An Introduction to Algorithmic Fairness

Lecture Notes 2IX30

Hilde Weerts
Eindhoven University of Technology
h.j.p.weerts@tue.nl

March 3, 2021

In recent years, several examples have shown how algorithmic systems can reproduce, amplify, or even introduce unfairness in our societies. In this chapter, we will introduce several ways in which machine learning can result in discrimination (Section 1), discuss notions of fairness proposed in the computer science literature (Section 2), and explore some of the underlying causes of unfair predictions (Section 3).

1 Algorithmic Fairness

Many machine learning applications make predictions about people. For example, algorithmic systems may be used to decide whether a resume makes it through the first selection round, judge the severity of a medical condition, or determine whether somebody will receive a loan. As that these systems are usually trained on massive amounts of data, they have the potential to be more consistent than human decision-makers with varying levels of experience. For example, consider a resume screening process. In a non-automated scenario, the likelihood to get through the resume selection round can depend on the personal beliefs of the recruiter who happens to judge your resume. On the other hand, the predictions of an algorithmic resume screening system can be learned from the collective judgement of many different recruiters.

However, the workings of a machine learning model heavily depend on how the machine learning task is formulated and which data is used to train the model. As such, prejudices against particular groups can seep into the model in each step of the development process. For example, if in the past a company has hired more men than women, this will be reflected in the training data. The machine learning model is likely to pick up this pattern. In fact, this is precisely what happened when Amazon tried to train a resume screening model[1]. The model did not explicitly take into account the applicant's gender. However, it turned out that the model penalized resumes that included terms that suggested that the applicant was female. For example, resumes that included the word "women's" (e.g., in "women's chess club captain") were less likely to be selected.

Notably, the characteristics that could potentially make algorithmic systems desirable over human-decision making, also amplify fairness related risks. One prejudiced recruiter can judge a few dozen resumes each day, but an algorithmic system can process thousands of resumes in the blink of an eye. As such, if an algorithmic system is biased in any way, harmful consequences will be structural and exceptionally scalable.

Even in applications where predictions do not directly consider individuals, people can be unfairly impacted [Barocas et al., 2019]. For example, a machine learning model that predicts the future value of houses can influence the actual sale prices. If some neighborhoods receive much lower house price predictions than others, this may disproportionately affect some groups over others.

Discrimination and bias of algorithmic systems is not a new problem. Well over two decades ago, Friedman and Nissenbaum [1996] analyzed problems of algorithmic fairness. However, with

[1] `https://www.reuters.com/article/us-amazon-com-jobs-automation-insight-idUSKCN1MK08G`

the increasing use of algorithmic systems, it has become clear that the issue is far from solved. Researchers from a range of disciplines have started working on unraveling the mechanisms in which algorithmic systems can undermine fairness and how these risks can be mitigated. This has given rise to the research field of *algorithmic fairness*: *the idea that algorithmic systems should behave or treat people fairly, i.e., without discrimination on the grounds of sensitive characteristics such as age, sex, disability, ethnic or racial origin, religion or belief, or sexual orientation.*

In this definition, a *sensitive characteristic* refers to a characteristic of an individual such that any decisions based on this characteristic are considered undesirable from an ethical or legal point of view. Note that our definition of algorithmic fairness is very broad. This is intentional. The concept is applicable to all types of algorithmic systems, including different flavors of artificial intelligence (e.g., symbolic approaches, expert systems, and machine learning), but also simple rule-based systems. In this introduction to algorithmic fairness, we will limit the discussion mostly to fairness of machine learning-based systems.

1.1 Types of Harm

The exact meaning of 'behaving or treating people fairly' heavily depends on the sociotechnical context of the algorithmic system. There are several different ways in which algorithmic systems can disregard fairness.

- *Allocation harm* can be defined as an unfair allocation of opportunities, resources, or information [Madaio et al., 2020]. In our resume selection example, allocation harm occurs when some groups are selected less often than others, e.g., the algorithm selects men more often than women.

- *Quality-of-service harm* occurs when a system disproportionately fails for certain (groups of) people [Madaio et al., 2020]. For example, a facial recognition system may misclassify black women at a higher rate than white men [Buolamwini and Gebru, 2018] and a speech recognition system may not work well for users whose disability impacts their clarity of speech [Guo et al., 2019]. In our resume selection example, quality-of-service harm occurs when some groups are more often wrongly rejected than others; e.g., qualified women are selected at lower rates than qualified men.

- *Stereotyping harm* occurs when a system reinforces undesirable and unfair societal stereotypes [Madaio et al., 2020]. Stereotyping harms are particularly prevalent in natural language processing and computer vision systems, as societal stereotypes are often deeply embedded in text corpuses and image labels. For example, an image search for "CEO" may primarily show photos of white men.

- *Denigration harm* refers to situations in which algorithmic systems are actively derogatory or offensive [Madaio et al., 2020]. For example, an automated tagging system may misclassify people as gorillas[2] and a chat bot might start using derogatory slurs[3].

- *Representation harm* occurs when the development and usage of algorithmic systems over- or under-represents certain groups of people [Madaio et al., 2020]. For example, some racial groups may be overly scrutinized during welfare fraud investigations or neighborhoods with a high elderly population may be ignored because data on disturbances in the public space (such as potholes[4]) is collected using a smartphone app. Representation harm can be connected to allocation harms and quality-of-service harms. However, a lack of diversity by itself can already be considered a violation of fairness. Moreover, representation harm can already occur even before the algorithmic system makes a prediction, which makes it important to consider from the start.

[2] `https://www.theverge.com/2015/7/1/8880363/google-apologizes-photos-app-tags-two-black-people-gorillas`
[3] `https://fortune.com/2020/09/29/artificial-intelligence-openai-gpt3-toxic/`
[4] `https://hbr.org/2013/04/the-hidden-biases-in-big-data`

- *Procedural harm* occurs when decisions are made in a way that violates social norms [see e.g., Rudin et al., 2018]. For example, penalizing a job applicant for having more experience can be considered a form of procedural harm. Procedural harm is not limited to the prediction-generating mechanisms of the model itself, but can also be extended to the development and usage of the system. For example, is it communicated clearly that an algorithmic decision is made? Do data subjects receive a meaningful justification? Is it possible to appeal a decision? This form of procedural harm is closely related to algorithmic accountablity.

Note that these types of harm are not mutually exclusive and that this list is not complete.

2 Notions of Algorithmic Fairness

With a rising interest in fairer machine learning systems, different notions of algorithmic fairness have been put forward in the computer science literature. Most of these notions focus on allocation harm and quality-of-service harm. Considerably fewer studies consider stereotyping harm, denigration harm, and procedural harm which are arguably even more difficult to formalize. Generally, we can distinguish two lines of work: *group fairness* and *individual fairness*.

2.1 Group Fairness

A straightforward way to approach fairness is to consider whether some groups are treated, on average, worse than other groups. *Group fairness* is a notion of fairness that requires group statistics to be equal across (sub)groups defined by sensitive characteristics. It is sometimes referred to as *statistical fairness*. Group fairness is an intuitive notion of fairness that is relatively easy to put into operation. For example, allocation harm can be quantified as a group fairness metric by comparing a classifier's selection rates across different groups. We can quantify quality-of-service harm and representation harm in a similar way. Note that group fairness is grounded in a *consequentialist* perspective, as it poses that the consequences of the model are crucial for ethical judgement.

2.1.1 No Fairness Through Unawareness

At this point, you may wonder: if we do not want to discriminate against certain groups, why don't we just remove the *sensitive feature* from the dataset? Unfortunately, it is not that simple. To see why, it can help to distinguish between direct and indirect discrimination.

In European Union law, *direct discrimination* refers to cases where (groups of) individuals are treated less favorably based directly on their membership of a protected-by-law group. In the United States, this is also referred to as disparate treatment. An example of direct discrimination is when a person is denied service in a restaurant based on their race. In the context of an algorithmic system, direct discrimination could occur when a machine learning model explicitly uses a sensitive feature to make a prediction. Following this definition of fairness, removing the sensitive feature will prevent discrimination.

Indirect discrimination (or disparate impact in United States labor law), refers to cases where groups or individuals are treated less favorably based on rules that seem neutral, but, as a side effect, disadvantage a protected group. A classic example of indirect discrimination is *redlining*. This refers to a practice in the United States where people were systematically denied services based on their postal code. Neighborhoods that were deemed 'too risky' were outlined on the map in the color red, hence the name redlining. Although postal code may appear to be a neutral feature, it is highly correlated with ethnicity. As services were mostly denied in older, predominantly black neighborhoods, black people were indirectly discriminated by this policy. Another example can be found in loan applications. Imagine we want to avoid allocation harms across genders. We decide to exclude the feature that represents gender from our data set, to will avoid any direct discrimination. However, if we do include occupation, an attribute which is highly gendered in many societies, the model can still identify historical patterns of gender bias. As such,

occupation acts as a *proxy variable* for gender. A proxy variable is a variable that can act as a stand-in variable. For example, the "quality" of a sales employee is impossible to measure directly, but you can measure their sales figures or customer satisfaction ratings. In the case of indirect discrimination, variables included in our model may unintentionally act as a proxy variable for a sensitive characteristics.

This is not just a hypothetical problem. Machine learning algorithms are specifically designed to identify relationships between features in a data set. Hence, if discriminatory patterns exist within the data, it is very likely that a machine learning model will replicate it. Removing all possible proxy variables is usually not a viable approach. First of all, it is not always possible to anticipate the patterns through which the sensitive feature can be approximated by the model. Several features that are slightly predictive of the sensitive feature might, taken together, be an accurate predictor of the sensitive feature. Second, the information proxy variables provide, apart from their relation with the sensitive feature, can be predictive of the target feature. Removing all features that are slightly related to the sensitive feature could therefore substantially reduce the predictive performance of the model.

Clearly, removing sensitive features is unlikely to prevent allocation harm, which can still occur in the form of indirect discrimination. Similarly, patterns of stereotyping and denigration can be deeply embedded in the data and removing the sensitive features will not remove these patterns. Additionally, quality-of-service, representation, and denigration harm can be caused by a lack of information on minority groups, which is not solved by removing the sensitive feature either. To conclude, removing sensitive features is only helpful in achieving a very narrow definition of fairness. The practical consequence is that it is unlikely that this approach will prevent real-world harms.

2.1.2 Conditional Group Fairness

In EU and US law, indirect discrimination in employment may not be unlawful if it is justified by a "legitimate aim". Examples of legal justifications for discrimination are genuine occupational requirement and business necessity. For example, a film producer is allowed to hire only male actors to play a male role, as this is a genuine occupational requirement. Apart from employment, there may be characteristics that, from an ethical perspective, legitimize differences between groups. Loosely inspired by these legal imperatives, Kamiran et al. [2013] put forward a notion of fairness we will refer to as *conditional group fairness*. This is a variant of group fairness that allows for differences between groups, if these differences are explained by a legitimate feature that can be justified by ethics and/or law.

Conditional group fairness can be best illustrated by an example. Imagine a scenario in which women have, on average, a lower income than men. This may imply that women are discriminated. However, in our scenario many women work less hours than men. As such, the observed indirect discrimination can be partly explained by the lower income. Therefore, equalizing income between men and women would mean that women are paid more per hour than men. If we believe unequal hourly wages to be unfair, we can instead equalize income only between women and men who work similar hours. In other words, we minimize the difference that is still present after conditioning on working hours. Conditional group fairness is particularly relevant considering *Simpson's paradox*. This paradox states that if a correlation occurs in several different groups, it may disappear or even reverse when the groups are aggregated (see Figure 1). In Example 1, we see that an analysis that does not consider all relevant characteristics might suggest discrimination in situations that would be considered morally acceptable if all information was known.

Example 1. *Simpson's Paradox: Berkeley University Admissions.* A classic example of Simpson's paradox is Berkeley's university admissions in 1973.

When considering all programs together, women were accepted less often than men, implying a gender bias. However, it turned out that women at Berkeley often apply for competitive programs with a relatively low acceptance rate. As a result, the overall acceptance rate of

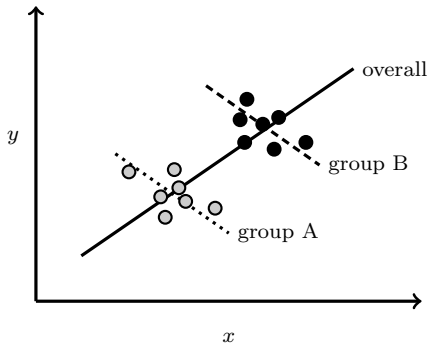

Figure 1: Visualization of Simpson's paradox. The correlation in both group A and group B is negative, whereas the overall regression line has a positive slope.

women in the aggregated data was lower – even though the acceptance rate of women *within* each program was higher than the acceptance rate of men. Hence, if the admission's office would have tried to equalize the overall acceptance rate between men and women, men would have received an even lower acceptance rate.

2.2 Individual Fairness

So far, we have considered fairness from a group perspective. Dwork et al. [2012] put forward an alternative notion of fairness: *individual fairness*. In this line of work, an algorithmic system is deemed fair if similar cases are treated similarly, irrespective of sensitive group membership.

This notion of fairness resembles *situation testing*, a traditional discrimination auditing approach. In this approach, pairs of individuals that are equal, except for their membership of a protected group, are put in the same situation. If the member of a protected group is treated less favorably, this is regarded as discrimination. For example, an auditor may send out pairs of fictional resumes that are identical apart from the gender of the fictional applicant. Notions of individual fairness can be used to quantify allocation harm and quality-of-service harm, by comparing the prediction or performance of the model between pairs of similar instances.

Putting individual fairness to practice can be challenging, as it requires a notion of similarity. A naive approach would be to simply change an instance's sensitive feature and see how this affects the machine learning model's prediction. However, such an approach has limitations similar to removing the sensitive feature from the data set. In particular, there may exist a relationship between the sensitive feature and other features, which mean that the perturbed instances would hardly ever (or never) occur in reality. When we limit ourselves to comparing cases that actually occur in reality, we require a more sophisticated notion of similarity. How can we incorporate the relative importance of features? How can we assess similarity across multiple features with different domains? How similar is similar enough? Defining a similarity metric is a highly non-trivial task.

2.2.1 Counterfactual Fairness

The notions of group fairness and individual fairness discussed so far approach fairness primarily from a statistical point of view. The notions acknowledge that there exist relationships between features, but the associated challenges are approached from a correlation-based perspective. In a different line of work, Kusner et al. [2017] propose to take an explicit causal approach leveraging causal models.

Causal models are mathematical models, typically graphs, that represent causal relationships between different features. Causal models cannot be learned from data alone: we cannot know from observations whether the crowing of the rooster caused the sun to rise, or the other way around. Instead, the structure of a causal model is based on assumptions about causal relationships,

grounded in our understanding of how the world works. Observational data can be used to learn the strength of relationships. An interesting feature of causal models is that they allow for answering counterfactual questions: *"what would have happened if..."?*

In the context of fairness, a causal model can help to determine what the model would have predicted if the individual had belonged to another group. *Counterfactual fairness* is a notion of fairness that requires that the treatment of an individual in the actual world is the same as the treatment of the individual in a counterfactual world where the individual belonged to a different sensitive group. Similar to individual fairness, counterfactual fairness considers fairness from an individual perspective. However, rather than considering a 'close-enough' world by means of a similarity metric, counterfactual fairness criteria are informed by counterfactual probabilities inferred from a causal model. By approaching individual fairness through a causal lens, assumptions about relationships between features can be made explicit. This alleviates some of the issues involved with defining a similarity metric. The challenge, of course, moves from defining a similarity metric to defining a causal model. As mentioned before, the structure of a causal model cannot be identified from data alone. Instead, it requires a deep understanding of the data generating process. Different assumptions about this process can lead to vastly different models and, consequently, different conclusions regarding fairness. A more philosophical challenge of notions of individual fairness is that sensitive group membership is often an important part of somebody's identity. Consequently, one may wonder what it means for an individual to "belong to a different sensitive group".

3 Biases as Sources of Unfairness

In the previous section we have already touched upon some of the mechanisms through which social bias may be replicated from historical decision-making. However, this is not the entire story. Algorithmic systems are an accumulation of design choices that embed the developers' explicit and implicit value judgements into the system [Wieringa, 2020]. Consequently, biases can seep into the system in many different places of the development process. In this section, we will explore biases as sources of unfairness in different parts of the machine learning development process.

3.1 Types of Bias

But first, let us define more precisely what we mean by bias. Generally speaking, *bias* is a systematic and disproportionate tendency towards something.

3.1.1 Social Bias

In everyday language, bias often considers prejudice against a person or a group, which we will refer to as *social bias*. Social bias is a form of *cognitive bias*: a systematic error in rational thinking that can affect judgment and decision-making. Most cognitive biases are a result of the limitations of information-processing capabilities of the human brain. From an evolutionary perspective, these biases are useful because they allow people to make quick decisions in critical scenarios. As a hunter-gatherer, you would probably rather be safe than sorry when encountering an unknown group of other humans. However, these shortcuts often come at the cost of the quality of decisions. In particular, stereotypes formed by social bias can be overgeneralized and inaccurate, especially on an individual level. When people act on social biases, they can result in discriminatory practices. It is important to realize that everybody has some degree of conscious or subconscious social bias. Awareness of cognitive biases, including social bias, can help to signal and mitigate their effects. In particular, a diverse team and critical self-reflection can help to signal social biases and avoid acting on them.

3.1.2 Statistical Bias

In statistics, bias refers to a systematic error in the estimation of parameters or variables. *statistical bias* can be the result of data collection practices that compromise the accuracy of the estimate, such as a particular sampling procedure. This form of statistical bias can be rooted in cognitive biases of the researcher or data subjects. Statistical bias can also refer to systematic errors caused by assumptions of the estimator. In the context of machine learning algorithms, this type of bias is often discussed in relation to the bias-variance trade-off. For example, some machine learning algorithms can only learn linear relationships between features, whereas the true underlying data distribution exhibits more complex relationships, resulting in an underfitting model. Note that this definition of bias is of a purely technical nature and does not connote cognitive or social biases.

Although all these different types of bias can result in fairness related harms, most issues arise at the intersection of social bias and statistical bias. In the remainder of this section, we will dive more deeply into different types of biases at each stage of the development process. Disclaimer: this list is not exhaustive. Moreover, as we will see, biases are hard to precisely dissect and often overlap. In practice, it is usually difficult if not impossible to know exactly which biases are at play. Luckily, a thoughtful development process that leaves room for self-reflection already goes a long way in mitigating harms.

3.2 Problem Understanding: Abstraction Traps

Translating a real-world problem into a machine learning task can be difficult. By definition, a model is a simplification of reality. A data scientist's task is to decide which elements of the real world need to be included in the model and which elements will be left out of scope. To this end, some amount of *abstraction* is required: removing details to focus attention on general patterns. The impact of your system, both positive and negative, highly depend on how you define the machine learning task. By abstracting away the context surrounding a model and its inputs and outputs, you may accidentally abstract away some of the consequences as well.

A mismatch between the machine learning task and the real-world context is referred to as an *abstraction trap* [Selbst et al., 2019]. Abstraction traps can amplify harmful consequences of your system, particularly those related to fairness. We will now discuss five abstraction traps: the *framing trap*, *portability trap*, *formalism trap*, *ripple effect trap*, and *solutionism trap*.

3.2.1 The Framing Trap

Machine learning models hardly ever operate in isolation. A decision-making process may incorporate (multiple) other machine learning models or human decision-makers. The *framing trap* considers the failure to model the relevant aspects of the larger system your machine learning model is a part of.

For example, consider a scenario in which judges need to decide whether a defendant is detained. To assist them in their decision making process, they may be provided with a machine learning model that predicts the risk of recidivism; i.e., the risk that the defendant will re-offend. Notably, the final decision of the judge determines the real-world consequences, not the model's prediction. Hence, if fairness is a requirement, it is not sufficient to consider the output of the model; you also need to consider how the predictions are used by the judges.

In many real-world systems, several machine learning models are deployed at the same time at different points in the decision-making process. Unfortunately, a system of components that seem fair in isolation do not automatically imply a fair system, i.e. *Fair + Fair* $\neq$ *Fair* [Dwork et al., 2020].

To avoid the framing trap, we need to ensure that the way we frame our problem and evaluate our solution includes all relevant components and actors of the sociotechnical system.

3.2.2 The Portability Trap

A system that is carefully designed for a particular context cannot always be directly applied in a different context. Taking an existing solution and applying it in a different situation without taking into account the differences between the two contexts is known as the *portability trap*. A shift in domain, geographical location, time, or even the nature of the decision-making process all impact the suitability of a system. For example, a voice recognition system optimized for speakers with an Australian accent may fail horribly when deployed in the United States. Similarly, the expectations of a good manager have changed considerably in the past few decades, including a stronger need for soft skills. A model trained on annual reviews in the 1960's will likely not be suitable to make predictions for current managers. The portability trap goes beyond performance issues due to differences in the data distribution. It also considers differences in social norms and actors. For example, a chat bot optimized to formulate snarky replies may be considered funny on a gaming platform, but inappropriate or even offensive in a more formal context, such as a website for loan applications. To avoid falling into the portability trap, we need to consider whether our problem understanding adequately models the social and technical requirements of the actual deployment context.

3.2.3 The Formalism Trap

In order to use machine learning, you need to formulate your problem in a way that a mathematical algorithm can understand. This is not a straightforward task: there are usually many different ways to measure something. Some may be more appropriate than others. You fall into the *formalism trap* when your formalization does not adequately take into account the context in which your model will be used. For example, machine learning problem formulations often simplify the decision space to a very limited set of actions [Mitchell et al., 2018]. In lending, the decision space of the machine learning model may consist of two options: reject or accept. In reality, there may be many more actions available, such as recommending a different type of loan.

The formalism trap is closely related to the statistical concept of *construct validity*: how well does the formalization measure the construct of interest? Business objectives often involve constructs such as "employee quality" or "creditworthiness" that cannot be measured directly [Jacobs and Wallach, 2021]. In such cases, data scientists may use a *proxy variable* instead. Every variable in a data set is the result of a decision on how a particular construct can be measured into a computer readable scale. For example, Netflix has chosen to measure a viewers' quality judgments with likes, rather than the more commonly used 1 - 5 star rating [Dobbe et al., 2018].

Not all variables measure the intended construct equally well. *Construct validity bias* is a statistical bias that occurs when a variable does not accurately measure the construct it is supposed to measure. For example, income measures the construct socioeconomic status to some degree, but does not capture other factors such as wealth and education [Jacobs and Wallach, 2021].

A mismatch between the choice of target variable and the actual construct of interest can be detrimental towards fairness goals. In particular, fairness concerns can arise when the measurement error introduced by the choice of formalization differs across groups. For example, you may be interested in predicting crime, but only have access to a subset of all criminal activity: arrest records. In societies where arrest records are the result of racially biased policing practices, the measurement error will differ across racial groups. Similarly, Obermeyer et al. [2019] found that due to unequal access to healthcare, historically less money has been spent on caring for African-American patients compared to Caucasian patients. Consequently, a system that used healthcare costs as a proxy for true healthcare needs systematically underestimated the needs of African-American patients.

Issues of construct validity are especially complex for social constructs such as race and gender. Almost paradoxically, measuring sensitive characteristics can introduce bias. In industry and academia, it is common to 'infer' these characteristics from observed data, such as facial analysis [Jacobs and Wallach, 2021]. This can be problematic because such approaches often fail to acknowledge that social constructs are inherently contextual, may change over time, and are mul-

tidimensional. For example, when talking about race, one may be referring to somebody's racial identity (i.e., self-identified race), their observed race (i.e., the race others believe them to be), their phenotype (i.e., racial appearance), or even their reflected race (i.e., the race they believe others assume them to be). Which dimension you measure will influence the conclusions you can draw (see Hanna et al. [2020] for a more detailed account).

To avoid falling into the formalism trap, data scientists should take into account whether the problem formulation handles understandings of (social) constructs in a way that matches the intended deployment context. To mitigate construct validity bias, ideally multiple measures are collected, especially for complex constructs. As we will see, many of the biases that can occur in later steps of the development process can be traced back to the problem of construct validity.

3.2.4 The Ripple Effect Trap

Introducing a machine learning model in a social context may affect the behavior of other actors in the system and, as a result, the context itself. This is known as the *rippl effect trap*. There are several ways in which a social context may change due to the introduction of a new technology. First, the introduction of a new technology might be used to argue for or reinforce power, which can change an organization's dynamics. For example, management may purchase software for monitoring workers, reinforcing the power relationship between management and subordinates. Second, the introduction of a prediction systems may cause reactivity behavior. For example, people might attempt to game an automated loan approval system by dishonestly filling out their data in the hope of a more favorable outcome. Third, a system that was developed for a particular use case, may be used in unintended, perhaps even adversarial, ways. To avoid falling into the ripple effect trap, it is important to consider whether the envisioned system changes the context in a predictable way.

3.2.5 The Solutionism Trap

As data scientists, we can be very excited about the possible benefits that machine learning solutions can bring. Unfortunately, machine learning is not the answer to everything (*what?!*). The belief that every problem has a technological solution is referred to as *solutionism*. We fall into the *solutionism trap* when we fail to recognize the machine learning is not the right tool for the problem at hand.

The solutionism trap is closely related to the *optimism bias*. This is a cognitive bias that causes people to overestimate the likelihood of positive events and underestimate the likelihood of negative events. In the context of algorithmic systems, optimism bias occurs when policy makers or developers are overly optimistic about a system's benefits, while underestimating its limitations and weaknesses. In particular, people might overestimate the objectiveness of data and algorithmic systems. If this happens, the system's goals, development, and outcomes might not be sufficiently scrutinized, which can result in systematic harms.

There are several reasons why machine learning may not be the right tool to solve a problem. In some scenarios, it may not be possible to adequately model the context using automated data collection. For example, consider eligibility for social welfare benefits in the Netherlands. Although the criteria for eligibility are set in the law, some variables, e.g. living situation, are difficult to measure quantitatively. Moreover, the Dutch legal system contains the possibility to deviate from the criteria due to weighty personal circumstances. It is impossible to anticipate all context-dependent situations in advance. As a result, machine learning may not be the best tool for this job. In other scenarios, machine learning may be inappropriate because it lacks human connection. For example, consider a person who is hospitalized. In theory, it may be possible to develop a robot nurse who is perfectly capable of performing tasks such as inserting an IV or washing the patient. However, the patient may also value the genuine interest and concern of a nurse – in other words, a human connection, something a machine learning model cannot (or even should not) provide. Furthermore, there may be cases where machine learning is simply overkill. For example, you may wonder whether spending several months on optimizing a deep learning

computer vision system to predict the dimensions of items in your online shop is a better approach than simply asking the person who puts the item on the website to fill out the dimensions.

To avoid falling into the *solutionism trap*, it is useful to consider machine learning as a means to an end. In other words, rather than asking "can we use machine learning", ask, "how can we solve this problem?" and then consider machine learning as one of the options.

3.3 Data Understanding and Data Processing

It may not come as a surprise that many fairness issues arise through biases in data collection and analysis.

3.3.1 Historical bias

Social biases can be encoded in data. If not accounted for, a machine learning model will reproduce these biases, resulting in unfair outcomes. Generally speaking, *historical bias* comes in two flavors.

Firstly, historical bias can arise due to social biases in human decision-making. This type of bias is particularly prevalent when target labels are based on human judgement. For example, if in the past more men have been hired than women, a model trained on historical decisions will likely reproduce this association. As we have seen in the previous section, simply removing the sensitive feature is not likely to remove this type of bias due to associations between features. Note that this type of historical bias is a form of *construct validity bias*: the historical hiring decisions are a biased proxy for actual suitability of the applicant. Similarly, inaccurate stereotypes can be embedded in texts, images, and annotations produced by people, resulting in systems that reinforce these stereotypes.

A second type of historical bias occurs when the data is a good representation of reality, but reality is biased. In our hiring example, the observed bias could be caused by actual differences in suitability, but these differences are in turn caused by structural inequalities in society. For example, people from lower socioeconomic backgrounds may have had fewer opportunities to get good education, making them less suitable for jobs where such education is required for job performance. Similarly, some stereotypes are accurate at an aggregate level (even if they can be very inaccurate at an individual level!). For example, in many societies female nurses still greatly outnumber male nurses.

Depending on your worldview, you might have a different definition of what is fair in each of these scenarios. In practice, it is usually impossible to distinguish between these two types of historical bias from observed data alone. Moreover, they can also occur simultaneously. Consequently, understanding historical bias and identifying a mitigation approach that is in line with your own moral values requires a deep understanding of the social context.

3.3.2 Representation bias

Representation bias occurs when some groups are underrepresented in the data [Suresh and Guttag, 2020]. A machine learning model might not generalize well for underrepresented groups, causing quality-of-service harm. Representation bias is especially risky when the data distribution of minority groups differs substantially from the majority group (see also aggregation bias). A well-known example of representation bias was uncovered by Buolamwini and Gebru [2018]. As it turns out, the data sets that were used to train commercial facial recognition systems contained predominantly images of white men. Consequently, the models did not generalize well to people with dark skin, especially women.

Representation bias is closely related to *selection bias*, a statistical bias that occurs when the data collection or selection results in a non-random sample of the population. If not taken into account, conclusions regarding the studied effect may be wrong. For example, young healthy people may be more likely to volunteer for a vaccine trial than less healthy older people. As a result, conclusions about the side effects may not be representative for the whole population. Notably, representation bias can occur even when a sample is truly random, as there may not be

sufficient information available for minority groups. Moreover, representation bias can be an issue in both training and testing data.

An underlying cause of representation bias is blind spots of the collectors. For example, a data science team that consists solely of women is less likely to notice that men are not well represented in the data than a more diverse team. Additionally, some data is easier to get then others. For example, collecting data on the interests of young adults, who often spend hours each day scrolling through their social media feeds, is much easier compared to the interests elderly people who are generally not as active online.

Representation bias is relatively easy to solve by getting more data. Additionally, data scientists can leverage techniques that were designed to deal with sampling errors, such as weighting instances. However, these approaches first require you to identify cases in which representation bias may occur. Therefore, a more durable solution is to invest in a diverse and inclusive development approach, which will help to avoid leaving groups out of the picture in the first place.

3.3.3 Measurement bias

Measurement bias is a statistical bias that occurs when data contains systematic errors, due to data collection practices. Measurement bias can be a cause of *construct validity bias*. If systematic errors are correlated to sensitive characteristics, measurement bias can become a source of unfairness.

Measurement bias can occur when the method of observation results in systematic errors. First of all, the measurement process may be different across groups [Suresh and Guttag, 2020], due to a combination of social bias and *confirmation bias*. Confirmation bias is a cognitive bias that refers to people's tendency to look for evidence of existing beliefs and disregard evidence that goes against it. In the context of data analysis, confirmation bias can lead to cherry picking data data support a conclusion. As the famous quote by Ronald Coase says: "if you torture the data long enough, it will confess to anything." For example, a fraud analyst might overly scrutinize some groups over others. Higher rates of testing will result in more positives, confirming the analysts biased beliefs and skewing the observed base rates. If not accounted for, these skewed numbers will be reproduced by the machine learning model.

Another example of measurement bias is when different observers interpret reality differently. In medical studies, for example, clinicians might arbitrarily round blood pressure readings up or down to the nearest whole number, depending on what they expect to see. In a machine learning context, this type of bias can occur when annotations reflect social biases, such as stereotypes, of the annotators or decision-makers.

Measurement bias can also occur at the side of the data subject. Data subjects might behave differently because they are being observed, especially when data is collected from memory or through self-reporting. Survey responses may be incomplete or inconsistent because participants try to present themselves in a way that is socially desirable. For example, consider self-reported height, scraped from a dating website. In many cultures, tallness is seen as an attractive trait in men. Consequently, men may have exaggerated their height to appear more attractive, resulting in measurement bias. Note that this is another example of how measurement bias can be a threat to construct validity.

Measurement bias can be mitigated by high-quality data collection procedures. Note that it is not possible to identify cases of measurement bias from observational data alone. For example, we cannot know the true underlying fraud rate if we only take into account data produced by a biased fraud detection approach. This highlights the importance of documenting the data collection procedure.

3.4 Modeling

Building a machine learning model includes many different choices, ranging from the class of machine learning algorithms that is considered to their hyperparameter settings. Different models may have different consequences related to fairness, depending on the task at hand.

3.4.1 Aggregation bias

Aggregation bias occurs when a single model is used for groups that have distinct data distributions [Suresh and Guttag, 2020]. If not accounted for, it may lead to a model that does not work well for any of the subgroups. For example, it is known that the relationship between hemoglobin levels (HbA1c) and blood glucose levels differs greatly across genders and ethnicities [Suresh and Guttag, 2020]. If these differences are not taken into account, a model that only uses a single interpretation of HbA1c will likely not work well for any of these groups. In combination with representation bias, it can lead to a model that only works well for the majority population.

Aggregation bias is related to the problem of *underfitting*. Machine learning can be seen as a compression problem that produces a mapping between input features and output variable. Some information is inherently lost because of the chosen mapping [Dobbe et al., 2018]. In particular, some model classes may not be able to adequately capture the different data distributions. Such an oversimplified model may come at the cost of predictive performance for minority groups, resulting in quality-of-service harm.

3.4.2 Omitted variable bias

Omitted variable bias is a statistical bias which occurs when one or more relevant features are left out of a linear regression model. Consequently, the model attributes the effects of the missing feature(s) to the included features, obscuring their true effects. Omitted variable bias was originally introduced in the context of statistical models that are used for causal inference. For example, consider a regression-based test for discrimination in loan applications. Imagine that loan officers consider payment history in their decision and that payment history correlates with race. If payment history is not recorded in the data, the results of the regression will attribute the effect of payment history to race, suggesting direct discrimination and potentially procedural harm when there is not [Jung et al., 2019]. This bias can also be relevant for prediction tasks. In particular, excluding sensitive features from the data may obscure a model's indirect dependence on this feature, attributing their effects to other related features. This makes it more difficult to detect and account for existing historical bias.

3.5 Evaluation

During the evaluation stage, the final model is scrutinized in more detail. *Evaluation bias* refers to the use of performance metrics and procedures that are not appropriate for the way in which the model will be used [Suresh and Guttag, 2020]. Mitchell et al. [2018] identify several underlying assumptions of performance metrics. First, these metrics assume that individual decisions are independent of each other. Note how this assumption is grounded in utilitarianism, in which overall utility is expressed as the sum of individual utilities. In practice, however, the impact of a decision may not be independent across instances. For example, denying one family member a loan may impact another family member's ability to repay their own loan. Additionally, it is typically assumed that decisions are symmetrical, i.e. the impact of the outcome is equal across instances. Again, this often does not hold in practice. For example, a rejection of a job application can have a very different impact, depending on whether that person is currently employed or unemployed.

3.6 Deployment

Once the system is deployed, it may be used, interpreted, or interacted with inappropriately, resulting in unfair outcomes [Friedman and Nissenbaum, 1996]. The underlying cause of these outcomes is a mismatch between the system's design and the context in which it will be applied. Indeed, biases in deployment can often be attributed to *abstraction traps* introduced in Section 3.2.

Usage The system may be used in a context for which it was not (properly) designed, in which case we fall into the *portability trap*. For example, a toxic language detection model trained on tweets may not be suitable for a platform such as TikTok, where the average user is much younger and may use different language (tone, words, etc.) than an average Twitter user. Note that this type of bias can also accrue over time due to changing populations and behaviors [Mehrabi et al., 2019], in which case it can be seen as a form of *concept drift*.

Interpretation Interaction of stakeholders with the system can be a source of unfairness. A decision-maker may interpret the model's output differently for different groups, due to social bias and confirmation bias. For example, a judge may weigh a high risk score more heavily for a black defendant compared to a white defendant, due to (unconscious) social bias. This bias, which can be attributed to falling into the *framing trap*, can be mitigated by taking into account stakeholder interactions during the system's design and evaluation.

Interaction In systems that learn from user interactions, users can introduce social bias. For example, consider a chat bot that learns dynamically. Without safeguards against toxicity, users might teach it to use obscene or otherwise offensive language, resulting in denigration harm. This type of bias can be avoided by putting checks in place to identify malicious intent towards the system.

Reinforcing feedback loop *Reinforcing feedback loops* are feedback mechanisms that amplify an effect. In the context of fairness, it refers to the amplification of existing (historical) biases when new data is collected based on the output of a biased model.

Example 2. *A Reinforcing Feedback Loop in Predictive Policing.* Lets imagine there is a police station that is responsible for two neighborhoods, A and B. Now lets imagine a predictive policing system that allocates police officers to the neighborhoods based on the predicted crime rate in each neighborhood. In this example, the true crime rates of the neighborhoods are equal. However, due to the randomness, we have collected slightly more crime data in neighborhood A than than in neighborhood B at the time the prediction model is trained. Consequently, the model predicts more crime in neighborhood A than in neighborhood B. Based on this prediction, we send more police officers to neighborhood A. Consequently, more crime will be detected in neighborhood A – even though the true crime rates are the same. If we retrain our model on the newly collected crime data, even more police officers will be allocated to neighborhood A and even more crime is detected. And so the feedback loop continues...

A consequence of these feedback loops is that people can form erroneous beliefs based on the data. For example, after the introduction of the predictive policing system in Example 2, police officers may believe that neighborhood A truly has a bigger crime problem than neighborhood B. A failure to anticipate on feedback loops can be particularly risky for automated decision-making systems, in which they can propagate quickly.

An instance of feedback loops in recommender systems is *popularity bias*. If people tend to click on highly ranked items more often, this can lead the algorithm to rank popular items even higher and disregard less popular items that may be just as valuable to the user.

One way to investigate feedback loops is through simulation, which requires a lot of domain expertise. Similarly, we may borrow approaches from the field of system dynamics [Martin et al., 2020]

4 Discussion

As we have seen in this chapter, there is no standard notion of fairness in literature on algorithmic fairness. This is a feature, not a bug. Different fairness criteria encode different value systems

[Hutchinson and Mitchell, 2019], which can all be reasonable ethical stances. Rather than falling into the *abstraction trap*, we need to acknowledge that fairness is an inherently contextual concept and no single definition of fairness will apply to all scenarios. Moreover, norms and laws shift over time and what is considered fair today may not be considered fair tomorrow. In practice, however, this means that it is difficult to choose the 'right' notion of fairness for the problem at hand.

As we have seen in Section 3, there are many different sources of unfairness, rooted in social biases we are not always aware of. Even a well-intended machine learning practitioner may inadvertently train a discriminatory model. Unraveling these biases is a complex task and it is never possible to fully "de-bias" a model [Madaio et al., 2020]. Instead, the goal of efforts in algorithmic fairness is to mitigate harms as much as possible. This requires careful attention throughout the entire development process.

Identifying the ways in which your system could be harmful to vulnerable populations can be helpful in choosing an appropriate notion of fairness. It is important to do this from the start of your project, as biases can seep in in each step of the way. Finally, it is very difficult, if not impossible, to debate the fairness of a system without being transparent on how the system was developed and which trade-offs were made along the way. As such, we cannot emphasize enough how important it is to document the process carefully, especially for high-impact systems.

References

S. Barocas, M. Hardt, and M. Narayanan. *Fairness in Machine Learning*. 2019. Retrieved from `https://fairmlbook.org`.

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

R. Dobbe, S. Dean, T. Gilbert, and N. Kohli. A Broader View on Bias in Automated Decision-Making: Reflecting on Epistemology and Dynamics. 2018. URL `http://arxiv.org/abs/1807.00553`.

C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science*, ITCS '12, page 214–226, New York, NY, USA, 2012. Association for Computing Machinery. ISBN 9781450311151. doi: 10.1145/2090236.2090255. URL `https://doi.org/10.1145/2090236.2090255`.

C. Dwork, C. Ilvento, and M. Jagadeesan. Individual fairness in pipelines. 2020.

B. Friedman and H. Nissenbaum. Bias in computer systems. *ACM Transactions on Information Systems*, 14(3):330–347, July 1996. doi: 10.1145/230538.230561. URL `https://doi.org/10.1145/230538.230561`.

A. Guo, E. Kamar, J. W. Vaughan, H. M. Wallach, and M. R. Morris. Toward fairness in AI for people with disabilities: A research roadmap. *CoRR*, abs/1907.02227, 2019. URL `http://arxiv.org/abs/1907.02227`.

A. Hanna, E. Denton, A. Smart, and J. Smith-Loud. Towards a critical race methodology in algorithmic fairness. In *FAT* 2020 - Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 501–512, dec 2020. ISBN 9781450369367. doi: 10.1145/3351095.3372826. URL `https://arxiv.org/abs/1912.03593`.

B. Hutchinson and M. Mitchell. 50 Years of Test (Un)fairness: Lessons for machine learning. *Proceedings of the 2019 Conference on Fairness, Accountability, and Transparency*, pages 49–58, 2019. doi: 10.1145/3287560.3287600.

A. Z. Jacobs and H. Wallach. Measurement and fairness. *Conference on Fairness, Accountability, and Transparency (FAccT '21), Virtual Event, Canada.*, mar 2021. doi: 10.1145/3442188.3445901. URL https://doi.org/10.1145/3442188.3445901.

J. Jung, S. Corbett-Davies, R. Shroff, and S. Goel. Omitted and included variable bias in tests for disparate impact, 2019.

F. Kamiran, I. Žliobaitė, and T. Calders. Quantifying explainable discrimination and removing illegal discrimination in automated decision making. *Knowledge and Information Systems*, 35 (3):613–644, 2013. ISSN 02191377. doi: 10.1007/s10115-012-0584-8.

M. Kusner, J. Loftus, C. Russell, and R. Silva. Counterfactual fairness. In *Advances in Neural Information Processing Systems 30*, 2017.

M. A. Madaio, L. Stark, J. Wortman Vaughan, and H. Wallach. Co-Designing Checklists to Understand Organizational Challenges and Opportunities around Fairness in AI. *Chi 2020*, pages 1–14, 2020.

D. Martin, Jr., V. Prabhakaran, J. Kuhlberg, A. Smart, and W. S. Isaac. Extending the machine learning abstraction boundary: A complex systems approach to incorporate societal context, 2020.

N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan. A Survey on Bias and Fairness in Machine Learning. aug 2019. URL http://arxiv.org/abs/1908.09635.

S. Mitchell, E. Potash, S. Barocas, A. D'Amour, and K. Lum. Prediction-based decisions and fairness: A catalogue of choices, assumptions, and definitions, 2018.

Z. Obermeyer, B. Powers, C. Vogeli, and S. Mullainathan. Dissecting racial bias in an algorithm used to manage the health of populations. *Science*, 366(6464):447–453, 2019. ISSN 0036-8075. doi: 10.1126/science.aax2342. URL https://science.sciencemag.org/content/366/6464/447.

C. Rudin, C. Wang, and B. Coker. The age of secrecy and unfairness in recidivism prediction. pages 1–46, 2018. URL http://arxiv.org/abs/1811.00731.

A. D. Selbst, D. Boyd, S. A. Friedler, S. Venkatasubramanian, and J. Vertesi. Fairness and abstraction in sociotechnical systems. *FAT* 2019 - Proceedings of the 2019 Conference on Fairness, Accountability, and Transparency*, pages 59–68, 2019. doi: 10.1145/3287560.3287598.

H. Suresh and J. V. Guttag. A framework for understanding unintended consequences of machine learning, 2020.

M. Wieringa. What to account for when accounting for algorithms. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. ACM, Jan. 2020. doi: 10.1145/3351095.3372833. URL https://doi.org/10.1145/3351095.3372833.