MSc Artificial Intelligence
Master Thesis

Causal Fairness Analysis with Automated Feature Engineering

by
Wietse van Kooten
14633094

June 29, 2024

36 ECTS
January till June

Supervisor:
Dr. E. Acar

Examiner:
Dr. E. Acar

Second reader:
Isabel Barberá

Universiteit van Amsterdam

Contents

Abstract

To understand how, and through which mechanisms one variable influences another variable is of great importance. It asks us the very question of why certain outcomes arise, and more importantly under what conditions. This is one of the key issues within data sciences. Automated systems that apply AI rely on methodologies to retrieve these mechanisms of how one variable affects the other. Unfortunately, a lot of these methodologies are not interpretable, making it hard to understand why outcomes arise, or what the relations are between variables. Especially now, the need for a better understanding and control of AI models is crucial, as we have seen often the harmful consequences of automated systems when there is a lack of understanding or control.

One of these unintended effects is discrimination. In today's society, discrimination occurs when biased decisions are made by humans, and these problems often persist or are even amplified when decisions are made by automated systems with little interpretability, and fairness. While theoretical advancements in methodologies for estimating causal mechanisms are significant, real-world applications often stick to methods that are not interpretable or fair. Quantifying bias in observed data, with unknown underlying causal mechanisms, remains a significant challenge across various scientific disciplines. However, we are at a point where we can provide a solution.

Causal fairness analysis aims to solve this challenge and provide an answer, by solving issues of fairness in decision-making settings. This thesis builds on the Standard Fairness Model (SFM) by Drago Plecko, a robust template designed to perform causal analysis. The SFM is a graphical model with fewer modeling assumptions than current structural causal models, while still being able to capture reality, and can calculate potential outcomes and counterfactuals. Such that it offers a novel approach to quantifying and explaining biases inherent in data and predictive algorithms. We will apply this model to the second task of the three main tasks within fair machine learning (fair ML): (1) Bias detection and quantification, (2) fair prediction, and (3) fair decision-making.

Additionally, We contribute to current knowledge by implementing automated feature engineering as a preprocessing step in the SFM. Automated feature engineering, the process of creating new features from existing data, has been shown to boost model accuracy, enrich the information provided by the data, and improve its interpretability. It is common practice to apply feature engineering in a variety of AI models. Despite this, it has never been used in the SFM, due to its novelty and stricter criteria to which it has to adhere.

This preprocessing extension addresses the challenge models face in detecting trends across multiple subgroups, known as Simpson's paradox. By using automated feature engineering we can better identify biases that arise in a similar trend. This preprocessing step modifies the SFM's structure, and we prove that the new structure still meets SFM structural criteria, ensuring its practical applicability. Additionally, our experiments show that this approach not only maintains usability but also improves performance in various scenarios. particularly the capability to remove bias increases, making the SFM more effective for removing biases, thereby advancing the ability to create fairness in an increasingly data-driven world.

Chapter 1

Introduction

Machine learning(ML) is increasingly central to decision-making across various real-world applications, such as hiring practices, law enforcement, university admissions, credit evaluations, healthcare, and other critical areas. Automated systems guided by ML can significantly impact individual lives and societal well-being. Examples of such applications include decision support systems for predicting recidivism, facial recognition, online advertising, and hiring. Given this, there is growing concern over the implications of replacing or supplementing current decision-making processes with automated systems.

Unfairness and discrimination are found in many settings where decisions are made by humans, implying an inherent bias. Research reveals that automated decision-making tools often inherit and even amplify the biases present in their training data, for example in the context of the gender pay gap and racial bias in criminal sentencing. This is a consequence of its biased input data, which reflects historical inequalities and discriminatory practices. This implies we need a model that can also mitigate pre-existing bias in the data, so creating a new "ground truth".

Causal inference is important in the advancement of fair ML, by providing effective methods to understand and manipulate the underlying mechanisms through which biases occur, and how those biases can lead to unfair outcomes in models [29]. This approach goes beyond traditional correlational (associational) analyses. Whereas correlation only identifies whether a consistent relationship exists between variables, causal inference can determine whether and how specific factors cause disparities, understanding their underlying mechanisms. This deeper understanding is crucial for calculating interventions and counterfactuals, enabling us to not only make fair predictions but also to make the whole decision-making process fair.

To illustrate, let us consider this small example about smoking, a debate that has been ongoing over the past decades. The harmful effects of smoking on health have been known for a long time. However, lobby groups of the tobacco industry (and professors with a smoking habit) claimed this was due to a confounding factor. They introduced a certain third variable: a gene causing a person to smoke and have worse health. At that time it was extremely hard to disprove this statement, as DNA research was not advanced enough to find these genes. Luckily, researchers advocating the harmful effects of smoking found another way; they used causal inference to prove that smoking negatively affects a person's health. They introduced another variable: the amount of tar inside the lungs, which could be measured at that time. They found a direct effect of smoking on the amount of tar, which affected the probability of developing lung cancer [17]. This made it able to reject the statement that smoking itself was not harmful. A simplistic, yet elegant approach to prove a causal mechanism. This also underlines the difference between correlation and causation, as the correlation between smoking and health was clear, but not the causation.

1.1 Correlation and Causation

Understanding the causal mechanisms between variables and the outcomes they influence is a fundamental aspect of data science, statistics, and ML. The pursuit of knowledge in these fields often begins with the detection of correlations; observing how variables move together. However, we know now: Correlation does not imply causation [16]. We will first introduce the change of thought from correlation to the more complex understanding of causality, with the use of the pioneering work of Judea Pearl and his Ladder of Causation [17], and we will once underline the necessity of this change.

At first, the study of relationships within data focuses on correlation, which simply identifies whether a consistent relationship exists between variables. Statistical methods have been developed to quantify the strength and direction of these relationships. Indeed, correlation is useful as a starting point for any analytical research where patterns in data suggest potential relationships. Even so, this is only the first rung on Pearl's *Ladder of Causation*, namely association [17]. It allows us to observe and describe patterns, but it does not give an understanding of whether one variable causes another or if their relationship is the product of a confounding factor. This has given a lot of space for ambiguity, as we have observed in the effect of smoking on cancer, and more recently in the redlining problem [26, 7]. The redlining problem is the effect of the location you live in on the financial stability that is assigned to you, due to confounding factors such as ethnicity. This is of course another example of a discriminatory practice we would rather not see; as the location you live in should not determine the height of your mortgage or other financial service.

As of today, still, a lot of ML models are only based on correlation, without a deeper understanding of the data. This is one of the reasons that caused a huge scandal in the Netherlands, the "Toeslagenaffaire" [9]. Here they explain on page 14, that indeed their classification system was discriminating and has caused families a lot of trouble. The system was discriminatory as there was a causal mechanism of nationality working on the outcome of the model. This causal mechanism between nationality and the outcome should have been removed, and it should have been seen during the testing of the model. However, associational measures are not able to detect these discriminatory practices in every situation, and the interpretation of the model was at that moment not clear.

The shift towards Causality

The need to move beyond correlation towards causality becomes clear when decisions need to be based not just on what is happening, but on why something is happening and what will happen if something is set to a certain value; an intervention. This is the second rung of Pearl's ladder, namely intervention. Intervention asks not just whether variables X and Y occur together, but what happens to Y when X is set to a certain value. Importantly note that interventions look at potential outcomes. This is where causal inference begins to play a crucial role, requiring more complex statistical tools and experimental designs to identify causal mechanisms rather than just associations [15].

Structural causal model; a causal framework

The development of structural causal models (SCM) provides the framework needed to make use of interventions. The SCM proposed by Judea Pearl, allows researchers and practitioners to simulate potential interventions and observe potential outcomes [16]. SCMs use a combination of statistical data and expert knowledge to construct diagrams that graphically represent causal mechanisms, offering a clear visualization of how variables interact. Figure 1.1 is an example of a causal diagram of an SCM.

SCMs not only simplify the understanding of causal mechanisms but also increase the ability to predict the consequences of changes in system components, as we now have learned to call interventions. The ability to intervene based on model predictions is a powerful advancement over correlational studies, as it enables actions that can actively shape outcomes rather than passively predicting them. Such capabilities are crucial for informed policymaking, scientific discovery, and strategic business planning [20]. In other words, SCMs make it possible to apply interventions and calculate their potential outcome.

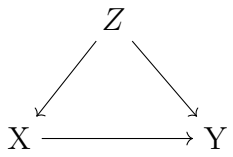


Figure 1.1: A causal diagram of an SCM, with the nodes X, Y, and Z, and the arrows indicating the existence of a causal mechanism.

Counterfactuals

The highest rung on Pearl's Ladder of Causation are counterfactuals, which involve understanding what could have happened but did not [17]. Counterfactual thinking allows us to consider hypothetical situations and to answer "what-if" questions. We, as humans are used to counterfactual thinking to determine the outcome of these hypothetical situations. We are perfectly able to imagine certain situations if some variables had been slightly different. This way of counterfactual thinking helps us to reevaluate our choices and makes us better able to adjust to new situations. For example, if you had taken a different way of transport today to work or university, you know perfectly well how it would go, due to experiences in the past. However, that situation never in reality occurred on this specific day, so the data does not exist.

This ability to calculate counterfactuals is essential for a deeper understanding of causal mechanisms because it enables the analysis of not only the effects of observed interventions but also the potential outcomes of hypothetical ones. This level of analysis is crucial for optimizing strategies in complex systems where multiple causal pathways may exist.

1.2 Fairness Regulation

As AI technologies become increasingly important to various sectors, regulatory frameworks such as the GDPR (General Data Protection Regulation), and the AI Act are of critical importance in determining the ethical, legal, and operational landscapes of AI implementation [19, 13]. These regulations are designed to address the growing concerns about privacy, fairness, transparency, and accountability in AI systems.

GDPR

The GDPR, known in the Netherlands as the AVG, is a comprehensive data protection regulation implemented by the European Union in 2018[19]. It sets strict guidelines for the collection, storage, and processing of personal data of individuals within the EU. The GDPR emphasizes the principles of transparency, data minimization, and user consent, providing individuals with greater control over their personal data. For AI, this regulation impacts how data can be used in training and operating AI models, enforcing the need for systems to be designed in ways that respect user privacy and data protection rights.

The AI Act

In contrast, the AI Act is a proposed regulation by the European Commission specifically made to govern the development and deployment of artificial intelligence across the EU[13]. This Act

categorizes AI systems into three risk levels, from minimal risk to high risk. Higher risk levels impose stricter requirements as the potential for harm increases. The AI Act focuses on critical areas such as high-risk AI systems, which require a thorough assessment of their accuracy, robustness, and cybersecurity measures. The Act also mandates transparency for AI systems that interact with humans or those used in ways that can significantly affect people's lives, ensuring that users are always aware they are interacting with AI and can understand the basis of AI-generated decisions.

Both the GDPR and the AI Act are important to the public debate on fair ML. While the GDPR ensures that data used in AI systems conforms to privacy standards, the AI Act goes deeper into the ethics of AI functionality, focusing on safety, transparency, and the potential for discrimination. We will try with our work to adhere to these legislations, for which Disparate treatment and impact are critical. These are two concepts within the anti-discrimination laws of the US [30].

Disparate treatment and impact

Disparate treatment enforces that the treatment of all groups should be equal. This regulation prohibits the use of sensitive attributes, such as race, gender, or age, when making decisions. As a result, it ensures that there is no direct effect from the sensitive attribute to the outcome, maintaining fairness in the decision-making process[1].

On the other hand, disparate impact is concerned with the equality of outcomes between different groups. This concept ensures that, even if sensitive attributes are not explicitly used in decision-making, the outcomes should not disproportionately disadvantage any particular group. Thus, it enforces that the sensitive attribute does not affect the outcome, striving for a fair distribution of results across all groups.

Business necessity

However, the law also states that some effect is tolerated due to its job-relatedness, also known as business necessity. So, business necessity is a justification for implementing a policy or practice that does have a disparate impact on certain groups of people. A practice or policy qualifies as a business necessity when it is essential for the operation of the business and there are no alternative practices that would serve the same purpose with a lesser discriminatory impact[4].

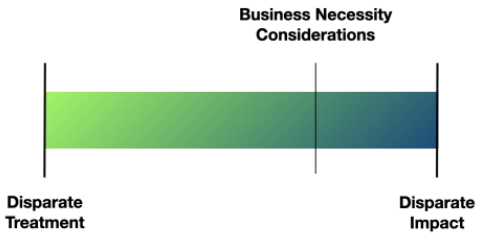


Figure 1.2: Spectrum of the two fairness notions disparate impact and treatment, with the business necessity deciding where we are on the spectrum.

So the law may not necessarily prohibit the usage of variables related to the sensitive attributes, due to their relevance to the business itself. This is of great importance as it indicates that variables that are related to sensitive attributes may still be used for predicting. This implies that the criteria of fairness depends on the business necessity. If all forms of effect besides direct effect are allowed, then we would only force disparate treatment. If no form of effect is allowed, then we would force both disparate impact and treatment. So we will obtain a spectrum of possibilities, in which the business necessity determines where we are at that spectrum. In figure 1.2 we made that spectrum, with the business necessity implying where we are on the spectrum.

1.3 Contribution

In conclusion, the evolving field of ML presents significant challenges and opportunities for achieving fairness. As these technologies become more integral to our daily lives and decision-making processes, the need to ensure they are fair grows. Causal fairness analysis fills this gap and can be seen as the intersection of causal inference and fair ML. Our causal fairness analysis will be based on the concepts of disparate impact and treatment, while also adhering to the AI act. We will use the SFM template to visualize and conduct our causal analysis. This thesis will make the following contributions to advance the goal of achieving quantifiable fairness while upholding accuracy.

1. We will extend causal fairness analysis by incorporating automated feature engineering. This will introduce new structures within the SFM framework.

2. We will demonstrate that the features developed through this extension enhance performance while upholding the SFM's structural fairness criteria.

3. We will implement this extension on the COMPAS dataset and perform an analysis with the original model to evaluate performance.

Chapter 2

Background

2.1 Causal Inference

Causal inference is a domain of statistics and data science that aims to determine causal mechanisms rather than mere correlations. It seeks to understand how and why changes in one variable influence another and is crucial in fields such as economics, epidemiology, social sciences, and ML. SCMs, introduced by Judea Pearl, provide a mathematical framework for causal inference. SCMs describe the relationships between variables in a system and make it possible to calculate potential outcomes. After introducing SCMs and the necessary definitions for causal inference we will introduce concepts of Fair ML and relate them to causal inference. Then, we will also introduce automated feature engineering.

Simpson's paradox

First, we will highlight the importance of causality, with an example. Simpson's paradox is a phenomenon where a trend appearing in several different groups of data disappears or reverses when these groups are combined. Figure 2.1 illustrates this simplistically, you can see that each group has a certain trend, which is comparable. But, when the groups are combined the trend is changed enormously. Essentially, Simpson's paradox shows how aggregated data can obscure or mislead about the true relationships between variables.

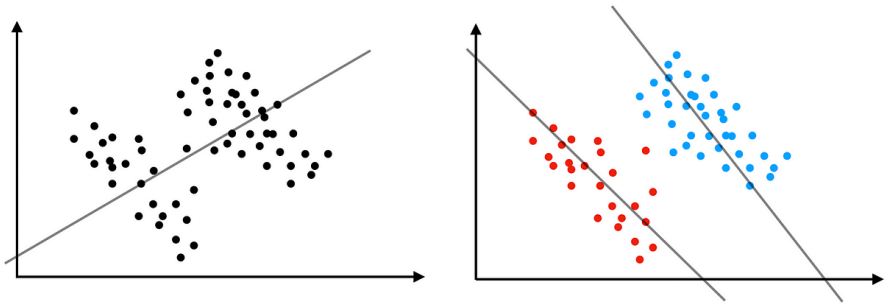


Figure 2.1: A graphical representation of Simpson's paradox

Causality can determine the effect on higher granularity, so for example on an individual or (sub)group level. If we can interpret the data on an individual or group level, we are better able to see trends within subgroups[17]. Nevertheless, without expert judgment, or prior knowledge

it is hard to always see the multiple trends within subgroups. Later, we will see how feature engineering can be a solution to this challenge.

The foundational framework of our analysis is based on SCMs, which allow the modeling of various tasks, making it the most flexible model available today[16]. We begin by introducing and illustrating SCMs with the following definition:

Definition 2.1 (Structural causal model (SCM)[16])**.** A structural causal model (SCM) is a 4-tuple $(V, U, F, P(u))$, where:

1. U is a set of exogenous (unobserved) variables, also called background variables, that are determined by factors outside the model;
2. $V = \{v_1, \ldots, v_n\}$ is a set of endogenous (observed) variables, that are determined by variables in the model (i.e. by the variables in $U \cup V$);
3. $F = \{f_1, \ldots, f_n\}$ is the set of structural functions determining V, $v_i \leftarrow f_i(p_A(v_i), u_i)$, where $p_A(v_i) \subseteq V \setminus \{v_i\}$ and $U_i \subseteq U$ are the functional arguments of f_i;
4. $P(u)$ is a distribution over the exogenous variables U.

With an SCM M we can calculate the potential outcomes of interventions and counterfactuals, on which our method is built. To perform these calculations we need some more insight into the workings of these graphs.

Markovian Graphs and semi-Markovian graphs

A Markovian graph, also known as a directed acyclic graph (DAG), is the base structure on which SCMs are built [31].

Definition 2.2 (Markovian graph[31])**.** Let V be the observed variables, U the unobserved variables, and E directed edges that denote causal relationships between these variables. We assume that no U variable is a descendant of any V variable. Furthermore, there are no cycles in these graphs, and there are no two-directional edges, and all the variables U are only directed to one variable V.

Because of the conditions of a Markovian graph, the product of conditional probabilities of each variable is given by its parents in the graph:

$$P(v) = \prod_i P(v_i \mid pa_i)$$

Here, pa_i is the set of parents of the variable v_i in the DAG. This factorization is crucial because it suggests that once the values of the parent variables are known, the value of the child variable is independent of any other variables in the system that are not descendants, also known as the Markov condition. If, however, we are dealing with an SCM with latent variables, which commonly is assumed, we have a Semi-Markovian graph.

Definition 2.3 (Semi-Markovian graphs[31])**.** Let V be the observed variables, U the unobserved variables, and E directed edges that denote causal relationships between these variables. We assume that no U variable is a descendant of any V variable. We allow for bidirected arrows, and therefore also cycles in the graph. And more, the unobserved variables U, can have a causal relation with multiple variables V

In a semi-Markovian model, the observed data distribution is typically a mixture, reflecting the influence of both observed and unobserved variables. The factorization of the joint probability in such graphs is more complex because it must account for these latent influences:

$$P(v) = \sum_{u \in U} \left(\prod_i P(v_i \mid pa_i, u^i) \right) P(u)$$

Here, u^i stands for the set of unobserved parents of v_i, and U is the set of all unobserved variables. The distribution $P(u)$ represents the joint distribution of the unobserved variables, which typically cannot be directly estimated from the observed data.

Our SCM M is semi-Markovian if there are latent variables, U. Now let us consider how we can apply interventions on our SCM.

Definition 2.4 (Potential Outcome / Response (Intervention) [16, 25])**.** Let X and Y be two sets of variables in V and let $u \in \mathcal{U}$ be a unit. The potential outcome/response $Y_u(v)$ is defined as the solution for Y of the set of equations $\mathcal{F}_x$, evaluated with $\mathcal{U} = u$. That is, $Y_u(v)$ denotes the solution of Y in the *submodel*[1] M_u of M.

In simple terms, $Y_u(v)$ is the value variable Y would take if X is set to x, for a specific unit u (possibly contrary). So suppose X is a variable representing the gender, Z a confounder, $Y_u(x)$ would then denote the outcome for the specific unit u, had their gender X been set to value x by intervention. Figure 2.2 graphically shows the implications of an intervention. Originally there was also a casual mechanism working from Z to X. However, by intervening on X, this arrow disappears, as the value of X is already set, and can not be changed.

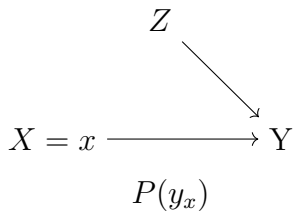


Figure 2.2: The causal graph G', with an intervention on the variable X. This causes the arrow from Z to X to disappear

So interventions are the act of modifying the state of attributes to a specific value and observing its subsequent effects. When intervening on an attribute $X = x$, it is similar to assigning the value x to the variable X in a modified causal graph G', essentially the same as the original graph G but with all incoming edges of X removed, as observed in figure 2.2. A do-operator is an alternative representation of an intervention. Specifically, an intervention represented as $P(Y|\text{do}(X = x)) = P(y_x) = Y_u(x)$.

Definition 2.5 (Counterfactual[16])**.** A counterfactual variable represents the potential outcome under a different set of conditions than what occurred. So a counterfactual outcome $X = x_1$ for a variable X, represents the hypothetical value that X would take x_1 instead of x. We express it as follows $P(y_{X=x_1}|X = x)$

Note, that a counterfactual is a combination of an intervention and a condition, which is counterfactual with the intervention. Graphically all incoming causal mechanisms of the newly intervened value, x_1, will be removed, as with a regular intervention. However, we now want the other variables to be acting as if the variable was not intervened. So we can imagine this as an extra node x, that does still have the prior causal mechanisms. In figure 2.4 we visualize a counterfactual on the SFM, which we will introduce shortly.

[1] see [25] for the definition of submodel

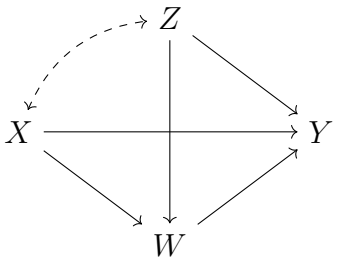
Interventions and counterfactuals are the most fundamental concepts of causal inference. They pave the way to obtain more than just associational data from a dataset. Interventions and counterfactuals will enable us to calculate direct, indirect, and spurious effects within a causal model, the perfect foundation for causal fairness analysis.

2.2 Causal Fairness Analysis

Causal fairness analysis is a method aimed at understanding, modeling, and addressing fairness issues in ML models. This involves decomposing the direct, indirect, and spurious effects to measure their impact, relying on the key concepts of decomposability and admissibility. To conduct these path-specific effect analyses, understanding the causal structure is essential [2].

Determining the correct causal structure of a dataset is challenging, and the field of causal discovery specializes in this area. However, rather than focusing heavily on finding these structures, we simplify the process by using a template model applicable to various scenarios. Instead of identifying a specific structure, we assume one and test its compatibility. The Standard Fairness Model (SFM) serves as a template to represent a range of causal diagrams, streamlining the modeling requirements.

Definition 2.6 (Standard Fairness Model(SFM)[25])**.** The SFM is the causal diagram G_{SFM} over endogenous, observed, variables {X, Z, W, Y} and given by



where the nodes represent:

- the *protected (sensitive) attribute*, labeled X (e.g., gender, race, religion),
- the set of *confounding variables* Z, which are not causally influenced by the attribute X (e.g., demographic information, zip code),
- the set of *mediator variables* W that are possibly causally influenced by the attribute X (e.g., educational level or other job-related information),
- the *outcome variable* Y (e.g., admissions, hiring, salary).

As the confounding variables Z and W are groups of features there exists a topological order such that the first element has a direct effect on all other elements in the set, but not the other way around. So we will obtain a structure that can be seen in figure 2.3

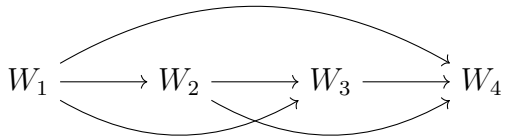


Figure 2.3: Topological order of features in a set

So, the SFM $\mathcal{M}$ is the 4-tuple $\langle V = \{X, W, Z, Y\}, U = \{U_X, U_W, U_Z, U_Y, U_{ZX}\}, \mathcal{F}, P(U)\rangle$, where $U_X, U_W, U_Z, U_Y, U_{ZX}$ represent the latent variables, outside the model, that affect the variables. The causal mechanisms $\mathcal{F}$ can be given as follows:

$$X \leftarrow 1(U_X < 0.5) + \delta_{XZ} U_{ZX}) \tag{2.1}$$
$$W \leftarrow 1(U_W < 0.5 + \lambda_{XW} X + \lambda_{ZW} Z) \tag{2.2}$$
$$Z \leftarrow 1(U_D < 0.5 + \delta_{XZ} U_{XZ}) \tag{2.3}$$
$$Y \leftarrow 1(U_Y < 0.1 + \lambda_{XY} X + \lambda_{ZY} Z + \lambda_{WY} W) \tag{2.4}$$

We introduced the concepts of the SFM and counterfactuals. Now, let us consider a counterfactual in the SFM. Figure 2.4 we illustrated the SFM with a counterfactual $P(y_{x_1} \mid x)$. As discussed in the prior section we see that all incoming edges of the newly intervened on value x_1 disappear, as the new value is set, as with a regular intervention. However, as we condition on x, this will still have incoming causal mechanisms working from the other variables.

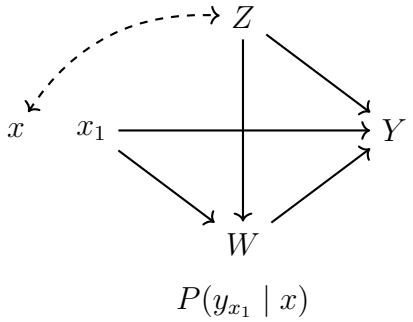


Figure 2.4: A graph of a causal structural model, with a counterfactual on the variable X. This causes the arrow from Z to X to disappear as the value of X is already set, so can not be changed by anything.

Intervening on variables could have consequences on the causal relation between variables. These causal relations can be determined by paths between the variables, and we have seen that paths can open up or close when intervening on a specific feature. This may seem a bit abstract, but consider this; suppose you would like to investigate the level of anxiety in students, due to the amount of coffee they are consuming. If you would, for example, intervene with students who study at the UvA, or a specific program, you could open up paths that are confounding between the university and anxiety. Maybe specific genders that are more prone to anxiety are studying at the UvA, or maybe the university attracts richer students, who could be prone to more anxiety. This indicated how important it is to keep track of paths that exist, and could open up when intervening (deliberately or not) on variables. So let us see how this path works, and how they open and close.

Definition 2.7 (d-sep[16])**.** A path is considered blocked if it meets any of the following criteria:

- It contains a chain $A \rightarrow B \rightarrow C$ or $A \leftarrow B \leftarrow C$ with the middle node B conditioned upon.
- It contains a fork $A \leftarrow B \rightarrow C$ with the middle node B conditioned upon.
- It contains a collider $A \rightarrow B \leftarrow C$ where the middle node B is not conditioned upon and no descendant of B is conditioned upon.

2.2.1 Structural Fairness Criteria

In section 1.2, we observed the importance of the distinction between direct, indirect, and spurious effects, due to disparate impact and treatment. Disparate treatment is fulfilled when there is no direct effect and disparate impact only when there is no direct, indirect, or spurious effect. So how do we qualitatively measure these distinctions? Remember that in real life the true SCM is not given, so we can't look up the parameters in the equations of 2.1 - 2.4. We define structural fairness criteria as qualitative assessments of the fairness in SCMs.

Definition 2.8 (Structural Fairness Criterion[25])**.** Let Ω be a space of SCMs. A structural criterion Q is a binary operator on the space Ω, that is a map $Q : \Omega \to \{0, 1\}$ that determines whether a set of causal mechanisms between X and Y exist or not, in a given SCM $\mathcal{M} \in \Omega$.

We focus on structural fairness criteria that capture direct, indirect, and spurious discrimination. Besides these path-specific fairness criteria, there are other well-defined fairness criteria in the literature, although the number is ever-increasing [2, 22]. Some worth mentioning are counterfactual fairness [11], and interventional fairness [28]. These concepts can also be incorporated in the SFM. Now, we need to know how to assess these structural fairness criteria. We will use fairness measures to assess if a structural fairness criterion is satisfied.

Definition 2.9 (Fairness Measure[25])**.** Let Ω be a space of SCMs. A fairness measure μ is a functional on the space Ω, that is a map $\mu : \Omega \to \mathbb{R}$, which quantifies the association of X and Y through any subset of causal mechanisms, in a given SCM $\mathcal{M} \in \Omega$.

A fairness measure is only suitable as a quantitative measure for the structural criteria if it upholds admissibility and decomposability. One can think of these fairness measures as, empirical measures for abstract ideas. A lot of fairness criteria are hard to exactly pinpoint to a specific calculation if it is even possible to calculate. That is also the idea behind admissibility; are these empirical fairness measures even able to quantify our criteria?

Definition 2.10 (Admissibility [25])**.** Let Ω be a class of SCMs on which a structural criterion Q and a measure μ are defined. A measure μ is said to be admissible w.r.t. the structural criterion Q within the class of models Ω, or (Q, Ω)-admissible, if:

$$\forall \mathcal{M} \in \Omega : Q(\mathcal{M}) = 0 \implies \mu(\mathcal{M}) = 0.$$

Definition 2.11 (Decomposability[25])**.** Let Ω be a class of SCMs and μ be a measure defined over it. μ is said to be Ω-decomposable if there exist measures $\mu_1, \ldots, \mu_k$ such that $\mu = f(\mu_1, \ldots, \mu_k)$,

$$f(0, \ldots, 0) = 0,$$

and where f is a non-trivial function vanishing at the origin.

In figure 2.5 we represented the idea of structural fairness criteria, fairness measures, admissibility, and decomposability. The figure shows that to detect bias we need at first a finer granularity for the TV, as the TV can't detect every form of bias. So we introduce structural fairness criteria to determine what kind of graphical criteria we want to detect. If these are defined we need fairness measures to quantify these criteria. These fairness measures need to uphold admissibility and decomposability, as otherwise, the calculations will not hold.

Now we will define in more detail these fairness measures (μ), also known as empirical fairness measures. These measures can be chosen in multiple levels of granularity within populations. We will stick to one form for simplicity.

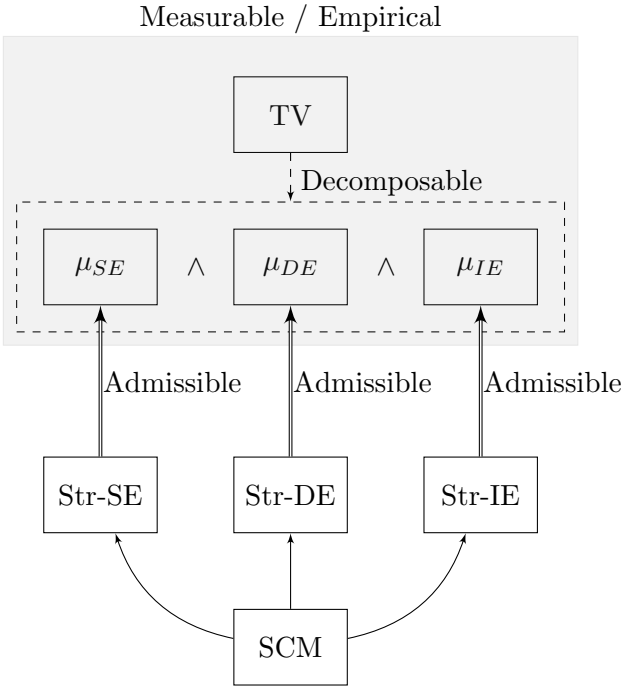


Figure 2.5: A graphical representation of the structural fairness criteria (Str-DE, Str-SE, and Str-IE) alongside their corresponding fairness measures (μ_{DE}, μ_{SE}, and μ_{IE}). The assumption of decomposability allows the total variation (TV) to be decomposed into μ_{DE}, μ_{SE}, and μ_{IE}. This decomposition illustrates the admissibility of each structural criterion to its respective fairness measure. The shaded area highlights the distinction between measurable and non-measurable elements.

2.2.2 Empirical Fairness Measures

In table 2.1 we summed up all the empirical fairness measures that we will discuss in this subsection. We will start with the total variation, which is the most intuitive and basic measure. This measure is also not an intervention or a counterfactual, so an associational measure.

Fairness Measure	**Definition**
Total Variation ($TV_{x_0,x_1}(y)$) *Parity gap*	$P(Y \mid x_1) - P(Y \mid x_0)$
Natural direct effect ($\text{NDE}_{x_1,x_0}(y)$) *nested counterfactual*	$P(y_{x_1,W_{x_0}}) - P(y_{x_0})$
Natural indirect effect ($\text{NIE}_{x_1,x_0}(y)$) *nested counterfactual*	$P(y_{x_1,W_{x_0}}) - P(y_{x_1})$
Exp-$\text{SE}_x(y)$	$P(y \mid x) - P(y_x)$

Table 2.1: Empirical Fairness Measures

Definition 2.12 (Total variation (TV))**.** The total variation (TV) measure, also referred to in the literature as the parity gap, is defined as the difference in conditional expectations:

$$TV_{x_0,x_1}(y) = P(Y \mid x_1) - P(Y \mid x_0),$$

where x_0 and x_1 are the two values of the sensitive attribute X, and Y the outcome.

Definition 2.13 (Direct, indirect and spurious effects)**.** The natural direct, the natural indirect, and experimental spurious effects are defined, respectively, as follows:

$$\text{NDE}_{x_1,x_0}(y) = P(y_{x_1,W_{x_0}}) - P(y_{x_0}), \tag{2.5}$$

$$\text{NIE}_{x_1,x_0}(y) = P(y_{x_1,W_{x_0}}) - P(y_{x_1}), \tag{2.6}$$

$$\text{Exp-SE}_x(y) = P(y \mid x) - P(y_x) \tag{2.7}$$

Further, we write $\text{NDE-fair}_X(Y)$ whenever $\text{NDE}_{X_0,X_1}(y) = 0$, or simply NDE-fair when X and Y are clear from the context. The other empirical fairness measures are defined analogously. Interestingly these concepts of effects have found their origin in the work of Chiappa, which introduced Path-Specific Counterfactual Fairness [2]. As the TV measure is not able to find a lot of forms of bias, as it is also just an associational measure, we want to decompose it into measures with higher granularity.

Theorem 1 (TV decomposition [25])**.** The total variation measure can be decomposed as follows:

$$TV_{x_0,x_1}(y) = \text{NDE}_{x_0,x_1}(y) - \text{NIE}_{x_1,x_0}(y) + (\text{Exp-SE}_{x_1}(y) - \text{Exp-SE}_{x_0}(y)). \tag{2.8}$$

Furthermore, the measures NDE, NIE, and Exp-SE are admissible for Str-DE, Str-IE, and Str-SE, respectively. We write

$$\text{Str-DE-fair} \implies \text{NDE-fair} \tag{2.9}$$

$$\text{Str-IE-fair} \implies \text{NIE-fair} \tag{2.10}$$

$$\text{Str-SE-fair} \implies \text{EXP-SE-fair} \tag{2.11}$$

$$\tag{2.12}$$

Importantly note, that Str-DE-fair implies NDE-fair, however the other way around does not hold. So, NDE-fair does not imply Str-DE-fair. To determine if a fairness measure is upheld, we need to perform a null hypothesis.

Definition 2.14 (Bias Detection under SFM[25])**.** Let Ω be a space of SCMs. Let Q be a structural fairness criterion, $Q : \Omega \to \{0, 1\}$, determining whether a causal mechanism within the SCM $\mathcal{M} \in \Omega$ is active ($Q(\mathcal{M}) = 0$ if mechanism not active, $Q(\mathcal{M}) = 1$ if active). The task of bias detection is to test the hypothesis

$$H_0 : Q(\mathcal{M}) = 0,$$

As a margin for the null hypothesis, we choose the standard deviation. This implies that if the mean of the fairness measure is within the standard deviation range of zero, the hypothesis is accepted. So suppose we want to test it for the NDE, we will get the following expression.

$$H_0 : \text{NDE}_{x_0,x_1}(y) = 0$$

Whether our model is compatible with the SFM model is a question we have to constantly ask ourselves. This sensitivity analysis on our assumptions we will make sure we are quantifying something sensible. We introduce the definition of identifiability, our sanity check to see if our model is usable.

Definition 2.15 (Identifiability[25])**.** Let $\mathcal{M} = \langle V, U, \mathcal{F}, P(u) \rangle$ be the true, generative SCM, $\mathcal{A}$ a set of assumptions, and $P(v)$ the observational distribution generated by $\mathcal{M}$. Let $\Omega_{\mathcal{A}}$ be the space of all SCMs compatible with $\mathcal{A}$. Let ϕ be a query that can be computed from $\mathcal{M}$. The quantity ϕ is said to be identifiable from $\Omega_{\mathcal{A}}$ and the observational distribution $P(V)$ if

$$\forall \mathcal{M}_1, \mathcal{M}_2 \in \Omega_{\mathcal{A}} : \mathcal{A}^{\mathcal{M}_1} = \mathcal{A}^{\mathcal{M}_2} \quad \text{and}$$

$$P^{\mathcal{M}_1}(V) = P^{\mathcal{M}_2}(V) \implies \phi(\mathcal{M}_1) = \phi(\mathcal{M}_2).$$

2.3 Fair Machine Learning

As the applications of ML expand, ensuring fairness within automated decisions becomes crucial. Fair ML is achieved by creating systems that operate in equal ways across all demographics, without discrimination toward any sensitive group. There are multiple interpretations of fairness, bias, and discrimination. We will adopt the definitions defined by Tiago Palma Pagano and Ninareh Mehrabi, as a non-causal interpretation.

Definition 2.16 (Bias[14])**.** A systematic mistake that modifies human behaviors or judgments about others due to their belonging to a group defined by distinguishing features such as gender or age.

General types of biases include dataset bias (pre-processing), model bias (in-processing), and emergent bias (post-processing). Pre-processing mitigation techniques aim to rebalance the data. In-processing mitigation focuses on the model itself, incorporating a bias correction term in the loss function or implicitly within the model, as seen in adversarial networks where the model predicts the sensitive attribute. Post-processing techniques aim to identify which sensitive attribute influenced the model's results and adjust the predictions accordingly.

Definition 2.17 (Fairness[12])**.** The absence of any prejudice or favouritism toward an individual or group based on their inherent or acquired characteristics

General forms of fairness are individual Fairness, group Fairness, and Subgroup Fairness (hybrid fairness). This granularity is of great importance, as seen in Simpon's paradox in section 2.1. Some fairness measures consider specific granularity, which causes them to detect or not detect certain biases. Later on, we will show how we will handle those differences in outcome.

Definition 2.18 (Discrimination[12])**.** A source of unfairness is due to human prejudice and stereotyping based on sensitive attributes, which may happen intentionally or unintentionally, while bias can be considered a source of unfairness due to the data collection, sampling, and measurement. Forms of discrimination are direct discrimination and indirect discrimination.

The fairness metrics in table 2.2 are used in assessing the performance of ML models, especially when considering the implications of their use in decision-making processes that affect various demographic groups.

Structural Fairness criteria	**Fairness measure**	**Definition**
Statistical Parity (SP)[25] *Demographic parity*[14]	$\text{TV}_{x_0,x_1}(\hat{y}) = 0$	$P(\hat{y} \mid x_1) = P(\hat{y} \mid x_0)$
Predictive Parity (PP) [25] *Calibration*[3]	All variations are in the BN	$P(y \mid x_1, \hat{y}) = P(y \mid x_0, \hat{y})$
Disparate Treatment[1]	$\text{NDE}_{x_0,x_1}(y) = 0$	Str-DE-fair
Disparate Impact[1]	$\text{NDE}_{x_0,x_1}(y) = \text{NIE}_{x_1,x_0}(y) = 0,$ $\text{Exp-SE}_{x_0}(y) = \text{Exp-SE}_{x_1}(y) = 0$	Str-DE-fair, Str-IE-fair & Str-SE-fair

Table 2.2: Structural fairness criteria, with their corresponding fairness measure, and definition

Definition 2.19 (Statistical and Predictive Parity [22])**.** Let X be the protected attribute, Y the true outcome, and $\hat{Y}$ the outcome predictor. The predictor $\hat{Y}$ satisfies statistical parity (SP) with respect to X if $Y \perp\!\!\!\perp X$, or equivalently if the statistical parity measure $\text{SPM}_{x_0,x_1}(\hat{y})$ satisfies:

$$\text{SPM}_{x_0,x_1}(\hat{y}) = P(\hat{y}|x_1) - P(\hat{y}|x_0) = 0.$$

Further, $\hat{Y}$ satisfies predictive parity (PP) with respect to X, Y if $Y \perp\!\!\!\perp X|\hat{Y}$, or equivalently if $\forall y$ we have

$$\text{PPM}_{x_0,x_1}(y) := P(y|x_1, \hat{y}) - P(y|x_0, \hat{y}) = 0.$$

Predictive parity occurs when the predictive accuracy of a decision-making model is consistent across different groups. In other words, a model achieves predictive parity if it predicts outcomes with equal precision for all demographic groups. Furthermore, *Demographic parity* is similar to statistical parity, although it commonly puts more emphasis that the sensitive attribute is demographic. *Calibration* is also similar to predictive parity, whereas predictive parity puts more focus on the positive outcome and the general likelihood.

These fairness concepts are of great importance for evaluating and improving the ethical implications of automated decision-making systems, ensuring they operate justly across all segments of society.

2.3.1 Fair ML tasks

In the literature, we see that there are commonly three distinct tasks in fair ML: bias detection and quantification, fair prediction, and fair decision-making[25, 14]. Our focus in the method will mainly be on the second task.

Task 1. Bias Detection and Quantification

The first and commonly most simple task of fair ML is detection and quantification. Let us consider a dataset D of past outcomes, or an infinite amount of samples with an observed distribution $P(V)$ over variables V. The task is to define a mapping[25]:

$$\mu : P \to \mathbb{R},$$

where P is the set of possible distributions $P(V)$, and μ is viewed as a *fairness measure*, with the aim that$\mu(P(V)) = 0$, which suggests the absence of discrimination.

Task 2. Fair Prediction

Fair prediction, usually, relies on the *fairness measure*. The task is to learn a distribution $P^*(V)$ while maximizing utility $U(P(V))$ and satisfying

$$|\mu(P^*(V))| \leq \epsilon,$$

with μ a *fairness measure*, as in Task 1. Within fair prediction, we also make the distinction between pre-processing, in-processing, and post-processing.

Task 3. Fair Decision-Making

With fair decision-making, we also keep in mind the well-being of certain groups over time. We might be interested in designing a policy π, which at every time step affects the observed distribution $P(V)$ (which now changes over time) so that we have

$$P_{t+1}(V) = \pi(P_t(V)),$$

and we are, perhaps, interested in controlling how $\mu(P_t(V))$ changes with t. In this work, we will mainly focus on the second task fair prediction, and then especially the pre-processing and the in-processing.

2.4 Feature Engineering

Feature engineering involves creating new features or modifying existing ones to make the data more suitable for modeling. Our focus will be only on creating new features without altering existing ones. Typically, feature engineering aims to boost model accuracy, enrich the information provided by the data, and improve its interpretability [6, 10, 27]. As we have noticed in Simpson's paradox (section 2.1) it is hard to find trends over aggregated data. Feature engineering has the potential to dissect this [6].

In a lot of applications, there is not a lot of prior knowledge or expert judgment to guide us on how to engineer these features. Therefore, we will apply automated feature engineering to see if we can introduce a method that can applied without prior knowledge or expert judgment. The creation of these new features will be done with multiple operators so that we can create a variety of features. The operations we will consider are: count, percent_true, sum, mean, num_unique, max, skew, min, std, median. We will use at max three operators to create new features, with the features that are in the business necessity (BN, definition 1.2). Thus, we can combine three operators in combination with the allowed variables to create a new variable. In the appendix A we show what the engineered features are for both experiments, and we discuss what the most used operators do.

It has been shown that it is possible to extract unbiased information from biased features by applying human-understandable transformations [27]. Feature construction by multiplication and group by aggregations are successful for extracting unbiased information. Furthermore, they show that to achieve both high accuracy and fairness, it is best to extract as much unbiased information as possible from inadmissible features using feature construction methods that apply nonlinear transformations. Thus, one can use feature construction first to generate more possible candidate features and then to drop inadmissible features and optimize for fairness and accuracy.

Chapter 3

Method

Understanding the causal mechanisms in the data more deeply is important, as we've learned from the demonstration of Simpson's Paradox in section 2.1. It has shown us that mere associations can be misleading. Thus, our goal is to extract additional insights from the existing features, by applying automated feature engineering, as discussed in section 2.4. We will show what the implications are of feature engineering in the SFM, and if it will still adhere to the structural criteria, introduced in 2.2.1. Then we will also show how we construct a fairer predictor (fair ML task 2), than the current ground truth in the data, based on the fairness measured defined in section 2.2.2.

3.1 Feature Engineering in the SFM

Our aim with automated feature engineering is to create new causal pathways through which the sensitive attribute can affect the outcome. We know that we can not alter the TV by adding new features to the dataset, as the TV measure solely depends on the X and Y values, as we can see in table 2.1. Thus, our goal is to alter the outcome of the empirical fairness measures (NDE, NIE, Exp-SE). For example, if the indirect effect is allowed, and the spurious effect is not. We can try making new paths from allowed features to create new allowed paths, to minimize the effect from the not allowed paths. To prevent creating a new form of discrimination, the newly created features must be in the BN set. For this to hold we need to make the following assumption.

Assumption 1 (Consistency of the business necessity)**.** If two features belong to the BN set, then any new feature engineered from these two will also be included in the BN set. Conversely, if either one or both of the original features are not part of the BN set, then the newly engineered feature will similarly be excluded from the BN set.[1]

This assumption serves as our justification for using the newly engineered features. By developing these new pathways, we aim to diminish the impact of discriminatory pathways, thereby reducing discrimination in our model. The reasoning behind this approach is that models tend to be more discriminatory when insufficient data is available. More comprehensive information can provide a truer representation of real-world scenarios and help minimize discriminatory effects.

In the process of engineering new features, we will also be creating a new kind of structure for the SFM. We can figure out what this new structure might look like and understand its potential effects on outcomes. As mentioned in assumption 1, the new features must be part of the BN, as these are the only features through which a causal effect on the outcome is allowed.

[1] In section 2.4, we also saw that it is possible to extract unbiased information from sensitive attributes[27]. We will leave that open for now, but is interesting for further research.

Furthermore, we will explore two different scenarios in which automated feature engineering is applied, distinguishing how each approach contributes to the model's development and effectiveness.

1. We assume that all features in W are in the BN, and so we will engineer new features from the features in W.

2. We assume that all features in W and Z are in the BN, and so we will engineer features from W and Z.

These two scenarios reflect real-world situations and are thus valuable for analysis. One might question why we do not explore a scenario where features are solely derived from Z, but it's important to note that Z primarily captures demographic information, while X may only have a spurious effect on Z. Conversely, W includes other mediators through which X can indirectly influence outcomes (see definition 2.6, SFM). Hence, a model excluding W from the BN would be quite unexpected, as that would mean that we are trying to predict an outcome only on demographic information. And, excluding both W and Z, would imply that we are predicting only with the use of a sensitive attribute, which is also not realistic.

It's also possible that only certain features from W or Z are included in the BN. However, the primary objective of these two scenarios is to demonstrate potential changes in the structures and assess whether they could enhance model performance. So if we can prove that both scenarios are beneficial, then we know that a scenario where a few features of W and Z are still beneficial. Let's now take a look at the newly created SFM structures, and if the defined fairness measures still uphold admissibility and decomposability with the new structure.

3.1.1 Scenario 1; W in the BN

The newly engineered features, W' are all from W, so there will be a direct causal path from W to W'. In figure 3.1 we see how the new structure at least should look like.

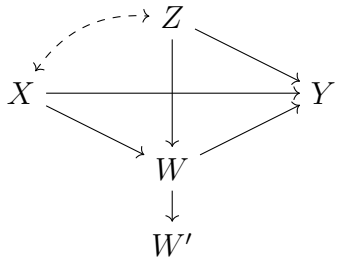


Figure 3.1: SFM, with automated feature engineering on W. The newly created features are in the group W'

The current question is whether there are additional causal mechanisms, particularly if there exists a path between W' and Y. To explore this, we will conduct conditional independence tests to identify further causal mechanisms. Initially, one might think to test $W' \perp\!\!\!\perp Y$, but this approach fails as the collider path $W' \leftarrow W \rightarrow Y$ remains open. We apply the concept of d-separation (see definition 2.7) to determine whether a path is open or closed. A subsequent logical step is to condition on W, resulting in the test $W' \perp\!\!\!\perp Y|W$. However, the situation complicates if there are potential causal paths from X to W', or Z to W', which we must consider possible. Thus, the paths $W' \leftarrow X \rightarrow Y$ and $W' \leftarrow Z \rightarrow Y$ remain open. Unfortunately, this approach also proves not usable. We then attempt to condition on both X and Z as well, leading us to test $W' \perp\!\!\!\perp Y|W, X, Z$. This test shows that no new paths open, and it represents the minimal set necessary for the analysis. We document these findings in table 3.1

This single conditional independence test offers insights into the causal relationship between W' and Y. However, even if this test suggests that they are not independent, the exact nature of their relationship remains uncertain. We will explore this further after conducting the conditional independence test.

Conditional Independence test	Open paths (backdoor paths)
$W' \perp\!\!\!\perp Y$	$W' \leftarrow W \rightarrow Y$
$W' \perp\!\!\!\perp Y \mid W$	$W' \leftarrow X \rightarrow Y$ & $W' \leftarrow Z \rightarrow Y$
$W' \perp\!\!\!\perp Y \mid W, X$	$W' \leftarrow Z \rightarrow Y$
$W' \perp\!\!\!\perp Y \mid W, Z$	$W' \leftarrow X \rightarrow Y$
$W' \perp\!\!\!\perp Y \mid W, X, Z$	-

Table 3.1: Conditional independent tests on the SFM extended with W', and the open (backdoor) paths that open up, while conditioning on specific features.

Conditional independence test

Pearson's X^2 test, also known as the Chi-square test for independence, is a statistical test used to determine whether there is a significant association between two categorical variables [18]. The test calculates a X^2 statistic that measures the discrepancy between observed counts and the counts one would expect if there was no association between the categories. A small p-value (< 0.05) leads to rejecting the null hypothesis of independence, indicating that there is a dependence between the variables. A large p-value suggests retaining the null hypothesis, indicating independence. In the second column of table 3.2 we noted the outcome of the Pearson's X^2 test. The degree of freedom for this test is 6896. This is a relatively high number,

Table 3.2: Pearson's correlation test results

Metric	$W' \perp\!\!\!\perp Y \mid W, X, Z$	$ZW \perp\!\!\!\perp Y \mid W, Z, X$
Correlation Coefficient	-0.060908	-0.052627
Degrees of Freedom (df)	6896	6896
P-value	4.14×10^{-7}	1.225×10^{-5}
Alternative Hypothesis	There is a correlation	There is a correlation

suggesting that the estimate of the correlation coefficient is based on a large sample size, which generally provides more reliable results.

The p-value is 4.14e-07, which is extremely small (much less than 0.05). A small p-value indicates strong evidence against the null hypothesis, which states there is no correlation between the variables. So we reject the null hypothesis, concluding that there is a statistically significant correlation between the variables.

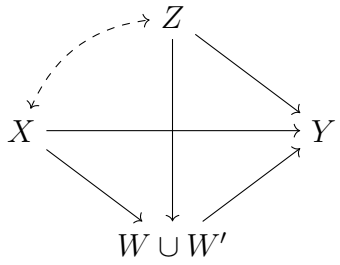


Figure 3.2: SFM, with automated feature engineering on W. The newly created features are in the group W'

We can conclude that $W' \not\perp\!\!\!\perp Y \mid W, X, Z$, so what now will be the true structure with the newly created features? We know for sure that it can't have a direct causal effect on either X or Z, as this will break the *semi-Markovian* (see definition 2.3) property. So the possibilities that still rest are a spurious or a direct causal link to Y. If there is a spurious relation between W' and Y, then the SCM is no longer identifiable (definition 2.15). This implies that there is a direct causal mechanism from W' to Y. Furthermore, there could be a causal mechanism going directly from either X or Z, but it would not have an effect on the identifiability or the structure criteria,

so we put them in just in case. So we will obtain the structure that can be seen in figure 3.2. Keep in mind that we will set the topological order such that the newly constructed features are higher in the index than the already existing ones, as in figure 2.3. This will give us the green light to see if the new features will also increase the accuracy of the in-processing optimization.

3.1.2 Scenario 2; Z and W in the BN

The newly engineered features, ZW are from W and Z, so there will be a direct causal path from W to ZW, and from Z to ZW. The proof will be analogues to the proof given in the prior section, so we already assume the following structure that can be seen in figure 3.3.

Once more, we need to assess whether there are additional causal relationships beyond those from Z and W. We will first determine if there is a causal mechanism working from ZW to Y, so we need to decide which conditional independence test to employ. In table 3.3, using a comparable approach as to our previous one, we have conducted an analysis to determine which features to condition on. We see that the minimal set to test the independence is $\{W, Z, X\}$, so we will test $ZW \perp\!\!\!\perp Y|W, Z, X$. Assuming there could be a path from X to ZW.

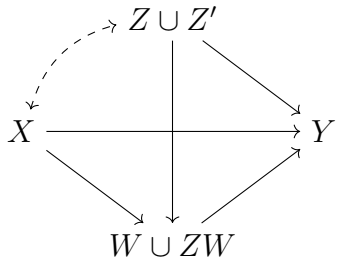


Figure 3.3: SFM, with automated feature engineering on W. The newly created features are in the group W'

Conditional Independence test	Open paths
$ZW \perp\!\!\!\perp Y$	$ZW \leftarrow W \rightarrow Y$ & $ZW \leftarrow Z \rightarrow Y$
$ZW \perp\!\!\!\perp Y\|W$	$ZW \leftarrow Z \rightarrow Y$
$ZW \perp\!\!\!\perp Y\|Z$	$ZW \leftarrow W \rightarrow Y$
$ZW \perp\!\!\!\perp Y\|W, Z$	$ZW \leftarrow X \rightarrow Y$
$ZW \perp\!\!\!\perp Y\|W, Z, X$	-

Table 3.3: The possible conditional independence tests to apply on the SFM extended with ZW. It outlines the paths that are open when applying the specified conditional independence test.

Conditional independence test

We see in table 3.2 that the p-value is 1.225e-05, which is small (less than 0.05). A small p-value indicates strong evidence against the null hypothesis, which states there is no correlation between the variables. So, we reject the null hypothesis, concluding that there is a statistically significant correlation between the variables, so $ZW \not\perp\!\!\!\perp Y|W, Z, X$.

We will make a distinction between two sets of features that we engineer. One set, Z', is solely engineered from features of Z, and the other set ZW is engineered from either Z and W, or only W. As the SFM is a semi-markovian graph, there can be no circles in the graph. So, there may be no path from ZW or Z' to X. So the possibilities that still rest are a spurious or a direct causal link to Y. So the possibilities that still rest are a spurious or a direct causal link to Y from both sets. If there is a spurious relation between ZW or W and Y, then the SCM

is no longer identifiable (definition 2.15). This implies that there is a direct causal mechanism from Z' and W to Y. Furthermore, there could be a causal mechanism going directly from either X or Z to ZW, but it would not have an effect on the identifiability or the structure criteria, so we put them in just in case. So we will obtain the structure that can be seen in figure 3.2. For Z' there could be a spurious causal mechanism coming from X, so we put them in just in case. So we will obtain the structure that can be seen in figure 3.2. Keep in mind that we will set the topological order such that the newly constructed features are higher in the index than the already existing one, as in figure 2.3. This will give us the green light to see if the new features will also increase the accuracy of the in-processing optimization.

3.2 Fair Prediction

In the second task and step, fair prediction, we will construct a predictor $\hat{Y}$ in the SFM. We will make use of an in-process approach in which the fairness constraints are incorporated in the learning process. We discussed that data could be inherently biased meaning human bias has been involved or measuring bias. Meaning that if this is the case, our predictor should find a new ground truth, such that these predictions are no longer biased.

Figure 3.4 illustrates the appearance of the SFM with the newly constructed predictor $\hat{Y}$. The green lines represent the newly added causal mechanisms between the features. Our objective is to mimic the outcome as closely as possible to achieve high accuracy while minimizing bias. This ensures that the new predictor complies with all our structural fairness criteria. This immediately highlights the inherent trade-off between accuracy and fairness [23]. Furthermore, we do not want a causal mechanism working between Y and $\hat{Y}$, as we want to create a predictor, which does not know the true outcome beforehand. In section 2.3.1 we defined that the fair prediction task is to learn a distribution $P^*(V)$ while maximizing utility $U(P(V))$ and satisfying

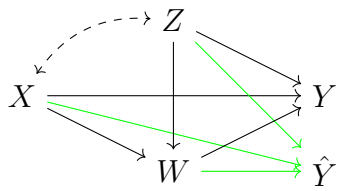


Figure 3.4: SFM, with a constructed predictor $\hat{Y}$ for fair prediction. All arrows in green are new because of the predictor.

$$|\mu(P^*(V))| \leq \epsilon,$$

with μ a *fairness measure*.

Figure 3.5 provides a comprehensive graphical representation of the structural fairness criteria and their corresponding fairness measures as utilized in our method. The figure is divided into two main sections: the measurable/empirical part and the SCM. In the top section, labeled "Measurable / Empirical," the TV is shown to be decomposable (theorem 1) into several components: (Exp-SE$_{x_2}$, Exp-SE$_{x_1}$), NDE, and NIE. These components are identified as admissible and are linked to their respective structural criteria in the SCM, specifically the Str-SE, Str-DE, and Str-IE. The arrows indicate the admissibility of the experimental effects and natural effects into the corresponding structural fairness criteria. This approach clarifies the decomposition of TV and describes the pathway from empirical measures to structural fairness criteria, supporting the methodological framework of our research.

As the predictor is constrained on the fairness measures, we need to determine how we will define the loss function and the constraints. Drago Plecko proposed an optimization problem for this predictor, which we will also use [25].

Theorem 2 (In-processing with Causal Constraints [25])**.** Let $\mathcal{M}$ be an SCM compatible with the SFM, and suppose none of the features are in the BN. Let the predictor $\hat{Y}$ be constructed

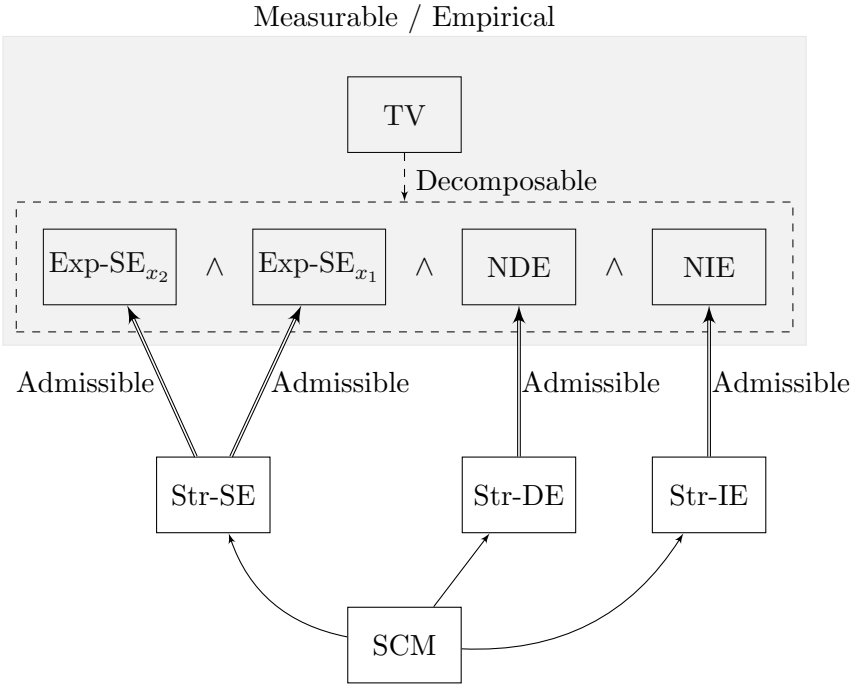


Figure 3.5: A graphical representation of the structural fairness criteria (Str-DE, Str-SE, and Str-IE) alongside their corresponding fairness measures (NDE, NIE and Exp-SE), as applied in our method.

as the optimal solution to

$$\hat{Y} = \arg\min_{f} \mathbb{E}[Y - f_{\hat{Y}}(X, Z, W)]^2$$

$$\begin{aligned} \text{subject to} \quad & \text{NDE}_x^{\text{sym}}(\hat{y} \mid x_0) = 0 && (3.1)\\ & \text{NIE}_x^{\text{sym}}(\hat{y} \mid x_0) = 0 && (3.2)\\ & \text{ExpSE}_{x_1,x_0}(\hat{y}) = 0 && (3.3)\\ & \text{ExpSE}_{x_0,x_1}(\hat{y}) = 0 && (3.4) \end{aligned}$$

With

$$\text{NDE}_x^{\text{sym}}(y \mid x) = \frac{1}{2}\left(\text{NDE}_{x_0,x_1}(y \mid x) - \text{NDE}_{x_1,x_0}(y \mid x)\right)$$
$$\text{NIE}_x^{\text{sym}}(y \mid x) = \frac{1}{2}\left(\text{NIE}_{x_0,x_1}(y \mid x) - \text{NIE}_{x_1,x_0}(y \mid x)\right)$$

Interestingly note that if we would just optimize for the TV measure, we would be able to keep the TV measure close to 0, but this would not imply Str-DE-fair, Str-IE-fair, or Str-SE-fair (fair prediction theorem [25]). Nevertheless, practitioners could adjust the fairness measures, as long as they uphold the admissibility and decomposability criteria. So one could also take one of the other fairness measures described in table 2.1.

This optimization problem gives us the loss function defined in 3.5. The parameter λ is a scaling factor that adjusts the overall contribution of the fairness measures to the loss function. So a higher λ implies more emphasis on the fairness restrictions. The goal of this loss function is to ensure that the predicted outcome $\hat{y}$ adheres to specified fairness criteria by minimizing the differences between the calculated effects and their respective thresholds.

$$L(y, \hat{y}) = \lambda\left(|\text{NDE}_{x_0,x_1}(\hat{y}) - \eta_1| + |\text{NIE}_{x_1,x_0}(\hat{y}) - \eta_2| + \left|\text{Exp-SE}_{x_1}(\hat{y}) - \eta_3\right| + \left|\text{Exp-SE}_{x_0}(\hat{y}) - \eta_4\right|\right) \quad (3.5)$$

Figure 3.6 illustrates the workflow of the model, highlighting both the normal workflow and the extension involving automated feature engineering. The normal workflow, shown in black, starts with the dataset D, which is used directly in the optimization process with fairness constraints, as specified in equations 3.1 to 3.4. The predictor $\hat{Y}$ is then derived as a function of x, z, and w. The extension, illustrated in green, introduces an additional step of automated feature engineering, which processes the data D before it enters the ML optimization phase. This automated feature engineering step aims to enhance the model by systematically generating new features that potentially improve the model's performance while adhering to fairness constraints, as described in section 2.4. This integrated approach ensures that the predictor not only achieves high accuracy but also mostly increases fairness in its predictions.

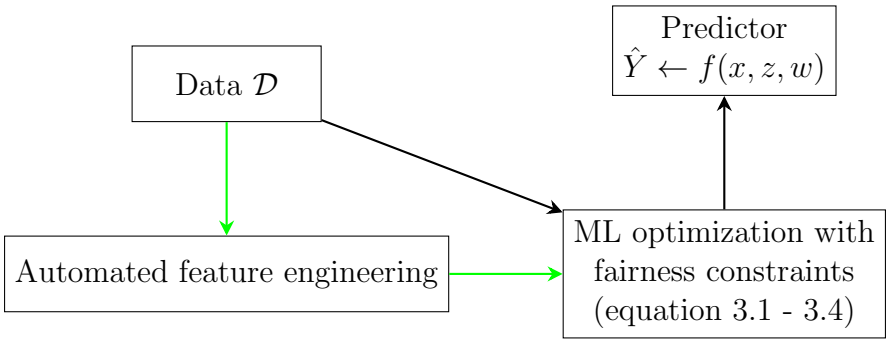


Figure 3.6: A schematic summary of the method's workflow, with the standard workflow in black and the extension with automated feature engineering in green.

Chapter 4

Experiments

In the experiment, we have two different models that create a fair predictor, task two of fair ML. The two different models are:

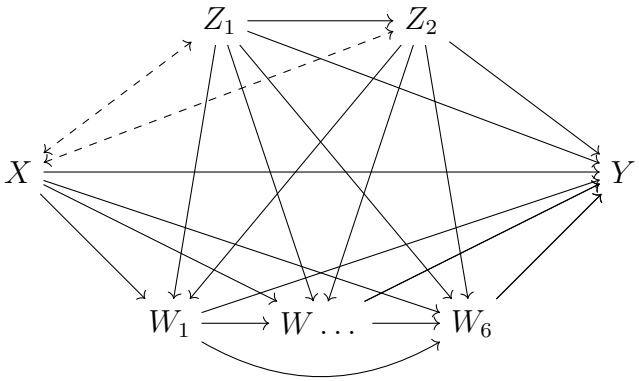
1. baseline; this model creates a fair predictor as described in section 3.2.
2. the extension with automated feature engineering; automated feature engineering will be performed as an extra pre-processing step, as described in section 3.1.

We aim to achieve a better performance by applying automated feature engineering as a preprocessing step. We will measure performance with accuracy and the fairness measures introduced in section 2.2.2; NDE, IDE, Exp-SE.

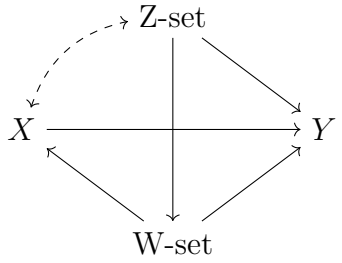
4.1 Dataset

COMPAS (Correctional Offender Management Profiling for Alternative Sanctions) is a popular commercial algorithm used by judges and parole officers for scoring a criminal defendant's likelihood of reoffending (recidivism). ProPublica concludes that the score (outcome) is performing differently among racial subgroups [8]. The data contains over 20 variables used by the COMPAS algorithm in scoring defendants, along with their outcomes (if they recidivate or not) within 2 years of the decision, for over 10,000 criminal defendants in Broward County, Florida.

We will group the variables, such that it is compatible with the SF. For X we will take *race*, as that is the sensitive feature we want to test for. Of course, one could also say that he would like to investigate possible discrimination based on *age* or *sex*, which is also possible. Furthermore, the outcome, Y will be *two year recid*, as that is the true outcome whether a person is a recidivism. *Age* and *sex* are in Z, and *juvenile felony count*, *juvenile misdemeanors*, *juvenile others*, *prior crimes*, *charge degree*. And we added one extra feature *days in prison*, subtracting the *day out* date from the *day in* date. This will lead to the following structure.



Notice, that there is a topological order in the variables, as shown in section 2.3. This topological order is of importance when incorporating automated feature engineering to uphold identifiability (definition 2.15). So if we group these variables, one can see the following structure:



This is exactly the SFM structure we have seen in definition 2.6. As mentioned in section 3.1 we will assume two approaches:

1. We will assume W in the BN
2. We will assume W and Z in the BN

If W is in the BN, then we will optimize such that the NDE, ExpSE_{x_0} and ExpSE_{x_1} will be minimized. If W and Z are in the BN, then only the NDE should be close to zero. If a variable is in the BN, there may be a path from the sensitive attribute through that variable to the outcome. That is why if W and Z are in the BN, the only path, which is not allowed is the direct path from X to Y, so the DE.

Furthermore, we will also look at the effect λ on the results, as that determines the strength of the fairness restrictions. So, a higher λ will have more fairness, but less accuracy [23]. This inherent trade-off is important to consider for the practitioner and the executioner.

4.2 Results

Both experiments have been done in both Rstudio and Python. The automated feature engineering is done in Pyton, and the optimization and conditional independence tests are in Rstudio. For the automated feature engineering, we only kept 7 (W in the BN) and 8 (W and Z in the BN) newly created features, which have shown to have the biggest effect on the prediction. In table A.1 and table A.2 we documented the features that were engineered and added.

4.2.1 Feature Engineering on W

Table 4.1 presents the results of the evaluation for λ values of 0.1, 0.5, and 1, comparing the baseline model to the Automated Feature Engineering (AFE) model. The key metrics evaluated are Accuracy, NDE, and ExpSE.

For $\lambda = 0.1$, the accuracy values are 0.683 (baseline) and 0.687 (AFE). The AFE model shows a slight increase in accuracy, indicating potentially better performance under this setting. $\lambda = 0.5$, the accuracy values are 0.683 (baseline) and 0.692 (AFE). Again, the AFE model demonstrates an increase in accuracy, suggesting improved performance. For $\lambda = 1$, the accuracy values are 0.665 (baseline) and 0.660 (AFE). Both models exhibit similar accuracy, with a slight decrease for the AFE model. So an overall increase in accuracy

For $\lambda = 0.1$, the NDE values are 0.066 (baseline) and 0.018 (AFE). The standard deviation suggests that the null hypothesis

$$H_0 : \text{NDE}_{x_0,x_1}(y) = 0$$

λ values	$\lambda = 0.1$		$\lambda = 0.5$		$\lambda = 1$	
	Baseline	**AFE**	Baseline	**AFE**	Baseline	**AFE**
Accuracy_{sd}	$0.683_{0.015}$	$0.687_{0.013}$	$0.683_{0.012}$	$0.692_{0.012}$	$0.665_{0.010}$	$0.660_{0.010}$
NDE_{sd}	$0.066_{0.010}$	$\mathbf{0.018_{0.010}}$	$0.028_{0.008}$	$\mathbf{0.015_{0.008}}$	$0.008_{0.012}$	$0.007_{0.007}$
ExpSE_{x_1}	$-0.024_{0.004}$	$-0.028_{0.004}$	$\mathbf{-0.015_{0.002}}$	$-0.019_{0.003}$	$-0.008_{0.002}$	$-0.007_{0.002}$
ExpSE_{x_0}	$0.013_{0.002}$	$0.012_{0.001}$	$0.010_{0.001}$	$\mathbf{0.007_{0.001}}$	$0.005_{0.001}$	$\mathbf{0.003_{0.001}}$

Table 4.1: The top row displays values of λ. Directly below each λ, the results for baseline and Automated Feature Engineering Extension are grouped. There are four metrics presented in the rows: Accuracy, NDE, ExpSE_{x_0}, and ExpSE_{x_0}. The standard deviation is noted in lowercase. If the values of the predictor for either baseline or Automated Feature Engineering(AFE) are significantly higher, then those numbers are highlighted in bold.

can be rejected for both, but the AFE model is less discriminatory. For $\lambda = 0.5$, the NDE values are 0.028 (baseline) and 0.015 (AFE). Similarly, the null hypothesis is rejected for both. however, the AFE model again outperforms the baseline. For $\lambda = 1$, the NDE values are 0.008 (baseline) and 0.007 (AFE). Both values are low, and the null hypothesis is not rejected, thus suggesting both are Str-DE-fair.

For $\lambda = 0.1$, the ExpSE values for x_1 are -0.024 (baseline) and -0.028 (AFE), while for x_0 they are 0.013 (baseline) and 0.012 (AFE). The slight differences suggest that the AFE model does not significantly improve this aspect over the baseline and for both the null hypothesis

$$H_0 : \text{Exp-SE}_x(y) = 0, \quad \text{for } x \in \{x_0, x_1\}$$

is rejected. For $\lambda = 0.5$, the ExpSE values for x_1 are -0.015 (baseline) and -0.019 (AFE), while for x_0 they are 0.010 (baseline) and 0.007 (AFE). Again, the AFE model shows similar results. For $\lambda = 1$, the ExpSE values for x_1 are -0.008 (baseline) and -0.007 (AFE), while for x_0 they are 0.005 (baseline) and 0.003 (AFE). The results suggest that the AFE model has a marginal improvement in fairness regarding ExpSE. We see that for all λ the null hypothesis is rejected, implying that it is not Str-SE-fair.

Overall, the AFE model demonstrates better accuracy and fairness in terms of NDE across different λ values. However, the improvements in ExpSE are marginal, suggesting that while AFE contributes to fairness, the effect is not substantial for all fairness measures. For $\lambda = 1$ both models suggest being Str-DE-fair, and so uphold disparate treatment.

4.2.2 Feature Engineering on W and Z

λ values	$\lambda = 0.1$		$\lambda = 0.5$		$\lambda = 1$	
	Baseline	AFE	Baseline	AFE	Baseline	AFE
Accuracy_{sd}	$0.683_{0.012}$	$0.688_{0.013}$	$0.683_{0.011}$	$0.685_{0.011}$	$0.682_{0.011}$	$0.687_{0.011}$
NDE_{sd}	$0.005_{0.011}$	$0.006_{0.011}$	$0.011_{0.016}$	$0.014_{0.015}$	$0.003_{0.013}$	$0.011_{0.012}$

Table 4.2: The top row displays values of λ. Directly below each λ, the results for baseline and Automated Feature Engineering (AFE) are grouped. There are four metrics presented in the rows: Accuracy, NDE. The standard deviation is noted in lowercase. If the values of the predictor for either baseline or automated feature engineering are significantly higher, then those numbers are highlighted in bold.

For $\lambda = 0.1$, the accuracy values are 0.683 (baseline) and 0.688 (AFE). The AFE model shows a slight increase in accuracy, indicating potentially better performance under this setting. For $\lambda = 0.5$, the accuracy values are 0.683 (baseline) and 0.685 (AFE). Again, the AFE model demonstrates a slight increase in accuracy, suggesting improved performance. For $\lambda = 1$, the accuracy values are 0.682 (baseline) and 0.687 (AFE). The AFE model shows a slight increase in accuracy, indicating potentially better performance.

For $\lambda = 0.1$, the NDE values are 0.005 (baseline) and 0.006 (AFE). The standard deviation suggests that the null hypothesis $H_0 : \text{NDE}_{x_0,x_1}(y) = 0$ cannot be rejected for either model, suggesting that both models are Str-DE-fair. This also holds for the other λ values. For $\lambda = 0.1$ we already fulfill disparate treatment, as the NDE values fall within the standard deviation range of 0. Consequently, this value will not be further optimized for higher λ values, making comparisons between NDE values redundant.

For both feature engineering on W alone and on W and Z together, maintaining disparate impact is impossible due to the assumed presence of features in the BN. With the current approach, the disparate impact cannot be upheld unless the following property holds:

$$X \perp\!\!\!\perp Y$$

This would instantly make the entire fairness analysis redundant, as there would be no bias or causal mechanism linking X to Y.

Chapter 5

Related work

Feature engineering is a frequently employed pre-processing step that has been demonstrated in many papers to have a positive effect on accuracy and interpretability, and enrich the information provided by the data [6, 27]. The work of Salazar is unique in its way that also incorporates causal feature selection [27]. Causal feature selection, which follows automated feature engineering, retains only those features that do not introduce bias [5]. This paper initially aimed to apply this concept to the SFM framework. However, feature engineering on features permitted to influence the outcome had not been previously undertaken. Therefore, this research focuses on the initial step: incorporating automated feature engineering. Future research could explore extending feature engineering to features not in the business necessity using Galhotra's approach[5]. After discussions with Galhotra and Salazar, it became evident that implementing these steps would be too time-consuming. Additionally, incorporating causal feature selection would significantly increase computational time, as observed in Salazar's work, making its application more challenging

The foundation of this research is built upon the pioneering contributions of Bareinboim, Pearl, and Plecko in the fields of causal inference and causal fairness analysis. My first inspiration came from Judea Pearl's book, Causal Inference in Statistics: A Primer, recommended by a colleague at the University of Amsterdam [15]. This book introduced fundamental concepts such as the ladder of causality, potential outcomes, and counterfactuals [15]. It immediately became clear that this work effectively quantified bias and discrimination and was easily interpretable, bridging the gap between data scientists and other professionals. Pearl's earlier works defined broader concepts such as causal graphs, SCMs, and interventions [16].

Drago Plecko's contributions have significantly advanced our understanding in several key areas. His research has demonstrated how to decompose spurious effects, making us able to create path-specific measures, and can also be used to apply feature engineering on specific features in a set [21]. how to create a fair predictor, as we have used in section 3.2 [20]. show similarities between multiple fairness notions [22]. the inherent trade-off between accuracy and fairness [23]. And, fair data pre-processing in the fair-adapt package [24]. Most of all he put all this work together creating a new field of causal fairness analysis [25]. Furthermore, Drago Plecko was of great support during my research by providing insights and explaining his own work. The extension of his work is due to the help of Drago Plecko himself, Sainyam Galhotra, and Ricardo Salazar.

Furthermore, to deepen my understanding of the field of fair machine learning, I have drawn insights from Pagano's and Mehrabi's surveys [14, 12]. These surveys have been of great help in shaping my understanding of bias, discrimination, and fairness concepts, and have underscored the great amount of fairness criteria that exist in the literature [14, 22]. This further emphasizes the need to clarify and unify the field's diverse notions, ensuring broader applicability across domains.

Chapter 6

Conclusion

We applied automated feature engineering as the preprocessing step in the creation of a fair predictor, task two of the fair ML tasks. We compared it to the baseline, the same process but without automated feature engineering. Generally, automated feature engineering improves slightly or maintains accuracy, and improves significantly in fairness, but not significantly. Fairness was measured with the natural direct effect (NDE), natural indirect effect (NIE) and experimental spurious effect (Exp-SE). The extension is especially beneficial in models more susceptible to bias as it introduces new causal paths to reduce the importance of discriminatory paths. The results indicate that this approach enhances model accuracy and the management of causal effects. Additionally, the computational complexity introduced by automated feature engineering is minimal. However, it's important to carefully examine the features that are engineered and to consult with domain experts to evaluate these features thoroughly.

We applied automated feature engineering in two different scenarios; (1) Automated feature engineering on features of W, and (2) Automated feature engineering on features of $W\&Z$. For both scenarios, we have shown that the newly created structure of the SFM adheres to its criteria. Also, the new structure indicates that it can dissect more information from the features, giving us the possibility to handle trends across different subgroups, better known as Simpson's paradox.

Automated feature engineering on features of W has shown to improve results for fairness and tends to perform better on accuracy. So it shows promising results in debiasing while having also a slightly better accuracy. The consistent improvements for increasing λ, suggest that automated feature engineering is robust for multiple parameter settings. This analysis implies that automated feature engineering might be a valuable extension to the baseline model, offering enhanced performance, particularly in terms of debiasing.

Automated feature engineering on features of both W and Z consistently performs slightly better than the baseline on accuracy, but not significantly. The NDE values for both methods are within the standard deviation range of 0, so the null hypothesis is accepted for the given values of λ, and this is seen as fair. So this value is not further optimized. As in the prior experiment, the performance is consistent for different values of λ. As both models are considered fair, there is no difference in which is more fair than the other. Making us unable to say anything regarding fairness improvements.

In summary, the SFM proposed by Drago Plecko represents an advancement in the domain of fair ML and causal inference. This model excels in identifying and controlling biases within data and the modeling process. Its well-defined structure not only improves the interpretability of outcomes but also makes them easier to manage. Furthermore, with the ability to quantify direct, indirect, and spurious effects, the model provides a clear representation of the biases that may arise, enhancing the transparency of the ML process. This transparency is necessary with the introduction of the AI act in Europe. Also, the model is based on the American law

concepts of disparate impact and treatment, thus it can verify if it upholds those laws or not. Therefore, the contribution of this work is not solely an academic one, but also a legal one.

Appendix A

Feature engineering

The operators used for feature engineering typically come from a data manipulation, related to aggregating or transforming data to uncover more insights or improve predictive models. Here's a general interpretation of each operator used in the feature names:

PERCENT_TRUE: This operator calculates the percentage of true or positive instances within a specified group. For example,
days_in_jail.PERCENT_TRUE(recidivism_data.c_charge_degree) computes the percentage of cases where c_charge_degree is true among all records grouped by days_in_jail.

NUM_UNIQUE: This operator counts the number of unique values in a specific column for each group. For instance, days_in_jail.NUM_UNIQUE(recidivism_data.priors_count) would determine the number of unique prior count values for each distinct number of days spent in jail.

COUNT: This operator totals the number of entries for each group.
priors_count.COUNT(recidivism_data) counts the total number of records grouped by the priors_count.

These operators help in transforming raw data into features that can better represent underlying patterns in data.

Table A.1: Newly Engineered Features for W

Feature Engineering on W
days_in_jail.PERCENT_TRUE(recidivism_data.c_charge_degree)
days_in_jail.NUM_UNIQUE(recidivism_data.priors_count)
priors_count.NUM_UNIQUE(recidivism_data.days_in_jail)
priors_count.COUNT(recidivism_data)
priors_count.PERCENT_TRUE(recidivism_data.c_charge_degree)
days_in_jail.NUM_UNIQUE(recidivism_data.juv_fel_count)
days_in_jail.NUM_UNIQUE(recidivism_data.juv_other_count)

Table A.2: Newly Engineered Features for W and Z

Feature Engineering on W and Z
ages.PERCENT_TRUE(recidivism_data.c_charge_degree)
ages.COUNT(recidivism_data)
ages.NUM_UNIQUE(recidivism_data.days_in_jail)
ages.PERCENT_TRUE(recidivism_data.sex)
ages.NUM_UNIQUE(recidivism_data.priors_count)
days_in_jail.PERCENT_TRUE(recidivism_data.c_charge_degree)
days_in_jail.COUNT(recidivism_data)
days_in_jail.PERCENT_TRUE(recidivism_data.sex)

Bibliography

[1] Solon Barocas and Andrew D. Selbst. Big data's disparate impact. *California Law Review*, 104:671, 2016.

[2] Silvia Chiappa and Thomas P. S. Gillam. Path-specific counterfactual fairness. *stat.ML*, arxiv(arXiv:1802.08139), 2018.

[3] Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments, 2017.

[4] Federal Trade Commission. Equal employment opportunity commission, 1974. https://www.ftc.gov/policy-notices/no-fear-act/protections-against-discriminatio.

[5] Sainyam Galhotra, Karthikeyan Shanmugam, Prasanna Sattigeri, and Kush R. Varshney. Causal feature selection for algorithmic fairness. *arxiv*, 2022.

[6] Jeff Heaton. An empirical analysis of feature engineering for predictive modeling. In *SoutheastCon 2016*. IEEE, mar 2016.

[7] Jesus Hernandez. Redlining revisited: Mortgage lending patterns in sacramento 1930-2004. *International Journal of Urban and Regional Research*, 33:291–313, 2009.

[8] Lauren Kirchner Jeff Larson, Surya Mattu and Julia Angwin. How we analyzed the compas recidivism algorithm. *ProPublica*, 2016.

[9] Chris van Dam et al. Jeroen van Wijngaarden, Roy van Aalst. Ongekend onrecht, brief van de parlementaire ondervragingscommissie, 2020.

[10] Udayan Khurana, Deepak Turaga, Horst Samulowitz, and Srinivasan Parthasrathy. Cognito: Automated feature engineering for supervised learning. *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, pages 1304–1307, 2016. Conference Name: 2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW) ISBN: 9781509059102 Place: Barcelona, Spain Publisher: IEEE.

[11] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[12] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM*, 54(6):115:1–115:35, 2021.

[13] Future of Life Institute (FLI), 2024. https://artificialintelligenceact.eu/.

[14] Tiago P. Pagano, Rafael B. Loureiro, Fernanda V. N. Lisboa, Rodrigo M. Peixoto, Guilherme A. S. Guimarães, Gustavo O. R. Cruz, Maira M. Araujo, Lucas L. Santos, Marco A. S. Cruz, Ewerton L. S. Oliveira, Ingrid Winkler, and Erick G. S. Nascimento. 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), 2023.

[15] J. Pearl, M. Glymour, and N.P. Jewell. *Causal Inference in Statistics: A Primer*. Wiley, 2016.

[16] Judea Pearl. *Causality*. Cambridge University Press, Cambridge, UK, 2 edition, 2009.

[17] Judea Pearl and Dana Mackenzie. *The Book of Why*. Basic Books, New York, 2018.

[18] Karl Pearson. X. on the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 50(302):157–175, July 1900.

[19] Authoriteit persoonsgegevens. Verkennend onderzoek gegevensbeschermingsbeleid, 2016. https://www.autoriteitpersoonsgegevens.nl/documenten/onderzoek-privacybeleid.

[20] Drago Plecko and Elias Bareinboim. Causal fairness for outcome control. *SIGMOD*, arxiv(arXiv:2306.05066), 2023.

[21] Drago Plecko and Elias Bareinboim. A causal framework for decomposing spurious variations. *SIGMOD*, arxiv(arXiv:2306.05071), 2023.

[22] Drago Plecko and Elias Bareinboim. Reconciling predictive and statistical parity: A causal approach. *SIGMOD*, arxiv(arXiv:2306.05059), 2023.

[23] Drago Plecko and Elias Bareinboim. Fairness-accuracy trade-offs: A causal perspective, 2024.

[24] Drago Plečko and Nicolai Meinshausen. Fair data adaptation with quantile preservation. *J. Mach. Learn. Res.*, 21(1), jan 2020.

[25] Drago Plečko and Elias Bareinboim. Causal fairness analysis: A causal toolkit for fair machine learning. *FNT in Machine Learning*, 17(3):304–589, 2024.

[26] Tracy A. Ruegg. Historical perspectives of the causation of lung cancer. *Global qualitative nursing research*, 2, 2015.

[27] Ricardo Salazar, Felix Neutatz, and Ziawasch Abedjan. Automated feature engineering for algorithmic fairness. *VLDB*, 14(9):1694–1702, 2021.

[28] Babak Salimi, Luke Rodriguez, Bill Howe, and Dan Suciu. Interventional fairness: Causal database repair for algorithmic fairness. In *Proceedings of the 2019 International Conference on Management of Data*, pages 793–810. ACM, 2019.

[29] Yishai Shimoni, Ehud Karavani, Sivan Ravid, Peter Bak, Tan Hung Ng, Sharon Hensley Alford, Denise Meade, and Yaara Goldschmidt. An evaluation toolkit to guide model selection and cohort definition in causal inference. *stat.ML*, arxiv(arXiv:1906.00442), 2019.

[30] Thomson Reuters. What is disparate treatment discrimination — and how is it proven?, 2022. Accessed: 2024-06-24.

[31] Jin Tian and Judea Pearl. A general identification condition for causal effects. In *Eighteenth National Conference on Artificial Intelligence*, page 567–573, USA, 2002. American Association for Artificial Intelligence.