Utrecht University

Master Thesis

Automatic Fairness Criteria and Fair Model Selection for Critical ML Tasks

First /Daily Supervisor

Dr. Heysem Kaya

Author

Selim Büyük

External Supervisor

Prof. dr. Ruud Wetzels

Second Examiner

Dr. Matthieu Brinkhuis

A thesis submitted to the Artificial Intelligence Graduate Program

in fulfillment of the requirements for the degree of

Master of Science

June 30, 2023

Abstract

Automatic Fairness Criteria and Fair Model Selection for Critical ML Tasks

by Selim Büyük

In recent years, Artificial Intelligence (AI) has seen a rapid development where it can aid, and sometimes completely replace, humans with decision making due to its superior computation and information processing skills. However, using AI in decision making tasks has not been without flaws, as researchers warn that blindly trusting AI could prove to have major adverse effects for humans. This is due to bias in AI, described as a prejudice of favoritism toward certain subjects, even if they are rationally unjustified. Examples range from university rankings to recidivism tests, where using AI resulted in damaging effects and perpetuation of bias. As a consequence, legislators over the world have introduced new acts to ensure transparency, explainability and fairness in AI, like the recent EU AI act and GDPR. To support this, the field of Responsible AI has set out to make AI more transparent and fair. However, we saw a gap in the current state of the art in fairness assessment toolkits. Researchers urgently called for the creation of a methodology in assisting users with fairness due to its complexity and context dependency. That is why we created this toolkit, in which users are automatically guided by interactive questions on selecting the most suitable model and fairness criteria for a given task, all openly and freely available in JASP. This methodology was created by identifying characteristics of fairness measures, creating a decision tree whose internal nodes composed of interactive questions, mutating this tree and generating candidate trees and subsequently evaluating the trees to select the best one. With this toolkit we hope the help the effort on making AI more transparent and fair.

Acknowledgements

First and foremost, I would like to thank my first supervisor, Heysem Kaya, for his guidance on this project. Heysem, you have given me invaluable insights, in and outside of academics. Our meetings were always guiding me to a better thesis, and I can not thank you enough for that. Matthieu, thank you for your extensive feedback and your suggestions, as this has undoubtedly led this project to be more polished. Ruud, I can not thank you enough for giving me the chance to be on this project. Thank you for your guidance at PwC and throughout this project. Koen and Mieke, thank you for freeing up time to discuss my thesis with you, your expertise in JASP and in this field made the discussions extremely fruitful. All my friends, but most notably Ro, Frits, Freek, Tim and Dirk, thank you for lending a sympathetic ear, even when you did not understand any of what I was doing. Lastly Yvette, thank you for always believing in me and and supporting me through all of this, this is for you.

TABLE OF CONTENTS

List of Tables

List of Figures

List of Acronyms/Abbreviations

AI	Artificial Intelligence
AP	Accuracy Parity
CF	Counterfactual Fairness
COMPAS	Correctional Offender Management Profiling for Alternative Sanctions
DP	Demographic Parity
DI	Disparate Impact
EO	Equalized Odds
EOp	Equal Opportunity
FN	False Negative
FNRP	False Negative Rate Parity
FP	False Positive
FPRP	False Positive Rate Parity
FTA	Fairness through Awareness
FTU	Fairness through Unwareness
GDPR	General Data Protection Regulation
GUI	Graphical User Interface
MCC	Matthews Correlation Coefficient
ML	Machine Learning
NPVP	Negative Predictive Value Parity
PP	Proportional Parity
PRP	Predictive Rate Parity
TN	True Negative
TP	True Positive
TPRP	True Positive Rate Parity
TE	Treatment Equality
XAI	eXplainable Artificial Intelligence

1. Introduction

In our lives, we have to make countless choices. Ranging from mundane, like what to have for breakfast or what to wear, all the way up to more impactful ones, like picking the best college to attend as a student or picking the best candidate for a vacancy in a company as a recruiter. In recent years, *Artificial Intelligence* (AI) has seen a rapid development where it can aid, and sometimes completely replace, humans with these decisions (Verma, 2019). This is due to the evolution of technology in the late twentieth century, AI is therefore able to compute faster and process more information than ever before. As a consequence, algorithms and models have been developed that have helped humans in many ways, from predicting and detecting criminal activity in certain cities to improving efficiency in transportation and logistics (Asaro, 2019; Kancevičienė, 2019; Ryan, 2020; Tilimbe, 2019). The incredible speed and amount of information that these models can go through, really emphasise the difference between AI and human; the capabilities of an AI are far superior in comparison. Another advantage that an AI has over humans, apart from the aforementioned ones, is that AI does not, in principle, make decisions based on emotions and can therefore be seen as completely objective: they learn from the data they are presented with and have no underlying emotions regarding them. Humans, however, tend to deviate from rational thinking when making decisions (Hewig et al., 2011). These deviations can be a result of emotions or be based on *biases*, which can be described as a prejudice or favoritism toward certain subjects, even if they are rationally unjustified (Mehrabi, Morstatter, Saxena, Lerman, & Galstyan, 2021).

However, the fact that AI does not operate with emotions, does not mean it does not have biases. Bias in AI can be attributed to an underlying problem in its decision making, as AI needs data to base its decisions on a subject. The data that has been provided to the AI can have biases in them. For example, we can give data that represents the two sexes in the workforce to an AI that has to hire someone. Since females are still underrepresented in certain positions, we can say that there is a disparity between males and females in the workforce. When AI uses this data, it might have a

certain bias towards males as opposed to females. These biases can have major impact on people's lives if the decisions of AIs are blindly trusted, as O'neil (2017) notes in her work *"Weapons of Math Destruction"*. In this work, O'neil provides examples of cases ranging from university ranking systems to recidivism tests, where using such AI resulted in damaging effects and perpetuation of bias (Verma, 2019). We will highlight a few other examples under different circumstances where blindly trusting AI resulted in catastrophic consequences for a lot of people by going over the different type of harms it can cause.

The different categories of harms can be derived from the use of AI (Crawford, 2017). When AI is used to aid humans with distributing (or *allocating*) resources, like loans in credit score systems or jobs in hiring systems, we speak of *allocative* harms. More specifically, allocative harms occur when an AI withholds certain groups a resource or an opportunity, or in other words, when there is an unfair distribution of resources across groups (Whittaker et al., 2018). Examples of this are gender and racial biases, such as racial bias in sentencing decisions and admissions tests, but also gender bias in assigning credit (Angwin, Larson, Mattu, & Kirchner, 2016; Lamb, 2010; Telford, 2019). On the other hand, AI aids humans by filtering the content we see. This is due to the increase of information on the internet. Models have been created to show the most "relevant" content to a user. By filtering content, the *representation* of certain topics can get biased. Hence why this type of harm is defined as *representational* harm (Whittaker et al., 2018). In other words, representational harms occur when some group is systematically represented in a negative light surrounding identity, gender, race, etc. (Baker & Hawn, 2022; Barocas, Crawford, Shapiro, & Wallach, 2017). Examples of this can be attributed to *stereotyping*, like the word "criminal" appearing more often in online advertisements in which typical black-sounding names were searched (Kay, Matuszek, & Munson, 2015). Or the depiction of more male figures when searching the internet with the keyword "CEO" (Langston, 2015).

1.1. Motivation

The above examples have one thing in common: negative consequences for humans due to bias in AI. To counteract this, a novel growing field of AI has emerged. This field, called *Responsible AI*, tries to provide more fair and transparent solutions while also focusing on accountability and privacy (Arrieta et al., 2020). In other words, by understanding the decision process of an AI, humans can get insights on whether or not a decision has been made responsibly or *fair*. Broadly, we define fairness as the absence of any bias regarding an individual or group's innate or learned characteristics (such as religion or gender) that are irrelevant in a given context of decision-making (Saxena et al., 2019).

The rise of Responsible AI is not by coincidence, legislators have made it a requirement to clarify clearly what the impact of AI is in their use case. Like the EU, who have introduced a new act called the *General Data Protection Regulation (GDPR)* to ensure transparency, among other things (Paka, 2022). Additionally, the recent EU AI act, where pursuing trustworthy AI has been at the forefront (Kop, 2021). Legislation for trustworthy, fair and explainable AI is not bounded to the EU. This is a global effort, as is made evident with a recent New York law. In this law, the use of AI was mitigated on cases, such as employment, if it was not audited on its bias first (Mulvaney, 2021). So while there has been an increase in interest across the globe in the field of Responsible AI, Panigutti et al. (2021) argue that quantitative and systematic assessment of fairness and explainability of AI is still in its infancy. This emphasises the need for such tools, even when these tools are not freely and openly available, if they exist at all. This is further substantiated by Pagano et al. (2023), in which they note that the need for a framework assisting users with fairness is needed.

1.2. Problem Statement

After carefully reviewing the current state-of-art in context of bias assessment toolkits, we found that all of them had some downsides. And yet, there was one major aspect that was consistently lacking across all of them. To be more precise, users in these toolkits did not have any help in the identification of fairness in their data (Deng et al., 2022). That is to say that users have to consider every aspect of fairness manually in order to assess the amount of bias in data and the models generated from them. Processes that range from selecting fairness criteria to selecting models. This is detrimental since most users who use these toolkits, frequently lack adequate knowledge for the tasks (Lee & Singh, 2021). Especially since there are a lot of contextual variations of bias and fairness, which makes manual identification even harder (Lee, 2019). To further emphasise this point, a systematic review of bias toolkits by Pagano et al. (2023) concluded that a methodology for assisting users with this is urgently needed. Because of this, we propose the concept of automating fairness criteria and fair model selection to fill in this gap and help users.

For this thesis, we will focus mainly on binary classification tasks, since most of the bias and unfairness in ML literature is done within this category (Berk et al., 2017). According to Caton and Haas (2020), this is due to two reasons, the first of which being that decisions surrounding binary classification are the most controversial and discussion sparking. Think of hiring vs. not hiring and re-offending vs. not re-offending. The second reason being the relative simplicity of binary classification as opposed to multi-class classification. Hence our focus on binary classification tasks first. This means that we will mainly go over binary classification performance measures and algorithms. By building a toolkit that is able to automatically select a fair model within this type of tasks, we can help a majority of the users and set a solid foundation. Future work can extend this toolkit with algorithms based on regression or other problems.

1.3. Thesis Outline

This thesis consists of five chapters. In Chapter 2, we will discuss the theoretical background and related work regarding this subject. Here, we will go over definitions and what measures and algorithms are more commonly used in the literature, as well as the current state of the art concerning fairness toolkits. After that, in Chapter 3, we can combine the garnered knowledge in the previous chapter and adapt this to a methodology. More specifically, our research approach, the algorithms, automatic model selection, fairness measures and the automatic fairness criteria will be discussed. By applying our methodology, we obtain results, which are shown in Chapter 4. In this chapter, we will also give an impression on how we adapted the result into JASP and discuss the code availability. Finally, in Chapter 5, the results with their implications and limitations are discussed. We end with contextualizing the results into the field of AI and its relevance.

2. Background and Related Work

In order to be able to make the best tool possible, we have to go over some information regarding the history of this problem and what has already been done. By doing so, we can create a tool that fills in the shortcomings of the current state of the art. We specifically want to go over some important subjects, viz. *Bias*, *Predictive Performance Measures*, *Fairness Measures*, *Mitigation Methods*, *Machine Learning Algorithms* and *Existing Tools*. These sections are in order on what aspects of this tool we have to deal with.

2.1. Bias

For the purpose of recognizing bias and successfully assessing it, we have to understand what bias is and how it can differ in varying circumstances first. Following the work of Das and Rad (2020), we define bias as the indication of prejudice, disproportionate weight, inclination or favor for the model towards subsets of data (like individuals in a set) due to biases underlying in algorithm deficiencies and (human) data gathering. These types of biases are identified by Suresh and Guttag (2021) as:

- **Data to Algorithm**: This type of bias resides in the data. Algorithms that use this data, might have biased outcomes.
- **Algorithm to User**: This type of bias resides in the algorithms. By using biased algorithms, user behavior may get affected.
- **User to Data**: This type of bias resides in the users. Since some of the data is gathered directly by surveying users, their underlying bias can affect the data.

As Mehrabi et al. (2021) visualize in their paper, these types of bias can be seen as a continuous feedback loop. This loop can be instantiated in a manner of ways. Suppose for instance that the data is biased, the algorithms that use this data will therefore be biased too. The algorithms themselves can then amplify and preserve

this bias further (Jensen & Neville, 2002; Fan, Davidson, Zadrozny, & Yu, 2005). When the results of an algorithm are then used in real-world applications and are published, they may affect user behaviour. Users that have their behaviour altered in such a way that they now have a bias, can propagate this bias into data and therefore future algorithms (Lerman & Hogg, 2014). There is an alternate case where the data is unbiased, but biased behaviour may still be present because of design/algorithmic choice (Friedman & Nissenbaum, 1996). The loop will still hold true. The same holds for the loop "originating" from the user instead of the algorithm (Olteanu, Castillo, Diaz, & Kıcıman, 2019). All of the scenarios above show us the continuous feedback loop. This loop with the three types of biases, in tandem with some specific examples of bias, can be seen in Figure 2.1.

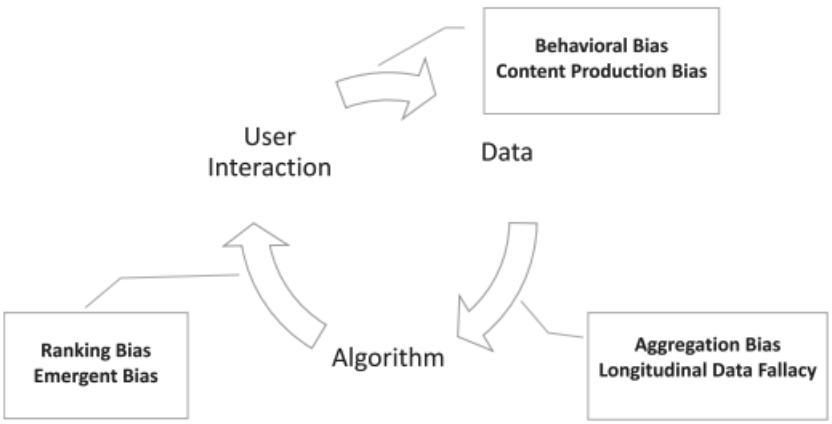

Figure 2.1: Bias cycle with some examples, figure adopted from (Mehrabi et al., 2021).

Since our toolkit may have to deal with all these categories, we will get a bit more in-depth into examples of these types of biases and what specific biases there are. We will start with the Data to Algorithm type.

2.1.1. Data to Algorithm

One form of introducing bias to the data is by misjudging how to measure or report particular features, which is called *measurement bias* (Suresh & Guttag, 2021). Oversimplifications of more complex constructs can be the cause of measurement bias.

Suppose for instance that we want to predict whether a student will be successful in college. Due to the complexity of this task, it may be impossible to predict this with just one attribute. And yet some researchers opt to use a single variable, like Grade Point Average (GPA), to capture this outcome (Kleinberg, Ludwig, Mullainathan, & Rambachan, 2018). However, by doing so, different parts of success are not represented and varying indicators of a student's success are thus ignored, which can lead to measurement bias (Suresh & Guttag, 2021).

Another form of bias underlying in the data we want to discuss, is the *omitted variable bias*. This bias occurs when (one or more) important variables are not considered/omitted, which can lead to spurious correlations (Clarke, 2005; Walsh, Stein, Tapping, Smith, & Holmes, 2021). In other words, it means that there is an important aspect missing in the dataset. As an example, we use the case given by Geiser (2020), where we consider the claim that standardized test scores are better at predicting a student's success in college than high-school grades. When comparing the two, the standardized scores strongly correlate with student demographics (like race and family income). Omitting the student demographics can lead to a higher predictive value for testing and result in the spurious claim that standardized scores are the better predictor. Since the findings are reversed when the these demographics are included.

2.1.2. Algorithm to User

The first form of this type of bias we want to discuss, is the *algorithmic bias*. In this form, bias emerges because of certain algorithmic choices, like use of certain optimization functions or estimators over others (Baeza-Yates, 2018). The importance of carefully deciding the design of algorithms is emphasised here. This form adds bias, even if there is none apparent in the data. As an example, Danks and London (2017) mention the bias-variance tradeoff in algorithms. In this example, researchers might opt to increase bias in a statistical estimator in return of less variance, and thus increase robustness and reliability in the future (Geman, Bienenstock, & Doursat, 1992).

Another way of introducing bias to the algorithm, is during evaluation due to the usage of disproportionate benchmarks that do not represent the population (Suresh

& Guttag, 2021). This form is called *evaluation bias* (Mehrabi et al., 2021) and is similar to representation bias. However, the main difference is that this bias occurs when trying to generalize/evaluate the models of an algorithm due to parameter tuning. An example of this is using IJB-A as a benchmark for facial recognition. Due to the disproportionate amount of dark-skinned woman, algorithms that use this data as a benchmark perform poorly on recognising dark-skinned woman, even when initially scoring high in the training phase (Buolamwini & Gebru, 2018; Ryu, Adam, & Mitchell, 2017).

As a final form of this type we want to discuss is *emergent bias*. This form gets instantiated because of the interaction with real users (Mehrabi et al., 2021). Which can be the case for populations that change in cultural values or societal knowledge after completion of the design (Feuz, Fuller, & Stalder, 2011). An example of this is the emerging bias in social media, where feeds are changed based on what the user *likes*. Showing interest in one subject, may give rise to the feed changing and showing more of that subject due to the constant feedback loop with the interaction with the user, which introduces emergent bias (Kirdemir & Agarwal, 2022).

2.1.3. User to Data

A form of bias that arises due to lexical, semantic or syntactic differences in the contents generated by the users of whom the data is gathered by, is called *content production bias* (Olteanu et al., 2019). To be more concise, users across different ages, cultures or genders can have varying ways of using language. Using just one word that only younger people use, will be biased if older people give their data too (Nguyen, Gravel, Trieschnigg, & Meder, 2013).

There may be cases where data is perfectly measures and sampled, but still consist of bias. A form of bias and socio-technical issues already existing (or existed) in the world, that can be augmented in datasets due to gathering data from users, is called *historical bias* (Suresh & Guttag, 2021). This form of bias is especially tricky to handle because it mimics a historically accurate real world. Take the (now dysfunctional) hiring tool made by Amazon in 2014 for reviewing job applicant's resumes for example.

Due to the algorithm being trained on historic data of the company's hirees, bias was introduced and resumes of females were wrongfully downgraded. This was because of male dominance in the company and the tech industry at that time (Kodiyan, 2019; Langenkamp, Costa, & Cheung, 2020).

As a final form of this type of bias type, *population bias* will be discussed. This type of bias becomes apparent due to a difference in demographics or user characteristics (everything related to a population) of the intended population in comparison to the actual population. Say for example that the toolkit is specifically tailored towards auditors, then there will be bias when other non-auditor users give their data.

2.2. Predictive Performance Measures

In order to understand the effectiveness of this tool and evaluate whether a dataset is fair, we have to consider a way to measure. To do this, we have to establish measures in two parts: the predictive performance of the algorithms, and the amount of bias in them. The first part consists of predictive performance measures. Since we want to select the best model possible, we would not want to consider models that perform worse. This is because the worse performing model (in comparison with another) would give incorrect impressions on the fairness measures. Hence why we need to define predictive performance measures that can be used to compare the models. While on the other hand, we also need good fairness measures to be in place. If we measure fairness with measures that are not applicable for a certain task, we would not be the wiser. If an algorithm is accurate but is not fair (and therefore full of bias), the conclusions of such an algorithm have to be reconsidered. Because of the above, we want good predictive performance- and fairness measures to track. We will go more in-depth about how we define fairness in the upcoming section, but first we want to define our predictive performance measures.

		Predicted Condition			
		Postive (PP)	Negative (PN)		
Actual Condition	Positive (P)	True Positive (TP)	False Negative (FN)	True Positive Rate (TPR) - Recall - Sensitivity (SEN) $\frac{TP}{P}$	False Negative Rate (FPR) $\frac{FN}{P}$
	Negative (N)	False Positive (FP)	True Negative (TN)	False Positive Rate (FPR) $\frac{FP}{N}$	False Negative Rate (FPR) - Specificity (SPC) $\frac{TN}{N}$
		Positive Predictive Value (PPV) - Precision $\frac{TP}{PP}$	Negative Predictive Value (NPV) $\frac{TN}{PN}$		
		Accuracy (ACC) $\frac{TP+TN}{P+N}$	F1 score $\frac{2TP}{2TP+FP+FN}$	Matthews Correlation Coefficient (MCC) $\frac{TP*TN-FP*FN}{(TP+FP)(TP+F)(TN+FP)(TP+FN)}$	

Figure 2.2: Overview of calculations for basic performance measures.

The evaluation of models will be done with standard ML measures in combination with a few additional measures. An overview on how these measures are calculated, like what *True Positives* and *Specificity* is, can be seen in Figure 2.2. We will highlight three measures that are most often used in comparing models in ML tasks:

- **Accuracy**: A simple and intuitive predictive performance measure, as we calculate it by getting the number of correct predictions divided by the total number of predictions. However, as a consequence of being simple, this measure is not capable of handling imbalanced datasets well (Weng & Poon, 2008).
- **F_1 Score**: A predictive performance measure that uses two different measures called Precision and Recall. This measure is especially useful in imbalanced data (Smelyakov, Hurova, & Osiievskyi, 2023).
- **Matthews Correlation Coefficient (MCC)**: This performance measure is, according to Chicco et al. (2021), a better way to measure performance than (balanced) accuracy and F_1 Score (Chicco et al., 2021; Yao & Shepperd, 2020).

2.3. Fairness Measures

Fairness is a context-dependent concept, in which varying views and contexts give rise to (slightly) modified definitions. The broad definition we have given in the introduction, where fairness is defined as the absence of any bias regarding an individual or group's innate or learned characteristics (such as religion, skin colour or gender) that are irrelevant in a given context of decision-making, is a good starting point, but not sufficient for fairness criteria (Saxena, 2019). Definitions of fairness can be put in one of three categories, to be more specific: *Statistical Fairness*, *Individual Fairness* and *Causal Fairness* (Verma & Rubin, 2018). These three categories all have a different view on fairness criteria.

To help define these fairness measures further, we have to go over some notations of the aspects playing a role in fairness, like *protected classes*. This aspect, which is also called *sensitive attribute*, is potentially used to treat individuals unfairly and prohibited to be used by the human, like gender, skin color or ethnicity (Abraham, Sundaram, et al., 2019). With these sensitive attributes, algorithms can classify groups with labels, in which the *favorable label* is the desired decision outcome. The favorable label gives an advantage to a certain group. This group is called the *privileged group*. Conversely, the group that has the unfavorable outcome, and thus is discriminated against by the sensitive attributes, is called the *protected* or *unprivileged group* (Ben-Porat, Sandomirskiy, & Tennenholtz, 2021). Attributes whose values can be used to derive the value of other attributes, are called *proxies* (Kilbertus et al., 2017). Finally, we speak of *disparity* when groups are treated differently based on the sensitive attributes. Contrarily, *parity* is the equal treatment of groups (Zafar, Valera, Rodriguez, Gummadi, & Weller, 2017). With these notions in place, we will go over the three categories previously mentioned, starting with statistical fairness measures.

2.3.1. Statistical Fairness Measures

This category is sometimes referred to as Group fairness, since its main characteristic can be reduced to: varying groups should be treated similarly (Mehrabi et al., 2021). This category uses the classes in a confusion matrix, more specifically, the TP, TN, FP and FP and its derivatives, like the Negative Predictive Value, which we will discuss further in this section. According to Verma and Rubin (2018), the measures in this category represent the simplest and most intuitive notion of fairness. For these measures, we use the following notation for a given dataset $D = (A, Z, Y, \hat{Y})$, with A being the protected attribute (like race or religion), other attributes Z, ground truth labels Y and the predicted binary class outcome $\hat{Y}$, derived from a classifier. For demonstration purposes, we suppose that 0 is the unfavorable outcome while 1 is the favorable outcome, so $y \in \{0, 1\}$ in binary cases.

2.3.1.1. Demographic Parity (DP). A definition that looks at the demographics of those receiving a specific classification (positive or negative) which are indistinguishable to the demographics of the whole population, is called *Statistical parity* or *Demographic parity* (Dwork, Hardt, Pitassi, Reingold, & Zemel, 2012). In other words, the probability of a positive outcome should be the same for across all groups (protected and unprotected). This definition is based solely on the predicted outcome, the actual outcomes do not play a role. Formally, demographic parity is defined as:

$$p(\hat{Y} = 1|A = 0) = p(\hat{Y} = 1|A = 1). \tag{2.1}$$

2.3.1.2. Disparate Impact (DI). This measure is similar to DP, but here we use the ratio instead of taking the differences. More specifically, we can say that a given dataset has *Disparate Impact*, sometimes called *Proportional Parity (PP)*, if the probability for the positive outcome class given the protected or sensitive attribute, divided by the positive outcome class given the unprotected attribute (or privileged group) is greater than or equal to 80% (Feldman, Friedler, Moeller, Scheidegger, & Venkatasubramanian,

2015). In other words, we will say that dataset D has disparate impact if:

$$\frac{p(\hat{Y} = 1|A = unprivileged)}{p(\hat{Y} = 1|A = privileged)} < 0.8. \tag{2.2}$$

When the ratio between the positive prediction rates of both groups is high, this measure ensures that the proportion of the positive predictions is similar across groups, just like DP. However, there is one big disadvantage that both of these measures have. An algorithm may be deemed unfair due to differing proportions of actual positive outcomes (base rates) of the various groups, even when a fully accurate classifier is used (Pessach & Shmueli, 2022). This definition is, just like DP, based solely on the predicted outcome.

2.3.1.3. Equalized Odds (EO). Designed by Hardt et al. (2016), *Equalized Odds* was initially created for the shortcomings of DP and DI. In this fairness measure, the probability of someone in the positive class being correctly assigned to a positive outcome and the probability of someone in the negative class being incorrectly assigned a positive outcome should be the same for people in the protected and unprotected groups. In other words, both groups should have the same probability for true positives and false positives. This definition is based on a combination of the predicted and actual outcomes. In equation form:

$$p(\hat{Y} = 1|A = 0, Y = y) = p(\hat{Y} = 1|A = 1, Y = y). \tag{2.3}$$

2.3.1.4. Predictive Rate Parity (PRP). With this measure, we aim to have the same positive predictive value, or precision, across sensitive groups (Castelnovo et al., 2022). This definition is based on a combination of the predicted and actual outcomes. We get:

$$p(Y = 1|A = 1, \hat{Y} = 1) = p(Y = 1|A = 0, \hat{Y} = 1). \tag{2.4}$$

2.3.1.5. Equal Opportunity (EOp). Also called True Positive Rate Parity (TPRP), is a measure that relaxes Equalized Odds, where its main difference is that it is focused on the positive outcome and disregards the negative outcome. This definition is based on a combination of the predicted and actual outcomes. Formally, this can be defined as:

$$p(\hat{Y} = 1|Y = 1, A = 1) = p(\hat{Y} = 1|Y = 1, A = 0). \tag{2.5}$$

2.3.1.6. Specificity Parity (SP). Also known as True Negative Rate Parity (TNRP). This definition is based on a combination of the predicted and actual outcomes.

$$p(\hat{Y} = 0|Y = 0, A = 1) = p(\hat{Y} = 0|Y = 0, A = 0). \tag{2.6}$$

2.3.1.7. False Positive Rate Parity (FPRP). In this measure the false positivity rates between groups should be equal. This definition is based on a combination of the predicted and actual outcomes.

$$p(\hat{Y} = 1|Y = 0, A = 1) = p(\hat{Y} = 1|Y = 0, A = 0). \tag{2.7}$$

2.3.1.8. Treatment Equality (TE). A form that looks specifically at the false negative rates, also called False Negative Rate Parity (FNRP), is the *Treatment Equality* measure. In this measure, the amount of false negatives are divided by the amount of false positives. If they are equal for both the protected group categories, we can speak of treatment equality (Mehrabi et al., 2021). This definition is based on a combination of the predicted and actual outcomes and can be formally defined as:

$$p(\hat{Y} = 0|Y = 1, A = 1) = p(\hat{Y} = 0|Y = 1, A = 0). \tag{2.8}$$

2.3.1.9. Accuracy Parity (AP). This measure requires equal parity across all groups. This measure is most often used when the effects of false negatives and false positives are similar (Fraenkel, 2020). This definition is based on a combination of the predicted and actual outcomes and can be formally described as:

$$p(\hat{Y} = Y|A = 1) = p(\hat{Y} = Y|A = 0). \tag{2.9}$$

2.3.1.10. Negative Predictive Value Parity (NPVP). In cases where we want to evaluate what the proportion is of negative cases that are correctly predicted to be in that negative class out of all negatively predicted cases, we use this measure. In other words, the probability of a subject with a negative prediction whom truly belongs in the negative class (Verma & Rubin, 2018). This definition is based on a combination of the predicted and actual outcomes and can be formalized by:

$$p(Y = 0|\hat{Y} = 0, A = 1) = p(Y = 0|\hat{Y} = 0, A = 0). \tag{2.10}$$

2.3.1.11. Consistency. A completely different way of measuring fairness, is with help of the k-nearest neighbors algorithm. This measure is called *consistency*, defined by Zemel et al. (2013). As its name suggest, this measure assesses the consistency of a model's classification of a given data item to its nearest neighbors. This definition is based on a combination of the predicted and actual outcomes. This is done with the following equation:

$$yNN = 1 - \frac{1}{N}\sum_{i=1}^{N}\left|\hat{y}_i - \frac{1}{k}\sum_{j \in kNN(x_i)}\hat{y}_j\right|. \tag{2.11}$$

2.3.2. Individual Fairness

Whereas measures in the statistical fairness category evaluated fairness based on statistical measures between two (or more) groups, individual fairness measures evaluate each individual using a similarity or distance measure. In other words, similar individuals are treated similarly (Binns, 2020).

2.3.2.1. Fairness through Awareness (FTA). Originally created by Dwork et al. (2012), Fairness through Awareness is a measure to assist and guarantee DP within demographic groups between individuals. This measure deems an algorithm fair if similar predictions are given to similar individuals, where similarity is calculated with help of a measure designed for a specific task (Dwork et al., 2012).

2.3.2.2. Fairness through Unawareness (FTU). If an algorithm does not use any protected attributes (A) during the decision-making process, then Fairness through Unawareness considers it to be fair (Kusner, Loftus, Russell, & Silva, 2017). One of the main disadvantages of this measure however, is its inability to detect bias when there are proxies in the dataset.

2.3.3. Causal Fairness Measures

Due to the relative recency of interest in fairness in AI, new measures and categories are still being created. Thinking of fairness criteria in new ways is how this category got developed (Spirtes, Meek, & Richardson, 2013). Instead of being completely data-driven, this category of fairness measures investigate the causal relationships between outcome labels and attributes. These measures require additional knowledge of the structure of the world in the form of causal models as a consequence of this (Loftus, Russell, Kusner, & Silva, 2018). This added knowledge is significant for understanding how changing an attribute can cause a change in the system, as it supplies additional information (Kilbertus et al., 2017).

2.3.3.1. Counterfactual Fairness (CF). Proposed by Kusner et al. (2017), this measure is derived from the notion that a decision is fair towards individuals when the decision is the same in the actual- and counterfactual world. In this counterfactual world, the individual belongs to a different group. In other words, a predictor $\hat{Y}$ is counterfactually fair under all circumstances where $X = x$ and $A = a$ and for all y and any value a' obtainable by A' in:

$$p(\hat{Y}_{A\leftarrow a}(U) = y|X = x, A = a) = p(\hat{Y}_{A'\leftarrow a'}(U) = y|X = x, A = a). \tag{2.12}$$

2.4. Mitigation Methods

After assessing datasets and coming to the conclusion that the dataset is biased, mitigation methods can be applied to alleviate this. There exist three kinds of such methods, each with a different idea on where to mitigate the bias (Caton & Haas, 2020). We can summarise them as:

- **Pre-Processing Methods**: Approach is applied before modelling.
- **In-Processing Methods**: Approach is applied during the modelling.
- **Post-Processing Methods**: Approach is applied after modelling.

However, even with these three distinct methods, there are approaches in these methods that can be appointed to more than one. We will name a few of these approaches when we go over them in their respective sections, starting with *Pre-Processing Methods*.

2.4.1. Pre-Processing Methods

These types of methods assume that bias is situated in the data itself, in other words, the distribution of sensitive variables is biased and/or imbalanced. Pre-processing techniques are introduced to tackle this problem. As a goal, the algorithms perform transformations to remove discrimination from the (training) data (Kamiran & Calders,

2012). It is important that the training data can be modified for this technique to be applied.

One of the most used approaches in this method, is *(Re)sampling* (Adler et al., 2018; Bastani, Zhang, & Solar-Lezama, 2019). One of the biggest reasons is its intuitive approach, as it gives new data points in the training data that we can use, which can eliminate bias. If there is another subset of unbiased data points, then this approach allows us to use them instead. This approach can also be modified to accommodate multiple classifiers. Combining each of these classifiers with a different subset of data points and one or more sensitive variables can be beneficial in terms of speed and accuracy for each subset (Dwork, Immorlica, Kalai, & Leiserson, 2018).

Another approach is *Reweighing* (Li & Liu, 2022; Sonoda, 2021). This approach assigns weights to instances of data. These weights indicate the importance of sensitive training samples, which help with classifier stability (Caton & Haas, 2020). On top of that, this approach can be defined as something in between pre-processing and in-processing. An example of a paper that does this, is from Jiang and Nachum (2020), in which they identified sensitive training instances, a pre-processing characteristic, and after which learned the weights to optimize for a fairness measure, an in-processing characteristic.

2.4.2. In-Processing Methods

These methods assume that fairness has to be achieved while considering utility and fairness criteria and is applied during modelling. They have to find a balance by going over multiple model objectives while learning. This involves changing the constraints of the classifier to include fairness constraints for example. In-processing focuses on fixing the classifier (Y. Wang & Singh, 2021).

The first approach we want to discuss, is *Regularization* (Rieskamp, Hofeditz, Mirbabaie, & Stieglitz, 2023; Halevy, Harris, Bruckman, Yang, & Howard, 2021). In (normal) regularization, machine learning algorithms are penalized for using too many features. As a consequence, this inhibits such algorithms to overfit. When regularization is applied in fairness, a penalty term is added specifically for inhibiting algorithms

for discriminatory practices (Caton & Haas, 2020). It is to be noted that not all fairness measures are affected the same by the strength of the regularization parameters, making it one of the challenges of this approach. Since regularization is used for going over multiple objectives while learning, it is characterised as an in-processing method. Another approach that is similar to regularization, is *Constrained Optimization* (Manisha & Gujar, 2018). In this approach, the classifier's loss function is adapted for optimization, the constraint in this case being the notions of fairness. This approach can also be used as a post-processing method when using a pre-trained classifier.

One other approach is *Adverserial Learning*, in which an adversary (a deliberate error) is introduced to a learning model to check its robustness (Xu, Yuan, Zhang, & Wu, 2018; Rajabi & Garibay, 2022). When adversarial learning is applied to fair learning, the adversaries are introduced to specifically check if the models are fair. The feedback of this adversary is given to modify the model to be more fair.

2.4.3. Post-Processing Methods

These methods recognize that an algorithm can introduce bias. That is to say that a Machine Learning algorithm may be unfair to protected variables within the protected variable with a given label. Therefore, these types of methods change the labels after a classifier has been trained based on rules and constraints (Castillo, 2019). They fix the predictions of algorithms and are characterised as having one of the more flexible approaches, since these approaches only need predictions and sensitive attribute (variable) information. That is to say that no information on the models is needed.

In order to ensure that the amount of positive predictions is the same as positive examples across all groups, *Calibration* can be used (T. Wang & Saar-Tsechansky, 2020; Pleiss, Raghavan, Wu, Kleinberg, & Weinberger, 2017). However, there are a few drawbacks to this approach. A randomization approach that uses calibration can be used to strike a balance between accuracy and fairness. Though by doing so, the overall fairness as opposed to the accuracy of a model is not impacted positively. This approach also does not guarantee optimality.

When making decisions, humans apply their own decision boundary called thresh-

olds. These thresholds differ for every situation and between people. The final approach we want to discuss, *Thresholding*, uses this concept (Riazy, Simbeck, & Schreck, 2020; Abebe, Lucchese, & Orlando, 2022). These approaches try to find the decision boundaries of a classifier where favored and protected groups have two classifications (positive and negative). Since there has not been made a decision there, it may be prone to bias. Measures like equalized odds are helpful to find these boundaries and handle them.

2.5. Machine Learning Algorithms

In this section, we want to go over standard algorithms with papers that use them in a bias and fairness context. Instead of giving the comprehensive theory behind these algorithms, we will simply go over the most notable aspects of them. We will refer to the previously mentioned existing tools, as well as literature, for reference on what standard algorithms to use. These algorithms are:

- **Support Vector Machine (SVM)**: This algorithm tries to optimize an objective function, where the learned linear projection is orthogonal to the optimally separating (maximum margin) hyperplanes (Cortes & Vapnik, 1995). Park et al. (2022) have used SVMs for privacy-preserving fair learning, while Martinez-Eguiluz et al. (2021) apply a mechanism in SVM to control the degree of fairness relatively precisely.
- **Logistic Regression (LR)**: This algorithm estimates the probability of an event occurring, with help of a given logistic function. This algorithm is used by Kamishima et al. (2012) in applying prejudice removal techniques in datasets with sensitive features (like gender and religion). While Choi and Rainey (2014) have used this algorithm for organizational fairness, as to increase the diversity in a company.
- **Neural Network (NN)**: By using nodes and layers in combination with weights and thresholds, these algorithms are able to classify data points even if the classes are not linearly separable (McCelloch & Pitts, 1943). These algorithms can maintain their high accuracy while also achieving fairness with help of frameworks,

such as the *Fair Neural Network Classifier (FNNC)* (Manisha & Gujar, 2018). On top of that, deep learning variants of this algorithm have also been modified in order to handle fairness, like the *RULER* algorithm (Tao, Sun, Han, Fang, & Zhang, 2022).

- **Random Forests (RF)**: With help of multiple Decision Trees, that consist of nodes that can branch into internal- and leaf nodes due to splitting criteria, this algorithm can classify data points by averaging out the trees and getting the majority vote (Hastie et al., 2009). Modified versions of this algorithm exist to handle fairness specific tasks (Zhang, Bifet, Zhang, Weiss, & Nejdl, 2021).
- **XGBoost**: This algorithm uses gradient boosted decision trees, which exceed the limits of normal decision trees. This algorithm is characterised for its execution speed (Chen & Guestrin, 2016). By using the speed of this algorithm, Liu et al. (2022) have proposed a framework in which fairness, efficiency and accuracy all are central.

2.5.1. Fairness-aware Machine Learning Algorithms

These types of algorithms are specifically made with fairness in mind. These algorithms are most often modified versions of standard machine learning algorithms. They use the original's performance characteristics and add fairness on to it. An overview of a few of these modified fairness algorithms are:

- **FairXGBooST**: This algorithm is a modified version of the XGBoost algorithm, that uses state-of-the-art bias mitigation methods with the scalability, robustness and transparency of the original (Ravichandran, Khurana, Venkatesh, & Edakunni, 2020).
- **Fair-AdaBoost**: This algorithm is an extension of the AdaBoost algorithm, which has its advantages in interpretability, scalability and accuracy. This algorithm can be optimised even further with Non-Dominated Sorting Genetic Algorithm-II (NSGA-II), which assists with hyper-parameter tuning (Huang, Li, Jin, & Zhang, 2022).

- **FairGBM**: Another gradient boosting algorithm that is modified to be able to handle fairness constraints (Cruz, Belém, Bravo, Saleiro, & Bizarro, 2022). Its main advantage is the speed in training time in comparison to other fairness-aware algorithms.

2.6. Existing Tools for Fairness Analysis and Mitigation

To be able to improve the current state of the art, we have to review existing tools. Moreover, we want to establish what has already been done, and what can be improved. We can go over the techniques and definitions used in these toolkits as a baseline for our toolkit, and supplement them where necessary. Instead of going over all toolkits that exist, we want to lay a focus on toolkits that comply to a few conditions, like in the work of Lee and Singh (2021), with reference to what their implications and considerations are in practice. The toolkits we consider have to be open-source, since we believe that open and freely available tools should be the norm and will help the most amount of people. Next, we want toolkits that are likely to be found by users searching for aid in fairness. In other words, the most used toolkits in these contexts. We can use literature reviews on fairness toolkits to achieve this, like the works of Mehrabi et al. (2021) and Wenink (2021). Finally, we want to select toolkits with relevant implementations of fairness methods (Lee & Singh, 2021). We have selected three toolkits that satisfy the requirements, starting with the *AI Fairness 360 (AIF360)* toolkit.

2.6.1. AI Fairness 360 (AIF360)

The first tool we want to discuss is created by Bellamy et al. (2019) within IBM. This toolkit can be used for detection and mitigation of ML models. It takes uploaded datasets as input. A set of bias measures is then checked for this dataset. It then asks the user to pick a bias mitigation algorithm to use. Afterwards a comparison is made between the original vs. mitigated results to see the impact the chosen mitigation algorithm has. The AIF360 toolkit can be improved in a few ways, the first of which is the automation of the bias measures. Currently, user's choices on what bias measures

they want to use, are not considered. This is because AIF360 uses a set list of measures. A lack of reporting and explanation of the measures is also a drawback we want to address. Users can observe the values of the measures and whether they are fair or biased, though an intuition on what that means is left to decide for the user.

2.6.2. FairLearn

The second tool, created by Bird et al. (2020) called *FairLearn*, is used for assessing and improving fairness in AI. It specifically focuses on negative impacts for groups of people and uses a variety of bias mitigation algorithms. The main difference with this toolkit that we want to underline is the lack of a (graphical) user interface (GUI). Though originally planned to be implemented, the GUI was scrapped. The use of GUI, especially with people with little knowledge on how to identify bias, is an important aspect to have.

2.6.3. Aequitas

The final tool we want to discuss, is the tool created by Saleiro et al. (2018). This tool is characterised as a bias and fairness toolkit for auditors. It takes uploaded data in which the user can select protected groups and fairness measures. It then evaluates the selected parameters by giving a bias report. Aequitas is in essence what we want to accomplish. In spite of that, this toolkit is missing some crucial parts that this thesis intends to implement. First and foremost, Aequitas relies on manual user input for selecting fairness measures. This project intends to fully automate that process. By asking users specific questions, our tool should automatically select the best bias measures fitted for the task. On top of that, Aequitas' reporting can also be improved. In our toolkit, we want to further explain the measures and make them more approachable. More notably, we want users to understand the impact of certain results. For instance, instead of simply showing the value of a fairness measure, we want to further discuss this result.

2.6.4. Summary, Literature Gap & Research Questions

To summarise, all of the current tools have some downsides that we want to solve, like lack of GUI or bias mitigation. The most important common deficiency of these tools that they all lack, is leaving the identification of bias entirely up to the users who frequently lack adequate knowledge for the tasks. The user has to consider every aspect of fairness manually in order to assess the amount of bias in data. This has shown to be difficult in part due to a lot of contextual variations of bias and fairness, which we went over in previous sections.

Due to the above, in combination with a systematic review of bias toolkits by Pagano et al. (2023), the need for a framework assisting users with this is needed. Therefore, we propose the concept of automating fairness criteria and fair model selection to fill in this gap as the main research goal of this project. We can formulate the goal as a research problem, as well as the research (sub)questions, all considering a binary classification task, as follows:

- **Research Problem**: Is it possible to automate fairness criteria and fair model selection for ML tasks via an interactive interface?
- **RQ 1**: How do we find the best suited machine learning model and how should we automate fair model selection?
- **RQ 2**: How should we evaluate and automate for the most suitable fairness criteria?
  - RSQ 1: How can we evaluate a decision tree whose internal nodes are composed of interactive questions that ensure maximum separation of fairness criteria?
  - RSQ 2: What kind of a decision tree is most suitable for automated interactive selection of fairness criteria?

3. Methodology

In this chapter, we will discuss the approach on how we want to implement the toolkit with the knowledge gained from Chapter 2. We will go over the cases our toolkit should handle and display how the final phase of the toolkit should like. The binary classification algorithms that are used for predicting labels and how the method of picking the best model for these cases will then be explained. After this, we will state the selection of fairness measures that are considered in automatic fairness criteria. Finally, we will examine a method for setting a suitable fairness criteria for the set of fairness measures.

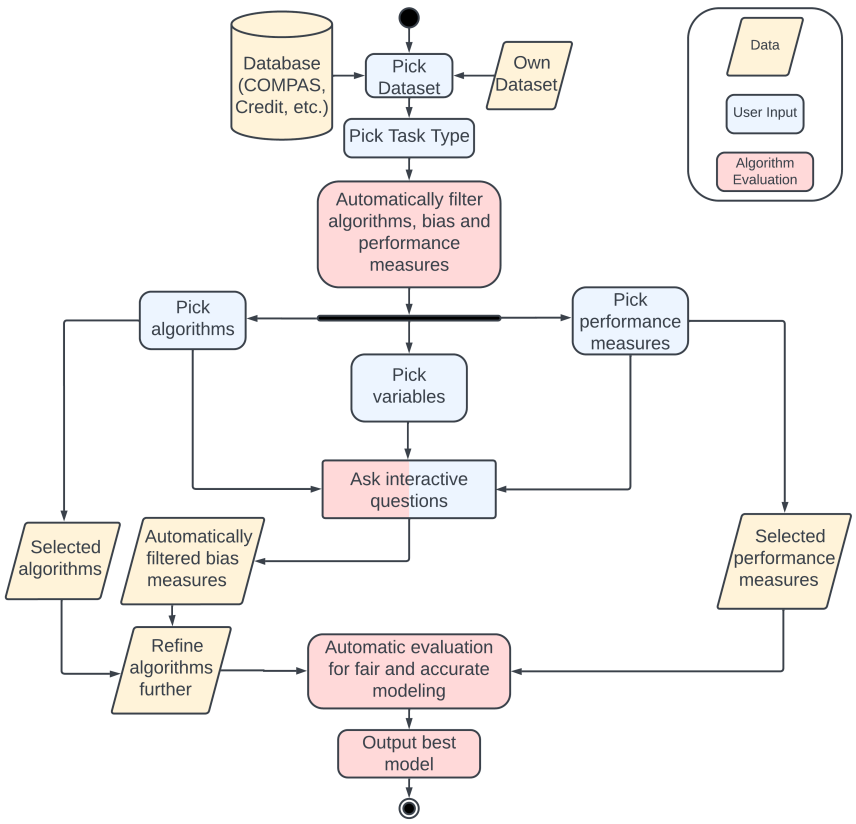

Figure 3.1: Flowchart of the toolkit.

3.1. Proposed Processing Pipeline

First, we want to showcase our vision of the toolkit with this methodology in a flowchart. Every aspect of what we need in this toolkit is depicted in Figure 3.1. In this flowchart, we see that users are able to input different datasets and pick algorithms, performance measures and variables to evaluate the dataset. By asking interactive questions, we can automatically evaluate for fair and accurate model selection. Finally, we can output the best model with a report. These aspects will be discussed in more detail in their respective subsections, starting with the algorithms.

3.2. Algorithms

We consider two cases for the use of our toolkit. In the first case, the data that the user wants to audit for fairness across sensitive attributes and subgroups already contain a predicted column of the target column. In other words, a Machine Learning classifier has already been used to gather the predictions. In contrast to this is the second case, where the data does not contain any predicted columns and needs classifiers in order to predict the target labels. For this case, we need algorithms that are specifically designed to do so. We can think of multiple types of algorithms, like binary classification, multiclass classification and regression. However, just like the fairness measures, we want to ensure that our toolkit works before we add more algorithm types. Hence the decision to focus on binary classification first, as we want to test our toolkit on the most used cases of fairness in the literature as a test. According to Caton and Haas (2020), this is done within binary classification tasks. The selection of algorithms to be implemented in our toolkit, have been decided after evaluating the literature on the most well known and used classifiers regarding fairness. According to Hattatoglu et al. (2021), these algorithms are:

- **Support Vector Machines**
- **Random Forest**
- **Logistic Regression**

Hence why we chose to implement these classifiers in our toolkit. If a user want to use more than one algorithm, for instance to get the best performing one, automatic selection of the best model will be performed by this toolkit. Which is what we will discuss next.

3.3. Automatic Model Selection

In order to get the best approximation of fairness in a dataset, we need a model that comes as close to the target labels as possible. In other words, we want the model with the best performance measures. Seeing as how each measure is useful in a different context, we will let the user decide on which performance measure we should focus to pick the best model. For testing purposes, we will pick models based on Matthew's Correlation Coefficient, as papers suggest that this measure is more informative and is a more truthful score for binary classification tasks over other measures, such as the F_1 measure (Chicco et al., 2021; Chicco & Jurman, 2020). In addition to that, standard Machine Learning preprocessing steps will be taken, like One-Hot encoding for categorical variables and feature scaling, before we use the data as input for the algorithms to improve the accuracy and reliability of the models.

If there are models with the same score for the selected performance measure(s), a tiebreaker is held by looking at the fairness measures. Since auditors can have different reasons for using this toolkit, we will also let the user decide whether the best- or worse case should be picked alongside an option to randomly pick a model out of the models with the best performance measure. Two models with the same performance measure can have different parities. If we automatically pick the model with the parities closer to each other, than we bias the toolkit. The same holds true if we automatically pick the worse model. That is why we let the users choose.

3.4. Fairness Measures

Despite our wish to include all fairness measures in our toolkit that we could find, like those mentioned in Section 2.3, we wanted to first make sure that automating fairness criteria is possible before including more fairness measures. Therefore, a selection of a few measures will be used to test whether we can automate fairness criteria. We have decided to select most of the statistical fairness measures described in Section 2.3.1. Because this category of fairness measures represent the simplest and most intuitive notion of fairness according to Verma and Rubin (2018). It is good to note that even though we will use a subset of all fairness measures, this toolkit will be made in such a way that adding new fairness measures should be done with relative ease. This selection of fairness measures is based on the most used fairness measures in the literature and currently existing toolkits. More notably, we will use the fairness measures used in Aequitas (Saleiro et al., 2018) and the fairness package in R (Kozodoi & V. Varga, 2021). This is done so that we can easily cross-check our values with the existing toolkits' values. We will do this by calculating all the fairness measures in our selection with the datasets mentioned in Section 4.1 and comparing these values to the values from Aequitas and the fairness package in R. Our selection of fairness measures, accompanied by the respective section in which they are discussed in more detail, is:

- **Demographic parity (Section 2.3.1.1)**
- **Proportional parity (Section 2.3.1.2)**
- **Equalized odds (Section 2.3.1.3)**
- **Predictive rate parity (Section 2.3.1.4)**
- **True positive rate parity (Section 2.3.1.5)**
- **True negative rate parity (Section 2.3.1.6)**
- **False positive rate parity (Section 2.3.1.7)**
- **False negative rate parity (Section 2.3.1.8)**
- **Accuracy parity (Section 2.3.1.9)**
- **Negative predictive value parity (Section 2.3.1.10)**

We can calculate the values of these measures and examine datasets on their fairness and evaluate whether there is parity (all is fair) or disparity (there is a difference in treatment and thus unfair). Parity between subgroups can be calculated in a number of ways but there is no benefit of using one over the other. There is the possibility of calculating the ratios between two subgroups, as well as the difference between them. For the former, we speak of parity when the value is equal to 1, as division of equal values will result in 1. For the latter, this is the case when the value is equal to 0, as the difference in equal values will result in 0. Both of these options use a reference group, which can also be seen as a baseline. Or in other words, it will be the denominator for the ratios, while it is the minuend for the differences.

In this toolkit, we opted for calculating the ratios between subgroups. Suppose for instance that we have the sensitive attribute Sex, with subgroups Male, Female and Other. If we consider Male to be the reference group, we can calculate the parity for a measure, like the demographic parity, in this way: $\frac{DemPar_F}{DemPar_M}$ and $\frac{DemPar_O}{DemPar_M}$. If the values are equal to 1, like the case where $\frac{DemPar_M}{DemPar_M}$, then we say that there is parity between the subgroups. We also know if there is disparity between subgroups, and in which direction they favor. Say for instance that we have a value larger than 1 in the case above. This means that the non-reference group (female or other) is larger/has a better demographic parity value than the reference group (male), and as such, that the non-reference group is the privileged group and vice versa.

3.5. Automatic Fairness Criteria

For the purpose of aiding the user in getting the best selection of the aforementioned fairness measures, we suggest the use of interactive questions which will ultimately result in a subset of fairness measures tailored towards the needs of the user. This is especially needed since fairness measures can be mutually exclusive, heavily context dependent and quite complex. So in short, by asking users guided questions, we can help them in selecting the best fairness measures for certain tasks. Creating these guided questions can be done with help of the following methodology:

(i) **Determine Characteristics** We want to focus on the characteristics of fairness measures in the questions, as this is the easiest way to divide the measures into smaller subgroups. The goal here is to find characteristics that are distinguishable, that is to say that these characteristics should be unique for a few measures. Overlap of these characteristics would put the measures in the same group. In other words, fairness measures with similar characteristics will be in the same group. Hence why the first step is to determine the characteristics of all the fairness measures in our selection. This can be done by going over their mathematical definitions or use cases for example.

(ii) **Create Base Decision Tree** By using the characteristics of the fairness measures, we can try to create the first decision tree that act like flowcharts. Starting at the root of the tree, we have the first question. With regard to this answer of this question, we pose another question. This is done until a subset of fairness measures is selected. These are the measures that are best suited for the given fairness criteria. Thus, the nodes of this tree represent the questions and the leaves represent the fairness measures. The questions need to be asked in such a way that there are two distinct groups. For instance, if we have the characteristic "ground truth", we can pose a question "Should the ground truth labels be considered?".

(iii) **Create Candidate Trees** In order to get the best set and order of questions, we need to generate more trees to determine if there is a better one than the base tree. Mutation can be done by swapping the nodes in depth d with depth $d + 1$ and adapting the fairness measures in the leaves accordingly. This process can start at the root (depth 0). The resulting tree will be a candidate tree.

(iv) **Evaluate Trees** After gathering candidate trees, an evaluation measure is need to check the candidate trees for their goodness of fit. This is an equation in which we can set variables and constraints such that the best tree corresponds to the best values. We can minimise and maximise in this equation. An example would be minimising the amount of leaf nodes to get a compact tree.

4. Results

In this chapter, the results of applying our methodology are discussed. More specifically, we will go over the datasets that we used for evaluating our methodology first. In addition, the characteristics of our fairness measures, the base tree, a few candidate trees and the evaluation of them will follow. Finally we will show the implementation of the best decision tree we obtained in JASP and describe the code and application availability.

4.1. Datasets

To ensure that this dataset is working as intended, we have evaluated a selection of datasets most commonly used in fairness assessment (in binary classification) in the literature. These datasets have been also selected on their size, as we wanted to test our toolkit on different sizes (=features) of datasets to further generalize our approach. As previously mentioned in Chapter 3, the outcomes of these datasets with our toolkit have been cross-checked with other toolkits and fairness libraries. The datasets we have used, along with their abbreviations in parentheses, are the Correctional Offender Management Profiling for Alternative Sanctions[1] (COMPAS), UCI Adult Income[2] (Adult), Taiwan Credit[3] (Taiwan) and German Credit[4] (German) datasets. An overview of these datasets can be seen in Table 4.1.

[1] https://github.com/propublica/compas-analysis
[2] https://archive.ics.uci.edu/ml/datasets/adult
[3] https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients
[4] https://archive.ics.uci.edu/ml/datasets/Statlog+%28German+Credit+Data%29

	COMPAS	Adult	German	Taiwan
***N* Features**	51	14	20	24
***N* Samples**	7918	48842	1000	30000
Sensitive Features	Gender, Race	Sex, Race	Sex, Age	Gender, Age
Privileged values	Female, Caucasian	Male, White	Male, Old	Male, Old
Desired label	Did not recidivate	High income (> 50K)	Good credit	Normal default rate
Undesired label	Did recidivate	Low income (≤ 50K)	Bad credit	Lower default rate
Area	Social	Financial	Financial	Financial

Table 4.1: Characteristics overview of datasets.

4.2. Characteristics

To determine characteristics to use, we primarily examined the mathematical definition and the use of TP, TN, FP and FN (classes in a confusion matrix). By doing so, we determined the following characteristics for our selection of fairness measures:

- **Absolute values vs. Proportional values** For calculating the fairness measures, either absolute values or proportional values can be used. It is explicitly noted that demographic parity uses absolute values, while proportional parity for example uses proportional values.
- **Ground Truth** Some values do not use the ground truth and can therefore be calculated by just using the predicted labels from a model. Examples of this are Demographic Parity and Proportional Parity.
- **Confusion Matrix** Some fairness measures need all of the classes in a confusion matrix (TP - FP - TN - FN) to calculate the values. An example of this is Equalized Odds, as True Positive Rate Parity (needs TP and FN) and False Positive Rate Parity (needs FP and TN) are in this measure.
- **Correctly vs. Incorrectly classified** In some fairness measures, there is a focus on correctly classified classes (TP + TN), whereas the opposite holds true for others where the focus is on incorrectly classified classes (FP + FN). An

example of the former is Predictive Rate Parity, while an example of the latter is False Negative Rate Parity. It may also be the case that both classes need to be used, this holds true for fairness measures in the previous characteristic where all of the classes in the confusion matrix should be used.

- **Class Specific Focus** Some fairness measures only use one class in the numerator. This characteristic is designed for those fairness measures. An example of this is the Specificity Parity, where only TN is in the numerator.

An overview of all of the fairness measures with the characteristics mentioned above can be seen in Table 4.2. With these characteristics, we can create questions that utilize them to get subsets of the fairness measures.

4.3. Base tree

For our application, we have a dichotomy, where on the one hand we want to have short and interpretable flowchart where compactness is key. While on the other hand we want to help the user in all circumstances and want to use a flowchart that is as detailed as possible to be able understand every decision we make, where understandability is key. If we do not set a constraint on the maximum amount of fairness measures in a leaf node, the most compact tree will always be a tree with one question. To counteract this, the $maxleaf$ parameter is introduced, which is required for generating compact and understandable trees. More specifically, this parameter is a constraint on the leaves of a decision tree. The value the maxleaf is set on, is used to depict the maximum amount of fairness measures in that leaf. For instance, if a maxleaf is set to 2, each leaf can have at most 2 fairness measures. Other values for maxleaf are also possible, but since we have ten measures, we do not want to increase the maxleaf too much, as this can result in splits with groups with a lot of fairness measures. Conversely, if we set maxleaf to one, each leaf will have only one fairness measure, making some of the leaves in the resulting decision tree superfluous. Some of the generated trees will not be eligible for the best tree because of this parameter.

	Absolute values	Ground Truth	Confusion Matrix	Cases Focus	Class Focus
Demographic Parity	Yes	No	No	Both	No Focus
Proportional Parity	No	No	No	Both	No Focus
Equalized Odds	No	Yes	Yes	Both	No Focus
Predictive Rate Parity	No	Yes	No	Correct	TP
True Positive Rate Parity	No	Yes	No	Correct	TP
False Positive Rate Parity	No	Yes	No	Incorrect	FP
False Negative Rate Parity	No	Yes	No	Incorrect	FN
Accuracy Parity	No	Yes	Yes	Both	No Focus
Negative Predictive Value Parity	No	Yes	No	Correct	TN
Specificity Parity	No	Yes	No	Correct	TN

Table 4.2: Characteristics overview of fairness measures.

To generate our base tree, we used a top-down approach to determine the questions. In other words, we wanted to think of questions at the top that were generally applicable for more fairness measures, while asking more specific questions the deeper the tree goes. For instance, we wanted the characteristic of all elements in the confusion matrix to be a question more to the root of the tree, as this is a more general characteristic. Questions such as focusing on a single class should occur deeper in the tree, as this is much more specific. Our implementation of a base tree can be seen in Figure 4.1, where the nodes represent the questions and the leaves represent the fairness measures. In this tree, we ask a minimum of 2 questions and a maximum of 5.

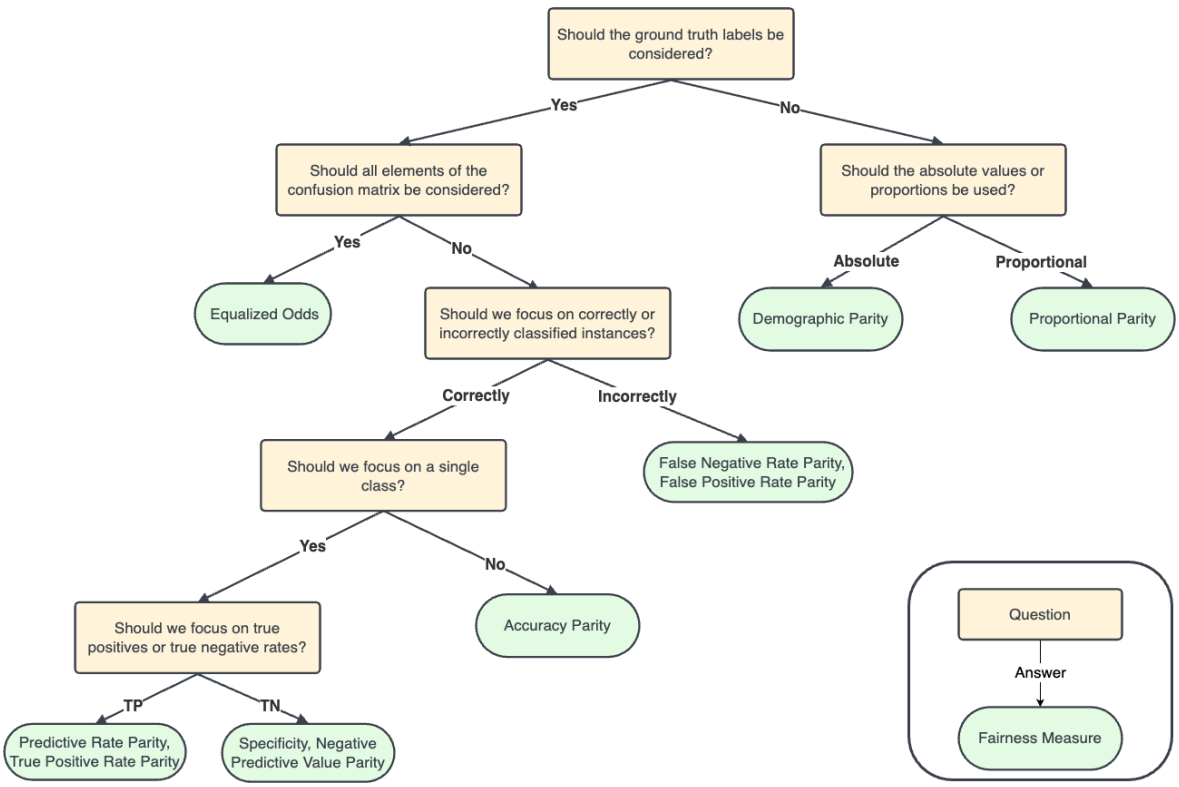

Figure 4.1: The base tree.

4.4. Candidate Trees

Instead of going over all the possible candidates, we will highlight a few interesting ones. The first tree we want to highlight is an ineligible one, to emphasise why we would not want to use questions that are too specific first tree. This tree can be seen

in Figure 4.2. Because of this specific question at the root, no more questions can be asked as none of the unaccounted for fairness measures have the same characteristics as one of the subgroups. In other words, if a measure can not be picked by the user through the questions, we deem that tree to be ineligible for the best tree.

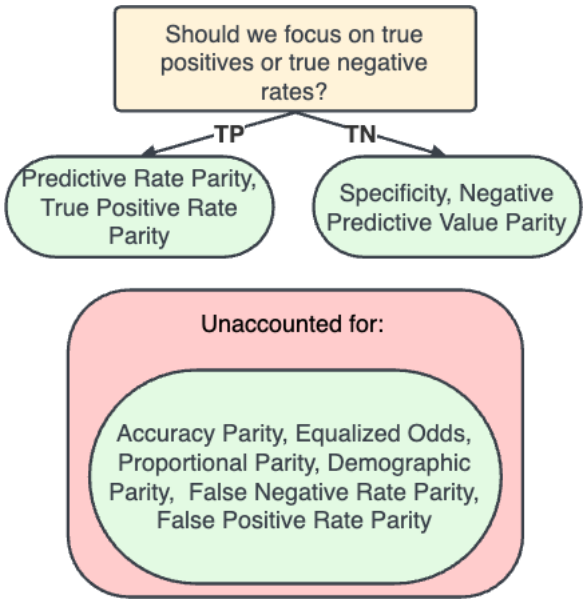

Figure 4.2: An ineligible candidate tree due to unaccounted fairness measures.

Next, we want to go over a tree that is made by mutating the first few questions of the base tree, to show how the mutation process looks like in the beginning phase. This candidate tree can be seen in Figure 4.3. We can see that this tree is rather deep and linear in depth. Each split has a result and a followup question, except for the question in the sixth and final depth, where both of the splits result in a subgroup of fairness measures.

Finally, we want to go over a tree that is more balanced, where the amount of questions is evenly distributed between the branches of the tree. This results in trees with less depth. An example of such a tree can be seen in Figure 4.4. This tree is a more balanced style of tree, where the questions in depth 1 have a follow-up question and a leaf node. The third and final depth has splits which both result in leaves, in comparison to Figure 4.3, where the tree is much more linear and much deeper.

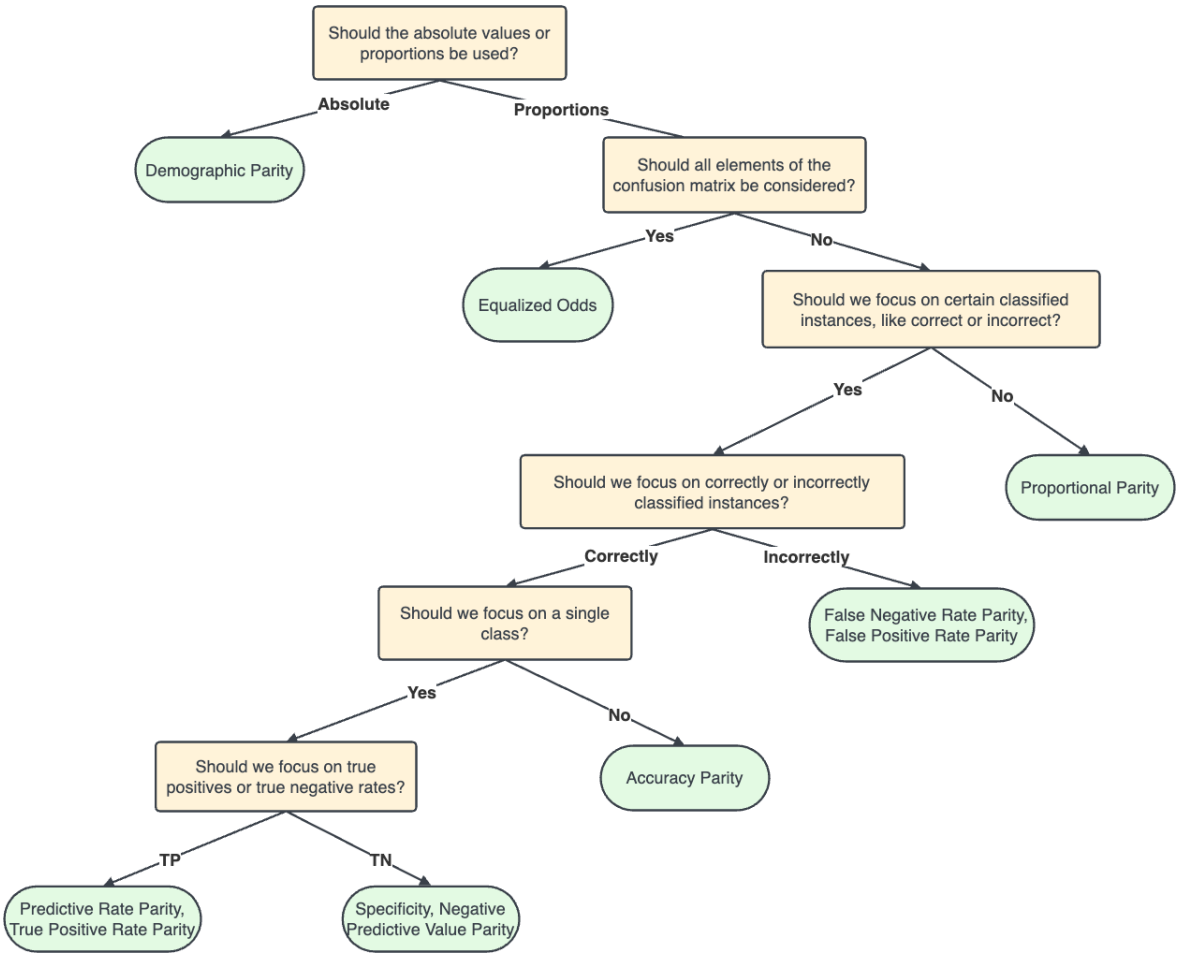

Figure 4.3: Linear candidate tree.

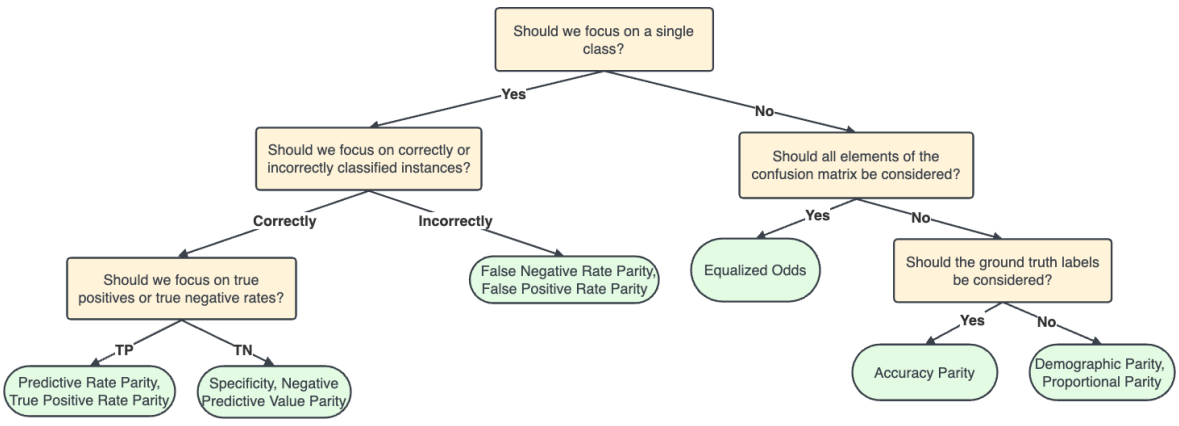

Figure 4.4: Balanced candidate tree.

4.5. Evaluation

As mentioned before, our goal is to select the most compact but understandable decision tree. We want an evaluation function in which this is rewarded. This is done by looking at a few variables with constraints. The first is the average *rule* length, or $\frac{\sum_r n_r}{n_{leaves}}$, where n_r is the number of questions in rule r. Rules in decision trees can be characterised as *if-then* statements, where the if-clause is a conjunct of the the conditions along the path and the then-clause is the outcome (Quinlan, 1987). Since we want a compact tree, we want to have a tree in which the average rule length is minimised. Next we want to punish the amount of leaves a tree has, or n_{leaves}. In the same trend as the previous constraint, minimising the amount of leaves will result in a more compact tree. The above, in combination with a good maxleaf parameter, would affect both the compactness and understandibility of a tree. Since we already used our maxleaf in generating candidate trees, our evaluation function becomes:

$$f(x) = \frac{\sum_r n_r}{n_{leaves}} + n_{leaves}. \tag{4.1}$$

With this function, we can pick the tree that has the lowest value. In general, this evaluation function promotes more balanced trees with great separation and punishes deeper decision trees with just one fairness measure in the leaves. Using this function for the trees, we get 10.29, 10.86, 8.67 for base, c1 and c2 (see Equations 4.2, 4.3 and 4.4) respectively.

$$f(base) = \frac{2+2+2+3+4+5+5}{7} + 7 = 10.29, \tag{4.2}$$

$$f(c1) = \frac{1+2+3+4+5+6+6}{7} + 7 = 10.86, \tag{4.3}$$

$$f(c2) = \frac{2+2+3+3+3+3}{6} + 6 = 8.67. \tag{4.4}$$

4.6. Implementation

As our evaluation measure showed, the tree with the best evaluation score was candidate 2 with a score of 8.67. This decision tree has also been implemented into JASP, as one of our goals in this work was to make this toolkit accessible and ready to use for users. It is good to note that for demonstration purposes, we have used the most commonly used dataset in the literature: the COMPAS dataset. This implementation can be seen in Figure 4.5. All the variables as seen in the implementation are from this dataset. Here, the user can pick a machine learning task and choose between two options: generating predictions by using binary classification algorithms, or use the predictions of their own dataset. If the user decides to generate the predictions by using more than one of the algorithms depicted (like SVM and RF), then we automatically help them with assessing the best model out of the selection by basing it on the selected metric, which is MCC by default. In this section, the user can also define the variables, like the sensitive attribute, the target variable and the predictions column if the user already has their own predictions. Otherwise, this last option is to provide the features for classification. Finally, after picking a reference group from the levels of the sensitive attribute (like picking "male" in "sex"), the user can continue with answering questions for automatically assessing the suitable fairness criteria for their needs.

The section for automatically assessing fairness criteria can be seen in Figure 4.6. The users see a question that, depending on the previous answer, can change the followup questions and selected fairness criteria. The full options that a user can see, is depicted in 4.4. Users will see three questions at most.

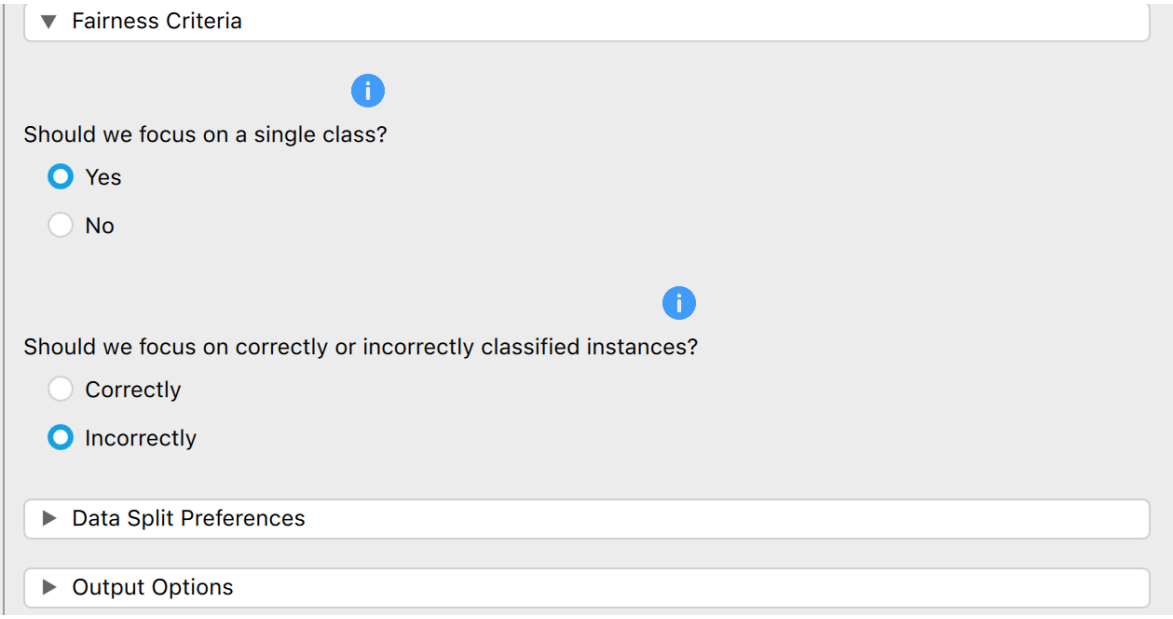

Figure 4.6: Dynamic questions for automatic fairness criteria.

After evaluating the questions and picking the answers best tailored for the user's case, we depict a few results on screen. The first of which being the raw values of the fairness measures that have been automatically selected, as can be seen in Figure 4.7. All the subgroups in the sensitive attribute, accompanied with the values of the specific subgroup and fairness measure combination are shown in this table. Notable in this table is that the reference group ("Caucasian" in this case) will always have a value of 1, as the ratio is calculated by dividing with itself.

Comparison of Fairness Measures Between Groups ▼

Sensitive Groups	FPRP	TPRP
African-American	1.2	1.1
Asian	0.43	1.3
Caucasian	1	1
Hispanic	1	0.92
Native American	0.58	1.3
Other	1	0.92

Figure 4.7: Resulting table of the audit with Caucasian as reference group in COMPAS.

To get a more intuitive feel for the results, we also show users a graph, an example of which can be seen in Figure 4.8. In this graph, the automatically selected fairness measures are depicted for the respective subgroups in the sensitive attribute. The fairness threshold range is also highlighted. With this, we can intuitively see whether there is parity, as all bars should be in the range of the fairness threshold if that is the case, which it is not here. Just like the table, the bars of the reference group will have a value of 1 here.

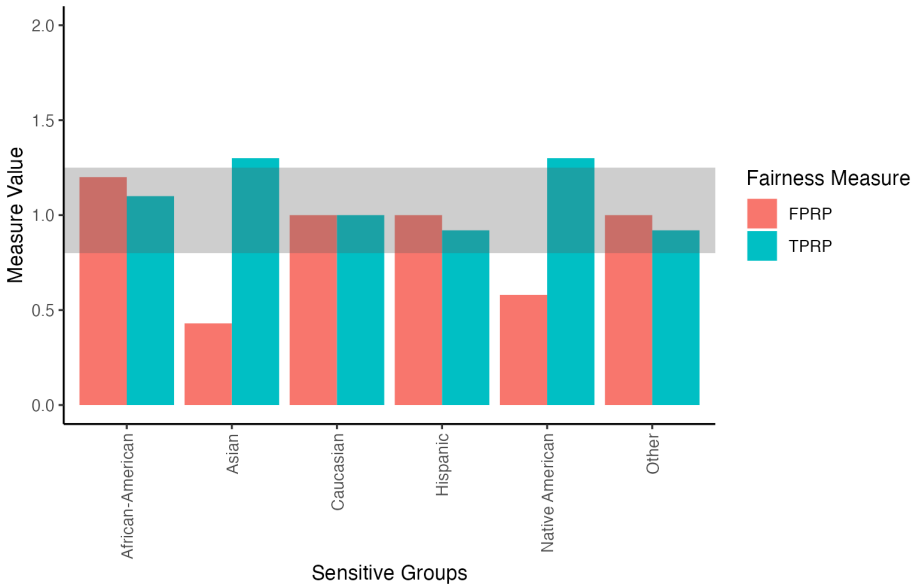

Figure 4.8: Resulting graph of the audit with Caucasian as reference group in COMPAS.

4.7. Code and Application Availability

Since we want our toolkit to be openly and freely available, we had to pick a program that allowed us to do that. We have chosen to use JASP for this purpose (JASP Team, 2023). We specifically chose JASP due to its public availability and transparency as it is open-source and free. The user-interface of JASP also aids the toolkit in the sense that this is completely interactive. This helps us with our implementation for our interactive questions. A version of JASP is needed to use this toolkit. The code for this toolkit is also openly available and can be found on the following GitHub Repository: `https://github.com/selimbuyuk/jaspAudit`.

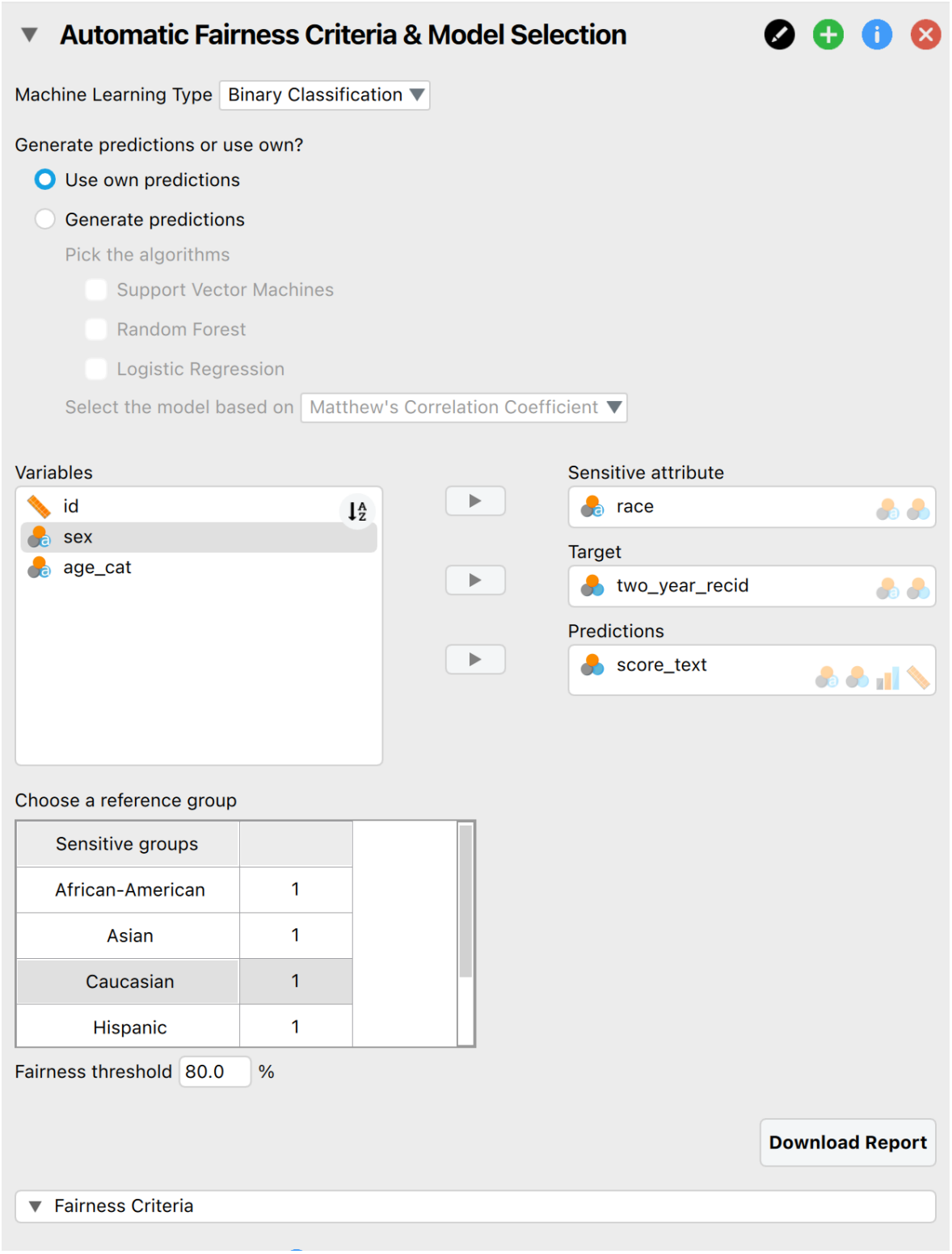

Figure 4.5: User interface of our implementation in JASP.

5. Discussion and Conclusion

In this chapter, we give a quick overview of our results while going over the implications of them. We then discuss the limitations this work has and what future work can do to strengthen the findings. Finally, we will discuss how the findings of our work associate with AI and what its relevance is.

5.1. Overview of the Results

This work was set out to fill the gap of current state of the art in fairness toolkit where automation of fairness criteria in composition with automation of model selection was the goal. We proposed a methodology where four critical steps had been depicted, viz. determining characteristics of the fairness measures, creating a base decision tree, mutating the base tree to create candidate trees and finally evaluating all the trees to get the best one. We have shown that this method works by going over results and showing the implementation in JASP. As such, we can go over the research question and research subquestions (which we posed in Section 2.6.4) and answer them, by starting with the first research subquestion.

This question was about finding the most suitable model for fair model selection. For this, we considered two cases. The first case being that there is just one model performing better than the rest. In this case, we pick the model solely based on a performance measure specified by the user, but MCC by default, as Chicco et al. (2021) argue that this is the most informative measure over others. This model is then used to calculate the fairness measures. On the other hand, we have a case where there are multiple models performing the best. In this case, we hold a tiebreaker and give the user to choice to pick between the best- or worse case scenario for the fairness measures if they are also the same. In short, we used performance metrics to determine the best model. And if there were any ties, that is to say, when the top $k > 1$ models are not statistically significantly better than one another, we let the user choose the best- or worse case regarding the fairness measures.

The second question was about the evaluation and automation for the most suitable fairness criteria. Here, we also posed how we could evaluate a decision tree whose internal nodes were composed of interactive questions, and know what kind of a decision tree was optimal for our purposes. By going over our results and implementation, we can provide the evaluation function that we have used for selecting the most suitable decision tree. Our evaluation function, which can be seen in Equation 4.1, uses the average rules with the amount of leaves in a decision tree. By adding these two variables, we get the evaluation value for a tree. By minimising this value, we get the best tree suited for our needs. This was a balanced tree in our case, since we sought after a compact tree and tailored our evaluation function specifically towards that goal.

Finally, with the methodology we laid out in Chapter 3 and the results we have obtained and shown in Chapter 4, we can answer the main research problem by saying that yes, it is possible to automate fairness criteria and fair model selection for ML tasks via an interactive interface.

5.2. Limitations and Future Work

What we have shown in this work is a working minimum viable product. This can be improved in a multitude of ways. By going over limitations, we can propose steps to improve this toolkit in the future. Firstly, this implementation has not been extensively tested with users due to time constraints. Since our goal is to lay some groundwork for helping users with assessing fairness in their datasets, a good next step for future work would be to test this methodology with actual users, like auditors. Their feedback would lead to a more polished toolkit, as explainability and understanding of definitions regarding fairness among users is one of our main goals. To get more valuable feedback, questionnaires like the System Usability Scale (Jordan, Thomas, McClelland, & Weerdmeester, 1996; Bangor, Kortum, & Miller, 2008) can be used, to test varying versions of the toolkit.

Another limitation in this work was the selection of 10 fairness measures. Since the literature is full of other fairness measures, future work can delve deeper into adapting them into this toolkit as this work has proven that it is possible to automate

automatic fairness criteria. This also holds true for the category of fairness measure we have used. As we have only selected a few measures from the statistical fairness measures. More advanced measure, like some of measures we discussed in the individual- or causal fairness measures can be used to make the toolkit more robust.

Next, a different approach to the evaluation function can be considered. In our evaluation function, we assumed that the fairness measures are equally as likely to be selected. That is to say that we operated with a naive approach and therefore assumed that the weights of all of the fairness measures were constant. However, future work can look into the distribution of fairness measures and reflect this into the function. For instance, if a fairness measure, like demographic parity, is used more often than accuracy parity, then this can be reflected in the evaluation function in the form of a *weighted* sum. A fairness measures that is used more often, should be given as an option after fewer questions. These have to be more to the root of the tree, instead of lower depths. Future work can look into adjusting these weights to get better decision trees and thus questions.

In our case, we used standard binary classification algorithms to predict labels. In Section 2.5.1, we described fairness-aware machine learning algorithms. These are machine learning algorithms specifically designed with fairness in mind. Thus, future work could implement this type of algorithm in the toolkit and evaluate their performance as opposed to using standard machine learning algorithms. If their performance is indeed better, users could opt to use these algorithms instead of the standard ones. In the same trend, we propose the implementation of other ML types for different tasks, like regression and multi-class classification to accommodate more problems this toolkit can assist with.

Lastly, we propose the ability to mitigate bias in future version of this toolkit. As the first step of this project was to assess whether there was any bias in a dataset, the next logical step would be to mitigate bias if there is any. For this purpose, we propose the use of methods and approaches discussed in Section 2.4.

5.3. Relevance for AI

With this work, we set out to provide evidence that assisting users automatically with selecting fairness criteria was possible. This has been made evident with a methodology to follow, as well as an implementation in an open-source environment, freely to use for anyone. By doing so, we helped to bridge fairness and users that wanted to audit their data, in an emerging and booming field called Responsible AI. Not only do we think that we have set out a stepping stone for future researchers and practitioners in responsible AI to adapt this toolkit for more and refined fairness criteria. We also believe that we have set a precedent by showing that automation in this field can be done.

References

Abebe, S. A., Lucchese, C., & Orlando, S. (2022). Eifffel: enforcing fairness in forests by flipping leaves. In *Proceedings of the 37th acm/sigapp symposium on applied computing* (pp. 429–436).

Abraham, S. S., Sundaram, S. S., et al. (2019). Fairness in clustering with multiple sensitive attributes. *arXiv preprint arXiv:1910.05113*.

Adler, P., Falk, C., Friedler, S. A., Nix, T., Rybeck, G., Scheidegger, C., ... Venkatasubramanian, S. (2018). Auditing black-box models for indirect influence. *Knowledge and Information Systems*, *54*(1), 95–122.

Angwin, J., Larson, J., Mattu, S., & Kirchner, L. (2016). Machine bias. In *Ethics of data and analytics* (pp. 254–264). Auerbach Publications.

Arrieta, A. B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., ... others (2020). Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, *58*, 82–115.

Asaro, P. M. (2019). Ai ethics in predictive policing: From models of threat to an ethics of care. *IEEE Technology and Society Magazine*, *38*(2), 40–53.

Baeza-Yates, R. (2018). Bias on the web. *Communications of the ACM*, *61*(6), 54–61.

Baker, R. S., & Hawn, A. (2022). Algorithmic bias in education. *International Journal of Artificial Intelligence in Education*, *32*(4), 1052–1092.

Bangor, A., Kortum, P. T., & Miller, J. T. (2008). An empirical evaluation of the system usability scale. *Intl. Journal of Human–Computer Interaction*, *24*(6), 574–594.

Barocas, S., Crawford, K., Shapiro, A., & Wallach, H. (2017). The problem with bias: Allocative versus representational harms in machine learning. In *9th annual conference of the special interest group for computing, information and society.*

Bastani, O., Zhang, X., & Solar-Lezama, A. (2019). Probabilistic verification of fairness properties via concentration. *Proceedings of the ACM on Programming Languages*, *3*(OOPSLA), 1–27.

Bellamy, R. K., Dey, K., Hind, M., Hoffman, S. C., Houde, S., Kannan, K., ... oth-

ers (2019). Ai fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM Journal of Research and Development*, *63*(4/5), 4–1.

Ben-Porat, O., Sandomirskiy, F., & Tennenholtz, M. (2021). Protecting the protected group: Circumventing harmful fairness. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 35, pp. 5176–5184).

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

Binns, R. (2020). On the apparent conflict between individual and group fairness. In *Proceedings of the 2020 conference on fairness, accountability, and transparency* (pp. 514–524).

Bird, S., Dudík, M., Edgar, R., Horn, B., Lutz, R., Milan, V., ... Walker, K. (2020). Fairlearn: A toolkit for assessing and improving fairness in ai. *Microsoft, Tech. Rep. MSR-TR-2020-32*.

Buolamwini, J., & Gebru, T. (2018). Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency* (pp. 77–91).

Castelnovo, A., Crupi, R., Greco, G., Regoli, D., Penco, I. G., & Cosentini, A. C. (2022). A clarification of the nuances in the fairness metrics landscape. *Scientific Reports*, *12*(1), 4209.

Castillo, C. (2019). Fairness and transparency in ranking. In *Acm sigir forum* (Vol. 52, pp. 64–71).

Caton, S., & Haas, C. (2020). Fairness in machine learning: A survey. *arXiv preprint arXiv:2010.04053*.

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785–794).

Chicco, D., & Jurman, G. (2020). The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. *BMC genomics*, *21*, 1–13.

Chicco, D., Tötsch, N., & Jurman, G. (2021). The matthews correlation coefficient

(mcc) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation. *BioData mining*, *14*(1), 1–22.

Choi, S., & Rainey, H. G. (2014). Organizational fairness and diversity management in public organizations: Does fairness matter in managing diversity? *Review of Public Personnel Administration*, *34*(4), 307–331.

Clarke, K. A. (2005). The phantom menace: Omitted variable bias in econometric research. *Conflict management and peace science*, *22*(4), 341–352.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine learning*, *20*, 273–297.

Crawford, K. (2017). The trouble with bias. In *Conference on neural information processing systems, invited speaker.*

Cruz, A. F., Belém, C., Bravo, J., Saleiro, P., & Bizarro, P. (2022). Fairgbm: Gradient boosting with fairness constraints. *arXiv preprint arXiv:2209.07850*.

Danks, D., & London, A. J. (2017). Algorithmic bias in autonomous systems. In *Ijcai* (Vol. 17, pp. 4691–4697).

Das, A., & Rad, P. (2020). Opportunities and challenges in explainable artificial intelligence (xai): A survey. *arXiv preprint arXiv:2006.11371*.

Deng, W. H., Nagireddy, M., Lee, M. S. A., Singh, J., Wu, Z. S., Holstein, K., & Zhu, H. (2022). Exploring how machine learning practitioners (try to) use fairness toolkits. In *2022 acm conference on fairness, accountability, and transparency* (pp. 473–484).

Dwork, C., Hardt, M., Pitassi, T., Reingold, O., & Zemel, R. (2012). Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference* (pp. 214–226).

Dwork, C., Immorlica, N., Kalai, A. T., & Leiserson, M. (2018). Decoupled classifiers for group-fair and efficient machine learning. In *Conference on fairness, accountability and transparency* (pp. 119–133).

Fan, W., Davidson, I., Zadrozny, B., & Yu, P. S. (2005). An improved categorization of classifier's sensitivity on sample selection bias. In *Fifth ieee international conference on data mining (icdm'05)* (pp. 4–pp).

Feldman, M., Friedler, S. A., Moeller, J., Scheidegger, C., & Venkatasubramanian, S. (2015). Certifying and removing disparate impact. In *proceedings of the 21th acm sigkdd international conference on knowledge discovery and data mining* (pp. 259–268).

Feuz, M., Fuller, M., & Stalder, F. (2011). Personal web searching in the age of semantic capitalism: Diagnosing the mechanisms of personalisation. *First Monday*.

Fraenkel, A. (2020). *Fairness amp; algorithmic decision making.* Retrieved from `https://afraenkel.github.io/fairness-book/content/05-parity-measures.html`

Friedman, B., & Nissenbaum, H. (1996). Bias in computer systems. *ACM Transactions on Information Systems (TOIS)*, *14*(3), 330–347.

Geiser, S. (2020). Sat/act scores, high-school gpa, and the problem of omitted variable bias: Why the uc taskforce's findings are spurious. research & occasional paper series: Cshe. 1.2020. *Center for Studies in Higher Education*.

Geman, S., Bienenstock, E., & Doursat, R. (1992). Neural networks and the bias/variance dilemma. *Neural computation*, *4*(1), 1–58.

Halevy, M., Harris, C., Bruckman, A., Yang, D., & Howard, A. (2021). Mitigating racial biases in toxic language detection with an equity-based ensemble framework. In *Equity and access in algorithms, mechanisms, and optimization* (pp. 1–11).

Hardt, M., Price, E., & Srebro, N. (2016). Equality of opportunity in supervised learning. *Advances in neural information processing systems*, *29*.

Hastie, T., Tibshirani, R., Friedman, J., Hastie, T., Tibshirani, R., & Friedman, J. (2009). Random forests. *The elements of statistical learning: Data mining, inference, and prediction*, 587–604.

Hattatoglu, B., Khwaileh, E., Qahtan, A., Kaya, H., & Velegrakis, Y. (2021). Coscfair: Ensuring subgroup fairness through fair classification framework.

Hewig, J., Kretschmer, N., Trippe, R. H., Hecht, H., Coles, M. G., Holroyd, C. B., & Miltner, W. H. (2011). Why humans deviate from rational choice. *Psychophysiology*, *48*(4), 507–514.

Huang, X., Li, Z., Jin, Y., & Zhang, W. (2022). Fair-adaboost: Extending adaboost method to achieve fair classification. *Expert Systems with Applications*, *202*,

117240.

JASP Team. (2023). *JASP (Version 0.17)[Computer software].* Retrieved from `https://jasp-stats.org/`

Jensen, D., & Neville, J. (2002). Linkage and autocorrelation cause feature selection bias in relational learning. In *Icml* (Vol. 2, pp. 259–266).

Jiang, H., & Nachum, O. (2020). Identifying and correcting label bias in machine learning. In *International conference on artificial intelligence and statistics* (pp. 702–712).

Jordan, P. W., Thomas, B., McClelland, I. L., & Weerdmeester, B. (1996). *Usability evaluation in industry.* CRC Press.

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

Kamishima, T., Akaho, S., Asoh, H., & Sakuma, J. (2012). Fairness-aware classifier with prejudice remover regularizer. In *Machine learning and knowledge discovery in databases: European conference, ecml pkdd 2012, bristol, uk, september 24-28, 2012. proceedings, part ii 23* (pp. 35–50).

Kancevičienė, N. (2019). Insurance, smart information systems and ethics. *ORBIT Journal*, *2*(2).

Kay, M., Matuszek, C., & Munson, S. A. (2015). Unequal representation and gender stereotypes in image search results for occupations. In *Proceedings of the 33rd annual acm conference on human factors in computing systems* (pp. 3819–3828).

Kilbertus, N., Rojas Carulla, M., Parascandolo, G., Hardt, M., Janzing, D., & Schölkopf, B. (2017). Avoiding discrimination through causal reasoning. *Advances in neural information processing systems*, *30*.

Kirdemir, B., & Agarwal, N. (2022). Exploring bias and information bubbles in youtube's video recommendation networks. In *Complex networks & their applications x: Volume 2, proceedings of the tenth international conference on complex networks and their applications complex networks 2021 10* (pp. 166–177).

Kleinberg, J., Ludwig, J., Mullainathan, S., & Rambachan, A. (2018). Algorithmic fairness. In *Aea papers and proceedings* (Vol. 108, pp. 22–27).

Kodiyan, A. A. (2019). An overview of ethical issues in using ai systems in hiring with

a case study of amazon's ai based hiring tool. *Researchgate Preprint*, 1–19.

Kop, M. (2021). Eu artificial intelligence act: the european approach to ai..

Kozodoi, N., & V. Varga, T. (2021). fairness: Algorithmic fairness metrics [Computer software manual]. Retrieved from `https://CRAN.R-project.org/package=fairness` (R package version 1.2.1)

Kusner, M. J., Loftus, J., Russell, C., & Silva, R. (2017). Counterfactual fairness. *Advances in neural information processing systems*, *30*.

Lamb, S. (2010). Toward a sexual ethics curriculum: Bringing philosophy and society to bear on individual development. *Harvard Educational Review*, *80*(1), 81–106.

Langenkamp, M., Costa, A., & Cheung, C. (2020). Hiring fairly in the age of algorithms. *arXiv preprint arXiv:2004.07132*.

Langston, J. (2015). Who'sa ceo? google image results can shift gender biases. *UW News, April*.

Lee, M. S. A. (2019). Context-conscious fairness in using machine learning to make decisions. *AI Matters*, *5*(2), 23–29.

Lee, M. S. A., & Singh, J. (2021). The landscape and gaps in open source fairness toolkits. In *Proceedings of the 2021 chi conference on human factors in computing systems* (pp. 1–13).

Lerman, K., & Hogg, T. (2014). Leveraging position bias to improve peer recommendation. *PloS one*, *9*(6), e98914.

Li, P., & Liu, H. (2022). Achieving fairness at no utility cost via data reweighing with influence. In *International conference on machine learning* (pp. 12917–12930).

Liu, H., Zhang, X., Shen, X., & Sun, H. (2022). A fair and efficient hybrid federated learning framework based on xgboost for distributed power prediction. *arXiv preprint arXiv:2201.02783*.

Loftus, J. R., Russell, C., Kusner, M. J., & Silva, R. (2018). Causal reasoning for algorithmic fairness. *arXiv preprint arXiv:1805.05859*.

Manisha, P., & Gujar, S. (2018). Fnnc: Achieving fairness through neural networks. *arXiv preprint arXiv:1811.00247*.

Martinez-Eguiluz, M., Irazabal-Urrutia, O., & Arbelaitz-Gallego, O. (2021). Towards fairness in classification: Comparison of methods to decrease bias. In *Advances in*

artificial intelligence: 19th conference of the spanish association for artificial intelligence, caepia 2020/2021, málaga, spain, september 22–24, 2021, proceedings 19 (pp. 86–95).

McCelloch, W., & Pitts, W. (1943). A logical calculus of the idea immanent in neural nets. *Bulletin of Mathematical Biophysics*, *5*, 115–133.

Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., & Galstyan, A. (2021). A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, *54*(6), 1–35.

Mulvaney, E. (2021). *Nyc targets artificial intelligence bias in hiring under new law.* Retrieved 2021-12-10, from `https://news.bloomberglaw.com/daily-labor-report/nyc-targets-artificial-intelligence-bias-in-hiring-under-new-law`

Nguyen, D., Gravel, R., Trieschnigg, D., & Meder, T. (2013). " how old do you think i am?" a study of language and age in twitter. In *Proceedings of the international aaai conference on web and social media* (Vol. 7, pp. 439–448).

Olteanu, A., Castillo, C., Diaz, F., & Kıcıman, E. (2019). Social data: Biases, methodological pitfalls, and ethical boundaries. *Frontiers in big data*, *2*, 13.

O'neil, C. (2017). *Weapons of math destruction: How big data increases inequality and threatens democracy.* Crown.

Pagano, T. P., Loureiro, R. B., Lisboa, F. V., Peixoto, R. M., Guimarães, G. A., Cruz, G. O., ... others (2023). Bias and unfairness in machine learning models: A systematic review on datasets, tools, fairness metrics, and identification and mitigation methods. *Big data and cognitive computing*, *7*(1), 15.

Paka, A. (2022). *Eu mandates explainability and monitoring in proposed gdpr of ai: Fiddler ai blog.* Retrieved from `https://www.fiddler.ai/blog/eu-mandates-explainability-and-monitoring-in-proposed-gdpr-of-ai`

Panigutti, C., Perotti, A., Panisson, A., Bajardi, P., & Pedreschi, D. (2021). Fairlens: Auditing black-box clinical decision support systems. *Information Processing & Management*, *58*(5), 102657.

Park, S., Byun, J., & Lee, J. (2022). Privacy-preserving fair learning of support vector machine with homomorphic encryption. In *Proceedings of the acm web conference*

2022 (pp. 3572–3583).

Pessach, D., & Shmueli, E. (2022). A review on fairness in machine learning. *ACM Computing Surveys (CSUR)*, *55*(3), 1–44.

Pleiss, G., Raghavan, M., Wu, F., Kleinberg, J., & Weinberger, K. Q. (2017). On fairness and calibration. *Advances in neural information processing systems*, *30*.

Quinlan, J. R. (1987). Simplifying decision trees. *International journal of man-machine studies*, *27*(3), 221–234.

Rajabi, A., & Garibay, O. O. (2022). Tabfairgan: Fair tabular data generation with generative adversarial networks. *Machine Learning and Knowledge Extraction*, *4*(2), 488–501.

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

Riazy, S., Simbeck, K., & Schreck, V. (2020). Fairness in learning analytics: Student at-risk prediction in virtual learning environments. In *Csedu (1)* (pp. 15–25).

Rieskamp, J., Hofeditz, L., Mirbabaie, M., & Stieglitz, S. (2023). Approaches to improve fairness when deploying ai-based algorithms in hiring–using a systematic literature review to guide future research. In *Hawaii international conference on system sciences.*

Ryan, M. (2020). The future of transportation: ethical, legal, social and economic impacts of self-driving vehicles in the year 2025. *Science and engineering ethics*, *26*(3), 1185–1208.

Ryu, H. J., Adam, H., & Mitchell, M. (2017). Inclusivefacenet: Improving face attribute detection with race and gender diversity. *arXiv preprint arXiv:1712.00193*.

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

Saxena, N. A. (2019). Perceptions of fairness. In *Proceedings of the 2019 aaai/acm conference on ai, ethics, and society* (pp. 537–538).

Saxena, N. A., Huang, K., DeFilippis, E., Radanovic, G., Parkes, D. C., & Liu, Y. (2019). How do fairness definitions fare? examining public attitudes towards algorithmic definitions of fairness. In *Proceedings of the 2019 aaai/acm conference*

on ai, ethics, and society (pp. 99–106).

Smelyakov, K., Hurova, Y., & Osiievskyi, S. (2023). Analysis of the effectiveness of using machine learning algorithms to make hiring decisions. *Proceedings http://ceur-ws. org ISSN*, *1613*, 0073.

Sonoda, R. (2021). A pre-processing method for fairness in ranking. *arXiv preprint arXiv:2110.15503*.

Spirtes, P. L., Meek, C., & Richardson, T. S. (2013). Causal inference in the presence of latent variables and selection bias. *arXiv preprint arXiv:1302.4983*.

Suresh, H., & Guttag, J. (2021). A framework for understanding sources of harm throughout the machine learning life cycle. In *Equity and access in algorithms, mechanisms, and optimization* (pp. 1–9).

Tao, G., Sun, W., Han, T., Fang, C., & Zhang, X. (2022). Ruler: discriminative and iterative adversarial training for deep neural network fairness. In *Proceedings of the 30th acm joint european software engineering conference and symposium on the foundations of software engineering* (pp. 1173–1184).

Telford, T. (2019). Apple card algorithm sparks gender bias allegations against goldman sachs. *The Washington Post*.

Tilimbe, J. (2019). Ethical implications of predictive risk intelligence. *The ORBIT Journal*, *2*(2), 1–28.

Verma, S. (2019). Weapons of math destruction: how big data increases inequality and threatens democracy. *Vikalpa*, *44*(2), 97–98.

Verma, S., & Rubin, J. (2018). Fairness definitions explained. In *2018 ieee/acm international workshop on software fairness (fairware)* (pp. 1–7).

Walsh, C., Stein, M. M., Tapping, R., Smith, E. M., & Holmes, N. (2021). Exploring the effects of omitted variable bias in physics education research. *Physical Review Physics Education Research*, *17*(1), 010119.

Wang, T., & Saar-Tsechansky, M. (2020). Augmented fairness: An interpretable model augmenting decision-makers' fairness. *arXiv preprint arXiv:2011.08398*.

Wang, Y., & Singh, L. (2021). Analyzing the impact of missing values and selection bias on fairness. *International Journal of Data Science and Analytics*, *12*(2), 101–119.

Weng, C. G., & Poon, J. (2008). A new evaluation measure for imbalanced datasets. In *Proceedings of the 7th australasian data mining conference-volume 87* (pp. 27–32).

Wenink, E. (2021). *Group fairness.* Retrieved from `https://edwinwenink.github.io/ai-ethics-tool-landscape/fairness/group-fairness/`

Whittaker, M., Crawford, K., Dobbe, R., Fried, G., Kaziunas, E., Mathur, V., ... others (2018). *Ai now report 2018.* AI Now Institute at New York University New York.

Xu, D., Yuan, S., Zhang, L., & Wu, X. (2018). Fairgan: Fairness-aware generative adversarial networks. In *2018 ieee international conference on big data (big data)* (pp. 570–575).

Yao, J., & Shepperd, M. (2020). Assessing software defection prediction performance: Why using the matthews correlation coefficient matters. *Proceedings of the evaluation and assessment in software engineering*, 120–129.

Zafar, M. B., Valera, I., Rodriguez, M., Gummadi, K., & Weller, A. (2017). From parity to preference-based notions of fairness in classification. *Advances in neural information processing systems*, *30*.

Zemel, R., Wu, Y., Swersky, K., Pitassi, T., & Dwork, C. (2013). Learning fair representations. In *International conference on machine learning* (pp. 325–333).

Zhang, W., Bifet, A., Zhang, X., Weiss, J. C., & Nejdl, W. (2021). Farf: A fair and adaptive random forests classifier. In *Advances in knowledge discovery and data mining: 25th pacific-asia conference, pakdd 2021, virtual event, may 11–14, 2021, proceedings, part ii* (pp. 245–256).