ 10 times fewer than Whites). Thus, many fairness measures suggest that Blacks receive the worst predictions, but on the other hand, the single most violated fairness criterion is equal opportunity, which suggests that Whites receive unfair predictions. Moreover, one could argue that the false negative rate is

most important, because it measures the proportion of individuals with cardiomegaly where the condition went undetected. However, the high relative false negative rate for Whites is caused by the low rate seen for Blacks, which is the reference group. The results of this fairness analysis is an opportunity to contemplate what makes the different definitions of unfairness point towards different unfairness issues in the predictions and how these issues affect patients in the specific case of cardiomegaly detection.

The last fairness analysis with sensitive groups based on the combination of sex and race was inconclusive because the prevalence of cardiomegaly for Black males in the test set did not reflect the prevalence in the training data. In order to perform a fairness analysis based on both sex and race with the CheXpert dataset, it would be necessary to stratify the sampling of the train, validation, and test set such that all combinations of race and sex had a similar prevalence of cardiomegaly in the sets. This emphasizes that caution should be taken when analyzing datasets with small sensitive groups, further highlighting the need for representative data.

5.4.2 Predictive Model

The results of the fairness analysis naturally depend on the predictive model. In this section, the choices made during the model development are discussed. Only roughly 12% of the images in the dataset are positive observations, which means the target variable is imbalanced, and the predictive model does not see many cases of cardiomegaly compared to cases without cardiomegaly. One way to handle this is to assign a higher weight to the positive samples with cardiomegaly during the training phase such that the model is more exposed to radiographs with instances of cardiomegaly.

Instead, the class imbalance was handled by adapting the threshold. The classification threshold was set based on the false positive rate such that $FPR = \frac{FP}{N} = 0.2$ across the training set. Consider a scenario where individuals classified with cardiomegaly get called in for further examinations. By choosing a threshold corresponding to $FPR = 0.2$, we thus allow 20% of the people without cardiomegaly to be called into additional examinations with the hope of consequently offering further examinations to a higher proportion of the patients with cardiomegaly. Instead of choosing the threshold based on the false positive rate, the threshold could be based on the false discovery rate, $FDR = \frac{FP}{PP}$. Choosing the threshold such that FDR is at a certain level would control how large a proportion of the people we call in for extra examination is healthy - this could be important if extra examinations require scarce resources. The threshold choice affects the model's predictions and fairness, and it is therefore important that data scientists cooperate with people with domain knowledge in the specific task area, e.g., doctors and hospital staff, when choosing the threshold. We did not collaborate with such experts, and instead, the choice of using FPR was inspired by the work in [Glocker and Winzeck, 2021], where race-specific thresholds were chosen such that $FPR_a = 0.2 \ \forall a \in A$. We did not choose race or sex-specific thresholds because the objective was to analyze the fairness of a model trained without considering sensitive groups.

5.4.3 Mitigation of Unfairness

Identifying and quantifying unfairness in predictive models is not the only research area within machine learning fairness. Another area is the mitigation of such biases and unfairness. Mitigation is a vast research field, and thoroughly explaining and using mitigation techniques is outside the scope of this thesis. However, it is relevant to briefly consider them in the context of the CheXpert case study as we saw unfairness in our model when analyzing it using BiasBalancer w.r.t. race. The available mitigation techniques mainly fall into three categories: Pre-processing, in-processing, and post-processing [Barocas et al., 2019].

Pre-processing consists of various data transformations applied before training a model on the data. One of these methods is reweighing, which weighs each group differently in order to ensure equal base rates and can thus be used to mitigate violation of the independence criterion [Kamiran and Calders, 2012]. We did not see any issues w.r.t. the independence criterion, and one might argue that, given the different prevalence of cardiomegaly across both sex and race in the dataset, this might not be a desirable approach.

An example of in-processing approaches is to constrain the optimization at training time [Barocas et al., 2019]. Recently, different approaches to using adversarial learning to mitigate biases have been developed. [Zhang et al., 2018] uses a neural network for the prediction task and the adversary to model the protected attribute. The optimization is carried out using an objective to maximize the accuracy of the prediction task while minimizing the ability to predict the sensitive attribute. [Madras et al., 2018] also uses adversarial learning by using an encoder structure to learn latent representations of the input data such that the target can be predicted by the classifier from the latent representation, but the adversary cannot predict the sensitive attribute from this latent representation. Such methods to mitigate bias could have been very interesting to investigate and evaluate using BiasBalancer. Especially since [Banerjee et al., 2021] found that algorithms could predict race from the CheXpert chest radiographs in various settings and resolutions of the images. Investigating the results of using such in-processing mitigation techniques is left as potential future work.

The last approach to mitigation is post-processing. Post-processing does not require re-training the model, which is an advantage if the training process is complex or we do not have access to it. [Hardt et al., 2016] presented a post-processing step along with the fairness criteria equalized odds and equal opportunity. The criteria can be satisfied by choosing group-specific thresholds. [Vyas et al., 2020] criticizes the use of race-specific correction of models for medical purposes and encourages researchers to think about whether such corrections relieve or exacerbate existing biases in society. The race-specific ROC curves showed that the current threshold results in a larger true positive rate for Blacks compared to other races. This means changing the threshold to satisfy separation would result in fewer correctly classified cases of cardiomegaly in a historically discriminated group.

The most common source of unfairness in predictive models is that minority groups are poorly represented in the training data. Hence, the best mitigation approach is to obtain representative data of high quality. This is not an easy task and may even be impossible in some cases. The results indicated that the small number of observations of Black individuals could have affected both the fairness analysis and accuracy for the group. Only 7.1% of the population in the state of California, which is where Stanford hospital is located, identify as African American [United States Census Bereau, 2021]. This is likely to impact the patient demography in the

CheXpert dataset and potentially the fairness of the model. Representative data is crucial when algorithms are distributed globally as the risk of them influencing the lives of the individuals poorly represented in the training data increases.

Chapter 6

Discussion

Throughout this thesis, we have seen how BiasBalancer enables making more nuanced fairness analyses of predictive algorithms. The tool comes with some design choices discussed in the first discussion section. The tool can not stand alone, and the inherent limitations of BiasBalancer are discussed in the second section.

6.1 Design Choices

The weighted misclassification quotient is constructed such that a perfect classifier is always evaluated to be fair. This behavior relies on the strong assumption that the provided labels are correct and fair. This may not always be the case, but the assumption is a necessary condition for all observational fairness criteria. Moreover, the weighted misclassification rate is a function of the chosen false positive weight, w_{FP}, which is allowed to be in the range $[0, 1]$. Hence, a misclassification is considered an unfavorable outcome. There may be contexts in which this is not always true. For example, an unwarranted college admission given to an unqualified individual could improve the individual's life drastically. However, one could argue that if the admission was a good outcome for the individual, the individual was actually qualified, and the prediction was therefore correct. Hence, this design choice again relies on the assumption that the outcome labels are considered correct and fair. Choosing a common false positive weight for all subgroups also assumes that every group values the potential outcomes similarly. Different people are likely to agree on the severity of a false positive when using models to predict potentially life-threatening situations, but the task becomes more difficult in situations where it is less clear what the negative outcome is.

The criteria used in the unfairness barometer are originally formulated as hard constraints enforcing exact equality between groups. When converting the criteria to a continuous, instead of binary, measure, we have chosen to use the relative rate. For a rate r and sensitive group a, the relative rate is calculated as $RR_a(r) = \frac{r_a - r_{min}}{r_{min}} \cdot 100\%$, where $r_{min} = \min_{a \in A} r_a$. Another natural choice would have been to use differences between the group-wise rates calculated simply as $\text{diff}_a(r) = r_a - r_{min}$. When media describes dissimilarities between two groups, percentages or ratios are often used to convey unfairness. The article about the COMPAS algorithm contains

an example of this [Angwin et al., 2016]: "Black defendants were still 77 percent more likely to be pegged as at higher risk of committing a future violent crime and 45 percent more likely to be predicted to commit a future crime of any kind." We have chosen to express the degree of disparity in a similar manner because we perceive percentages to be intuitive to understand. However, this also means that one should be aware that small absolute differences in the rates can yield large relative differences. It is especially relevant if only a single subgroup has a weighted misclassification rate of zero while the remaining groups have positive values. In that case, the weighted misclassification quotient becomes very large and can blow up for the remaining groups.

BiasBalancer measures unfairness relative to the group with the most favorable rate value. This is in contrast to other available tools where unfairness typically is calculated relative to a privileged group specified by the user [Bellamy et al., 2019, Johnson et al., 2020, Bird et al., 2020, Saleiro et al., 2019]. We have decided to always measure unfairness relative to the group with the most favorable rate because this increases the interpretability since all calculated relative rates will be non-negative. Moreover, there will be situations where the privileged group is unknown before using the toolkit. Choosing the reference group in this way is not ideal when all groups have similar rates, except for one group with a significantly lower rate. The "outlier" group then becomes the reference group, and the impression would be that all but one group receive unfair predictions instead of the arguably more correct interpretation that the predictions favor the outlier group over the remaining others. For this reason, the second-level visualization includes both the actual rates and the derived relative rates to supplement the unfairness barometer. In the unfairness barometer, we choose to show the maximum relative rate along with the color of the corresponding sensitive group. We have chosen to highlight the group with the maximum value because it increases the interpretation of the barometer. It depicts when a single group is treated unfairly according to several measures, and it clarifies that the presented fairness criteria seldom agree, which means we should be cautious when choosing a single criterion over another. We think it is important to highlight the group with the most unfair predictions regardless of the size of the group. Whenever a criterion is composed of two criteria, the mean of the value for each criterion is used. The barometer shows this using colored stripes of the bars. We made this choice to showcase that not only can the different criteria disagree, but a single criterion, dependent on two rates, can also point toward disfavoring of different groups.

BiasBalancer does not include any tools for mitigating potential unfairness found in the predictions made by the predictive algorithm because mitigation algorithms were outside the scope of this thesis and because many of the existing fairness toolkits already include a large suite of mitigation algorithms [Bird et al., 2020, Bellamy et al., 2019].

Fairness in machine learning is a relatively new and very active field of research. Because of this, several different terms are used in articles to describe the same concept, and some fairness definitions vary slightly from author to author. We have chosen to use the terminology in the book *Fairness and Machine Learning* [Barocas et al., 2019] for both this report and in BiasBalancer. This relatively new book written by prominent researchers within the field covers a wide range of fairness concepts. It gathers existing material and is a useful reference for those interested in fairness in machine learning.

6.2 Limitations

All fairness criteria in BiasBalancer are based on the assumption that the provided labels represent the ground truth and are fair. This will not be the case in many datasets since labels are prone to simple errors, possibly reflect historical biases in society, and can be a poorly chosen proxy for the outcome of interest. If the assumption of correct and fair labels is not satisfied, the results from BiasBalancer are likely to be flawed or even completely misleading. Domain knowledge and knowledge of how the data is collected is needed to correct erroneous labels and hence cannot be performed using any general-purpose tool, including BiasBalancer. Therefore, it is important to assess the validity of the labels before using BiasBalancer, especially when the algorithms are used in the real world.

When using BiasBalancer, care needs to be taken when the dataset consists of few observations or when the dataset includes sensitive groups with few observations. Just like in other analyses of data with limited observations, events can occur by chance. For this reason, the visualizations and data output from BiasBalancer contain confidence intervals when possible. These confidence intervals are calculated under the assumption of independent observations. When this assumption does not hold, the true confidence intervals are at least as wide as the ones produced by BiasBalancer.

The unfairness barometer in BiasBalancer only includes observational fairness criteria, also called group fairness or statistical measures, and the section 2.2.6 motivates this choice. Including only observational fairness criteria in the unfairness barometer naturally comes with limitations. Observational fairness criteria only depend on the joint distribution of the predictions, sensitive attributes, true labels, and the data features used for classification. However, this joint distribution typically does not contain all knowledge available in a given context, and hence observational fairness criteria assess the fairness of the predictive algorithm based on limited information. There are examples of scenarios with an identical joint distribution where one could be fair while the other would be unfair due to different causal structures in the data [Barocas et al., 2019, p. 57-61].

BiasBalancer makes it easier to make nuanced and comprehensive analyses of unfairness in a predictive model. However, the analysis from the tool cannot and should not stand alone. Building fair predictive algorithms is not a straightforward task, and many considerations have to be taken into account. Therefore, fair machine learning is difficult to achieve when only including model developers with mathematical or computer science backgrounds in the process. [Binns, 2017] writes about relevant lessons from political philosophy in the context of fair machine learning, and it is inevitable that the road to fairer algorithms not only includes the model developers. In the process, we also need to include individuals with extensive knowledge of the domain in question, sociologists, philosophers, or others with knowledge of the workings of ethics, discrimination, and the effects of the algorithmic predictions on individuals. We also can not rely on companies to make fairness the top priority and therefore need legislation ensuring that algorithms meet some commonly agreed upon standards.

6.3 Outlook

BiasBalancer currently only includes observational fairness criteria, and it could be beneficial to integrate other types of fairness definitions, such as individual or causal fairness criteria, into the toolkit. This would greatly improve the toolkit because it could further broaden the fairness analysis obtained using BiasBalancer. It could also be interesting to extend BiasBalancer such that it can aid the user in finding appropriate mitigation techniques for any potentially found unfairness. Care should be taken to ensure that such an extension does not facilitate "automated" fairness analyses and mitigation but instead require the user to think carefully about which kind of unfairness needs mitigation and which mitigation technique is best suited for the task.

The analysis performed by BiasBalancer is constrained to only look for unfairness across the sensitive attribute specified by the user. Algorithms for finding subgroups receiving unfavorable predictions based on some fairness metric exist, and incorporating such an algorithm into BiasBalancer to find relevant sensitive attributes could improve the tool [Zhang and Neill, 2016].

In this thesis, we have showcased and tested BiasBalancer on a wide variety of predictive algorithms. Using BiasBalancer in practice generated many ideas for improving the toolkit, and these ideas were added to the toolkit along the way. Using BiasBalancer on other algorithms, using different sensitive groups, or constructing synthetic data designed to test the toolkit could generate new ideas for improvement. A concrete idea of such improvement is to address the exploding relative rates if a single subgroup has a value of zero in one of the rates. To alleviate this issue, one could consider the idea of adding a threshold for when the absolute rates become so small that they should not be a concern. However, choosing a suitable threshold would require collaboration with, e.g., a philosopher and someone with domain knowledge.

Chapter 7

Conclusion

In this report, we presented the Python toolkit BiasBalancer used for making nuanced and comprehensive fairness analyses of predictive algorithms for binary classification problems. BiasBalancer creates a fairness analysis in three levels, where each level increasingly nuances the fairness analysis. The first level calculates the proposed fairness metric weighted misclassification quotient, which allows for a unified assessment of unfairness, taking the severity of false positives relative to false negatives into account. The second level visualization gives a comprehensive overview of disparities across sensitive groups, including a barometer showing violations of several fairness criteria. The third-level methods enable further investigation into potential unfairness identified in level two. The use of BiasBalancer was showcased on four example datasets, including canonical fairness datasets concerning predictions of credit score and risk of recidivism.

The medical case study consisted of an in-depth fairness analysis of a deep convolutional neural network, which we built and optimized without taking fairness considerations into account. The prediction network was trained to predict the presence of the heart condition cardiomegaly in chest radiographs from the CheXpert dataset. The fairness analysis using BiasBalancer showed no indications of unfair predictions with respect to sex. When considering race, the fairness analysis using BiasBalancer suggested unfair treatment of both Whites and Blacks depending on the fairness criterion used. This fairness analysis demonstrated how different fairness criteria potentially lead to different conclusions and showed how BiasBalancer helped uncover all these conclusions.

Central fairness criteria are mutually exclusive under common conditions, which means optimizing to satisfy one criterion can exacerbate unfairness according to another. BiasBalancer facilitates nuanced fairness analyses taking several fairness criteria into account, thereby enabling the user to get a fuller overview of the potential interactions between the criteria. Fair machine learning is a new and fast-developing research area, which means performing fairness analyses can be a daunting challenge for model developers. The level structure of BiasBalancer makes fairness analysis more accessible for model developers by guiding the user into increasingly complex analyses. We hope BiasBalancer can make fairness analyses based on more than a single criterion more accessible for model developers and thereby encourage the use of such detailed fairness analyses.

Bibliography

[Akiba et al., 2019] Akiba, T., Sano, S., Yanase, T., Ohta, T., and Koyama, M. (2019). Optuna: A Next-generation Hyperparameter Optimization Framework. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2623–2631.

[Angwin et al., 2016] Angwin, J., Larson, J., Mattu, S., and Kirchner, L. (2016). Machine Bias. *propublica.org*. (Date Accessed 2021-09-17).

[Banerjee et al., 2021] Banerjee, I., Bhimireddy, A. R., Burns, J. L., Celi, L. A., Chen, L.-C., Correa, R., Dullerud, N., Ghassemi, M., Huang, S.-C., Kuo, P.-C., Lungren, M. P., Palmer, L., Price, B. J., Purkayastha, S., Pyrros, A., Oakden-Rayner, L., Okechukwu, C., Seyyed-Kalantari, L., Trivedi, H., Wang, R., Zaiman, Z., Zhang, H., and Gichoya, J. W. (2021). Reading Race: AI Recognises Patient's Racial Identity In Medical Images. *arXiv e-prints*, page arXiv:2107.10356.

[Barenstein, 2019] Barenstein, M. (2019). ProPublica's COMPAS Data Revisited. *arXiv e-prints*.

[Barocas et al., 2019] Barocas, S., Hardt, M., and Narayanan, A. (2019). *Fairness and Machine Learning*. fairmlbook.org.

[Bellamy et al., 2019] Bellamy, R. K., Mojsilovic, A., Nagar, S., Ramamurthy, K. N., Richards, J., Saha, D., Sattigeri, P., Singh, M., Varshney, K. R., Zhang, Y., Dey, K., Hind, M., Hoffman, S. C., Houde, S., Kannan, K., Lohia, P., Martino, J., and Mehta, S. (2019). AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM Journal of Research and Development*, 63(4-5).

[Binns, 2017] Binns, R. (2017). Fairness in Machine Learning: Lessons from Political Philosophy. In *Proceedings of Machine Learning Research 81*, pages 1–11.

[Bird et al., 2020] Bird, S., Dudík, M., Edgar, R., Horn, B., Lutz, R., Milan, V., Sameki, M., Wallach, H., and Walker, K. (2020). Fairlearn: A toolkit for assessing and improving fairness in AI. Technical report, Microsoft.

[Bishop, 2006] Bishop, C. (2006). *Pattern Recognition and Machine Learning*. Springer.

[Bogen and Rieke, 2018] Bogen, M. and Rieke, A. (2018). Help wanted: an examination of hiring algorithms, equity, and bias. Technical report, Upturn.

[Bohr and Memarzadeh, 2020] Bohr, A. and Memarzadeh, K. (2020). *The rise of artificial intelligence in healthcare applications*. Academic Press.

[Brock, 2021] Brock, T. (2021). Credit Scoring. https://www.investopedia.com/terms/c/credit_scoring.asp. (Date Accessed 2021-11-04).

[Brown et al., 2001] Brown, L. D., Cai, T. T., and DasGupta, A. (2001). Interval Estimation for a Binomial Proportion. *Statistical Science*, 16(2):101–133.

[Centre d'Estudis Jurídics i Formació Especialitzada, 2015] Centre d'Estudis Jurídics i Formació Especialitzada (2015). Recidivism in juvenile justice Description template of the variables. http://cejfe.gencat.cat/en/recerca/opendata/jjuvenil/reincidencia-justicia-menors/.

[Centre d'Estudis Jurídics i Formació Especialitzada, 2016] Centre d'Estudis Jurídics i Formació Especialitzada (2016). Recidivism in juvenile justice. http://cejfe.gencat.cat/en/recerca/opendata/jjuvenil/reincidencia-justicia-menors/. (Date Accessed 2021-09-06).

[Chowdhury, 2021] Chowdhury, R. (2021). Sharing learnings about our image cropping algorithm. https://blog.twitter.com/engineering/en_us/topics/insights/2021/sharing-learnings-about-our-image-cropping-algorithm. (Date Accessed 2021-12-09).

[Deng et al., 2009] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.

[Directorate-General for Communications Networks Content and Technology, 2021] Directorate-General for Communications Networks Content and Technology (2021). Regulatory framework proposal on artificial intelligence.

[Dua and Graff, 2017] Dua, D. and Graff, C. (2017). UCI Machine Learning Repository. http://archive.ics.uci.edu/ml/index.php.

[Dwork et al., 2012] Dwork, C., Hardt, M., Pitassi, T., Reingold, O., and Zemel, R. (2012). Fairness Through Awareness. *ITCS 2012 - Innovations in Theoretical Computer Science Conference*, pages 214–226.

[Esteva et al., 2017] Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., and Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542. (Date Accessed 2021-12-09).

[Fuglsang-Damgaard and Zinck, 2021] Fuglsang-Damgaard, C. A. and Zinck, E. (2021). Fairness oriented interpretability of predictive algorithms - Github. https://github.com/elisabethzinck/Fairness-oriented-interpretability-of-predictive-algorithms. (Date Accessed 2021-01-14).

[Fuglsang-Damgaard and Zinck, 2022] Fuglsang-Damgaard, C. A. and Zinck, E. (2022). BiasBalancer Documentation. https://elisabethzinck.github.io/Fairness-oriented-interpretability-of-predictive-algorithms/html/biasbalancer.html. (Date Accessed 2021-01-14).

[Glocker and Winzeck, 2021] Glocker, B. and Winzeck, S. (2021). Algorithmic encoding of protected characteristics and its implications on disparities across subgroups.

[Hardt et al., 2016] Hardt, M., Price, E., and Srebro, N. (2016). Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems*, pages 3323–3331.

[Hastie et al., 2009] Hastie, T., Friedman, J. H., and Tibshirani, R. (2009). *The Elements of Statistical Learning : Data Mining, Inference, and Prediction.* Springer, 2nd edition.

[Hinton et al., 2012] Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv e-prints*, pages 1–18.

[Huang et al., 2018] Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. (2018). Densely connected convolutional networks. In *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, pages 2261–2269.

[Ioffe and Szegedy, 2015] Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *32nd International Conference on Machine Learning, ICML 2015*, 1:448–456.

[Irvin et al., 2019] Irvin, J., Rajpurkar, P., Ko, M., Yu, Y., Ciurea-Ilcus, S., Chute, C., Marklund, H., Haghgoo, B., Ball, R., Shpanskaya, K., Seekins, J., Mong, D. A., Halabi, S. S., Sandberg, J. K., Jones, R., Larson, D. B., Langlotz, C. P., Patel, B. N., Lungren, M. P., and Ng, A. Y. (2019). CheXpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *33rd AAAI Conference on Artificial Intelligence, AAAI 2019, 31st Innovative Applications of Artificial Intelligence Conference, IAAI 2019 and the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019*, pages 590–597.

[Johnson et al., 2020] Johnson, B., Bartola, J., Angell, R., Keith, K., Witty, S., Giguere, S. J., and Brun, Y. (2020). Fairkit, Fairkit, on the Wall, Who's the Fairest of Them All? Supporting Data Scientists in Training Fair Models. In *arXiv e-prints*.

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

[Kingma and Ba, 2015] Kingma, D. P. and Ba, J. L. (2015). Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*.

[Klare et al., 2012] Klare, B. F., Burge, M. J., Klontz, J. C., Vorder Bruegge, R. W., and Jain, A. K. (2012). Face recognition performance: Role of demographic information. *IEEE Transactions on Information Forensics and Security*, 7(6):1789–1801.

[Larrazabal et al., 2020] Larrazabal, A. J., Nieto, N., Peterson, V., Milone, D. H., and Ferrante, E. (2020). Gender imbalance in medical imaging datasets produces biased classifiers for computer-aided diagnosis. *Proceedings of the National Academy of Sciences of the United States of America*, 117(23):12592–12594.

[Liu et al., 2019] Liu, L. T., Simchowitz, M., and Hardt, M. (2019). The implicit fairness criterion of unconstrained learning. In *36th International Conference on Machine Learning, ICML 2019*, volume 2019-June, pages 7146–7155.

[Madras et al., 2018] Madras, D., Creager, E., Pitassi, T., and Zemel, R. (2018). Learning adversarially fair and transferable representations. *35th International Conference on Machine Learning, ICML 2018*, 8:5423–5434.

[Mehrabi et al., 2019] Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., and Galstyan, A. (2019). A Survey on Bias and Fairness in Machine Learning. *ACM Computing Surveys*.

[Nielsen, 2015] Nielsen, M. A. (2015). *Neural Networks and Deep Learning*. Determination Press, (Date Accessed 2021-01-12).

[Northpointe, 2015] Northpointe (2015). Practitioner's Guide to COMPAS Core. Technical report, Northpointe.

[Northpointe Inc., 2011] Northpointe Inc. (2011). Risk Assesment. https://www.documentcloud.org/documents/2702103-Sample-Risk-Assessment-COMPAS-CORE.html.

[Obermeyer et al., 2019] Obermeyer, Z., Powers, B., Vogeli, C., and Mullainathan, S. (2019). Dissecting racial bias in an algorithm used to manage the health of populations. *Science*, 366(6464):447–453.

[Paszke et al., 2019] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). PyTorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32(NeurIPS).

[Pedregosa et al., 2011] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, É. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(85):2825–2830.

[Pleiss et al., 2017] Pleiss, G., Raghavan, M., Wu, F., Kleinberg, J., and Weinberger, K. Q. (2017). On fairness and calibration. *Advances in Neural Information Processing Systems*, 2017-Decem(Nips):5681–5690.

[Pro-publica, 2017] Pro-publica (2017). compas-analysis. https://github.com/propublica/compas-analysis. (Date Accessed 2021-09-21).

[Rajpurkar et al., 2017] Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Langlotz, C., Shpanskaya, K., Lungren, M. P., and Ng, A. Y. (2017). CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning. *arXiv e-prints*, 1711.05225.

[Saleiro et al., 2019] Saleiro, P., Kuester, B., Hinkson, L., London, J., Stevens, A., Anisfeld, A., Rodolfa, K. T., and Ghani, R. (2019). Aequitas: A Bias and Fairness Audit Toolkit. *arXiv e-prints*, 1811.05577.

[Stanford AIMI, 2021] Stanford AIMI (2021). CheXpert Demo Data. https://stanfordaimi.azurewebsites.net/datasets/192ada7c-4d43-466e-b8bb-b81992bb80cf. (Date Accessed 2021-11-23).

[United States Census Bereau, 2021] United States Census Bereau (2021). CALIFORNIA: 2020 Census. https://www.census.gov/library/stories/state-by-state/california-population-change-between-census-decade.html. (Date Accessed 2022-01-04).

[U.S. Equal Employment Opportunity Commission, 1964] U.S. Equal Employment Opportunity Commission (1964). Title VII of the Civil Rights Act of 1964. https://www.eeoc.gov/statutes/title-vii-civil-rights-act-1964. (Date Accessed 2021-11-15).

[U.S. Equal Employment Opportunity Commission, 1979] U.S. Equal Employment Opportunity Commission (1979). Questions and Answers to Clarify and Provide a Common Interpretation of the Uniform Guidelines on Employee Selection Procedures. https://www.eeoc.gov/laws/guidance/questions-and-answers-clarify-and-provide-common-interpretation-uniform-guidelines.

[Verma and Rubin, 2018] Verma, S. and Rubin, J. (2018). Fairness definitions explained. In *Proceedings - International Conference on Software Engineering*, volume 18, pages 1–7.

[Vyas et al., 2020] Vyas, D. A., Eisenstein, L. G., and Jones, D. S. (2020). Hidden in Plain Sight-Reconsidering the Use of Race Correction in Clinical Algorithms. *The New England Journal of Medicine*, 383(9):874–882.

[Wasserman, 2004] Wasserman, L. (2004). *All of Statistics: A Concise Course in Statisical Inference*. Springer, 2nd edition.

[Yeh and Lien, 2009] Yeh, I.-C. and Lien, C.-H. (2009). The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert Systems With Applications*, 36:2473–2480.

[Zhang et al., 2018] Zhang, B. H., Lemoine, B., and Mitchell, M. (2018). Mitigating Unwanted Biases with Adversarial Learning. *AIES 2018 - Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 335–340.

[Zhang and Neill, 2016] Zhang, Z. and Neill, D. B. (2016). Identifying Significant Predictive Bias in Classifiers. *arXiv e-prints*.

Appendix A

Appendix

A.1 Derivation of Normalization Constant

The weighted misclassification rate forms the basis for our developed fairness measure: weighted misclassification ratio. The weighted misclassification rate is defined as

$$WMR = c(w_{FP}) \cdot \frac{w_{FP}FP + (1 - w_{FP})FN}{n}, \tag{A.1}$$

where $c(w_{FP})$ is the normalization constant which will be derived in this section, w_{FP} the false positive weight, FP the number of false positives, FN the number of false negatives, and n the total number of observations. The normalization constant is defined such that the weighted misclassification rate fulfills the following two criteria:

1. WMR reduces to the misclassification rate when $w_{FP} = 0.5$.
2. WMR $\in [0, 1]$.
3. For all $w_{FP} \in [0, 1]$, WMR uses entire span of $[0, 1]$

The misclassification rate is defined as $MR = \frac{FP+FN}{n}$. Inserting $w_{FP} = 0.5$, the first criterion is satisfied when

$$c(0.5) \cdot \frac{0.5FP + 0.5FN}{n} = \frac{FP + FN}{n} \Leftrightarrow c(0.5) = 2 \tag{A.2}$$

The second criterion limits the domain of WMR to the interval $[0, 1]$. The non-negativity is ensured by having $c(w_{FP}) \geq 0$ for all $w_{FP} \in [0, 1]$. Next task is to ensure that $WMR \leq 1$ at all times. It is noted that the weighted misclassification rate attains the largest values when either $FP = n$ and w_{FP} is large or when $FN = n$ and w_{FP} is small. These two scenarios give rise to the following equations:

$$WMR = c(w_{FP})\frac{w_{FP}n + (1 - w_{FP}) \cdot 0}{n} = c(w_{FP})w_{FP} \qquad \leq 1 \tag{A.3}$$

$$WMR = c(w_{FP})\frac{w_{FP} \cdot 0 + (1 - w_{FP})n}{n} = c(w_{FP})(1 - w_{FP}) \leq 1 \tag{A.4}$$

Hence, to ensure that $WMR \leq 1$, the normalization constant must satisfy $c(w_{FP}) \leq \frac{1}{w_{FP}}$ and $c(w_{FP}) \leq \frac{1}{1-w_{FP}}$. Furthermore, it is seen that when $w_{FP} = 0.5$, then $\frac{1}{w_{FP}} = \frac{1}{1-w_{FP}} = 2$, which satisfies the first criterion. Hence the normalization constant is defined as

$$c(w_{FP}) = \min\left(\frac{1}{w_{FP}}, \frac{1}{1-w_{FP}}\right). \tag{A.5}$$

The third criterion says that regardless of the value of w_{FP}, the WMR should be able to take on values in the entire range of $[0, 1]$. The perfect classifier will yield $WMR = 0$ irregardless of w_{FP} since $FP = FN = 0$. Without loss of generality assume that $w_{FP} > 0.5$ such that $c(w_{FP}) = \frac{1}{w_{FP}}$. By letting $FP = n$ and $FN = 0$, the WMR is

$$WMR = \frac{1}{w_{FP}} \cdot \frac{w_{FP} \cdot n + (1 - w_{FP}) \cdot 0}{n} = 1 \tag{A.6}$$

Hence, the WMR can take on values in the entire span of $[0, 1]$, and the choice of the normalization constant also satisfies the third criterion.

A.2 Additional Tables and Figures

A.2.1 Example Datasets

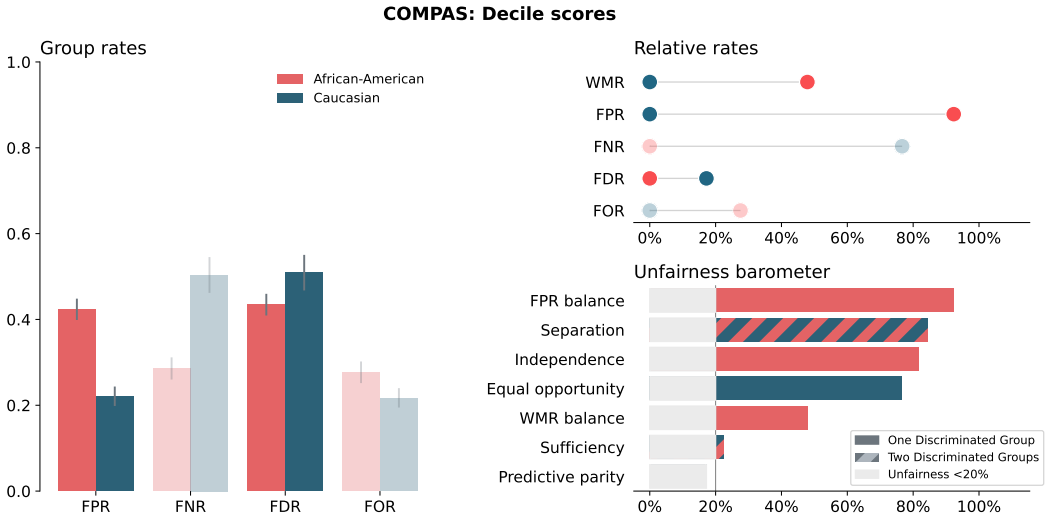


Figure A.1: The BiasBalancer second level visualization of the COMPAS decile scores with $w_{FP} = 0.9$.

	English Variable Name	Explanation	Variable Values
Individual	id	Unique id numbers	Integer
	V1_sex	Sex of juvenile	{Male, Female}
	V4_area_origin	Area of origin of the juvenile	{Espanya, Maghreb, Latin America, Other, Europa}
	V6_province	Province of the juvenile	{Lleida, Barcelona, Girona, Tarragona}
	V8_age	Age at time of crime	Integer, [14, 17]
	V9_age_at_program_end	Age at end of program (2010)	Integer, [14, 27]
	V10_date_of_birth_month	Birth month	{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}
	V10_date_of_birth_year	Bith year	Integer, [1982-1996]
Criminological	V11_criminal_record	Any previous records?	True/False
	V12_n_criminal_record	How many previous records?	{0, 1-2, 3-5, 5+}
	V13_n_crime_cat	Number of crimes	{1, 2, 3+}
	V15_main_crime_cat	Category of main crime	{Against People, Against Property, Other}
	V16_violent_crime	Was the crime violent?	True/False
	V17_crime_classification	Does it classify as a crime	True/False
	V19_committed_crime	What is the comitted crime?	23 unique strings
	V20_n_juvenile_records	Number of records at Juvenile Justice	Integer, [0, 39]
	V21_n_crime	Number of crimes in current case	Integer [1, 12]
	V22_main_crime_date_month	Month of commission of crime	{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}
	V22_main_crime_date_year	Year of commission of crime	Integer, [2000, 2010]
Program	V23_territory_of_execution	Province of followed program	{Lleida, Barcelona, Girona, Tarragona}
	V24_finished_program	What program has the juvenile finished	16 unique strings
	V26_finished_measure_grouped	Categorization of the program	{Internment, Probation, ATM, Other, Community Service, MRM}
	V27_program_duration_cat	Duration of program	{<6 moths, 6 months <1 year, >1 year}
	V28_days_from_crime_to_program	Days between crime and program	Integer, [1, 2390]
	V29_program_duration	Duration of assigned program in days	Integer, [0, 2257]
	V30_program_start_month	Month of program start	{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}
	V30_program_start_year	Year of program start	Integer, [2004, 2010]
Target	V115_RECID2015_recid	Are they recidivists by 2015	True/False

Table A.1: List of attributes used for modeling recidivism of Catalan juveniles. The English attribute names can be linked to the original attribute using the number in front of each variable. Each attribute is described, and the domain of the attribute in the processed dataset is listed.

Dataset	Fold	Train size	Val size	Test size	N layers	Lr	P dropout	N hidden
German Credit	0	640	160	200	2	0.010	0.4	(20, 5,)
German Credit	1	640	160	200	2	0.001	0.0	(20, 10,)
German Credit	2	640	160	200	3	0.001	0.0	(5, 15, 20)
German Credit	3	640	160	200	1	0.010	0.1	(15, ,)
German Credit	4	640	160	200	3	0.001	0.2	(10, 15, 15)
Taiwanese Credit		19200	4800	6000	3	0.001	0.0	(15, 5, 5)
Catalan Recidivism	0	2976	745	931	3	0.001	0.0	(5, 20, 20)
Catalan Recidivism	1	2976	745	931	1	0.001	0.0	(15, ,)
Catalan Recidivism	2	2977	745	930	2	0.001	0.1	(10, 20,)
Catalan Recidivism	3	2977	745	930	1	0.001	0.0	(10, ,)
Catalan Recidivism	4	2977	745	930	2	0.001	0.0	(10, 15,)

Table A.2: Train size, val size, and test size specify the size of the train, validation, and test datasets, respectively. N layers denotes the number of hidden layers, lr the learning rate, p dropout the chosen dropout probability, and finally N hidden is the number of hidden units in each of the hidden layers.

A.2.2 CheXpert Case Study

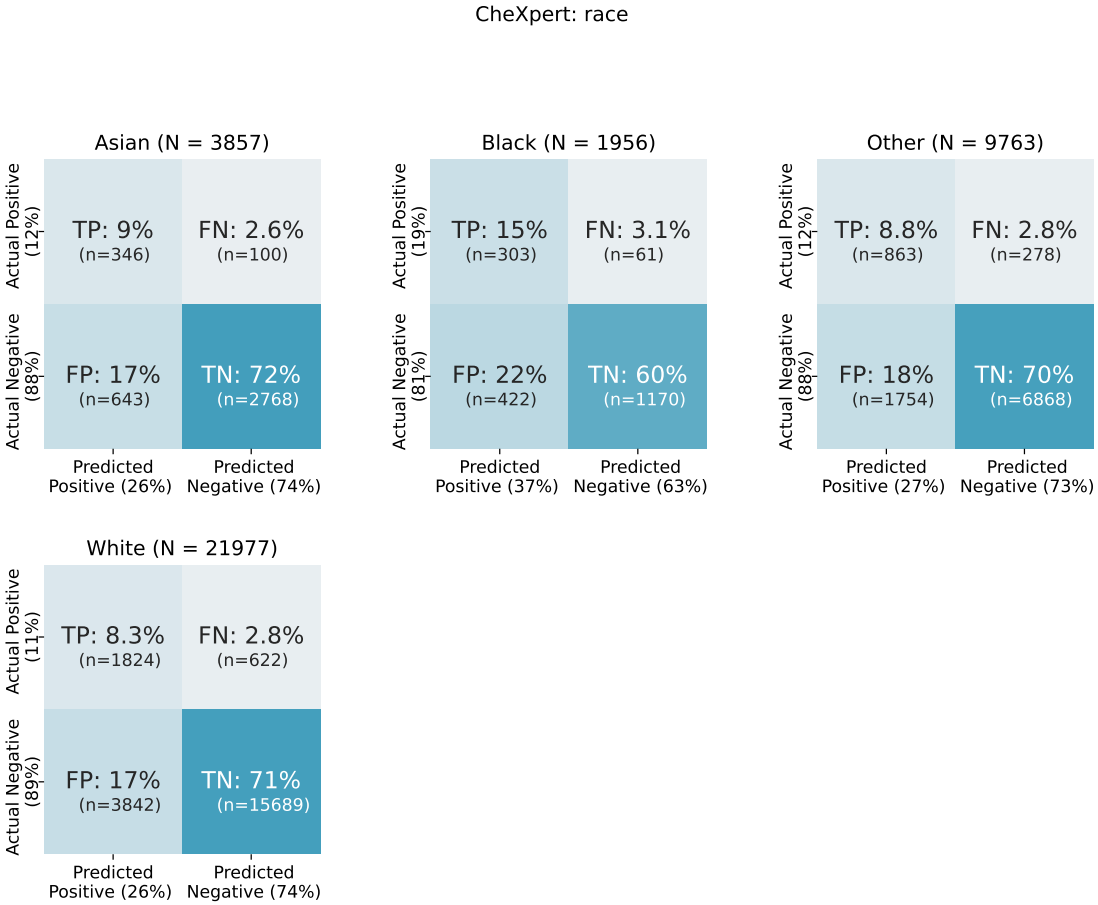


Figure A.2: Confusion matrix for each subgroup from the third level of BiasBalancer used to analyze the fairness of the predictions of cardiomegaly from DenseNet w.r.t. race.

Sensitive Attribute	Group	Split	N	Has Cardiomegaly	CI
Race	Asian	test	3857	446 (11.56%)	[10.59%, 12.61%]
		train	12372	1589 (12.84%)	[12.27%, 13.44%]
	Black	test	1956	364 (18.61%)	[16.95%, 20.40%]
		train	6653	1321 (19.86%)	[18.91%, 20.83%]
	Other	test	9744	1137 (11.67%)	[11.05%, 12.32%]
		train	34378	4028 (11.72%)	[11.38%, 12.06%]
	White	test	21975	2446 (11.13%)	[10.72%, 11.55%]
		train	68694	8183 (11.91%)	[11.67%, 12.16%]
Race and Sex	Asian_Female	test	1681	195 (11.60%)	[10.16%, 13.22%]
		train	5457	662 (12.13%)	[11.29%, 13.02%]
	Asian_Male	test	2176	251 (11.53%)	[10.26%, 12.95%]
		train	6915	927 (13.41%)	[12.62%, 14.23%]
	Black_Female	test	1069	237 (22.17%)	[19.78%, 24.76%]
		train	3323	690 (20.76%)	[19.42%, 22.18%]
	Black_Male	test	887	127 (14.32%)	[12.17%, 16.78%]
		train	3330	631 (18.95%)	[17.65%, 20.32%]
	Other_Female	test	3986	494 (12.39%)	[11.41%, 13.45%]
		train	14208	1658 (11.67%)	[11.15%, 12.21%]
	Other_Male	test	5758	643 (11.17%)	[10.38%, 12.01%]
		train	20170	2370 (11.75%)	[11.31%, 12.20%]
	White_Female	test	8896	929 (10.44%)	[9.82%, 11.10%]
		train	27822	2777 (9.98%)	[9.63%, 10.34%]
	White_Male	test	13079	1517 (11.60%)	[11.06%, 12.16%]
		train	40872	5406 (13.23%)	[12.90%, 13.56%]
Sex	Female	test	15632	1855 (11.87%)	[11.37%, 12.38%]
		train	50810	5787 (11.39%)	[11.12%, 11.67%]
	Male	test	21900	2538 (11.59%)	[11.17%, 12.02%]
		train	71287	9334 (13.09%)	[12.85%, 13.34%]

Table A.3: The table shows the prevalence and number of observations in test and training data for each of the three chosen sensitive attributes.

A.3 BiasBalancer Documentation

The documentation of BiasBalancer has been created using Spinx version 4.3.2 and is accessed through the Github repository, where the Readme file contains a link to the documentation. A screenshot of part of the documentation is seen in figure A.3.

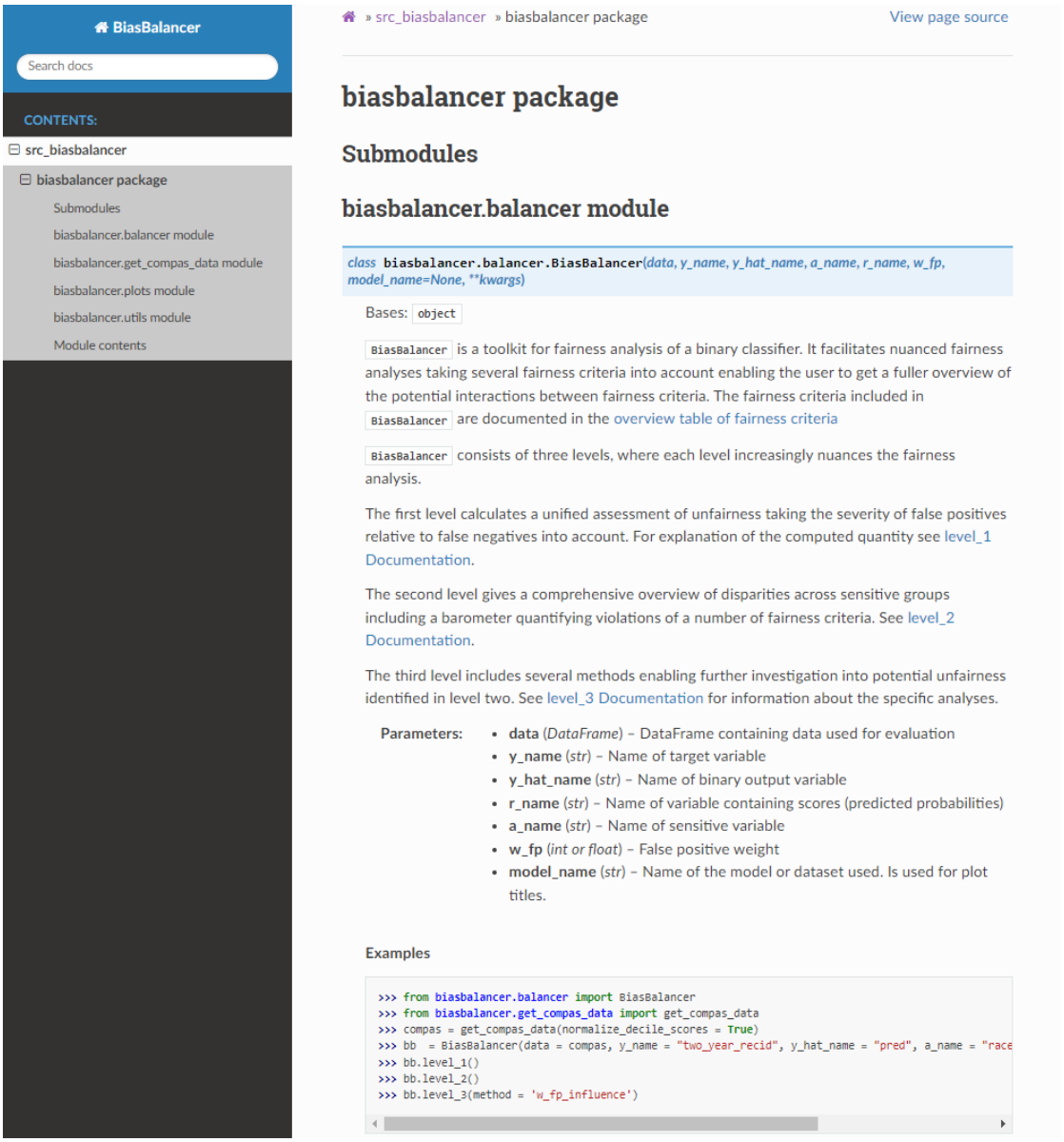


Figure A.3: Screenshot from BiasBalancer documentation [Fuglsang-Damgaard and Zinck, 2022].