# Addressing Interpretability, Fairness and Privacy in Machine Learning Through Combinatorial Optimization Methods

Julien Ferry


# THÈSE

**En vue de l'obtention du**

## DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

**Délivré par :** *l'Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)*

---

**Présentée et soutenue le** *09/10/2023* **par :**

**Julien Ferry**

**Addressing Interpretability Fairness & Privacy in Machine Learning Through Combinatorial Optimization Methods**

---

**JURY**

| | | |
|---|---|---|
| Josep Domingo-Ferrer | Full Professor | Rapporteur |
| Pierre Schaus | Professeur | Rapporteur |
| Elisa Fromont | Professeure des Universités | Examinatrice |
| Mathieu Serrurier | Maître de Conférences | Examinateur |
| Sylvie Thiébaux | Professor | Présidente du jury |
| Thibaut Vidal | Associate Professor | Examinateur |
| Sébastien Gambs | Professeur | Directeur de thèse |
| Marie-José Huguet | Professeure des Universités | Directrice de thèse |
| Mohamed Siala | Maître de Conférences | Co-encadrant de thèse |
| Ulrich Aïvodji | Assistant Professor | Invité |

---

**École doctorale :**
*Ecole Doctorale Mathématiques, Informatique et Télécommunications de Toulouse*
**Spécialité :**
*Informatique et Télécommunications*
**Unité de Recherche :**
*Laboratoire d'Analyse et d'Architecture des Systèmes*
**Directeur(s) de Thèse :**
*Marie-José Huguet*, *Sébastien Gambs* et *Mohamed Siala*
**Rapporteurs :**
*Josep Domingo-Ferrer* et *Pierre Schaus*

# Remerciements

Je tiens tout d'abord à remercier l'intégralité des membres de mon jury de thèse, pour leurs questions, leurs retours constructifs et pour le temps qu'ils m'ont accordé: Mme Sylvie Thiébaux, qui l'a présidé; Mr Pierre Schaus et Mr Josep Domingo-Ferrer, rapporteurs de ma thèse; et enfin Mme Elisa Fromont, Mr Thibaut Vidal et Mr Mathieu Serrurier, examinateurs.

Je remercie aussi grandement ceux qui m'ont accompagné depuis mes débuts dans le monde de la recherche: Marie-Jo, Mohamed, Sébastien et Ulrich. Le 06 Décembre 2018, je contactais Mohamed pour réaliser un stage de M1 dans le milieu de la recherche - et nous voici presque 5 ans (et un doctorat) plus tard! Merci de m'avoir fait découvrir (et aimer) ce milieu, de m'avoir suivi pendant ces 4 mois de stage et bien sûr ces trois ans de thèse. Merci pour vos qualités scientifiques, mais surtout pour votre confiance, votre aide, et votre présence pendant tous ces moments importants: premières soumissions, premiers papiers acceptés, premières conférences... Je pense aussi à tous nos moments de convivialité - en terrasse à Toulouse, mais aussi autour d'une tarte praline à Lyon, d'une poutine à Montréal, d'un bon steack à Durham, d'une tarte flambée à Strasbourg, d'une galette à Rennes, d'un burger à Los Angeles!

Mes pensées vont également à toute l'équipe ROC du LAAS-CNRS, au sein de laquelle j'ai eu la chance d'évoluer pendant ces trois années. J'ai trouvé à vos côtés un cadre de travail idéal, tant sur le plan professionnel que sur le plan personnel. Pour tous les échanges scientifiques, toutes les pauses café, toutes les séances de sport, et bien sûr pour tous les moments de partage à des heures plus ou moins tardives de la journée ou de la nuit, merci. J'ai une pensée particulière pour Carla, Alexandre, Aloïs, et Louis, camarades doctorants de la promotion 2020/21 - pour tous les instants partagés au LAAS ou pendant nos premières conférences. Je pense aussi et non sans nostalgie à la salle Arbizon au sein de laquelle, à cause des travaux dans nos bureaux, nous avons vécu ensemble la belle expérience de la rédaction! Quelques mercis particuliers à Christian (et Mohamed) pour les sorties nocturnes, à Cyrille pour sa présence et nos discussions, à Emmanuel, Hannes, Camille, Matthieu, Théo pour le vélo, la course à pied, le rugby, à Valentin et Tom, les anciens qui nous ont montré le chemin, à Théo (encore!) l'infatigable animateur de l'équipe et à Julien - mon tout premier stagiaire! Merci aussi à l'équipe PrivSec de l'UQAM, et à tous ses membres que j'ai pu voir régulièrement en visio - mais également à Montréal lors de ma visite entre Mai et Juillet 2022.

Enfin, j'aimerais remercier ma famille, pour leur support indéfectible depuis que j'ai quitté mon Aveyron natal, ses rivières, ses prairies et ses forêts, pour mener mes études dans la ville Rose - il y a déjà 8 ans! Merci aussi à mes amis - les Aveyronnais bien sûr, avec lesquel.les nous avons su rester proches malgré la distance, mais également tous ceux qui ont croisé ma route depuis 2015, et notamment les INSAïen.nes et les médecin.es. Tous nos moments partagés sont précieux!

A toutes les personnes mentionnées ci-dessus: je vous remercie une nouvelle fois, et je vous souhaite le meilleur pour l'avenir.

## Abstract

Machine learning techniques are increasingly used for high-stakes decision making, such as college admissions, loan attribution or recidivism prediction. It is thus crucial to ensure that the trained models can be audited or understood by human users, do not create or reproduce discrimination or bias, and do not leak sensitive information regarding their training data. Indeed, interpretability, fairness and privacy are key requirements for trustworthy machine learning, and all three have been studied extensively during the last decade. However, they are often considered in isolation.

The objective of this manuscript is precisely to investigate the interactions between the three different fields, using tools from combinatorial optimization and operational research. For each pairwise intersection, we review the literature on the previously observed compatibilities, tensions and synergies, and draw new insights from our proposed contributions - either highlighting an identified tension or proposing a conciliation mechanism. We first propose an Integer Linear Programming (ILP) based pruning technique for a learning algorithm producing fair and inherently interpretable models. By jointly leveraging accuracy, sparsity and fairness, it enhances the exploration of the algorithm's search space and helps conciliating fairness and interpretability. Motivated by the empirical observation that fairness often does not generalize well, we further introduce a novel framework to improve fairness robustness with respect to the training set sampling.

We then show how information regarding a model's fairness can be exploited to reconstruct its training set sensitive attributes. To this end, we propose efficient MILP and Constraint Programming (CP) models directly encoding the fairness information to improve the reconstruction performed by any baseline adversary. This highlights the intrinsic tension between enforcing fairness with respect to some sensitive attributes, and ensuring such attributes' privacy. Finally, we demonstrate how the structure of a released interpretable model can be used to reconstruct a probabilistic version of its training set. By precisely quantifying the amount of information a model encodes regarding its training data, we illustrate an apparent conflict between interpretability and privacy.

**Keywords:** Artificial Intelligence, Machine Learning, Combinatorial Optimization, Interpretability, Fairness, Privacy.

## Résumé

Les approches d'apprentissage automatique sont de plus en plus utilisées pour des problématiques de prise de décisions impactant nos vies, telles que l'admission à l'université, l'attribution de prêts ou la prédiction de récidive. Ainsi, il est crucial de s'assurer que les modèles entraînés peuvent être audités et compris par leurs utilisateurs, ne reproduisent pas ni ne créent de biais discriminatoires et ne divulguent pas d'informations sensibles sur leurs ensembles d'entraînement. Ainsi, l'interprétabilité, l'équité et la protection de la vie privée sont des propriétés indispensables pour le développement de techniques d'apprentissage dignes de confiance. Toutes trois ont été largement étudiées durant la dernière décennie mais elle sont le plus souvent considérées séparément les unes des autres.

L'objectif de cette thèse est précisément de caractériser les interactions entre ces trois domaines, en utilisant des outils d'optimisation combinatoire et de recherche opérationnelle. Considérant ces trois domaines deux à deux, nous passons tout d'abord en revue la littérature sur leurs compatibilités, tensions et synergies. Nous nous concentrons sur certaines de ces tensions et proposons soit un mécanisme de conciliation, soit des techniques permettant de mettre en exergue ou de quantifier ce conflit. Plus précisément, nous proposons d'abord une technique d'élagage basée sur la programmation linéaire en nombres entiers pour un algorithme d'apprentissage produisant des modèles équitables et intrinsèquement interprétables. En encodant conjointement précision, taille du modèle et équité, elle améliore l'exploration de l'espace de recherche de l'algorithme et aide à concilier équité et interprétabilité. Forts de la constatation expérimentale que l'équité généralise souvent mal une fois les modèles appliqués sur de nouvelles données, nous proposons une nouvelle approche visant à améliorer la robustesse de l'équité vis-à-vis de l'échantillonnage du jeu de données.

Par la suite, nous montrons par la suite comment l'information relative à l'équité d'un modèle peut être utilisée pour reconstruire les attributs sensibles de son ensemble d'entraînement. A cet effet, nous proposons des modèles de programmation linéaire en nombres entiers et de programmation par contraintes encodant directement l'information de l'équité afin d'améliorer une reconstruction effectuée en amont par un attaquant quelconque de la littérature. Ce travail illustre une tension intrinsèque entre le fait d'assurer l'équité par rapport à certains attributs sensibles et la nécessité de protéger l'information relative à ces attributs. Enfin, nous expliquons comment la structure d'un modèle interprétable peut être utilisée pour reconstruire une version probabiliste de son ensemble d'entraînement. En quantifiant précisément la quantité d'information qu'un modèle encode sur ses données d'entraînement, nous illustrons un conflit apparent entre l'interprétabilité et la protection de la vie privée.

**Mots-clés :** Intelligence artificielle, Apprentissage automatique, Optimisation combinatoire, Interprétabilité, Équité, Protection de la vie privée.

# Contents

# Introduction

Machine learning approaches were introduced more than half a century ago, with the term *machine learning* being popularized in 1959 by Arthur Samuel [Samuel 1959], an IBM employee and pioneer in self-learning programs. During the last decade, the use of these techniques has increased dramatically, driven by several factors. First, the amount of collected data has considerably grown during the past years, led by the development of the Internet of Things and sensor networks, as well as a tremendous monitoring of online behaviors. Each year, several tens of zettabytes ($10^{21}$ bytes) are being collected and stored [Øverby & Audestad 2021], and this number is expected to keep rising. Second, significant progress was done both in hardware and software, allowing to gather and process these huge amounts of data in a more efficient manner.

Machine learning methods have many useful and promising applications. For instance, they can help analyzing medical data, which is becoming increasingly complex due to the improvements in medical analysis tools. Thus, they can be used for a faster and/or more accurate medical diagnosis. In addition, they have a great business value in many fields such as advertisement targeting or recommendation algorithms. However, their growing use for high-stakes decision-making tasks - such as college admissions, recidivism prediction, credit scoring or even kidney exchange [Aziz *et al.* 2021] - raises significant ethical, philosophical and societal challenges. Furthermore, their use is also directly regulated by several legal texts, such as the recent European Union General Data Protection Regulation[1] [Voigt & Von dem Bussche 2017] or forthcoming AI Act[2].

Three main issues have been identified, each corresponding to a key concern that should be addressed to both comply with these new legal frameworks and lay the foundations towards an ethical and responsible AI. First, machine learning algorithms require large amounts of data, which often contains personal information. Thus, it is of paramount importance to ensure that the privacy of the involved individuals is not harmed while also being able to extract useful generic patterns from this data. Second, it was shown that data-driven decision-making mechanisms can create or reproduce biases that systematically disadvantage specific individuals or groups. Measuring but also reducing or eliminating these biases to promote fairness is hence an important challenge. Third, while common models such as deep neural networks can reach high predictive performance, their underlying logics and representation are often too complex or hidden, preventing users to fully understand their decisions. This raises significant concerns, regarding their auditability, certifiability and trust, thus calling for the need to explain their predictions.

These three topics, namely privacy, fairness, and interpretability, have been extensively studied during the last decade [Cristofaro 2020, Barocas *et al.* 2019,

[1] https://gdpr-info.eu/

[2] https://artificialintelligenceact.eu/

Guidotti *et al.* 2018] with an emphasis on how they trade-off with utility. However, while they are often considered in isolation, it is necessary to enforce them all *simultaneously*. Characterizing their mutual interplays is hence an important research avenue, which has attracted some attention in the last years. Indeed, these concerns often conflict [Datta *et al.* 2023], and compromises between them, as well as with utility, generally have to be done.

In this thesis, we investigate the pairwise interactions between fairness, privacy and interpretability. They intuitively correspond to the three edges of the graph represented in Figure 1. This graph will later be used at the beginning of each chapter to visually position its content. More precisely, our contributions are twofold. First, we survey the literature on the different compatibilities, synergies and tensions identified between each pair of desiderata. Second, for each such pair, we focus on one of the observed tensions and either propose a mitigation mechanism or a technique to highlight or quantify the conflicting relationship.

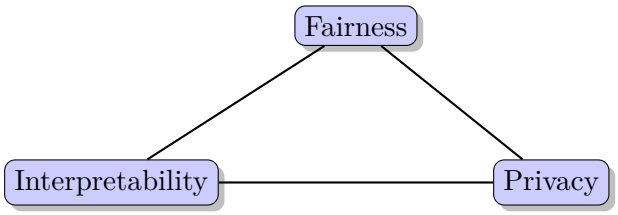



Figure 1: Graph representation of three identified dimensions of trustworthy machine learning. The objective of the thesis is to study their pairwise mutual interplays, which correspond to the edges of this graph.

## Outline of the Thesis

The manuscript includes four chapters, described in more details hereafter. The first one provides the necessary background while the following three detail our contributions. More precisely, each of the three contribution chapters focuses on the interplays between two of the three identified areas of trustworthy machine learning, namely fairness, interpretability and privacy. They are all organized in the same way: first we survey the literature and summarize the different compatibilities, synergies and tensions identified between the two concerned notions. Afterwards, we focus on one of these identified aspects before presenting a novel contribution whose objective is to highlight or address a tension. In the Appendix A, we further propose a graphical summary of our literature review on the identified interplays between fairness, interpretability and privacy, depicted in the first parts of Chapters 2, 3 and 4.

**Chapter 1** introduces the necessary background regarding the considered machine learning setup, useful tools from combinatorial optimization as well as the three identified issues for trustworthiness: fairness, interpretability and privacy. In addition, we perform two focused literature reviews that will be useful later in our three contribution chapters. More precisely, we survey the literature on improving fairness generalization and on performing reconstruction attacks against machine learning models. We also briefly describe two other contributions that will not be discussed further in the manuscript. More precisely, the first contribution considers the learning of optimal interpretable models under fairness constraints for multi-class classification tasks [Rouzot *et al.* 2022]. The second contribution is about hybrid interpretable models, which are machine learning models composed of both an interpretable and a black-box components [Ferry *et al.* 2023d].

**Chapter 2** studies the interactions between fairness and interpretability in machine learning. In this chapter, we investigate the inherent difficulty of learning optimal interpretable models under fairness constraints. More precisely, we build on a learning algorithm we introduced in early works [Aïvodji *et al.* 2019b, Aïvodji *et al.* 2021c] and propose an integer linear programming based pruning technique to enhance the learning of fair rule lists [Aïvodji *et al.* 2022]. By jointly considering fairness, accuracy and sparsity, this method enables the learning of optimal fair rule lists, effectively solving a conflict between interpretability and fairness desiderata. Motivated by our empirical findings that fairness often does not generalize well, we propose a new framework to quantify or improve fairness robustness [Ferry *et al.* 2023b]. This framework includes an exact approach based on integer programming as well as a heuristic one that is more efficient and scalable.

**Chapter 3** considers the interplays between fairness and privacy in machine learning. While most of the literature studies the connections between statistical fairness notions and differential privacy, we take a different direction and illustrate an intrinsic conflict between enforcing fairness with respect to some sensitive attributes and protecting such attributes' privacy. More precisely, we show that information regarding a model's fairness (either publicly known or easily inferred) can be leveraged to improve any baseline adversary reconstruction of the model's training set sensitive attributes. The proposed reconstruction correction process [Ferry *et al.* 2023a] is implemented with either an integer linear programming model or a constraint programming one. It is empirically shown effective in exploiting the fairness information to improve a baseline reconstruction.

**Chapter 4** focuses on the connections between interpretability and privacy in machine learning. While many works showed that post-hoc explainability frameworks can be leveraged to infer private information regarding a model's training data, we rather focus on interpretability by design. More precisely, we illustrate and theoretically quantify an intrinsic conflict between learning and releasing an

interpretable model on the one side and protecting the privacy of its training set on the other side [Ferry *et al.* 2023c]. We leverage tools from information theory to precisely measure the amount of information an interpretable model inherently encodes, via its structure, about its training data.

## Publications

Our contributions on investigating the connections between interpretability, fairness and privacy in machine learning led to several international publications, corresponding to the works depicted within Chapters 2, 3 and 4:

- Ulrich Aïvodji, Julien Ferry[3], Sébastien Gambs, Marie-José Huguet and Mohamed Siala - *FairCORELS, an Open-Source Library for Learning Fair Rule Lists* - ACM International Conference on Information and Knowledge Management (**CIKM 2021**), November 1-5, 2021.

- Ulrich Aïvodji, Julien Ferry[3], Sébastien Gambs, Marie-José Huguet and Mohamed Siala - *Leveraging Integer Linear Programming to Learn Optimal Fair Rule Lists* - International Conference on the Integration of Constraint Programming, Artificial Intelligence, and Operations Research (**CPAIOR 2022**), June 20-23, 2022.

- Julien Ferry, Ulrich Aïvodji, Sébastien Gambs, Marie-José Huguet and Mohamed Siala - *Improving fairness generalization through a sample-robust optimization method. Machine Learning* - **Machine Learning** journal, 2022. Also presented at the AAAI Conference on Artificial Intelligence (**AAAI 2023**) within the Journal Track and the Bridge on Constraint Programming and Machine Learning.

- Julien Ferry, Ulrich Aïvodji, Sébastien Gambs, Marie-José Huguet and Mohamed Siala - *Exploiting Fairness to Enhance Sensitive Attributes Reconstruction* - International Conference on Secure and Trustwor- thy Machine Learning (**SATML 2023**), February 8-10, 2023.

- Julien Ferry, Ulrich Aïvodji, Sébastien Gambs, Marie-José Huguet and Mohamed Siala - *Probabilistic Dataset Reconstruction From Interpretable Models* - **Preprint** (submitted).

- Julien Ferry, Ulrich Aïvodji, Sébastien Gambs, Marie-José Huguet and Mohamed Siala - *SoK: Taming the Triangle - On the Interplays between Fairness, Interpretability and Privacy in Machine Learning* - **Preprint** (submitted).

[3]First author.

Related to the topics of the thesis, other international publications were also realized, which are briefly summarized in Chapter 1, respectively in Sections 1.3.5 and 1.4.5:

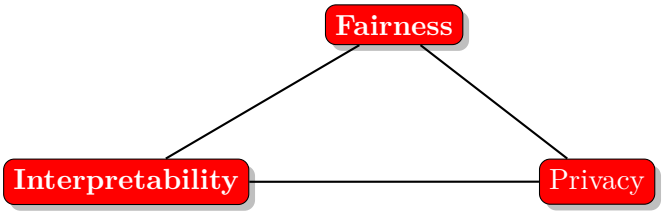
- Julien Rouzot, Julien Ferry and Marie-José Huguet - *Learning Optimal Fair Scoring Systems for Multi-Class Classification* - International Conference on Tools with Artificial Intelligence (**ICTAI 2022**), October 31-November 2, 2022.
- Julien Ferry, Gabriel Laberge and Ulrich Aïvodji - *Learning Hybrid Interpretable Models: Theory, Taxonomy, and Methods* - **Preprint** (submitted).

National communications were also presented at the ROADEF (the annual congress of the French society of operational research and decision support) and RJCIA (Meeting of Young Researchers in Artificial Intelligence, within the French Artificial Intelligence Platform, PFIA) conferences:

- **ROADEF 2021** and **RJCIA 2021**: French versions of (part of) our Machine Learning journal article.
- **ROADEF 2022**: French version of our CPAIOR 2022 article.
- **ROADEF 2023** and **RJCIA 2023**: French versions of our SATML 2023 article.

Chapter 1

# Background & Trustworthy Machine Learning




In this background chapter, we introduce key notions regarding three pillars of trustworthy machine learning, namely fairness, interpretability and privacy. While the objective of the thesis is to study their interactions, we first discuss each of them in isolation. More precisely, the objective of this chapter is not to perform an exhaustive review on each of these topics, but rather to provide a brief overview of the existing techniques and challenges. Furthermore, we overview concepts and tools from combinatorial optimization and operational research that we leverage in our works. We additionally perform some *focuses* on specific aspects of the literature that are useful in the following chapters (Sections 1.3.6 and 1.5.3). We also briefly highlight some additional *contributions* that are not detailed further in the remainder of the manuscript (Sections 1.3.5 and 1.4.5).

## Contents

**Outline of the Chapter.** First, in Section 1.1, we describe the considered supervised machine learning setup, along with the associated notations. We motivate the need for trustworthy machine learning, and highlight three main ethical issues: fairness, interpretability and privacy. Afterwards, we introduce in Section 1.2 tools from combinatorial optimization, which we will later use in our contributions. We then provide key notions regarding the three aforementioned ethical pillars in respectively Sections 1.3, 1.4 and 1.5. We also briefly summarize other *contributions* that are not detailed further in the manuscript in Sections 1.3.5 and 1.4.5. More precisely, we discuss fairness for multi-class classification and our proposed framework for learning optimal scoring systems in this setting in Section 1.3.5 before presenting a method for learning (optimal) hybrid interpretable models in Section 1.4.5. Finally, we conduct two focused literature reviews in Sections 1.3.6 and 1.5.3, regarding respectively, fairness generalization and reconstruction attacks.

## 1.1 Trustworthy Supervised Machine Learning

In this section, we first introduce the machine learning notions and notations that will be used throughout the manuscript. We then motivate the need to ensure trustworthiness in machine learning through historical use cases and applications.

### 1.1.1 Classification & Notations

Traditional taxonomies of machine learning methods usually distinguish three paradigms: reinforcement learning, unsupervised learning and supervised learning [Russell & Norvig 2020]. In the former, an autonomous agent interacts with its environment and takes actions for which it can get rewards. It ultimately learns a desired behavior, characterized by its actions, which is called a policy. Unsupervised learning approaches consist in learning patterns from unlabeled data, for instance by clustering similar sets of elements or by reducing the dimensionality of the data. Hereafter, we focus on *supervised learning* tasks that aim at learning to predict a given value from a set of provided attributes. Throughout the manuscript, we will consider the case in which the target value is discrete. In such case, the associated task is coined *classification*.

Formally, let $M$ be a number of *non-sensitive attributes* (*i.e.*, attributes that can legitimately be used for decision-making) characterizing an example (*e.g.*, an individual). For $m \in \{1..M\}$, $\mathcal{X}_m$ denotes the domain of possible values for attribute $m$, which can be either categorical or numerical, and $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \ldots \times \mathcal{X}_M$. Similarly, let $\mathcal{S}$ be the domain of a (categorical) *sensitive attribute*. Such sensitive attribute corresponds to personal information such as age, gender or race [Ding *et al.* 2021], which should not be used for a decision-making process due to legal, ethical, social or philosophical reasons [Barocas *et al.* 2019]. Finally, let $\mathcal{Y}$ be the domain of a *label*. For instance, for the aforementioned recidivism prediction task, we could have: $\mathcal{Y} = \{yes, no\}$, with the two classes indicating whether the offender actually committed a recidivism or not.

$\mathcal{D} = (X, S, Y)$ is a dataset drawn from the true (unknown) distribution over $\mathcal{X} \times \mathcal{S} \times \mathcal{Y}$. Let $N$ be the number of *examples* (*i.e.*, datapoints) in $\mathcal{D}$, with $e_{j \in \{1..N\}} = (x_j, s_j, y_j) \in \mathcal{X} \times \mathcal{S} \times \mathcal{Y}$. The objective of a supervised machine learning algorithm is to learn a *classifier* $\mathcal{L}(\mathcal{D}) = h$ mapping the attributes space to the label space. The explicit use of a sensitive attribute is usually prohibited by law to avoid *disparate treatment* [Barocas & Selbst 2016].

Thus, we assume that the sensitive attribute is not used for inference, which means that $h : \mathcal{X} \mapsto \mathcal{Y}$, with $\hat{Y} = h(X)$ being the predictions of the machine learning model. For the particular case of binary classification, we have: $\mathcal{Y} = \{0, 1\}$. The classifier $h$ belongs to some hypothesis space $\mathcal{H}$, which constitutes the range of the learning algorithm $\mathcal{L}$. More precisely, the hypothesis space corresponds to the set of candidate models, which can be for instance, the set of possible decision trees or rule lists.

For a specific training dataset $\mathcal{D}$ drawn from some distribution $\mathcal{P}$, the desired model $h$ is the solution to the following problem, in which $\mathsf{obj}(h, \mathcal{P})$ is the expected objective function under distribution $\mathcal{P}$:

$$\underset{h \in \mathcal{H}}{\arg\min} \quad \mathsf{obj}(h, \mathcal{P}) \tag{1.1}$$

In practice, the true underlying distribution $\mathcal{P}$ is often unknown, and we only

get a limited number of observations from it that forms the dataset $\mathcal{D}$. The optimal solution to Problem (1.1) is commonly approximated via Empirical Risk Minimization. More precisely, the objective of a supervised learning algorithm $\mathcal{L}$ is to learn a model $h = \mathcal{L}(\mathcal{D})$ solution to Problem (1.2).

$$\underset{h \in \mathcal{H}}{\arg\min} \quad \mathsf{obj}(h, \mathcal{D}) \tag{1.2}$$

In this manuscript, as the examples in $\mathcal{D}$ will usually correspond to individuals and the predictions $\hat{Y} = h(X)$ of the classifier may be used in high-stakes decision-making (*e.g.*, college admissions, credit attribution ...), it is crucial to ensure the trustworthiness of machine learning techniques. This is discussed in more details in the next subsection.

### 1.1.2 High-Stakes Applications and the Need for Trustworthy ML

The use of machine learning (ML) techniques for real-world high-stakes decision making systems motivates the need to ensure trustworthiness of these approaches. Hereafter, we illustrate these desiderata through some popular use cases.

A famous example concerns recidivism prediction. More precisely, it was demonstrated that the Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) tool led to discriminating black people by consistently predicting higher recidivism risk for them [Angwin *et al.* 2016]. In this particular use case, this bias led to higher false positive rates for black people than for other demographic groups. In other terms, black people were incarcerated unnecessarily more often, which illustrates the need to define some form of *fairness* and to ensure that the models' decisions satisfy it.

Another popular situation is when a person applies for a bank loan. Then, the bank may use credit scoring methods to accept or deny the credit. In case of a denial, the applicant may ask for an explanation, to understand the decision but also possibly to adapt its personal situation to modify the decision. This requirement for an explanation is both legitimate and legally stated: *explainability/interpretability* is another key requirement for trustworthiness in machine learning.

Finally, machine learning systems are increasingly used for medical data analysis, thanks to their ability to handle large amounts of complex data. For instance, an hospital could learn a machine learning model to detect several rare diseases. The model, if deployed and accessible through a prediction API, could then help other hospitals for their diagnosis. However, its prediction API could also be leveraged by an adversary to perform inference attacks, for instance to infer whether a given profile was part of the model's training data. This could indicate that the person is at a higher risk of having a rare disease, and health insurance companies could leverage this information to increase their fees for this particular person. Thus, it is crucial to preserve the *privacy* of the data used in machine learning.

These three main pillars, namely fairness, interpretability and privacy, have

been identified and extensively studied in recent years [Datta *et al.* 2023]. Several useful tools can be leveraged to address them, which are sometimes implemented in the form of public libraries. For instance, `Fairlearn`[1] [Bird *et al.* 2020] contains fairness-enhancing methods intervening at the different stages of the learning pipeline. It was later incorporated into `AI Fairness 360`[2] [Bellamy *et al.* 2019], an open toolkit for analyzing and mitigating biases. Several methods were also proposed to enforce explainability and interpretability in machine learning. For instance, the `iNNvestigate`[3] library [Alber *et al.* 2019] gathers tools for computing post-hoc explanations of neural networks while the `imodels`[4] library proposes several exact or approximate algorithms to learn different types of interpretable models. In addition, several libraries provide insightful frameworks for privacy protection. Among others, `Google Differential Privacy`[5] implements several differential privacy mechanisms and `Diffprivlib`[6] provides useful building blocks for differentially-private machine learning. Other works focus on attacks against privacy, such as the `PrivacyRaven`[7] repository, which aggregates many popular inference attacks frameworks.

In the next section, we introduce key notions and frameworks from combinatorial optimization, which will be used as a toolbox in our contributions. We then cover the three identified pillars of trustworthy machine learning in Sections 1.3, 1.4 and 1.5.

## 1.2 Combinatorial Optimization

Our objective is to leverage tools from the combinatorial optimization and operational research fields to study the interactions between interpretability, fairness and privacy in machine learning. Consequently, the purpose of this section is to provide a sufficient overview of such tools *from a user perspective*. First, we introduce the key ideas underlying combinatorial optimization. We then describe the principle of tree search, a popular way of representing and exploring a combinatorial search space. Afterwards, we depict several declarative programming approaches before defining key notions regarding multi-objective optimization.

### 1.2.1 General Principle

The aim of combinatorial optimization is to explore a finite space of elements and find the one(s) optimizing a given criterion. This criterion can be quantified using

[1] `https://fairlearn.org/`
[2] `https://ai-fairness-360.org/`
[3] `https://github.com/albermax/innvestigate`
[4] `https://github.com/csinva/imodels`
[5] `https://github.com/google/differential-privacy`
[6] `https://github.com/IBM/differential-privacy-library`
[7] `https://github.com/trailofbits/PrivacyRaven`

an objective function obj. The desired solution hence minimizes (or maximizes) obj while also satisfying eventual constraints.

Combinatorial optimization problems can be solved using either exact or approximate approaches. The former have the ability to find the optimal solution while also proving that no better solution can exist. In the remainder of this section, we will focus on this category. The later can be used to find good solutions within a predefined time frame, but cannot guarantee that the returned solution is optimal. They include, for instance, greedy algorithms (which iteratively make local choices) and local search techniques (which generate new solutions by perturbating former ones).

Popular interpretable models, such as decision trees or rule lists, inherently have a combinatorial structure, and training them corresponds to solving a combinatorial optimization problem (with accuracy often being the optimized objective). For instance, greedy algorithms such as `CART` [Breiman *et al.* 1984] were originally proposed to learn decision trees. However, they do not provide optimality guarantee and usually produce sub-optimal trees. Optimal methods were later proposed. For instance, `GOSDT` [Lin *et al.* 2020] is a branch-and-bound algorithm (using dynamic programming) to build optimal sparse decision trees. The philosophy underlying such tree search techniques is introduced in the next subsection. Declarative programming approaches can also be leveraged to learn interpretable models. More precisely, they encode a problem within a given syntax before letting a solver find a solution. For instance, [Aghaei *et al.* 2019] and [Verhaeghe *et al.* 2020] respectively use integer programming and constraint programming to learn optimal decision trees. Such declarative programming approaches are further detailed in Section 1.2.3.

### 1.2.2 Tree Search

A popular way of representing a combinatorial search space is to encode it using a tree structure, in which each node corresponds to a decision (*e.g.*, fixing the value of a variable). Its children nodes then define different sub-problems with the entire tree encoding the complete set of possible solutions. Because, it usually has an exponential size with respect to the decision variables' cardinalities, it is often impossible to explore it explicitly. Rather, common tree search techniques (coined *branch-and-bound*) compute lower and upper bounds to *prune* parts of the tree when it is possible to certify that they do not contain any better solution. Symmetry-breaking mechanisms can also be leveraged to reduce the size of the search space, for example when different branches represent different permutations of the same decisions, if order does not matter.

Furthermore, the order in which the tree is explored can highly influence the duration of the process and several search heuristics exist. Such heuristics consist in ordering the nodes belonging to the exploration frontier (*i.e.*, the next nodes to be expanded) following some criteria. Popular strategies include Breadth-First Search (BFS), in which the shallowest nodes are explored first, Depth-First Search (DFS),

in which the deepest nodes are explored first, and best-first search, which order the nodes according to some objective function (or bounds on this value). However, the search heuristic used does not affect the resulting objective function value if the method is exact and the entire search space is explored (possibly implicitly through pruning).

In Section 2.2.2, we introduce a branch-and-bound algorithm of the literature for learning optimal sparse rule lists. We describe its fairness-aware adaptation that we proposed in early work in Section 2.2.3, before later showing how to improve its exploration of the search space. The search space of this algorithm is represented using a prefix tree, as illustrated in Figure 2.1.

### 1.2.3 Declarative Programming

While ad-hoc algorithms can be designed for particular combinatorial optimization problems, declarative programming approaches aim at separating the expression of the problem and its resolution. More precisely, a given problem must first be encoded within a given syntax. The resulting problem can then be solved using off-the-shelf general purpose solvers.

**Mixed-Integer Linear Programming.** A Mixed-Integer Linear Program (MILP) is defined by a number of variables, some of them taking values within a finite (discrete) domain and some of them being continuous (*i.e.*, taking values in $\mathbb{R}$). The goal is to find an assignment of the variables that minimizes a given objective function $\mathsf{obj}$ defined on these variables while also satisfying constraints stated as inequalities. Importantly, the objective needs to be a linear combination of the decision variables, as well as the left and right terms of the inequalities.

In practice, MILPs can be solved using off-the-shelf solvers, such as `CPLEX`[8]. While this task is NP-hard in general, state-of-the-art solvers leverage several computational tricks and are able to handle large problems. In a nutshell, these solvers often use branch-and-bound techniques to deal with discrete variables. Furthermore, they iteratively solve (easier) linear relaxations of the MILP (*i.e.*, allowing some discrete variables to take a continuous value) to efficiently get objective bounds and prune the search tree.

We refer to Integer Linear Programming (ILP) when all variables are integer. In Section 2.3.1, we introduce an ILP that we later use to prune the search space of an interpretable and fair machine learning algorithm. Furthermore, we also propose a general ILP technique to perform sensitive attributes reconstruction correction in Section 3.2.2. Finally, when some of the constraints are not linear, we will simply refer to the resulting problem as Integer Programming (IP). For instance, this is the case of our IP used to quantify fairness sample-robustness, introduced in Section 2.5.3 and provided in details in Appendix C.

[8] https://www.ibm.com/products/ilog-cplex-optimization-studio

**Constraint Programming.** A constraint programming (CP) model is defined by a set of variables each taking value within a given (discrete) domain, a set of constraints on these variables, and (eventually) an objective function `obj`. The objective is to find an assignment of the variables which satisfies all the constraints while minimizing `obj`. Contrary to MILP formulations, the constraints (and the objective) need not be linear, and advanced expressions can be encoded. Furthermore, some of them are called *global constraints*. They correspond to a union of simple constraints that facilitate modeling while speeding up the resolution.

In practice, CP models can be solved using off-the-shelf solvers, such as the `OR-Tools` CP-SAT solver[9][Perron & Furnon 2019]. These solvers also leverage tree search to explore the space of solutions. In a nutshell at each node, a decision is made to reduce the domain of a variable. This decision is propagated to the other variables involved in the constraints, to *filter* their domains. A feasible solution is reached when the domain of each variable contains exactly one single value. If previously made decisions make the problem infeasible (*i.e.*, the domain of a variable becomes empty), then the algorithm *backtracks* and goes up in the tree. Again, while this task is theoretically NP-hard, state-of-the-art solvers are able in practice to handle large scale problems.

As aforementioned, CP allows the encoding of arbitrary expressions. For instance in Section 3.2.3, we leverage the `element` constraint to formulate a CP model equivalent to the aforementioned ILP for reconstruction correction, but with polynomial (*vs.* exponential) search space (*w.r.t.* the number of examples). Such constraints are used to access a data array $T$ at index given by the value of a variable $z$: $T[z] = \texttt{element}(T, z)$. Linearizing such constraint (*i.e.*, using a set of linear constraints and/or variables that produce equivalent behavior) is possible but would introduce a prohibitive number of linear constraints.

While we restrict our attention to MILP and CP which are both later used in this manuscript, one can note that other declarative programming approaches exist. For instance, maximum satisfiability (MaxSAT) aims at finding an assignment of given Boolean variables maximizing a number of satisfied clauses. In the literature, MaxSAT was used to learn different types of interpretable models, including optimal binary decision diagrams [Hu *et al.* 2022a] or decision trees [Hu *et al.* 2020].

Finally, several dedicated or general-purpose exact methods can be used to solve combinatorial optimization problems. When several objectives are considered, one should find a way to optimize them jointly and compute trade-offs between them as detailed in the next subsection.

### 1.2.4 Multi-objective Optimization

In many optimization problems, several criteria can be used to evaluate the quality of the solution. For instance, in the context of machine learning, one can evaluate the produced models in terms of accuracy, sparsity (*i.e.*, size), fairness, privacy, ... In such *multi-objective* problems, one can leverage the notion of *Pareto dominance*

[9] https://github.com/google/or-tools

to build sets of *non-dominated solutions*, called *Pareto frontiers*. More precisely, a solution is *non-dominated* (and belongs to the Pareto frontier) if there does not exist a solution at least as good as it on all considered objectives, and strictly better in at least one.

Several methods can be used to build sets of trade-offs between the different objectives. A trivial solution consists in optimizing a weighted sum of the different criteria. One can then vary the values of the coefficients of the different objectives to obtain different trade-offs. Furthermore, if the decision-maker can rank the different objectives in terms of priority (*i.e.*, any improvement on one objective is preferred to any improvement on another one), then the different objectives can be optimized sequentially, which is coined as lexicographic optimization. Finally, another possibility is to optimize a single objective, and integrate the other ones through constraints. In such *$\varepsilon$-constrained* [Haimes 1971] strategies, varying the values of the coefficients within the different constraints allows the building of approximations of the Pareto frontier. For instance, we use this method in the formulation of our fair learning problem (1.3). In this context, the two optimized values are the learning objective $\mathsf{obj}(h, \mathcal{D})$ (for instance, accuracy) and the unfairness violation $\mathsf{unf}(h, \mathcal{D})$. We directly optimize the former and integrate the later through a constraint. Varying the value of the unfairness tolerance $\varepsilon$ then allows to build the sets of trade-offs between accuracy and fairness, as described in Section 2.2.3 and shown in Figure 2.2.

## 1.3 Fairness

The objective of this section is to provide a brief overview of the fairness literature. More precisely, in Section 1.3.1, we first review the possible causes of bias in machine learning and their origins. In Section 1.3.2, we then present the most popular notions of fairness as well as their associated metrics. Afterwards in Section 1.3.3, we provide a taxonomy of the methods that can be used to enforce fairness. We also discuss some challenges regarding the compatibility and the applicability of existing fairness notions in Section 1.3.4. Motivated by the observation that fairness has almost only been studied in the context of binary classification, we discuss its application to the more general multi-class classification setup in Section 1.3.5. We briefly introduce our contribution on learning optimal interpretable fair models for multi-class classification leveraging Mixed Integer Linear Programming techniques. Finally in Section 1.3.6, we survey the literature on the methods proposed to improve fairness generalization. Indeed, it was identified as a major challenge in fair learning [Cotter *et al.* 2019], which motivated the formulation of our new fairness robustness framework, introduced at the end of Chapter 2.

### 1.3.1 Bias in Machine Learning

To learn a classifier $h \in \mathcal{H}$, a learning algorithm $\mathcal{L}$ identifies correlations in the training data $\mathcal{D}$ that allow to predict the target label $y$ from the provided fea-

tures. However, datasets are finite and incomplete representations of the real-world and commonly contain incorrect correlations. Such correlations can correspond to discrimination(s), and are often referred to as *biases*. Many sources of bias have been identified in the literature [Mehrabi *et al.* 2022] and we discuss some of them hereafter.

On the one side, *dataset bias* [Tommasi *et al.* 2017, Torralba & Efros 2011] has been studied extensively, and can emerge for several reasons. An example is *capture bias*, which refers to the data acquisition process. For instance, two images could be distinguished based on the type of camera used to take them rather than on their actual content (due to a difference in resolution, in exposition...). *Sampling bias* and *representation bias* occur when the dataset distribution does not match the real-world distribution due to a non-representative sampling. The process of choosing which features to report and how to measure them can lead to *omitted variable bias* or *measurement bias*. Even if the data collection was totally unbiased, there can still be biases in the real-world that would be reflected in the data. Such *historical bias* corresponds to historical discrimination and is the main source of data bias targeted in our work.

On the other side, the learning algorithm can further introduce biases, for example by focusing on the majority group and neglecting minorities for average accuracy purposes. Such *algorithm bias* is attributed to the design of the algorithm itself. The model may also be optimized using the wrong metrics or benchmarks, which is deemed as an *evaluation bias*.

Several legal texts are relevant to the concerns of fairness for decision making. For instance, one can refer to the U.S. Equal Employment Opportunity Commission and Title VII of the Civil Rights Act of 1964[10] [Barocas & Selbst 2016] or the European Union General Data Protection Regulation[11] [Voigt & Von dem Bussche 2017, Malgieri 2020] or AI Act[12]. Then, two main types of discrimination can be identified with respect to the predictions of a machine learning model [Zafar *et al.* 2017, Kilbertus *et al.* 2018, Aghaei *et al.* 2019, Hajian & Domingo-Ferrer 2013].
*Disparate treatment* (also called *direct discrimination*) consists in treating individuals differently based (explicitly) on sensitive characteristics[13]. Such discrimination can be avoided by preventing the use of sensitive attributes for inference. However, because sensitive attributes can often be accurately predicted based on the non-sensitive ones, avoiding disparate treatment is necessary but often not sufficient [Ekstrand *et al.* 2018]. Disparate impact (also called *indirect discrimination*) refers to practices that do not explicitly use sensitive features for decision making but result in disproportionately advantaging or disadvantaging groups with particular sensitive attribute settings. As observed in [Kilbertus *et al.* 2018], there is an

[10] https://www.eeoc.gov/statutes/title-vii-civil-rights-act-1964
[11] https://gdpr-info.eu/
[12] https://artificialintelligenceact.eu/
[13] This definition is broadly used in the ML community. Nevertheless, in law, it was originally proposed for human decision-makers (and not machine learning models' predictions), and the notion of intent to discriminate is key - but hard to define for an algorithm [Xiang & Raji 2019]. This legal aspect is however out of the scope of this manuscript.

intrinsic conflict between the goals of avoiding disparate treatment (*e.g.*, not explicitly using sensitive attributes for decision making) and avoiding disparate impact (which often requires the use of the sensitive attributes for training a fair classifier).

As we have seen, there are multiple sources of bias in machine learning systems. Because such bias can harm particular individuals or subgroups of the population, ensuring a fair decision making is legally and ethically required. Several approaches pursuing this goal have been proposed, as summarized in the next subsection.

Finally, while we focus on fairness for learning machine learning models, it is worth noting that a line of works targeting *anti-discrimination* in data mining preceded the literature in this area, considering very close notions and metrics [Pedreschi *et al.* 2008, Hajian & Domingo-Ferrer 2013]. More precisely, we restrict our attention to supervised learning tasks as defined in Section 1.1, even though fairness was considered in different machine learning setups, such as clustering [Chierichetti *et al.* 2017, Chhabra *et al.* 2021].

### 1.3.2 Notions of Fairness

Different approaches to fairness have been proposed in the literature, which can be grouped into three main categories [Verma & Rubin 2018]. We detail each of them hereafter, with a particular focus on statistical fairness notions, which are used in our contributions of Chapters 2 and 3.

**Statistical Fairness.** The rationale of *statistical fairness*, also coined *group fairness*, is to ensure that a given statistical measure has similar values between several *protected groups*, defined by the value(s) of some sensitive feature(s) in $\mathcal{S}$. The underlying principle is that such sensitive features (*e.g.*, race, gender, ...) should not influence predictions. The exact formulation of such metrics ensures that the probabilistic difference for the given measure over the protected groups on the entire data distribution is no greater than a given *unfairness tolerance* $\varepsilon$. A common relaxation consists in bounding the empirical difference measured on the training set. Depending on the particular value being equalized across groups, several metrics have been proposed in the literature. Hereafter, we consider four commonly used metrics: Statistical Parity [Dwork *et al.* 2012] (SP), Predictive Equality [Chouldechova 2017] (PE), Equal Opportunity [Hardt *et al.* 2016] (EOpp) and Equalized Odds [Hardt *et al.* 2016] (EO). We summarize them in Table 1.1, while indicating for each metric which values of the confusion matrix of the learnt classifier they try to match across the different protected groups. For instance, the Equal Opportunity metric ensures that the true positive rate of each protected group ($\forall s \in \mathcal{S}$) is no further than some tolerance $\varepsilon$ from that of the other groups. On the one hand, predictive equality, equal opportunity and equalized odds are aligned with accuracy. In particular, a perfectly accurate model will also be perfectly fair for such *bias preserving* [Wachter *et al.* 2020] metrics. They indeed target *algorithm bias*, but if the dataset is biased, this bias can still be reflected in the model's predictions. On the other hand, statistical parity is a *bias transforming*

metric. It does not look at the true labels and addresses *dataset bias*. Thus, the more the data is biased, the more this metric conflicts with accuracy.

Table 1.1: Summary of the considered statistical fairness metrics, along with the related statistical measures to be equalized across protected groups.

| **Metric** | **Equalized statistical measure** |
|---|---|
| Statistical parity (SP) | Probability of being assigned the positive class |
| Predictive equality (PE) | False Positive rate |
| Equal opportunity (EOpp) | True Positive rate |
| Equalized odds (EO) | False Positive rate and True Positive rate |

Two main implementations of these fairness metrics coexist in the literature. We report in Table 1.2 their associated expressions. The first one, that we coin the *one-vs-one* formulation, bounds the difference of the statistical measure (*e.g.*, true positive rates) between each pair of protected groups. While being convenient when dealing with two protected groups, this approach requires a number of constraints quadratic with respect to the number of protected groups. In Chapter 2, we restrict our attention to the binary protected group case and focus on such notions. The second one, that we call the *one-vs-all* formulation, bounds the difference between each group and the overall dataset. This notion is more convenient when dealing with more than two protected groups. We use it in Chapter 3. Finally, one can observe that the two notions imply each other, with carefully chosen values of the unfairness tolerance $\varepsilon$.

Table 1.2: Summary of the considered statistical fairness metrics, along with the associated constraints expressions, using either the *one-vs-one* (pairwise) or the *one-vs-all* formulations.

| **Metric** | **Constraint Expression - *one-vs-one* (pairwise) formulation** |
|---|---|
| SP | $\forall s,\ \forall s',\ \lvert\mathbb{P}(\hat{y}=1 \mid s) - \mathbb{P}(\hat{y}=1 \mid s')\rvert \leq \varepsilon$ |
| PE | $\forall s,\ \forall s',\ \lvert\mathbb{P}(\hat{y}=1 \mid s,\ y=0) - \mathbb{P}(\hat{y}=1 \mid s',\ y=0)\rvert \leq \varepsilon$ |
| EOpp | $\forall s,\ \forall s',\ \lvert\mathbb{P}(\hat{y}=1 \mid s,\ y=1) - \mathbb{P}(\hat{y}=1 \mid s',\ y=1)\rvert \leq \varepsilon$ |
| EO | Conjunction of PE and EOpp |
| **Metric** | **Constraint Expression - *one-vs-all* formulation** |
| SP | $\forall s,\ \lvert\mathbb{P}(\hat{y}=1) - \mathbb{P}(\hat{y}=1 \mid s)\rvert \leq \varepsilon$ |
| PE | $\forall s,\ \lvert\mathbb{P}(\hat{y}=1 \mid y=0) - \mathbb{P}(\hat{y}=1 \mid s,\ y=0)\rvert \leq \varepsilon$ |
| EOpp | $\forall s,\ \lvert\mathbb{P}(\hat{y}=1 \mid y=1) - \mathbb{P}(\hat{y}=1 \mid s,\ y=1)\rvert \leq \varepsilon$ |
| EO | Conjunction of PE and EOpp |

We let $\mathsf{unf}(h, \mathcal{D})$ be the empirical unfairness violation of classifier $h$ on dataset $\mathcal{D}$, measured using one of the aforementioned metrics. This value is precisely the one that we want to be no greater than the unfairness tolerance $\varepsilon$. The fair learning

problem can then be formulated as a constrained optimization problem:

$$\begin{aligned} \arg\min_{h \in \mathcal{H}} \quad & \mathsf{obj}(h, \mathcal{D}) \\ \text{s.t.} \quad & \mathsf{unf}(h, \mathcal{D}) \leq \varepsilon. \end{aligned} \tag{1.3}$$

**Individual Fairness.** Individual fairness notions build on the idea that similar individuals should be treated similarly. It was first introduced in [Dwork *et al.* 2012], as the authors observe that statistical fairness may be desirable but can lead to decisions appearing unfair at the level of single individuals (for example majority group applicants being rejected while less qualified applicants from minority group are accepted). Furthermore, fairness through unawareness, which simply consists in not using the sensitive attributes for decision making, was shown ineffective due to the presence of proxy features (*i.e.*, non-sensitive attributes that are correlated with the sensitive one). Authors then define fairness through awareness as a Lipschitz condition on the classifier, given a distance metric that defines the similarity between the individuals with respect to a particular task. The main difficulty lies in the definition of the distance function, which is related to the context and should be designed carefully by experts and policymakers, depending on the task at hand. Some approaches relax the need to explicitly define such metric. For instance, [Zemel *et al.* 2013] learn a mapping from individuals to a set of clusters, each represented by "prototypes". Such prototypes are then used in the decision-making process, which ensures that all individuals belonging to the same cluster are treated similarly. [Lahoti *et al.* 2019] also propose to build an individually fair representation of the data by performing a clustering-based mapping from individuals to prototypes. However, they use a different objective function, emphasizing that the resulting representation is task-agnostic. Considering that the distance metric can not be defined manually not learnt automatically without annotations, [Jung *et al.* 2021] first ask a panel of stakeholders to indicate for a given set of pairs of individuals whether their outcomes should be similar, ordered or unconstrained. The resulting pairwise constraints are then added to the learning problem. In a previous work, [Ilvento 2020] propose to learn a distance metric leveraging a reduced number of answers from a domain expert. The approach of [Joseph *et al.* 2016] takes a different direction as it does not rely explicitly nor implicitly on a similarity metric. It rather enforces that *a better applicant is never disadvantaged over a worse one*. More precisely, the method ensures that the probability of an individual being assigned to a positive outcome reflects the true probability of this event.

**Causal Fairness.** Causal fairness approaches analyze the causal relationships between the sensitive features, the non-sensitive ones and the target decision, leveraging causal graphs [Kilbertus *et al.* 2017]. In particular, they often look for a (direct or indirect) path from a sensitive attribute to the prediction, which indicates unfair decision making. For instance, [Nabi & Shpitser 2018] distinguish between *legitimate* and *illegitimate* paths in a causal graph, highlighting that the later should

not be found in the causal graph built from a fair model. A framework using causal Bayesian networks to model correlations contained in a given dataset is proposed in [Chiappa & Isaac 2018]. This network can then be leveraged to detect unfairness, defined as the *presence of an unfair causal path*. [Zhang *et al.* 2018b] use two causal graphs. A first one is human-defined, and identifies the correct semantic relationships that should exist between the different non-sensitive attributes, the sensitive ones and the label. The second one is built using for each class the averaged activation matrices of a learnt neural network. Comparing the two graphs allows for the discovery of unwanted and discriminatory correlations.

Finally, fairness can be defined using several notions and quantified using many metrics. There are also different ways of integrating such notions and metrics into the machine learning pipeline, as discussed in the next subsection.

### 1.3.3 Fairness-Enhancing Methods

Depending on which step of the machine learning pipeline they intervene, fairness-enhancing methods can be divided into three main categories [Bellamy *et al.* 2019, Friedler *et al.* 2019]. For each of them, many methods were proposed in the literature and recent surveys provide more complete overviews (for instance, Figures 3, 4 and 5 of [Caton & Haas 2023]). Hereafter, we introduce the key principles of the three identified categories, summarize their intrinsic advantages and drawbacks and briefly describe example methods.

**Pre-processing** methods aim at removing undesired correlations from the training data before applying standard learning techniques on the sanitized data. The key advantage of such approaches is that they are agnostic to the hypothesis class and learning algorithm, and one sanitized dataset can be used for multiple downstream tasks. However, these techniques usually come with no guarantee regarding the final model's fairness as they only aim at removing bias contained in the data (but not the bias which may be added by the learning procedure). Because of the modifications they perform on the training data to sanitize it, pre-processing techniques may also incur significant utility losses. Example of approaches include [Feldman *et al.* 2015], which modifies the training data non-sensitive attributes so that each attribute's marginal distribution is identical over subsets of examples with different sensitive attributes. [Kamiran & Calders 2012] also introduce several strategies to remove discrimination from the training data. They propose to use different techniques, including modifying class labels, re-weighting the examples, suppressing some of the attributes or re-sampling the data.

[Jiang & Nachum 2020] observe that approaches modifying the examples' attributes or labels might face a legal issue as such practice can be interpreted as training on falsified data. To address this, they propose a re-weighting approach, modifying the examples' distribution to satisfy fairness constraints. [Zemel *et al.* 2013] introduce a framework to Learn intermediate Fair Representations (LFR) with two competing goals: obfuscating knowledge regarding the sensitive attributes while preserving as much information as possible. The learnt representations jointly ad-

dress individual and statistical fairness notions. [Calmon *et al.* 2017] also aim at building a transformation of the data that simultaneously removes discrimination, preserves utility and limits individual examples' distortion. They formulate this task as an optimization problem and show that it is convex under some assumptions. Finally, other approaches [Aïvodji *et al.* 2021b] use local adversarial debiasing, with a discriminator trying to infer the sensitive attribute information from the learnt representation and a sanitizer trying to hide it. One can then apply the built sanitizer before releasing new data. Such methods consider a fairness notion targeting the inability to infer the sensitive attribute from the non-sensitive ones.

**Post-processing** techniques modify the outputs of a trained classifier to achieve fairness. An example method is the ThresholdOptimizer [Hardt *et al.* 2016], which uses Linear Programming to compute per-protected-group probabilities of modifying the original model's predictions. The resulting randomized classifier then satisfies statistical fairness criteria (equalized odds) in expectation. In a follow-up work, [Pleiss *et al.* 2017] apply similar techniques but considers calibrated classifier outputs. [Kamiran *et al.* 2012] re-establish fairness by attributing unfavorable outcomes to members of privileged groups and favorable outcomes to members of unprivileged groups, focusing on the examples with low confidence (*i.e.*, lying close to the decision boundary). [Lohia *et al.* 2019] modify the predictions of some examples of the protected group(s) to improve some given statistical fairness criterion. While doing so, the method selects the examples susceptible to yield a bad individual fairness score, and so both individual and statistical fairness values are improved jointly. Model-specific methods were also proposed, such as [Kamiran *et al.* 2010], which modifies the labels within the leaves of a trained decision tree to satisfy fairness criteria. Still, most of the post-processing approaches are model-agnostic, and they are particularly well suited in contexts were an unfair model is already trained. One common drawback is that they often require access to the sensitive attributes at inference time, which can be seen as a form of disparate treatment and is prohibited in many applications [Barocas & Selbst 2016, Zafar *et al.* 2017]. Furthermore, because the fairness corrections are performed after the model training, the resulting trade-offs can be highly sub-optimal [Woodworth *et al.* 2017].

**In-processing** (also called **algorithmic modification**) techniques directly adapt the learning procedure to produce inherently fair models. For instance, [Kamishima *et al.* 2012] add a fairness-aware regularizer to the objective of a learning algorithm and integrate it into logistic regression models. [Raff *et al.* 2018] build fair decision trees and random forests by modifying the computation of the information gain used by greedy tree induction mechanisms. The modified splitting criterion penalizes the splits correlated with the sensitive attributes values. The ExponentiatedGradient method [Agarwal *et al.* 2018] approximates the fairness-constrained learning problem (1.3) through its Lagrangian relaxation and solves it by finding the equilibrium of a two-player min-max game. More precisely, a learner minimizes the objective function while an auditor (owning the sensitive attributes) maximizes it by spotting the largest fairness violations. The learner iteratively updates the model's parameters by solving a cost-sensitive classification problem,

in which costs are set according to the unfairness violation coefficients set by the auditor. [Zafar *et al.* 2017] introduce a convex relaxation of traditional fairness constraints, coined decision boundary unfairness and expressed as the covariance between the examples' sensitive attributes and their signed distance to the decision boundary. They integrate it into the learning of convex margin-based classifiers, namely logistic regression and support vector machines.

Adversarial training techniques can also be leveraged to ensure that the sensitive attribute cannot be inferred. Indeed, in [Beutel *et al.* 2017], an additional adversarial classifier is connected to a deep learning model's latent representation. It aims at retrieving the sensitive attribute, while the learnt network tries to hide such information. [Zhang *et al.* 2018a] propose a similar approach, but try to prevent correlations directly between the outcome and the sensitive attributes. Thus, the adversarial component is rather connected to the learnt model's output layer. In-processing methods constitute the most studied category in the literature [Friedler *et al.* 2019], and many methods were proposed in this category. In particular, they generally lead to the best fairness/utility trade-offs, because all the available information is provided to the learning procedure, which can search for the best trade-offs between fairness and utility all at once [Barocas *et al.* 2019]. Their main drawback is that they require the design of specific algorithms, increase the problem complexity and lead to more difficult training [Jiang & Nachum 2020].

The last two subsections provided an overview of the plurality of the existing fairness notions, of the metrics within each type of notion and of the types of approaches to enforce them. Hereafter, we also discuss some associated challenges, in particular related to the compatibility between the different proposed notions.

### 1.3.4 Compatibility & Applicability of Fairness Notions

Several challenges towards applying the aforementioned fairness metrics have been pointed out in the literature. For instance, recent works suggest that some of the proposed notions are jointly incompatible. Furthermore, their formulations were designed mainly by the computer science community, with the purpose of operationalizing and quantifying fairness. Thus, there is some doubt on whether they actually capture some meaningful form of fairness or not. Finally, using them in practice also raises several questions. In the following paragraphs, we further discuss these aspects.

**(In)compatibility of individual and statistical fairness.** [Friedler *et al.* 2016] suggest that statistical and individual fairness often cannot be achieved simultaneously, and provide impossibility results. Indeed, both notions were shown to empirically conflict. However, some works address them jointly.

***Both notions can be jointly enforced.*** Several frameworks were designed to jointly tackle individual and statistical notions of fairness. For instance,

[Zemel *et al.* 2013] include statistical fairness constraints into their process of learning fair representations (ensuring that an individual's chance of being part of a prototype's cluster is not influenced by their membership in a protected group). [Lohia *et al.* 2019] propose a post-processing approach modifying the predictions of some examples of the protected group(s) to improve statistical fairness. As it focuses on the examples yielding a bad individual fairness score, the method is able to optimize both individual and statistical fairness simultaneously.

***Both notions can be aligned or not.*** [Dwork *et al.* 2012] show that individual fairness (formulated as a Lipschitz condition) implies statistical parity, if members of the different protected groups are close enough. More precisely, with carefully chosen distance metric between individuals, they demonstrate that the statistical parity violation can be upper-bounded using both the original distance between the protected groups' distributions and the enforced Lipschitz condition for individual fairness. When the protected groups' distributions are very different, the theoretical upper-bound is loose and it is necessary to achieve *fair affirmative action* ( *i.e.*, preferential treatment). In such cases, statistical and individual fairness can strongly conflict and even lead to trivial classifiers. Indeed, the later forces close examples to have close outcomes, while the former might force more distant examples to have similar outcomes, overall forcing all examples to have similar outcomes. A proposed alternative method is to enforce individual fairness only internally within the protected groups (and not between them) and enforce statistical fairness. Another related approach to addressing preferential treatment is to adjust the metric so that the Lipschitz condition will imply statistical parity. [Lahoti *et al.* 2019] have observed that their individual fairness technique also improves group fairness metrics. However, they note, in line with [Dwork *et al.* 2012], that unless the distribution of features and labels is the same across different protected groups, jointly enforcing individual and group fairness implies some trade-off.

***Both notions are conceptually compatible.*** [Binns 2020] suggests that the observed conflict between individual and group fairness notions may be attributed to the used technical notions but is not inherent to the underlying concepts. On the one side, individual fairness corresponds to the norm of *consistency* as a conception of justice, defined by Aristotle and stating that *when presented with identical cases, judges ought usually to come to the same answer*. On the other side, statistical fairness implements the *luck egalitarianism* principle, whose key idea is that inequalities between people can not be justified by attributes or circumstances they are not responsible for. The principles of consistency and egalitarianism do not conflict and "can even be seen as mutually implied". Furthermore, the author suggests that individual and statistical fairness notions can be applied interchangeably, by selecting appropriate metrics.

**On the alignment with non-technical fairness notions.** Another interesting result of [Binns 2020] is that both individual and statistical fairness notions fail to implement the notion of individualized justice as defined by Aristotle. In fact, as any machine learning systems, fair models rely on previously seen examples and generalization to make new predictions, which conflicts with the idea of a fully individualized treatment. [Selbst *et al.* 2019] emphasize that existing fairness notions are not adapted to the societal context inherent to fairness applications. Indeed, fairness is often problem-specific, which conflicts with the proposed mathematical formulations. Furthermore, a deployed fair model should be seen as one component within a more complex system. Taking into account the entire system when optimizing the model and anticipating the effect of the model on its environment are aspects that are largely ignored by the actual formulations. These works illustrate intrinsic limitations of existing fairness notions: they may not align with nontechnical conceptions of fairness [Datta *et al.* 2023], and may additionally fail to implement legal requirements, which are usually not limited to a single statistical measure [Watkins *et al.* 2022]. Finally, fairness-enhancing methods are known for having potential waterfall effects: mitigating unfairness at one stage of the machine learning pipeline may induce other biases in further stages [Krco *et al.* 2023], and static fairness interventions may have undesired effects on the long run [Liu *et al.* 2019]. This emphasizes the need for a thorough characterization of the effects of fairness interventions over time, in the entire development and deployment pipelines.

**On the applicability of existing fairness metrics.** As mentioned previously, several fairness notions exist, and many metrics were proposed for each of these notions. [Ignatiev *et al.* 2020] identify a list of desired properties for a fairness metric and prove that only fairness through unawareness can satisfy them. However, this approach was shown ineffective in the presence of proxy features (*i.e.*, non-sensitive attributes that are correlated with the sensitive one(s)) [Dwork *et al.* 2012]. For instance, it was shown that racial membership can accurately be inferred from geographic location [Fiscella & Fremont 2006, Long & Albert 2021]. This demonstrates that no fairness metric is perfect. Furthermore, several works theoretically prove impossibility results, suggesting that many popular group fairness metrics are incompatible and applying more than two or three of them together is not feasible [Defrance & Bie 2023]. Nevertheless, some may be more adapted to a particular situation than others. Therefore, when facing a decision making problem, one can wonder *which particular fairness metric should be used for a given scenario.* [Makhlouf *et al.* 2021] precisely study this question. They first identify the key characteristics of the problem at hand, among which the existence of explaining variables (*i.e.*, proxy features that can legitimately explain an unfair outcome), likelihood of intersectionality and the importance of false/true positives/negatives on the application. They ultimately propose a decision diagram to determine which fairness notion should be used based on these characteristics.

An additional limitation is that most of the fairness literature considers the particular task of binary classification. Defining and enforcing fairness for other tasks, such as multi-class classification or regression, or even other paradigms, such as reinforcement learning, is still in its infancy. In the next subsection, we discuss the definition and use of fairness notions for multi-class classification problems.

### 1.3.5 Contribution: Fairness Beyond Binary Classification

A wide majority of works on fairness for machine learning consider the particular case of binary classification [Barocas *et al.* 2019, Caton & Haas 2023]. Indeed, in this setting, the notions of true/false positives/negatives are easily defined with respect to the model's predictions. In the more general multi-class classification setting, these numbers can be computed for each class, and generalizing statistical fairness metrics may be non-trivial. More precisely, most metrics can easily be extended by applying the binary formulation to each label [Alghamdi *et al.* 2022]. For instance, the statistical parity metric enforces, for each label, that the rate of prediction of this label does not differ by more than the unfairness tolerance $\varepsilon$ between the different protected groups. The equal opportunity metric ensures that, for each label, the true positive rates of the different protected groups are no further than $\varepsilon$. Interestingly, the equalized odds metric can have two distinct multi-class formulations [Putzel & Lee 2022]. *Term-by-term equality of odds* aims at equalizing the probabilities of being predicted $\hat{y} \in \mathcal{Y}$ given true label $y \in \mathcal{Y}$, for all pairs $(\hat{y}, y) \in \mathcal{Y}$, which requires $|\mathcal{Y}|^2$ fairness constraints. In contrast, *classwise equality of odds* handles all false positive classifications altogether, only considering the probabilities of being predicted $\hat{y} \in \mathcal{Y}$ given true label either $y = \hat{y}$ (true positives for class $\hat{y}$) or $y \neq \hat{y}$ (false positives for class $\hat{y}$) which only requires $2 \cdot |\mathcal{Y}|$ constraints.

In recent work [Rouzot *et al.* 2022], we survey the literature on fairness in the context of multi-class classification, synthesizing existing metrics in a unified notation. We introduce the flexible notion of *sensitive labels*, which allows the fairness constraints to apply only on a subset of the possible outcomes. We also propose a method to learn optimal fair and interpretable models for multi-class classification. More precisely, we build on `SLIM` (*Supersparse Linear Integer Models*) [Ustun & Rudin 2016, Rudin & Ustun 2018], a Mixed Integer Linear Programming (MILP) model for learning optimal sparse scoring systems for binary classification. We propose `FAIRScoringSystems`, a MILP generalizing `SLIM` to multi-class classification and integrating multi-class fairness constraints. The resulting method, named FAIRScoringSystems, is available online[14], and generates optimal sparse and fair scoring systems for multi-class classification. We empirically evaluate its effectiveness to learn interpretable sparse models achieving good trade-offs between accuracy and fairness in multi-class classification problems. We report and explain an example scoring system learnt using `FAIRScoringSystems` in Figure 1.1.

[14] https://gitlab.laas.fr/roc/julien-rouzot/fairscoringsystemsv0

| Score for | A |
|---|---|
| starts with | 9 pts |
| Graduated | 9 pts |
| Work experience <= 1 | -9 pts |
| Cat == Cat_6 | -9 pts |

| Score for | B |
|---|---|
| starts with | -9 pts |
| Ever_Married | -9 pts |
| Graduated | 9 pts |
| Work experience <= 1 | 6 pts |

| Score for | C |
|---|---|
| starts with | 2 pts |
| Graduated | 9 pts |
| Cat == Cat_6 | 9 pts |

| Score for | D |
|---|---|
| starts with | 1 pts |
| Spending_Score == Low | 9 pts |
| Work experience <= 1 | -4 pts |
| Cat == Cat_6 | 9 pts |

Figure 1.1: Example multi-class scoring system [Rouzot *et al.* 2022] generated by `FAIRScoringSystems` for customer segmentation with sparsity and fairness constraints. More precisely, the number of lines in each scoring system must be less or equal to 4 and the multi-class equal opportunity violation is restricted to be lesser than 0.01. The considered task consists in attributing customers' profiles to one of four categories: $\mathcal{Y} = \{A, B, C, D\}$. Each class has an associated scoring system through which each example goes. For each line of the scoring system, if the example satisfies the stated condition then the corresponding number of points is added to the local score. The final prediction for an example is the class whose scoring system has the highest score. Note that "starts with" is a constant bias determined for each scoring system.

Finally, while most fairness approaches consider binary classification, some methods also handle multi-class one, including our proposed `FAIRScoringSystems`, whose produced models are also interpretable. In any case, the learnt classifiers may not always produce fair decisions when applied on unseen data, and the fairness constraints that were satisfied at training time can be violated on a separate test set. In the next subsection, we review the literature on the proposed methods to enhance fairness generalization in machine learning.

### 1.3.6 Focus on Fairness Generalization: a Literature Review

As aforementioned, many methods were proposed in the literature to enhance the fairness of machine learning models [Caton & Haas 2023, Barocas *et al.* 2019]. However, models that are fair with respect to their training data may still exhibit unfairness when applied to previously unseen data. Indeed, *fairness constraint overfitting* [Cotter *et al.* 2018, Cotter *et al.* 2019] can occur, and fairness generalization has been identified as an open challenge for trustworthy machine learning [Chuang & Mroueh 2021, Huang & Vishnoi 2019, Mandal *et al.* 2020].

To improve the generalization of statistical fairness, several approaches have been designed based on the method proposed by [Agarwal *et al.* 2018], who formu-

lated the problem of learning an accurate classifier under fairness constraints as a two-player zero-sum game. Considering the Lagrangian relaxation of this constrained optimization problem, the first player ($\theta$-player) optimizes the model's parameters for the objective function with current Lagrange multipliers, while the second player ($\lambda$-player) approximates the strongest Lagrangian relaxation by updating the Lagrangian multipliers. In their original contribution, [Agarwal *et al.* 2018] analyzed the fairness generalization error of the models trained using this framework. In order to avoid the *fairness constraints overfitting*, in [Cotter *et al.* 2018, Cotter *et al.* 2019] the $\lambda$-player updates the Lagrangian multipliers based on fairness violations measured on a separate validation set (instead of the training set itself). In [Mandal *et al.* 2020], the $\lambda$-player uses linear programming to compute the worst-case fairness violation among a set of re-weightings of the training set. This approach falls into the category of *Distributionally Robust Optimization (DRO)* techniques. We briefly introduce our DRO-inspired robustness fairness framework in Section 2.5 within Chapter 2.

Other methods also leverage DRO approaches. In [Sagawa *et al.* 2020], a model is learnt while minimizing the maximum error over a set of protected groups defined by the value of some biased attributes. This is motivated by the observation that when training to minimize average error, decision boundaries are often learned for majority groups, and average loss can hide disparities across subgroups in the training set. In a same line of work, [Slowik & Bottou 2021] study the use of DRO with calibration to mitigate such disparities. The approach first computes, for each protected group, the best achievable performance. Then, DRO equalizes the gaps between the actual model's accuracy and this value (rather than the absolute accuracy) across protected groups. Several approaches have been proposed to tackle the worst-group error minimization problem. In particular, different methods do not require the full training set protected groups knowledge. Indeed, annotating protected groups membership for each training point can be costly in real-world settings [Duchi *et al.* 2020, Nam *et al.* 2020, Liu *et al.* 2021a]. Such methods do not reach the performances levels of the standard DRO approach with groups knowledge but constitute interesting alternatives. For example, [Nam *et al.* 2020] and [Liu *et al.* 2021a] use two-stage approaches, in which they first train a model before leveraging its errors to train another more robust one. [Duchi *et al.* 2020] applies a DRO technique to approximate and optimize for a worst-case subpopulation above a certain size, without any group annotations.

In [Taskesen *et al.* 2020], distributionally robust and fair logistic regression models are trained by optimizing the fairness-regularized objective function for a worst-case distribution. This most adversarial distribution is considered within an ambiguity set characterized as a Wasserstein distance-based ball around the original training distribution. [Rezaei *et al.* 2020] also leverage the principles of DRO to optimize a robust logarithmic loss under fairness constraints. Their approach uses a minimax formulation, in which a fair predictor minimizes the training loss while a worst-case approximator of the population distribution (subject to statistic-matching constraints) maximizes it. In a similar line of work, [Wang *et al.* 2021]

propose a distributionally robust measure of unfairness for the Equality of Opportunity metric. Robustness is achieved by computing the worst-case unfairness over a set of neighbouring distributions, within a type-$\infty$ Wasserstein ambiguity set. Taking into account this measure enables the training of distributionally robust fair Support Vector Machines (SVM).

[Du & Wu 2021] propose two algorithms for fair and robust learning under sample selection bias. These two methods aim at estimating the sample selection probabilities, by leveraging (or not) the availability of unlabeled unbiased data. The key point is that knowledge of these biased sample selection probabilities can be used to re-weight the training dataset to make it representative of the true distribution. As an approximation error exists, a minimax approach is used to optimize the objective function for the worst-case sample selection probabilities in a given radius around the estimated ones. The proposed method can only handle the statistical parity metric, which is approximated using decision boundary fairness and included as a regularization term to the objective function. One consequence is that robustness is enforced jointly for error and fairness. Nonetheless, the fairness constraints may not be strictly satisfied.

Measuring prediction stability on the training set, [Huang & Vishnoi 2019] propose the addition of a regularization term to the objective function of a fair learning algorithm. This regularization term aims at ensuring that the predictions of the built model do not vary too much when the training dataset is perturbed. In addition, this method theoretically bounds the generalization error.

In a different line of work, [Slack *et al.* 2020a] study the scenario in which a model trained to be fair may behave unfairly on related but slightly different tasks. This paper introduces two contributions, namely `Fairness Warnings` and `Fair-MAML`. On the one side, `Fairness Warnings` aims at predicting whether shifts in the features' distributions may result in violating fairness. This is achieved by generating perturbed versions of the training set (they only consider mean-shifting of the features), measuring the resulting fairness violation and training an interpretable model to predict such violation given the features' shifts. On the other side, `Fair-MAML` has for objective to learn a fair model that can be adapted to particular new tasks using minimal (and possibly biased) task-specific data. This is done by adding a fairness regularizer (for either the Statistical Parity or Equal Opportunity metrics) to the loss of the Model Agnostic Meta Learning (`MAML`) framework.

More recently, [Chuang & Mroueh 2021] proposed a data augmentation strategy improving the generalization of fair classifiers. This method leverages existing data augmentation strategies to generate interpolated distributions between two given sensitive groups. During training, a regularisation term penalizes changes in the model's predictions between the interpolated distributions. The goal here is to ensure that the model has a smooth behavior along the "path" formed by the interpolated distributions between the two sensitive groups. This approach theoretically and empirically improves the fairness generalization of the built models.

Furthermore, fairness robustness has also been studied in other settings, such as multi-source learning [Iofinova *et al.* 2021] or for other notions of fairness such

as individual fairness [Yurochkin *et al.* 2020].

## 1.4 Interpretability

The objective of this section is to introduce the key notions regarding the different approaches for enhancing the understanding of machine learning models. First, we motivate the need for such approaches in Section 1.4.1. We then try, in Section 1.4.2, to formalize the different dimensions of explainable AI before highlighting some intrinsic challenges. In Section 1.4.3, we introduce a taxonomy of the methods designed to help understanding machine learning models, and distinguish between two types of approaches, namely crafting post-hoc explanations for a black-box model or learning inherently interpretable ones. Afterwards, in Section 1.4.4, we summarize the limitations of the two identified paradigms. Finally in Section 1.4.5, we introduce our contribution on learning optimal hybrid interpretable models, a particular type of model composed of both an interpretable component and a black-box one.

### 1.4.1 Understanding Machine Learning Models

Machine learning models make decisions that increasingly impact our everyday lives. However, their decisions are often opaque. Indeed, the model used to make predictions is usually hidden to the end-user. Furthermore, this model may be too complex anyway to be understood by a human even if entirely revealed. Finally, we may not even be conscious that a machine learning model was used.

To face this situation and protect users from potential harms, legal frameworks have been proposed in recent years, such as the European Union's General Data Protection Regulation (GDPR)[15] [Voigt & Von dem Bussche 2017]. These frameworks define a "right to an explanation": when a decision taken by a ML system produces legal effects concerning a person, or significantly affects her, explanations must be provided. One main limitation of these texts is that there is no clear requirement on the explanations' properties, such as faithfulness (to the explained system), accuracy or completeness. Furthermore, for several applications (*e.g.*, credit scoring, medical applications ...), legal texts can constrain human decision-makers to explain their decisions. Then, if an automated system was used to produce recommendations for the decision-maker, the later must be able to understand and explain the system's predictions [Freitas 2014].

Understanding a model's internal logic facilitates its audit and is important to confirm many desiderata introduced previously [Doshi-Velez & Kim 2017] such as fairness, privacy or causality. As emphasized in [Guidotti *et al.* 2018], it can also be required for ethics, safety and industrial liability. In particular, trained models may learn spurious correlations in the training data and behave unexpectedly in some

[15] https://gdpr-info.eu/

situations. Understanding the model's reasoning can help ensuring that it targets correct patterns.

Comprehensibility of computer-induced models is also a crucial property for the end users to trust and use such models [Guidotti *et al.* 2018]. This is particularly true for critical applications. In such cases, the ability to understand the model can increase its impact, while on the contrary, users are often not prone to use and trust models they do not understand [Freitas 2014].

Finally, [Doshi-Velez & Kim 2017] argue that interpretability can help face an *incompleteness* in the problem formulation. For instance, it can be used for scientific knowledge discovery, systems' safety assessment (in particular, when an end-to-end system is not completely testable), or ethics (when notions of discrimination/fairness are too abstract to be encoded into the system). When optimizing jointly several (possibly incompletely defined) objectives, it can also help understanding the "dynamics of the trade-off".

Overall, many aspects motivate the need to understand machine learning models and their predictions. In the next subsection, we try to formalize the notion of explainability/interpretability, and explain why such task is inherently difficult.

### 1.4.2 Formalizing Explainable AI

[Doshi-Velez & Kim 2017] define interpretability as "the ability to explain or to present in understandable terms to a human". This definition is large enough to include a variety of techniques, and may correspond to different requirements depending on the task at hand and on the audience receiving the explanation. Indeed, interpretability is domain-specific and can be assimilated to a set of application-related constraints [Rudin 2019]. Hence, [Dziugaite *et al.* 2020] formulate interpretability as an abstract notion (whose instantiation depends on the precise context) corresponding to a set of constraints over the learning process, restricting the space of possible classifiers. [Aghaei *et al.* 2019] further argue that interpretability is subjective, and motivate the interest of letting the end-user customize the obtained model or explanation, to increase his adherence and trust to the explained concepts by maximizing its own comprehension.

Overall, the notion of interpretability is difficult to define in the general case, and therefore [Doshi-Velez & Kim 2017] and [Guidotti *et al.* 2018] propose to decompose it into several latent dimensions to ease its characterization. In particular, *time limitations* encode the idea that in some applications, a shorter (but possibly incomplete) explanation can be preferable, while in others a longer but exhaustive explanation may be required. The *nature of user expertise* relates to the fact that the content, granularity and form of explanations should be adapted to the user that will be given such explanations. Furthermore, *global interpretability* refers to methods explaining an entire model's logic, while *local interpretability* only aims at explaining specific decisions.

[Doshi-Velez & Kim 2017] also propose several *task-related* dimensions, which characterize the problem being considered rather than any specific explanation or

model. On the opposite, *method-related* latent dimensions of interpretability aim at assessing an explanation's quality based on its particular form. These basic units of explanations are called *cognitive chunks*. Beyond their number, the form of the cognitive chunks and the way they are connected also matter. In particular, *monotonicity* facilitates the models' understanding and enhances their acceptance.

Finally, [Doshi-Velez & Kim 2017] identify several approaches to evaluate and quantify interpretability. First, *application-grounded evaluation* ("real humans, real tasks") requires domain experts to evaluate the quality of an explanation in the context of its end-task. It is the more thorough evaluation process, but also the most expensive one, hence other approaches may be preferred. Second, *human-grounded metrics* ("real humans, simplified tasks") evaluate the quality of an explanation without a specific end-goal. They are easier and cheaper. Finally, *functionally-grounded evaluation* ("no human, proxy tasks") use some mathematical definition of interpretability as a proxy for explanation quality.

While defining interpretability is a difficult task, we introduce in the next subsection a taxonomy of the methods designed to improve human understanding of machine learning models.

### 1.4.3 Taxonomy of Explainable AI Methods

Two main approaches to explainable AI can be distinguished: computing *post-hoc* explanations that are either global or local approximations of a trained black-box model, or learning an inherently interpretable one [Lipton 2018]. We detail these two main families of approaches in the next paragraphs. One can note that different taxonomies of explainable AI methods were proposed in the literature, distinguishing them using different criteria [Molnar 2020]. A more complete overview can be found in a recent survey [Speith 2022] synthesizing the different existing taxonomies, and proposing a general framework, encompassing all the previous ones.

**Post-hoc explanations of black-box models.** There are two main reasons for which a model can be considered as a black-box: it is either too complicated to be understood, even with full knowledge of its parameters, or it is proprietary and its internals are not publicly accessible (for privacy, security or economical reasons) [Rudin 2019]. In these situations, to provide some understanding to such model, one has to craft *post-hoc explanations*. These explanations can target different problems [Guidotti *et al.* 2018]:

- *Black-box model explanations*, also called global explanations, aim at providing an interpretable surrogate of the black-box model. The objective is to explain the behaviour of the entire black-box. Interpretable surrogates of the black-box can be trained simply by fitting the black-box predictions (as if they were labels), hence optimizing the surrogates' *fidelity*. Other more sophisticated techniques exist, relying on a deeper analysis of the black-box internals. For instance, [Vidal & Schiffer 2020] transform a random forest into a single yet

functionally equivalent decision tree. Intuitively, the key idea is to determine the different decision regions (within the feature space) and to compute a tree whose splits define the same regions.

- *Black-box inspection* intends to provide a (visual or textual) representation for understanding either how the black-box model works or why it returns some predictions more likely than others. Several techniques can be leveraged to realize this. For instance, sensitivity analysis gives insights about the features importance within the black-box logic. Partial dependence plots aim at visualizing the functional relationship between a small number of input variables and the black-box predictions. While these methods are model-agnostic, other approaches rely on the black-box internals. For example, the activation patterns of a neural network can be analyzed to understand its computations or identify the features influencing the most the probability of predicting a given class.

- *Black-box outcome explanations*, also called local explanations, can be used to explain particular predictions of the black-box model.

Depending on their form, different types of such explanations can be defined, among which:

- *Example-based explanations* are simply datapoints, belonging to the same space as the model's training set examples. For instance, they can be highly influential training examples [Koh & Liang 2017], nearest neighbours or prototypes. Counterfactual explanations also fall into this category, as they are datapoints close to the explained example but exhibiting a different prediction from the considered black-box. For instance, [Parmentier & Vidal 2021] formulate the problem of finding optimal counterfactual explanations of a random forest using a Mixed Integer Linear Programming model.

- *Feature-based explanations* take the form of a vector in the feature space, in which each coordinate is the degree to which the associated feature influences a model's prediction. Feature-based explanations can be computed using several mechanisms, including gradient-based or perturbation-based methods. The former compute the gradients of a model (*e.g.*, a deep neural network) with respect to the input features, either for a given class or for some intermediate component(s) of the network. This enables to determine which features contribute the most to a particular prediction. For instance, `Grad-CAM` outputs a localization map highlighting the regions of an image that most explain a chosen prediction [Selvaraju *et al.* 2017]. The later perturb the input provided to the black-box and observe the resulting changes in the model's outputs. An example technique is `LIME` (Local Interpretable Model-agnostic Explanations) [Ribeiro *et al.* 2016], which first samples a set of datapoints by perturbating the explained example. `LIME` uses these examples to train an interpretable model locally approximating the black-box decision boundary.

As mentioned in the previous subsection, identifying which properties of an explanation are important (*e.g.*, compactness, comprehensibility, completeness ...) is not a well-defined problem, because it highly depends on the application context. Furthermore, characterizing these properties is also challenging. For instance, the degree of comprehensibility of an explanation depends on the user receiving the explanation and must be assessed in a context-dependent, interdisciplinary manner.

**Transparent-box design.** The key idea underlying transparent-box design is to provide a model which is locally or globally interpretable on its own.

Although the notion of interpretability does not admit a general, simple definition, rule sets, rule lists, decision trees, scoring systems and linear models are commonly considered as interpretable if they have a reasonable size [Lipton 2018, Guidotti *et al.* 2018]. While the meaning of a *reasonable size* is ill-defined and context-specific, it indicates that sparsity (or model *simplicity*) is an important property to consider while building these models.

For instance, one can consider the number of rules within a rule list, as in done in the `CORELS` algorithm described in Section 2.2.2. Furthermore, shorter rules are preferable because they are easier to interpret [Chikalov *et al.* 2013]. Other model-specific measures can be used, such as the number of non-zero weights for linear models or the depth of a decision tree. To generalize sparsity to a model-agnostic metric of interpretability, the notion of decision complexity was proposed [Jo *et al.* 2023]. It is measured as the minimum number of parameters needed for the model to make a prediction given a new example, and allows for comparisons across different types of models (including non-interpretable ones such as neural networks). More elaborated metrics exist, for example counting the number of decision regions to quantify a model's complexity [Agarwal 2021a]. Complexity can also be evaluated with respect to the model's representation: models with similar behavior should be as simple as possible, which meets the Occam's razor principle. While such complexity or sparsity measures can be used as proxies for interpretability, one should keep in mind that they are just syntactical aspects, while comprehensibility should be related to semantics.

An other important design choice concerns the type of interpretable model considered. Indeed, depending on the chosen representation, some of them may be more comprehensible than others [Freitas 2014, Guidotti *et al.* 2018]. For instance, the graphical structure of decision trees, along with the fact that they usually use only a subset of attributes, eases their understanding. Rule lists can produce more compact representations, but the resulting models are also more difficult to interpret, as a rule makes sense only in the context of all the previously unmatched ones. This was also observed in a user study [Lakkaraju *et al.* 2016] in which human users interpret more easily rule sets than rule list models, because the rules ordering affects the users' understanding.

Prototype selection methods can also be considered as interpretable. A prototype is defined as an object that is representative of a set of similar instances

(observed data point or built artifact). Methods linking each instance to its best prototype and assigning its prototype's label work in an interpretable manner, with the prototype acting like an explanation.

While these terms have sometimes been used interchangeably in the literature, from now on, we will refer to any techniques aimed at providing some understanding of a machine learning model as *explainable AI*. On the one hand, *post-hoc explainability* encompasses all methods designed to craft explanations of a trained machine learning model. On the other hand, *interpretability* will specifically designate the use of inherently understandable models.

In the next subsection, we discuss intrinsic limitations of the two previously identified paradigms, namely post-hoc explainability and interpretable models learning and highlight key challenges.

### 1.4.4 Some Limitations of Existing Paradigms

As aforementioned, defining and quantifying explainability are important challenges. In addition, the two paradigms introduced in Section 1.4.3, namely post-hoc explainability and interpretability, also exhibit inherent drawbacks. We detail them hereafter.

**Post-Hoc Explanation Methods are not Trustworthy.** One commonly mentioned drawback of black-box explainability methods is their lack of reliability [Rudin 2019]. Indeed, while post-hoc explanations aim at enhancing the comprehension of a black-box model's internals, there is usually no guarantee that the crafted explanation really reflects the underlying black-box reasoning. In particular, post-hoc explanations can have a high fidelity to the black-box while using totally different features [Lakkaraju & Bastani 2020]. Furthermore, post-hoc explainability methods were shown to be unstable with respect to small perturbations of the input and not robust to distribution shifts [Ghorbani *et al.* 2019]. Methods were proposed to compute robust explanations [Lakkaraju *et al.* 2020], but they do not solve the key limitation of these approaches: optimizing (even well-generalizing) fidelity does not guarantee faithfulness to the actual black-box reasoning. These limitations were exploited in the literature and it was shown that several types of explanations and popular frameworks to compute them can be manipulated by a malicious entity to hide unfair decision-making [Aïvodji *et al.* 2021a]. This aspect if further discussed in Section 2.1.2, in which we review the tensions between explainability and fairness.

Finally, post-hoc explainability frameworks, although model-agnostic, are not suitable for every scenario. For instance, many popular methods rely on building a local linear surrogate to explain an example's classification. However, in situations in which the black-box's decision boundary is not linear around the specific examples, such techniques are not suitable anymore. This is illustrated in [Delaunay *et al.* 2022], in which the authors propose to first sample example

points belonging to the different classes around the explained example. Afterwards, they consider that local linear explanations are suitable only if the different sampled examples are linearly separable. Otherwise, they use a different explainability mechanism based on rules. This brings an improvement in the resulting explanations' fidelity, but the approach sometimes fails to identify unsuitable scenarios or to compute appropriate explanations for the identified ones.

**Interpretability Inherent Trade-Offs.** [Dziugaite *et al.* 2020] propose a theoretical analysis of the trade-offs resulting from *enforcing interpretability* - *i.e.*, restricting the range of a learning algorithm to the set of interpretable models, whatever definition of interpretability is considered. A first effect is a reduction of the space of possible hypothesis, directly impacting the number of different trade-offs (between utility and any other desiderata), which may possibly penalize accuracy. Indeed, many papers assume that interpretability necessarily has to be traded-off with accuracy [Pan *et al.* 2020]. However, this is not theoretically true, as shown in [Dziugaite *et al.* 2020], neither empirically verified in many practical situations [Rudin 2019, Bell *et al.* 2022]. On the contrary, the ability to interpret a model's results and internal logic can be leveraged to improve the learning pipeline, overall benefiting to the model's utility. It is however important to inform the end users about a possible accuracy loss if it is empirically observed, and in such case to let them choose between the interpretable model and a more accurate black-box.

Enforcing interpretability can additionally increase a problem's complexity, or even render it unfeasible. Indeed, building interpretable models often necessitates more efforts, both computationally due to the intrinsic combinatorial nature of common interpretable models, but also in terms of expertise for the human designer. On the same line, [Rudin 2019] states that building interpretable models can bring an additional effort "in terms of both computation and domain expertise". Nevertheless, recent algorithmic and hardware progress mitigate this trend. [Weller 2019] further suggest that interpretability is often a means to an end (*i.e.*, safety, certification, reliability, trust ...) rather than a goal itself - and the effort put into enforcing it should not come at the expense of the final target.

It is also important to note that interpretable models are *only as interpretable as their features* [Guidotti *et al.* 2018, Zytek *et al.* 2022], and all the challenges associated to interpretability, including domain- and user-specificness, apply to the features composing an interpretable model. This is one of the greatest challenges towards the development of interpretability for applications in which examples' features are not interpretable. For instance in computer vision, input examples are images whose attributes are pixel values [Rudin *et al.* 2022]. In such cases, interpretability notions have to be reconsidered. For example in [Chen *et al.* 2019], a neural network matches parts of an image that explain its prediction with parts of prototypes images that are provided to the user.

Finally, we saw in this section that while complex black-boxes may reach high predictive accuracy, post-hoc explainability frameworks used to understand them

are not trustworthy. On the other hand, learning inherently interpretable models does not suffer from this drawback, but can be more challenging and may conflict with utility. To take the best of both worlds, hybrid interpretable models were proposed as discussed in the next subsection.

### 1.4.5 Contribution: Learning Hybrid Interpretable Models

As aforementioned, two distinct paradigms consist in either explaining black-box models or learning inherently interpretable ones. However, rather than treating them as dichotomies, some approaches rather explore the continuum between the two philosophies. More precisely, Hybrid Interpretable Models [Wang 2019, Pan *et al.* 2020, Wang & Lin 2021] are systems that involve the cooperation of an interpretable model and a complex black-box one. At inference time, any input of the hybrid model is assigned to either its interpretable or complex component based on a gating mechanism, as illustrated in Figure 1.2a. The intuition behind this type of modeling is that not all examples in a dataset are hard to classify, and that a potentially large part of them can be accurately classified by a simple model. Transparency is then defined as the ratio of samples that are sent to the interpretable part. The higher the transparency, the more model predictions one can actually understand and possibly certify. However, it is possible that the interpretable component makes more errors on average meaning that the overall system suffers a performance loss. Therefore, an integral part of hybrid modeling is to empirically explore the accuracy-transparency trade-off and find the best compromises, as reported for several state-of-the-art approaches in Figure 1.2b. Despite their high potential, hybrid models remain under-studied and under-used in the interpretability/explainability literature. One of the reasons for this under-exploration could be that learning interpretable models is hard in general (often NP-Hard), and fitting a Hybrid Model on top can only make the task harder. To address this issue, past studies have optimized such models using local search heuristics [Wang 2019, Pan *et al.* 2020]. Nevertheless, the inherent stochasticity of these local search algorithms hinders the ability of practitioners to consistently attain a target level of transparency.

In a recent work [Ferry *et al.* 2023d], we conducted a fundamental investigation of such models from three perspectives: Theory, Taxonomy and Methods. From the theory point of view, we explore Probably-Approximately-Correct (PAC) generalization guarantees of hybrid models. A consequence of our PAC guarantee is the existence of a *sweet spot* for the optimal transparency of the system. When such a sweet spot is attained, a hybrid model can potentially perform better than a standalone black-box. Secondly, we provide a general taxonomy for the different ways of training hybrid models: the *Post-Black-Box* and *Pre-Black-Box* paradigms. These approaches differ in the order in which the interpretable and complex components are trained. We show where state-of-the-art hybrid models fall in this taxonomy. Thirdly, we implement the two paradigms in a single method: `HybridCORELS`, which extends the `CORELS` algorithm to hybrid modeling. By leveraging `CORELS`, `HybridCORELS` provides a certificate of optimality of its interpretable component and

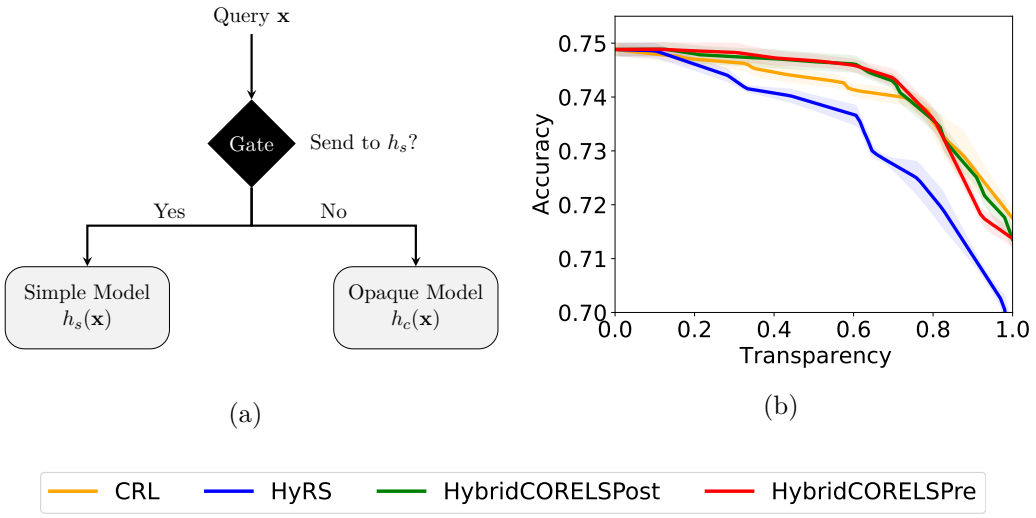



Figure 1.2: Overview of Hybrid Interpretable Modeling. (a) General schematic of a Hybrid Model in which, at inference time, a gating mechanism determines whether to send the instance to the interpretable component $h_s$ or to the complex one $h_c$. (b) Letting transparency be the ratio of samples sent to the interpretable component $h_s$, the trade-off between accuracy and transparency can be measured and compared across different Hybrid Models of the literature [Ferry *et al.* 2023d], on an experiment using the ACS Employment dataset [Ding *et al.* 2021].

precise control over transparency. We finally show empirically that `HybridCORELS` is competitive with existing hybrid models, and performs just as well as a standalone black-box (or even better) while being partly transparent. An example evaluation result for experiments on the ACS Employment dataset [Ding *et al.* 2021] is provided in Figure 1.2b. It shows that our proposed methods $\texttt{HybridCORELS}_{\texttt{Pre}}$ and $\texttt{HybridCORELS}_{\texttt{Post}}$ (respectively based on the *Pre-Black-Box* and *Post-Black-Box* paradigms) provide interesting trade-offs between transparency and accuracy, compared to the state-of-the-art methods `HyRS` and `CRL`.

## 1.5 Privacy

In this section, we introduce key notions regarding the protection of privacy in machine learning. We first discuss, in Section 1.5.1, different methods that can be used to protect the privacy of data. In Section 1.5.2, we then review popular inference attacks against machine learning models. Afterwards in Section 1.5.3, we survey the literature on a particular type of attacks, coined reconstruction attacks, that we later consider in our contributions of Chapters 3 and 4. These attacks motivate the need for privacy-preserving mechanisms. Indeed, in Section 1.5.4, we present a popular framework to protect against inference attacks, namely differential privacy. We discuss some of its applications to machine learning tasks in Section 1.5.5. Finally, in Section 1.5.6, we highlight some limitations and challenges regarding differential

privacy and the current approaches to privacy protection.

### 1.5.1 Achieving Data Privacy

The meaning of privacy has changed over time [Ekstrand *et al.* 2018]. Nowadays, key aspects of this concept include the *limitation theory*, calling for a "limited and contextually bounded" access to individuals' data, and the *control theory*, stating that individuals should be able to choose which information they want to share and which they want to keep private. These concepts relate to the principles of *data minimisation* (personal data collection should be limited to what is necessary and relevant for the considered task) and *data sovereignty* (the data generated within a country is subject to its laws and regulations). New technologies, and in particular the collection and use of huge amounts of data, exacerbate the need to consider these desiderata. Legal texts, such as Title 13 for the Census data in the U.S.[16] or the European Union General Data Protection Regulation[17] [Voigt & Von dem Bussche 2017] in the EU, constrain the use and release of data related to individuals, making privacy a legal requirement. Indeed, several types of techniques targeting data protection exist, addressing different concerns. Hereafter, we discuss two complementary approaches, before focusing on privacy-enhancing notions that aim at protecting the output of a computation against inference attacks.

**Protecting input through cryptography.** Cryptography and privacy target the same high-level objective: protecting the data $\mathcal{D}$ used to perform some computation. More precisely, cryptography ensures that no information about $\mathcal{D}$ is leaked during the computation process [Dinur & Nissim 2003]. Common techniques include Multi-Party Computation (MPC) and Homomorphic Encryption (HE) [Cristofaro 2020]. In a nutshell, MPC allows a set of entities to jointly perform some computation (such as training a machine learning model) while keeping their input data private. HE can also be used to perform computations using encrypted data, without having to first decrypt it. For example, it can be used to compute the predictions of an online machine learning model by only providing it the encrypted data. In both cases, the data used to perform the computation (either training or inference) is kept private. Indeed, the intrinsic goal of such tools is to make sure that *no information other than what could be learnt from the output of the computation is leaked*. However, the result of the computation itself is usually expected to be the same as if it was performed with the original (decrypted) data and it may leak information about it.

Several privacy-enhancing tools precisely address this issue and ensure that the (possibly publicly available) output of the computation cannot be used to retrieve private information about $\mathcal{D}$. Both approaches are complementary, and frameworks

[16] https://www.census.gov/history/www/reference/privacy_confidentiality/title_13_us_code.html

[17] https://gdpr-info.eu/

exist to address them jointly [Kairouz *et al.* 2015], hence ensuring that the input of a computation is not leaked during the computation itself and cannot be retrieved from its outputs neither. Hereafter, we focus on privacy attacks that are possible even in the setup in which an entity owns some data and trains a machine learning model by itself. In such setting, the entity usually does not need to use cryptographic tools. However, if its model is released or queried by external users, appropriate protection methods need to be apply to protect against inference that can be done on the training data from the model itself.

**Privacy-enhancing notions to prevent inference attacks.** A first approach to protect the privacy of individuals whose data is released is de-identification. This method simply consists in removing directly identifying attributes, such as the names or addresses. However, this approach was shown ineffective and highly vulnerable to re-identification attacks, which consist in matching publicly available information to the released data in order to determine the identity of the involved individuals. For instance, this is possible by using *quasi-identifiers*, which are a combination of attributes that are each not directly identifying but can be unique in a population when consider together (*e.g.*, such as the combination of ZIP code, age and gender). A well-known empirical demonstration of such *linkage attack* was the discovery of the Massachusetts governor's personal health information, in a (so-called) anonymized public database, made possible by merging overlapping records with a voter registry [Sweeney 2002].

To counter this risk, several syntactic models of anonymity were proposed, which rely on the key idea of *generalizing a profile to hide him within a group of similar ones* [Clifton & Tassa 2013]. More precisely, these approaches consist in grouping examples within *blocks* so that the profile of a user is indistinguishable among those belonging to the same block. The first introduced notion to realize this is $k$-anonymity [Sweeney 2002, Samarati 2001], which requires that each block contains at least $k$ examples. The examples' *quasi-identifier* (private) features are then replaced with their closure. More precisely, rather than their original, single value, they are defined as sets of possible values, computed as the union of the values for the examples of the block. A remaining issue is that if all individuals within one block have the same value(s) for their private attribute(s), then the value can be determined with certainty, which makes the protection ineffective. $p$-sensitivity [Truta *et al.* 2007] addresses this issue by requiring that at least $p$ different values of the private attribute are represented within each block. $\ell$-diversity [Machanavajjhala *et al.* 2007] is a stronger notion enforcing that within each block, at least $\ell$ different values of the private feature are *well represented*. $t$-closeness [Li *et al.* 2007] additionally ensures that the distribution of these values within each block is sufficiently close to that of the entire dataset. Overall, many frameworks were proposed to improve the original $k$-anonymity notion. However, they were proved to be still vulnerable against several background attacks [Clifton & Tassa 2013].

Another approach, coined differential privacy, was proposed to precisely bound the amount of information the output of a computation leaks regarding its inputs [Dwork *et al.* 2006]. Due to the strong theoretical guarantees it provides, to the interesting properties it exhibits, and to the availability of several mechanisms to enforce it, it has now been widely adopted. Example of recent applications of DP include the 2020 release of the U.S. Census Bureau[18] [Abowd 2018], but also its use by companies such as Google [Aktay *et al.* 2020], Facebook [Herdagdelen *et al.* 2020] and Apple [Team 2017].

The design and use of privacy-preserving mechanisms for machine learning is widely motivated by the flourishing literature on inference attacks against trained models, which are introduced briefly in the next subsection.

### 1.5.2 Inference Attacks against Machine Learning Models

One fundamental objective in privacy protection is to ensure that the output of a computation over a dataset $\mathcal{D}$ cannot be used to retrieve private information about this dataset [Dinur & Nissim 2003]. Inference attacks [Dwork *et al.* 2017] precisely aim at retrieving information regarding the dataset $\mathcal{D}$ by only observing the outputs of the computation. In the machine learning field, the computation being performed is usually a learning algorithm whose output is a trained model.

Inference attacks against machine learning often consider two distinct adversarial settings [Cristofaro 2020, Rigaki & Garcia 2023]. In the *black-box setting*, the adversary does not know the actual trained model's parameters and can only query it through an API. In contrast, in the *white-box* setting, the adversary has full knowledge of the model parameters. Between these two scenarios, different *gray-box* settings are possible. Depending on their objective, different types of inference attacks have been proposed against machine learning models [Cristofaro 2020, Rigaki & Garcia 2023], among which:

- ***Membership inference attacks*** try to infer whether given individuals were used to train a model or not. [Kulynych *et al.* 2022] attribute the success of these attacks to a bad distributional generalization (*i.e.*, when a model's outputs are distributed differently on sets of examples inside and outside its training set) but such attacks are sometimes possible even under well-generalized models. Such attacks constitute an elementary building-block for detecting privacy leaks and possibly build more elaborated privacy attacks. They were proposed in [Shokri *et al.* 2017], and have been studied in various settings. A review of this literature can be found in [Hu *et al.* 2022b]. A recent work [Carlini *et al.* 2022] summarizes the key mechanisms that were proposed in the literature and builds on them to propose new evaluation mechanisms and a more effective attack. One can note that differential privacy (introduced hereafter in Section 1.5.4) precisely upper-bounds the success of membership

[18] https://www.census.gov/programs-surveys/decennial-census/decade/2020/planning-management/process/disclosure-avoidance/differential-privacy.html

inference attacks, by theoretically bounding the contribution of each individual example to the result of the computation [Kulynych *et al.* 2022].

- ***Reconstruction attacks*** aim at reconstructing part of a model's training data. Various settings can be considered [Dwork *et al.* 2017] and Chapters 3 and 4 propose different types of reconstruction attacks. For this reason, we survey the literature with more details in the next subsection.

- ***Model extraction attacks*** aim at stealing a black-box model's internal functionalities or parameters. [Tramèr *et al.* 2016] introduced these attacks, which often consider the black-box setting as the model is proprietary and accessed only through a dedicated prediction API.

- ***Property inference attacks*** often involve building a meta-classifier, which, given a trained model, predicts whether its training data exhibited some property of interest or not [Ateniese *et al.* 2015].

- ***Model inversion attacks*** try to retrieve a model's inputs by only observing the associated outputs [Fredrikson *et al.* 2015]. Such attacks hence do not necessarily target the training data, but rather data provided at inference time[19].

Recent surveys [Rigaki & Garcia 2023, Cristofaro 2020] provide more complete summaries of existing attacks. In the next subsection, we review the literature on a particular type of inference attack against machine learning models, namely reconstruction attacks. Indeed, we will be interested in this type of attacks within Chapters 3 and 4.

### 1.5.3 Focus on Reconstruction Attacks: a Literature Review

Reconstruction attacks have been studied in the context of database access mechanisms since the early 2000s. In the considered setup, a database contains records about individuals, with each record being composed of non-private information along with a private bit (one per individual) [Dwork *et al.* 2017]. The adversary performs queries to a database access mechanism, whose outputs are aggregate and noisy statistics about private bits of individuals in the database. Such reconstruction attacks were introduced and formalized in [Dinur & Nissim 2003], along with some fundamental reconstruction results based on the adversary's capabilities. An efficient linear program for reconstructing private bits of a database leveraging counting queries was also proposed. This linear program was later improved and extended to handle different query types [Dwork *et al.* 2007]. The practical effectiveness of the proposed attacks was demonstrated by a large-scale study carried

[19]The term *model inversion* has nonetheless been used to designate attacks with different objectives. Here, we refer to the taxonomy introduced in [Cristofaro 2020], in which model inversion attacks do not specifically target members of the training set. This is in line with the particular model inversion attacks we later mention in the manuscript.

out by the US Census Bureau in 2018 [Garfinkel *et al.* 2018] and was part of its motivation to adopt differential privacy for future data releases. The linear reconstruction program was also used successfully to break the Diffix commercial database access mechanism [Cohen & Nissim 2020]. Pursuing the same goal, another attack [Gadotti *et al.* 2019] exploited Diffix's data-dependent noise (*i.e.*, sticky noise as well as the addition of static and dynamic noise) to infer private attributes of individuals in a dataset.

One fundamental difference between this line of work and the machine learning privacy literature lies is the nature of the mechanism accessing the private data. In the machine learning (respectively, database access) setup, such mechanism is the learning algorithm (respectively, database access mechanism), and its output is the trained model (respectively, answers to queries). Indeed, database access mechanisms use the private information to compute the answer to each query. On the contrary, in our setup, the training set sensitive attributes are not accessed anymore at inference time, and all the information regarding them is released at once (with the model itself or its predictions). However, our objective is similar to these works: we aim at retrieving a *column* of the dataset by leveraging the output of some computation involving this column (query answers in the previously depicted works, trained fair model in ours).

Other previous works have also tackled reconstruction problems in various settings. For example in the context of online learning, a reconstruction attack was proposed to infer the *updating set* (newly-collected data used to re-train the deployed model) information using a generative adversarial network leveraging the difference between the model before and after its update [Salem *et al.* 2020]. In collaborative deep learning, it was also shown that an adversarial server can exploit the collected gradient updates to recover parts of the participants' data [Phong *et al.* 2017]. In the pharmacogenetics field, machine learning models are learnt to propose medical treatments specific to a patient's genotype and background. In this sensitive context, a reconstruction attack was proposed, taking advantage of the correlation between the sensitive attributes, the non-sensitive ones, and the output of a trained model. More precisely, the attack takes as input a trained model and some demographic (non-private) information about a patient whose records were used for training and predicts the patient's sensitive attributes [Fredrikson *et al.* 2014]. Subsequent work proposed attacks leveraging confidence values output by several ML models to infer private information about training examples given some information about them [Fredrikson *et al.* 2015]. The attack has been shown to be effective against several models and applications, namely decision trees for lifestyle surveys and neural networks for facial recognition. While being different both in terms of techniques and objectives, such inference attack still lies in the category of reconstruction attacks. Finally, other works have studied the intended [Song *et al.* 2017] and unintended [Carlini *et al.* 2019] training data memorization of machine learning models, along with different ways to exploit it in a white-box or black-box setting.

Recent works have considered the special case of training set *sensitive attributes* reconstruction [Aalmoes *et al.* 2022, Hu & Lan 2020, Hamman *et al.* 2022]. The

key challenge here is that while *sensitive attributes* are usually known at training time to ensure the resulting model's fairness, they cannot be used explicitly for inference to avoid disparate treatment [Barocas & Selbst 2016, Zafar *et al.* 2017]. More precisely, [Aalmoes *et al.* 2022] propose a machine learning based attack leveraging an auxiliary dataset whose sensitive attributes are known. [Duddu & Boutet 2022] propose a similar approach, but also leverages *feature-based model explanations* computed with different types of methods. They show that these explanations can be exploited to increase the attack success. Other works [Hamman *et al.* 2022, Hu & Lan 2020] consider a particular setup, in which a fair training process is done in a distributed manner, with a learner wanting to build a fair model on some training dataset for which it does not know the sensitive attributes, and a third-party which owns them. The learner iteratively sends models parameters to the third-party, which then tells him whether the current model is fair. The learner then knows, for an entire set of models, whether they satisfy the fairness constraint or not. [Hamman *et al.* 2022] show that the learner can adversarially query the auditor to retrieve individual sensitive attributes within the training data. [Hu & Lan 2020] propose to use Integer Programming techniques to encode this information and perform the reconstruction of the training set sensitive attributes. This line of works is closely related to the attack we introduce in Chapter 3.

Finally, in the white-box setting, an attack was introduced that exploits the structure of an interpretable machine learning model to reconstruct a probabilistic (uncertain) version of a database [Gambs *et al.* 2012]. We extend this work in Chapter 4.

Overall, reconstruction attacks, as well as the other inference attacks introduced in Section 1.5.2, motivate the need for privacy-preserving mechanisms in machine learning. In the next subsection, we introduce a popular approach to protect the output of a computation from inference attacks: differential privacy.

### 1.5.4 Differential Privacy

Differential privacy (DP) [Dwork *et al.* 2006, Dwork & Roth 2014] is a formal privacy model, ensuring that the output of a computation $\mathcal{L}$ over a database (*i.e.*, dataset) does not depend too much on any single datapoint (*i.e.*, example). More precisely, DP enforces that for any two neighboring datasets $\mathcal{D}$ and $\mathcal{D}'$ (*i.e.*, differing by at most one single example), the probability of observing any particular output to the algorithm must not differ by more than a given factor. This factor is exponential in a $\varepsilon_{DP}$ privacy parameter and intuitively bounds each example's individual contribution to the result of the computation. Another parameter, $\delta_{DP}$, quantifies a risk of failure, which happens if the probability of observing some output to the algorithm differs by more than the aforementioned factor between two neighboring datasets. This is mathematically stated in Definition 1. When $\delta_{DP} = 0$, the algorithm yields *pure* DP guarantees. Otherwise, if $\delta_{DP} > 0$, it satisfies *approximate* DP. Crucially, if $\delta_{DP} << \frac{1}{|\mathcal{D}|}$, the probability of failure is negligible and the privacy protection is meaningful.

**Definition 1.** ***(Differential Privacy)*** *[Dwork & Roth 2014] A randomized algorithm $\mathcal{L} : \mathbb{N}^{|\mathcal{X}|} \mapsto \mathcal{H}$ is ($\varepsilon_{DP}$,$\delta_{DP}$)-differentially private if for any two neighbouring datasets $\mathcal{D}$ and $\mathcal{D}'$ such that $\|\mathcal{D} - \mathcal{D}'\|_1 \leq 1$ and for all $H \subseteq \mathcal{H}$:*

$$\mathbb{P}(\mathcal{L}(\mathcal{D}) \in H) \leq e^{\varepsilon_{DP}} \mathbb{P}(\mathcal{L}(\mathcal{D}') \in H) + \delta_{DP}$$

In addition of providing formal privacy guarantees, differential privacy also exhibits several strong and appealing theoretical properties. Of particular importance is the *resilience to post-processing*: the output of a differentially-private algorithm remains differentially private, whatever computation (not depending on the original private data) is performed. Furthermore when several differentially-private algorithms are applied sequentially or in parallel, *simple and advanced composition* theorems exist that can be used to compute the privacy cost of the entire process. DP is also suitable to *protect groups of examples* as any ($\varepsilon_{DP}$,$\delta_{DP}$)-differentially private algorithm is ($N\varepsilon_{DP}$,$Ne^{N\varepsilon_{DP}}\delta_{DP}$)-differentially private for groups of size $N$.

Several mechanisms have been introduced to enforce pure or approximate DP guarantees. We briefly describe some of the most popular ones hereafter. The key idea underlying them is to perturb the output of the computation with noise calibrated to the function's sensitivity. This ensures that the contribution of any single example is hidden by the perturbation.

One of the simplest mechanisms one can design to satisfy differential privacy is called ***randomized response*** [Warner 1965] and was proposed decades before DP was invented. This technique simply consists in returning a random answer with some probability (otherwise, the true answer is returned). It was proposed to tackle evasive answer bias. For instance, when asking individuals whether they already did some illegal activity, they may not want to answer positively even if it is the case.

The ***Laplace mechanism*** [Dwork *et al.* 2006] consists in adding noise directly to the computed quantity. Such noise is randomly drawn from a Laplace distribution whose magnitude is scaled to the computation's $\ell_1$-sensitivity. This approach has been proven to satisfy pure DP guarantees. The ***Gaussian mechanism*** [Dwork & Roth 2014] consists in adding noise drawn from a Gaussian (normal) distribution whose magnitude is scaled to the computation's $\ell_2$-sensitivity. The Gaussian distribution has lighter tails then the Laplace distribution, hence exhibiting a stronger concentration around the true (un-noised) output. This can result in a better utility, but the Gaussian mechanism only satisfies approximate-DP.

The ***functional mechanism*** [Zhang *et al.* 2012] first approximates an arbitrary function using its polynomial Taylor expansion. Indeed, analyzing and bounding the sensitivity of arbitrary functions can be challenging and/or result in large overestimates. The coefficients of the resulting polynomial form can then be perturbed with noise drawn from a Laplace distribution to satisfy pure DP.

Unlike the aforementioned noise addition techniques, the ***exponential mechanism*** [McSherry & Talwar 2007] consists in drawing an output from a probability distribution. More precisely, it tackles scenarios in which one wants to output an

element with highest utility among a finite set of candidates. Using the aforementioned noise addition mechanisms to perturb the utility function is possible, it could result in some scenarios in significantly harming the utilities comparisons and outputting an element with a low utility. Instead, the exponential mechanism maintains a probability distribution over the set of elements, with the probability of an element to be output by the procedure being related to its utility score. Indeed, the noise here is integrated through the randomness of the probability distribution rather than in the objective function computation.

While it was first proposed in the context of database access mechanisms, differential privacy has also been integrated into machine learning algorithms to ensure privacy of a model's training data, which we discuss in the next subsection.

### 1.5.5 Differentially-Private Machine Learning

Several frameworks leverage the building blocks introduced in the previous subsection to ensure differential privacy during a learning process. Hereafter, we describe some popular methods that are later referred to in this manuscript. Recent surveys [Ji *et al.* 2014, Gong *et al.* 2020] propose a more complete overview of existing differentially-private methods for building privacy-preserving machine learning models.

A first approach is to directly perturb the learning objective through the addition of noise, for example using the functional mechanism. As stated in the previous subsection, it consists in computing the Taylor expansion of the objective, and perturbating its coefficients with random noise. For example, this is done in [Zhang *et al.* 2012] for both linear regression and logistic regression. Even before, [Friedman & Schuster 2010] have proposed a procedure to learn differentially-private decision trees in a greedy manner. This method consists in using the exponential mechanism to determine which attribute to split on at each iteration. They also introduce an error based pruning strategy using noisy counts. `DP-SGD` [Abadi *et al.* 2016] was proposed to train deep learning models with differential privacy. More precisely, the authors modify the traditional Stochastic Gradient Descent (SGD) by clipping the norm of the computed individual gradients (to bound each example's contribution to the computation), and perturbating them with Gaussian noise. The resulting deep neural network parameters then satisfy differential privacy with parameters computed using an "accountant" procedure, which calculates and sums (using composition theorems) the privacy cost at each iteration of the gradient descent.

Ensemble methods can also be leveraged to achieve differential privacy. For example, `PATE` ensures differential privacy in a particular setup, with a private training set and a public unlabeled one [Papernot *et al.* 2017, Papernot *et al.* 2018]. First, the (private) training set is partitioned into a number of non-overlapping subsets used to train a set of *teacher* models. The predictions of the teachers (*i.e.*, vote histograms) are then made differentially private by adding Laplace noise. The public data is labeled using these noisy predictions, and used to train a

differentially-private *student* model. The Bootstrap aggregating (Bagging) ensemble method was also shown to intrinsically yield (weak) differential privacy guarantees [Liu *et al.* 2021b]. This is explained by the performed bootstrap sampling: the method builds a number of base learners, each trained on a dataset generated using random sampling with replacement from the original training set.

Finally, two paradigms can be considered. On the one side, *Global Differential Privacy* consists in using a differentially-private training procedure directly on the original data. However, it requires to trust the entity training the model with the complete dataset. This approach may not be applicable, for instance if the central aggregator (which gathers all the data) is not trusted. Then, another possibility for the different entities producing the data is to release differentially-private versions of their own subsets and to provide them to the aggregator. The resulting dataset can then be processed using traditional learning algorithms. By resilience to post-processing property, the trained models also satisfy differential privacy. This approach is coined *Local Differential Privacy* [Duchi *et al.* 2013], and usually requires the addition of more noise, resulting in more harms in terms of utility.

Differential privacy constitutes a strong theoretical privacy-preserving property for the trained models. However, it also has limitations and is sometimes misused, as discussed hereafter.

### 1.5.6 Differential Privacy: Limitations

As discussed in Section 1.5.1, differential privacy has been widely adopted as a gold standard in machine learning, replacing the former syntactic models of anonymity. However, it was shown [Clifton & Tassa 2013] that these approaches can still be useful. More precisely, the authors explain that syntactic approaches (such as $k$-anonymity or its later extensions) are designed for privacy-preserving data publishing (PPDP) while differential privacy is best suited for privacy-preserving data mining (PPDM). Indeed, while PPDP is possible using differential privacy, the added amount of noise is likely to harm utility significantly. A key difference is that in PPDM, the query whose answer has to be computed on the private data is known in advance, while in PPDP, the released dataset can be used for any purpose (although its privatization may be optimized for some task). While such general-purpose PPDP may constitute a higher danger in terms of privacy, implementation of a specific, interactive query-based mechanism for each particular application is not realistic.

Likewise, differential privacy was originally proposed in the context of database access mechanisms, to answer interactive queries. However, data collection, data release and machine learning tasks correspond to a different setting. While differential privacy still provides privacy protection in these applications thanks to its resilience to post-processing property, its use is such different setups should be carefully analyzed [Domingo-Ferrer *et al.* 2021]. More precisely, the authors point out that differential privacy is currently used by large companies with meaningless privacy parameters. This is largely due to the inherent nature of the performed tasks:

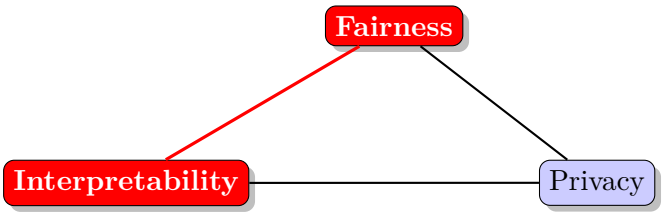
record-level data release or collection requires unreasonable levels of noise to be added to ensure privacy protection, which could harm utility significantly. Furthermore, they often do not respect some key properties such as composition (as respecting it while collecting data sequentially many times is not reasonable), which completely destroys the promised privacy protection. Indeed, the release of individual-level data records intrinsically conflicts with the principles of DP, and significantly harms to utility of the downstream tasks - which in the case of the US Census can have important societal impacts [Pujol *et al.* 2020]. Finally, training differentially-private models (and in particular, deep learning models) is very costly in terms of privacy budget. To maintain utility, common approaches [Abadi *et al.* 2016] enforce the approximate version of differential privacy. However, it comes with a significant price: in particular if $\delta_{DP}$ is in the order of $\frac{1}{|\mathcal{D}|}$, then it is possible to output the complete records of a few number of participants.

Furthermore, differentially-private mechanisms scale the magnitude of the noise added to the computation (directly or indirectly) to the query sensitivity. However, calculating or upper-bounding such sensitivity can be challenging for complex expressions or algorithms, and large upper-bounds result in adding a significant amount of noise, hence harming utility. In addition, such sensitivity has to be worst-case and can reach very high values due to a few number of outliers [Clifton & Tassa 2013]. In such case, large scale noise usually has to be added for all data samples. This is illustrated in [Sarathy & Muralidhar 2011], for computing mean revenue: hiding the contribution of a limited number of examples with very high income leads to the addition of noise whose magnitude is on the order of the actual answer, which affects utility. Determining the values of the chosen privacy parameter $\varepsilon_{DP}$ is also non trivial. This is exacerbated by the fact that in practice, the records within a dataset may not be independent. While group differential privacy still applies to this setup, the resulting privacy loss is affected.

Finally, while differential privacy can be useful and has been widely adopted as a standard in the machine learning community, it may not match the societal and legal expectations regarding privacy protection [Datta *et al.* 2023]. Yet, these expectations are at the crux of the notions of privacy, through the *control theory* mentioned in Section 1.5.1. Pursuing research on other privacy frameworks, improving differentially-private mechanisms, as well as monitoring their impacts on utility, are important research avenues for the development of useful and privacy-preserving learning techniques.

Chapter 2

# Conciliating Fairness & Interpretability through IP




In this chapter, we first provide a large literature review on the intersection between fairness and interpretability in machine learning. We then focus on one specific computational challenge: learning optimal interpretable models under fairness constraints. More precisely, we propose to learn optimal fair rule lists using an Integer Linear Programming based pruning technique. By leveraging jointly fairness and accuracy, this combinatorial optimization technique enables the learning of optimal models that are simultaneously fair and interpretable. Finally, we propose a new robustness framework for fairness, leveraging Integer Programming to improve fairness generalization.

## Contents

As mentioned in Chapter 1, learning machine learning models whose decisions can be understood by human users is a key requirement for trustworthy machine learning. To avoid the drawbacks of post-hoc explainability techniques as depicted in Section 1.4.4, one can learn inherently interpretable models[1] [Rudin 2019]. While many heuristic approaches for learning have been proposed, exact approaches offer a considerable advantage as a lack of optimality can have societal implications [Angelino *et al.* 2018], for example if incorrect classifications result in wrongly disadvantaging people. For instance, `CORELS` is an exact method producing rule lists that are certifiably optimal in terms of accuracy and sparsity [Angelino *et al.* 2017, Angelino *et al.* 2018]. It relies on a branch-and-bound algorithm leveraging several dedicated bounds to prune the search space efficiently. In an early work, we proposed `FairCORELS`, a bi-objective extension of `CORELS` handling both statistical fairness and accuracy [Aïvodji *et al.* 2019b, Aïvodji *et al.* 2021c]. `FairCORELS` consists in an $\varepsilon$-constraint method that leverages `CORELS`' original search tree and bounds for the accuracy objective and considers the fairness objective as a constraint. However, handling such constraints modifies the set of feasible solutions, which makes the exploration considerably harder. Indeed, learning optimal interpretable machine learning models under constraints (*e.g.*, fairness constraints) has been identified as one of the main technical challenges towards interpretable machine learning [Rudin *et al.* 2022].

We leverage combinatorial optimization to address this issue and propose a method that harnesses the fairness constraints to efficiently prune the search space and optionally guide exploration. More precisely, we argue that `CORELS`' original bounds are not sufficient to efficiently explore the search space in this bi-objective setup. To address this, we design Integer Linear Programming (ILP) models combining both accuracy and fairness requirements for well-known statistical fairness metrics. These models are incorporated into `FairCORELS` through effective pruning mechanisms and can also be used to guide the exploration towards fair and accurate rule lists. They can also be combined with a new symmetry-breaking data structure to enhance the scalability of the method while maintaining the optimality guarantee. Our large experimental study, using three datasets with various fairness

[1] As discussed in Section 1.4.2, interpretability is not simply a property of a given hypothesis class, and it should be assessed based on the considered context (and in particular, on the user at hand). However, rule lists of reasonable size are commonly considered as interpretable models [Lipton 2018, Guidotti *et al.* 2018] and we adopt this simplification throughout this chapter.

measures and requirements, demonstrates clear benefits of the proposed approaches in terms of search exploration, memory consumption and learning quality.

However, as discussed in Section 1.3.6, models that are fair with respect to their training data may still exhibit unfairness when applied to previously unseen data. Indeed, *fairness constraint overfitting* [Cotter *et al.* 2018, Cotter *et al.* 2019] can occur, and fairness generalization has been identified as an open challenge for trustworthy machine learning [Cotter *et al.* 2018, Cotter *et al.* 2019, Huang & Vishnoi 2019, Mandal *et al.* 2020, Chuang & Mroueh 2021]. Recent work on fairness generalization targets integrating different techniques for improving robustness into existing fair learning algorithms. While such methods have been shown (theoretically and empirically) to improve fairness generalization, they often induce a considerable computational overhead (*e.g.*, solving an additional problem to determine a worst-case unfairness [Mandal *et al.* 2020]), and thus have limited scalability. Some methods do not suffer from this drawback but instead require additional splitting of the data [Cotter *et al.* 2018, Cotter *et al.* 2019], hence possibly penalizing utility, as the amount of data used to update the model is reduced. Finally, other approaches have limited applicability, as they are designed for a particular learning algorithm or hypothesis class [Taskesen *et al.* 2020, Wang *et al.* 2021], or require some special property of the underlying algorithm (*e.g.*, access to a cost-sensitive classification oracle [Mandal *et al.* 2020]). To tackle these issues, we propose a new framework for statistical fairness robustness. Intuitively, our approach consists in ensuring fairness over a variety of samplings of the training set. We show that this notion can be quantified precisely, and leveraged to audit or train fair and robust machine learning models in practice. We additionally design a flexible and efficient heuristic method for learning robust and fair models, which can easily be integrated into existing fair classification methods, formulated as constrained optimization problems. We summarize the key notions of our fairness robustness framework within this chapter.

**Outline of the chapter.** In Section 2.1, we survey the literature on the connections between fairness and interpretability in machine learning. We then focus on one of the identified tensions: the computational difficulty of learning optimal interpretable models under fairness constraints. Indeed, we present our contribution on using an Integer Linear Programming pruning approach to learn optimal fair rule lists (that are inherently interpretable models). More precisely, we introduce rule lists and the baseline algorithms to learn them in the context of fairness in Section 2.2. We focus on `FairCORELS`, a method for learning fair rule lists that we proposed in an early work. We then describe our proposed pruning approach in Section 2.3, before providing an experimental evaluation in Section 2.4. While performing all these experiments, we noticed that, as already mentioned in the literature, fairness generalization is a fundamental issue. To address this challenge, we proposed a sample-based robustness framework for fairness, which we briefly summarize in Section 2.5.

## 2.1 Connections between Fairness and Interpretability

In this bibliography section, we review the literature at the intersection between fairness and interpretability in machine learning. We first highlight some synergies, before discussing the identified tensions between the two notions.

### 2.1.1 Synergies

**Interpretability eases model audit.** As mentioned in [Rudin 2019], it is easier to detect and debate possible biases or unfairness with an interpretable model than with a black-box. This is a benefit in terms of fairness but sometimes also for accuracy, as it makes it possible to detect and correct possible incorrect preprocessings or problems in the training data - which is more difficult with black boxes. Following the same line of research, [Doshi-Velez & Kim 2017] claims that interpretability can be used to qualitatively ascertain whether other desiderata - such as fairness - are met. Post-hoc explainability tools can also facilitate fairness audit: to gain insight regarding the causes of a model's unfairness, [Begley *et al.* 2020] propose *fairness explanations*. Such explanations are based on Shapley values and aim at attributing a model's overall unfairness to individual input features. In a recent work, [Mougan *et al.* 2023] show that post-hoc explanations can be leveraged to audit fairness properties of a black-box. They propose a "Demographic Parity Inspector" which detects and quantifies existing fairness violations, but can also provide insights regarding the features causing such disparities.

**Fairness may act as a regularizer.** It was observed in the literature that enforcing fairness constraints can have a regularizing effect and reduce overfitting [Kilbertus *et al.* 2018]. More precisely by preventing over-complex models, this can lead to sparser and more interpretable models.

### 2.1.2 Tensions

We first elaborate on the theoretical and empirical tensions between fairness and simplicity, which is often considered as a proxy for interpretability (as mentioned in Section 1.4.3). We then discuss challenges in the joint pursuit of interpretability and fairness desiderata. Finally, we enumerate different unfair effects of providing post-hoc explanations.

#### 2.1.2.1 Tensions between Fairness and Simplicity

**Simplicity and fairness intrinsically conflict.** [Agarwal 2021a] proposes a framework, adapted from the one proposed in [Kleinberg & Mullainathan 2019], to theoretically study the implications of enforcing interpretability. In these works, simplicity is considered as a proxy for interpretability. More precisely, a machine learning model is represented as a set of cells partitioning the input space. Simplifying a model consists in merging some of its cells (hence diminishing their num-

ber and the model's complexity). The authors prove that, for every non-trivial group-agnostic simplification, there exists a more complex classifier that simultaneously strictly improves both accuracy and statistical fairness notions. This classifier can be constructed by carefully selecting some examples from chosen protected groups and splitting their associated cells. Overall, these results suggest that interpretability/simplicity comes at some cost in terms of accuracy/fairness. [Kleinberg & Mullainathan 2019] present similar results, further analyzing that simplicity is *fundamentally inconsistent* with statistical fairness notions. As described in Section 1.4.4, [Dziugaite *et al.* 2020] model interpretability as an abstract notion while noting that enforcing it can only reduce the set of admissible machine learning models. A consequence is that interpretability can only harm (training) accuracy. This result can be extended to fairness: by limiting the space of classifiers, the enforcement of interpretability reduces the number of possible trade-offs, which can be an obstacle to fair and accurate learning.

**Empirical trade-offs are complex.** [Jabbari *et al.* 2020] propose an empirical study of the trade-offs between interpretability and fairness. The number of features available to a classifier is used as a measure of its complexity and acts as a proxy for interpretability. Varying this number, the authors report the variations of statistical fairness notions (namely, Statistical Parity and Equal Opportunity (*cf.* Section 1.3.2)). Experiments on synthetic and real-world datasets show several trends, that mainly depend on the correlation between protected attributes, non-protected ones and class labels. Unsurprisingly, when the sensitive attribute is correlated (even moderately) to the labels, using it explicitly for decision making greatly increases the model's unfairness. These results rely strongly on the chosen notion of interpretability and as such cannot be considered generic. However, they demonstrate that the trade-off between fairness and interpretability is, in practice, complex and data-dependent.

#### 2.1.2.2 Combining Fairness and Interpretability is Challenging

**Learning optimal interpretable models under fairness constraints is computationally challenging.** Due to their combinatorial nature, learning optimal interpretable machine learning models under constraints (*e.g.,* fairness constraints) has been identified as one of the main technical challenges towards interpretable machine learning [Rudin *et al.* 2022]. One can note that there exist approaches producing optimal interpretable and fair machine learning models in the literature. For instance, [Aghaei *et al.* 2019] propose an Integer Programming formulation for learning optimal fair decision trees. The approach is however computationally expensive, and the reported empirical run-times are quite large. Our pruning method leveraging jointly accuracy and fairness to learn optimal fair rule lists (as described in Section 2.3) provides an example technique to mitigate such tension.

**Explanations may not preserve fairness properties of a model.** It was observed [Dai *et al.* 2021] that popular explainability frameworks do not reliably reflect the fairness properties of the explained models. For example, it is possible to compute fair explanations of an unfair model [Aïvodji *et al.* 2019a], and the explanations of a fair model's decisions may (wrongly) rely on sensitive features and exhibit discrimination [Manerba & Guidotti 2022]. In addition, the specific explanation method chosen as well as the type of explanation it produces both impact the users' perceived fairness [Dodge *et al.* 2019]. [Dai *et al.* 2021] investigate the fairness of post-hoc explanations generated from a fair model's decisions. Using group fairness notions, they formulate the fairness of an explanation similarly to that of a classifier (an explanation being seen as a local surrogate model). Then, fairness is computed on a neighbourhood of the explained example. For such artificial points, no label is known and so only the statistical parity metric can be used. These researchers show that the fairness property of the explained model may not be reflected in the generated explanations and propose a framework for producing fairness-preserving explanations.

**Fairness-enhancing methods may require non-interpretable transformations, hence harming interpretability.** In a study on interpretable, fair and accurate ML for criminal recidivism prediction, [Wang *et al.* 2022a] observe that fairness-enhancing methods often require non-interpretable transformations, which are not compatible with interpretability desiderata. Indeed, pre-processing methods usually perform complex transformations of the input features, which harm their original semantic [Kamiran & Calders 2012, Zemel *et al.* 2013]. The resulting representation hence can not be used to produce an understandable model. Furthermore, the corrections performed to a model's outputs by post-processing techniques [Pleiss *et al.* 2017] can also lead to non-interpretable processes.

#### 2.1.2.3 Other Unfair Effects of Explainability Methods

**Post-hoc explanations affect individuals' privacy in a disparate manner.** As discussed further in Section 3.1, minority groups often suffer from increased privacy risks. Interpretability can also exhibit this trend, as noted by [Shokri *et al.* 2020, Shokri *et al.* 2021]. For instance, when investigating whether membership information (*i.e.*, whether an example was part of a model's training set) can be inferred from post-hoc explanations, they observed that outliers and certain "hard to generalize minorities" are at a higher risk of being revealed than majority groups. This is partly due to the fact that they are more susceptible to being part of the generated explanations. In such case, interpretability tools can penalize minorities by leaking more information about disadvantaged groups.

**Post-hoc explanation frameworks can introduce unfairness by providing lower-quality explanations to minority groups.** [Dai *et al.* 2022] investigate group-based disparities in explanation quality. More precisely, they first identify

key characteristics that define the quality of an explanation (*e.g.*, fidelity, stability, consistency and sparsity). They then conduct a large experimental study demonstrating that there is often a disparity in the quality of the produced explanations, disproportionately affecting minority groups. Such quantitative disparity is identified to depend on the type of model being explained and on the particular post-hoc explanation framework considered. Using several real-world applications (*e.g.*, finance, healthcare, college admissions and the US justice system) and post-hoc explanation frameworks, [Balagopalan *et al.* 2022] also show that the fidelity of the produced explanations varies significantly across the different identified subgroups of the population. Finally, they suggest that robustness techniques can help reduce the observed disparity - but emphasize that communicating details regarding such disparity to end-users is critical.

**Counterfactual explanation frameworks can harm subgroups of the population by consistently providing higher-cost recourse.** In the specific case of counterfactual explanations, the *cost of recourse* is defined as the amount of effort a user has to do to implement the provided recourse and change the model's decisions. It was then shown that counterfactual explanation frameworks may provide lower-cost recourse for some subgroups of the population and harm some others [Ustun *et al.* 2019, Sharma *et al.* 2020]. For instance, this means that some protected groups may consistently have to make more effort to implement the provided recourse after a loan refusal. To face this issue, *recourse fairness* was studied [Gupta *et al.* 2019, Karimi *et al.* 2023] and frameworks equalizing the cost of recourse across protected groups were proposed.

**Post-hoc explanations can be manipulated.** Explainability tools are designed to facilitate model audit and enhance the users' understanding. However, because the explanation generation process can be opaque, post-hoc explanations can also be leveraged by some black-box model holder to hide unfair decision-making processes by providing manipulated fair explanations. Indeed, it was shown that black-box explanations can be misleading, in particular because they can achieve high fidelity with respect to the explained model while using entirely different features, leveraging correlations in the feature space [Lakkaraju & Bastani 2020]. The authors demonstrate that this can be exploited and extend the `MUSE` framework [Lakkaraju *et al.* 2019] to generate explanations favoring some given features while avoiding others. They finally conduct a user study and find out that misleading explanations can increase the user trust in black-box models illegitimately.

In fact, several works show that malicious entities can manipulate explainability techniques to hide the true reasoning of the underlying model. For example, they can directly craft manipulated explanations, such as local surrogate models [Aïvodji *et al.* 2019a, Aïvodji *et al.* 2021a] that appear fair but actually explain the output of a globally unfair black-box, with such practice being coined as "fairwashing". They can also manipulate explanation frameworks, for instance by detect-

ing artificial examples generated by input-perturbation based methods and giving them a chosen output value [Slack *et al.* 2020b]. This can be leveraged to hide a black-box model's unfairness by crafting and providing fair explanations to a fairness auditor [Slack *et al.* 2021b]. Furthermore, [Heo *et al.* 2019, Dimanov *et al.* 2020] show that it is possible to fine-tune (*i.e.*, slightly modify) a pre-trained model to manipulate the output of feature importance explanation methods while having little impact on the model's accuracy. Considering sequence classification and sequence-to-sequence tasks (*i.e.*, in which the input of the model is a sequence of words), [Pruthi *et al.* 2020] propose a method to train a model with significantly reduced attention mass over some chosen words (*e.g.*, gender-related prefixes) while still using them for prediction. A user study shows that the proposed method is effectively able to mislead users into thinking that the underlying model is fair, while it is actually biased against gender minority.

It was also shown to be possible to learn a model so that the counterfactual explanations generated by some off-the-shelf algorithm look *recourse fair* across subgroups of the population (*i.e.*, the cost of the recourse associated to the counterfactual explanations does not vary too much between individuals from the different subgroups), while also being able to generate lower-cost recourse explanations for some privileged subgroup(s) by simply adding a small adversarial perturbation [Slack *et al.* 2021b, Slack *et al.* 2021a]. [Zhang *et al.* 2020] show how an adversary can generate adversarial examples with chosen prediction by the black-box model that also fool popular interpretability tools. This illustrates the fact that post-hoc explainability techniques are not a reliable way to detect adversarial inputs manipulation. Finally, [gabriel laberge *et al.* 2023] consider the setup of a fairness audit in which the data is private and owned solely by the malicious model holder, which provides subsamples to the external auditor. They show that the former can manipulate the auditor's explainability methods to hide unfair decision-making (such as the influence of a protected attribute) by providing adversarially-selected data samples. Such malicious practices are in addition particularly difficult to detect in a remote setting, in which the explanation is provided by a third-party API [Merrer & Tredan 2019].

We identified some synergies and several tensions between interpretability and fairness, among which a technical conflict: learning optimal interpretable models under fairness constraints is computationally challenging. In the next subsections, we show how Integer Linear Programming-based pruning techniques can be used to mitigate this tension for learning optimal fair rule lists.

## 2.2 Learning Fair Rule Lists

In this section, we first introduce a particular type of interpretable models, namely rule lists. We then present a state-of-the-art algorithm for learning certifiably optimal rule lists called `CORELS`. Finally, we describe `FairCORELS`, an extension of

`CORELS` that we proposed in an early work to learn optimal fair rule lists. `CORELS` will serve as a baseline for our pruning algorithm.

**Notation.** As mentioned in Section 1.1.1, $\mathcal{D}$ denotes a training set and $h$ a classifier. Throughout this chapter, we assume that $\mathcal{D}$ is partitioned into two groups: a protected group $\mathcal{D}^p$ and an unprotected group $\mathcal{D}^u$. This partition depends on the value of the sensitive feature(s) and so we can write $\mathcal{S} = \{p, u\}$. Thus for $s \in \{p, u\}$, we have $\mathcal{D}^s = \{e_j \mid s_j = s\}$. Furthermore, we restrict our attention to the binary classification case (*i.e.*, $\mathcal{Y} = \{0, 1\}$) as the baseline algorithms we consider and introduce in this subsection handle this particular problem. The examples in $\mathcal{D}$ are partitioned into $\mathcal{D}^+$ and $\mathcal{D}^-$, which correspond respectively to positive examples and negative ones. Precisely, we have $\mathcal{D}^+ = \{e_j \mid y_j = 1\}$ and $\mathcal{D}^- = \{e_j \mid y_j = 0\}$.

### 2.2.1 Rule Lists

We consider classifiers that are expressed as *rule lists* [Rivest 1987], which are formed by an ordered list of *if-then* rules, followed by a default prediction. More precisely, a *rule list* can be represented as a tuple $RL = (\delta_{RL}, q_0)$ in which $\delta_{RL} = (r_1, r_2, \ldots, r_K)$ is $RL$'s *prefix*, and $q_0 \in \{0, 1\}$ is a *default prediction*. A prefix is an ordered list of $K$ distinct association rules $r_i = a_i \rightarrow q_i$. Each rule $r_i$ is composed of an *antecedent* $a_i$ and a *consequent* $q_i \in \{0, 1\}$. Each antecedent $a_i$ is a Boolean assertion over $\mathcal{X}$ evaluating either to true or false for each possible input $x_j$. If $a_i$ evaluates to true for example $e_j$, that rule $r_i$ is said to *capture* $e_j$. Similarly, if at least one of the rules in $\delta_{RL}$ captures $e_j$, that prefix $\delta_{RL}$ is said to capture example $e_j$. For example, rule list 2.1 predicts whether a given individual has a [low] or [high] salary. Its prefix is composed of five rules, and its default decision is [low]. Using a rule list $RL = (\delta_{RL}, q_0)$ to classify an example $e$ is straightforward as rules in $\delta_{RL}$ are applied sequentially. If $e$ is not captured by prefix $\delta_{RL}$, then the default prediction $q_0$ is returned. Finally, remark that rule list $((), q_0)$ is well defined, and simply consists of a default prediction (hence representing a constant classifier).

Rule list 2.1: Example rule list found by `FairCORELS` on the Adult Income dataset.

```
if [occupation:Blue-Collar] then [low]
else if [occupation:Service] then [low]
else if [capital gain: > 0] then [high]
else if [not(workclass:Government)] then [low]
else if [education:Masters/Doctorate] then [high]
else [low]
```

### 2.2.2 `CORELS`

`CORELS` [Angelino *et al.* 2017, Angelino *et al.* 2018] is a state-of-the-art supervised learning algorithm that outputs a certifiably optimal rule list minimizing the following objective function on a given training dataset $\mathcal{D}$:

$$\mathsf{obj}_{\texttt{CORELS}}(RL, \mathcal{D}) = \mathsf{misc}(RL, \mathcal{D}) + \lambda \cdot K_{RL}, \tag{2.1}$$

in which $\mathsf{misc}(RL, \mathcal{D}) \in [0, 1]$ denotes the training classification error rate of the rule list $RL$, $K_{RL}$ is the length of $RL$ (*i.e.*, number of association rules in $RL$) and $\lambda$ is a regularization hyper-parameter for sparsity. `CORELS` is a branch-and-bound algorithm, representing the search space of rule lists $\mathcal{R}$ as a prefix tree. Each node is a prefix in this tree, and each child node is an extension of its parent, obtained by adding exactly one rule at the end of the parent's prefix. Finally, the root node corresponds to the empty prefix. Each node is a possible solution (*i.e.*, rule list), obtained by adding a default decision (based on majority prediction) to the prefix associated with this node. While this search space corresponds to an exhaustive enumeration of the candidate solutions, `CORELS` leverages several bounds to prune it efficiently. Thanks to these bounds, along with several smart data structures, `CORELS` is able to find optimal solutions with a reasonable amount of time and memory. The set of antecedents $A$ is pre-mined and given as input to the algorithm[2]. While `CORELS` is agnostic to the rule mining procedure used as preprocessing, an overview of existing techniques can be found in [Chikalov *et al.* 2013]. For the sake of illustration, we provide an example prefix tree for a toy dataset with a set of three pre-mined antecedents within Figure 2.1.

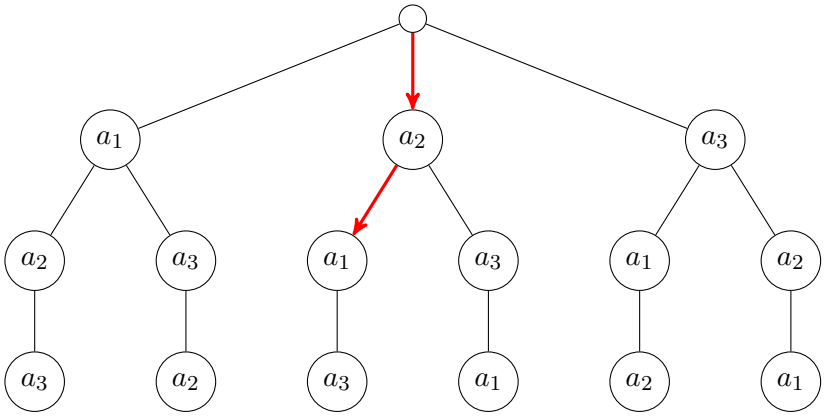



Figure 2.1: Example of prefix tree for a dataset with set of pre-mined antecedents $A = \{a_1, a_2, a_3\}$. The highlighted red path corresponds to prefix $\delta = ((a_2 \rightarrow q_2), (a_1 \rightarrow q_1))$, where consequents $q_2$ and $q_1$ are set using majority prediction. Associating a default prediction $q_0$ to $\delta$ (using majority prediction), we obtain the associated rule list $(\delta, q_0)$. Clearly, each node within this tree defines a unique prefix, which can be associated to a default prediction to form a rule list.

[2]This means that `CORELS` produces optimal rule lists *for the given pre-mined antecedents*. Some approaches that do not use rules pre-mining exist but usually they do not scale well. For instance, [Dash *et al.* 2018] propose a MILP model for learning optimal rule sets. Instead of pre-mining the rules, they dynamically produce new rules using column generation. This process is able to generate certifiably optimal rule sets (with respect to the original features) for small datasets. However, it requires the use of heuristics to solve the pricing problem (which generates candidate rules for improving the current solution) at scale. [Yu *et al.* 2020] do not require rules pre-mining neither, as they leverage declarative programming (Maximum Boolean Satisfiability) to build optimal rule lists, with the rules' making being part of the search. Again, as noted by the authors, the approach does not scale well compared to `CORELS`.

### 2.2.3 FairCORELS

FairCORELS is a bi-objective extension of CORELS jointly addressing accuracy and statistical fairness, integrating several metrics from the literature. Formally, given a statistical fairness notion, whose violation by a rule list $RL$ on dataset $\mathcal{D}$ is quantified by an unfairness function $\mathsf{unf}(RL, \mathcal{D})$ and a maximum acceptable unfairness violation $\varepsilon$, FairCORELS solves the following optimization problem (which instantiates the generic constrained optimization problem 1.3):

$$\begin{aligned} \underset{RL \in \mathcal{R}}{\arg\min} \quad & \mathsf{obj}_{\texttt{CORELS}}(RL, \mathcal{D}) \\ \text{such that} \quad & \mathsf{unf}(RL, \mathcal{D}) \leq \varepsilon \end{aligned} \tag{2.2}$$

FairCORELS is presented in Algorithm 1. In this algorithm, $RL^c$ denotes the current best solution and $z^c$ is its objective value. Moreover, a priority queue $Q$ of prefixes is used to store its exploration frontier. The priority queue ordering defines the exploration heuristic. The function $\mathsf{lb}(\delta, \mathcal{D})$ (coming from the CORELS algorithm) gives an objective lower bound for any rule list built upon prefix $\delta$ on the dataset $\mathcal{D}$. At each iteration of the main loop, a prefix $\delta$ is removed from the priority queue (Line 4). When the lower bound of $\delta$ is less than the current best objective value (Line 5), two operations are considered. First, the rule list $RL$ formed by prefix $\delta$ along with a default prediction is accepted as a new best solution if it improves the current best objective value while respecting the unfairness tolerance (Line 9). Second, extensions of $\delta$ using the antecedents not involved in $\delta$'s rules are added to the queue (Line 12).

**Fairness metrics considered.** Let $\varepsilon \in [0, 1]$ denote an unfairness tolerance value (*i.e.*, the maximum acceptable value for the unfairness measure). Thus, the fairness requirement gets harder as $\varepsilon$ gets smaller. For a classifier $h$, among a group $\mathcal{D}^s$, with $s \in \{p, u\}$, we denote by $TP^h_{\mathcal{D},s}$ the number of true positives, $TN^h_{\mathcal{D},s}$ the number of true negatives, $FP^h_{\mathcal{D},s}$ the number of false positives and $FN^h_{\mathcal{D},s}$ the number of false negatives. Table 2.1 gives the definition of the four metrics considered within FairCORELS. One can observe that the provided formulations bound the *pairwise difference* of the statistical measure across protected groups, following the *one-vs-one* formulation (*cf.* Table 1.2). Indeed, because we restrict our attention to the binary sensitive attribute case (as mentioned at the beginning of this section), such formulations result in a single linear constraint, as explained in Section 1.3.2.

**Building sets of trade-offs between accuracy and fairness.** The constrained optimization formulation of the fair learning problem used in FairCORELS (Problem 2.2) allows for the construction of different trade-offs between accuracy and fairness using a simple $\varepsilon$-constraint method [Haimes 1971]. More precisely, it is possible to build a set of non-dominated solutions, also called Pareto frontier, by varying the value of the unfairness tolerance $\varepsilon$. We provide an example of such

**Algorithm 1** `FairCORELS`

**Input**: Training data $\mathcal{D}$ with set of pre-mined antecedents $A$; unfairness tolerance $\varepsilon$; initial best known rule list $RL^0$ such that $\mathsf{unf}(RL^0, \mathcal{D}) \leq \varepsilon$

**Output**: $(RL^*, z^*)$ in which $RL^*$ is a rule list with the minimum objective function value $z^*$ such that $\mathsf{unf}(RL^*, \mathcal{D}) \leq \varepsilon$

$(RL^c, z^c) \leftarrow (RL^0, \mathsf{obj}(RL^0, \mathcal{D}))$
$Q \leftarrow queue(())$ ▷ Initially the queue contains the empty prefix ()
**while** $Q$ not empty **do** ▷ Stop when the queue is empty
  $\delta \leftarrow Q.pop()$
  **if** $\mathsf{lb}(\delta, \mathcal{D}) < z^c$ **then**
    $RL \leftarrow (\delta, q_0)$ ▷ Set default prediction $q_0$ to minimize training error
    $z \leftarrow \mathsf{obj}_{\texttt{CORELS}}(RL, \mathcal{D})$
    **if** $z < z^c$ **and** $\mathsf{unf}(RL, \mathcal{D}) \leq \varepsilon$ **then**
      $(RL^c, z^c) \leftarrow (RL, z)$ ▷ Update best rule list and objective
    **for** $a$ in $A \setminus \{a_i \mid \exists r_i \in \delta, r_i = a_i \rightarrow q_i\}$ **do** ▷ Antecedent $a$ not involved in $\delta$
      $r \leftarrow (a \rightarrow q)$ ▷ Set $a$'s consequent $q$ to minimize training error
      $Q.push(\delta \cup r)$ ▷ Enqueue extension of $\delta$ with $r$
$(RL^*, z^*) \leftarrow (RL^c, z^c)$

Table 2.1: Summary of four statistical fairness metrics widely used in the literature, using a *one-vs-one* (pairwise) formulation as implemented within `FairCORELS`.

| Metric | Constraint Expression |
|---|---|
| Statistical Parity (SP) | $\left\lvert \frac{TP^h_{\mathcal{D},p} + FP^h_{\mathcal{D},p}}{\lvert\mathcal{D}^p\rvert} - \frac{TP^h_{\mathcal{D},u} + FP^h_{\mathcal{D},u}}{\lvert\mathcal{D}^u\rvert} \right\rvert \leq \varepsilon$ |
| Predictive Equality (PE) | $\left\lvert \frac{FP^h_{\mathcal{D},p}}{\lvert\mathcal{D}^p \cap \mathcal{D}^-\rvert} - \frac{FP^h_{\mathcal{D},u}}{\lvert\mathcal{D}^u \cap \mathcal{D}^-\rvert} \right\rvert \leq \varepsilon$ |
| Equal Opportunity (EOpp) | $\left\lvert \frac{TP^h_{\mathcal{D},p}}{\lvert\mathcal{D}^p \cap \mathcal{D}^+\rvert} - \frac{TP^h_{\mathcal{D},u}}{\lvert\mathcal{D}^u \cap \mathcal{D}^+\rvert} \right\rvert \leq \varepsilon$ |
| Equalized Odds (EO) | Conjunction of PE and EOpp |

frontier in Figure 2.2. For the four considered datasets, we highlight three specific points within the entire Pareto frontier. Those correspond respectively to the most accurate or most fair models as well as the one minimizing the sum of classification error and unfairness (*best delta*). Depending on the task at hand, the end user can then select his favorite model within the frontier, considering its accuracy and unfairness, but also the models themselves as they are interpretable.

One important challenge within `FairCORELS` is that the fairness constraints modify the set of feasible solutions and the resulting search space is considerably more difficult to explore. First, because of the fairness requirement, the objective function value is updated less often as many potential solutions may not satisfy the fairness constraint. In addition, a fairness-accuracy trade-off is often observed,

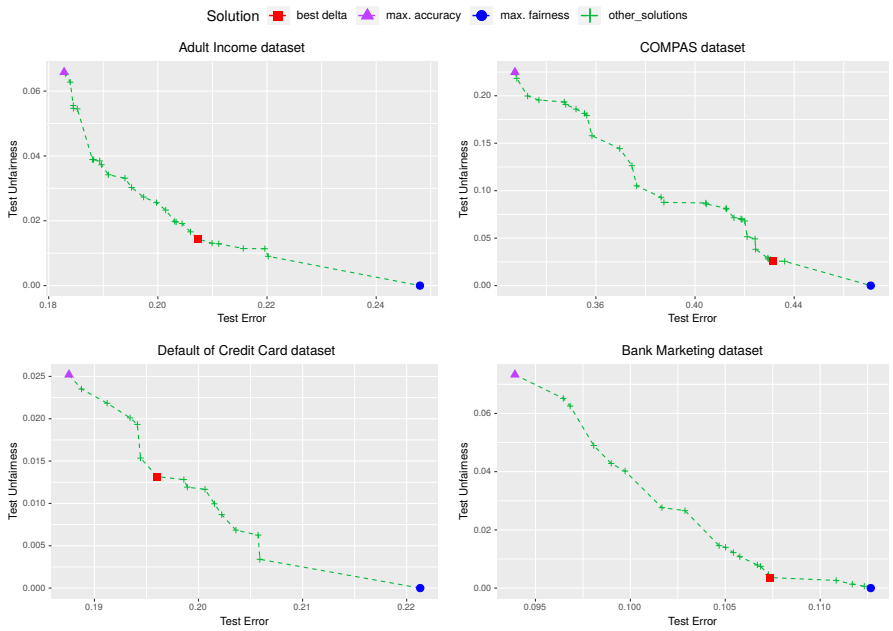



Figure 2.2: Example test set Pareto frontiers (unfairness/classification error trade-offs) built with `FairCORELS` for the statistical parity fairness metric on four different datasets. Bottom-left (low unfairness, low error) is preferable.

which results in achieving lower objective function values. Indeed, `CORELS`' original bounds are less efficient as the fairness constraint gets stronger. Furthermore, some data structures used by `CORELS` to speed up the exploration are no longer usable. For instance, the prefix permutation map, which reduces considerably the running time and the memory consumption [Angelino *et al.* 2017, Angelino *et al.* 2018], does not apply anymore. This symmetry-aware map ensures that only the best permutation of each set of rules containing the same antecedents is kept. However, it cannot be used within `FairCORELS` without sacrificing optimality. Indeed, a given permutation may allow for better objective function values than others but may not lead to solutions meeting the fairness requirement. In this situation, one could miss solutions that exhibit lower objective function values and meet the fairness requirement. Since we are interested in preserving the guarantee of optimality, we cannot use such a data structure. However, we note that a weaker permutation map can be designed and used by implementing a more restrictive symmetry criterion (additionally enforcing that two prefixes implying the same antecedents are not equivalent if they do not have the same confusion matrix). Although performing a less effective reduction of the search space, this approach (which we present in Section 2.4.3) preserves the guarantee of optimality. Overall, both observations motivate the need for a new pruning approach, leveraging both the objective function value and the fairness constraint to efficiently explore `FairCORELS`' search space.

## 2.3 Proposed Pruning Approach

This section presents our proposition to prune the search space of `FairCORELS` by reasoning jointly about the number of well-classified examples and fairness. The main idea is to discard prefixes that cannot improve the current objective while satisfying the fairness requirement before being treated. To realize this, one has to guarantee that for any prefix discarded, none of its extensions can satisfy both requirements, which is the purpose of Section 2.3.1. Afterwards, Section 2.3.2 exploits this property in the presentation of our proposition.

### 2.3.1 A Sufficient Condition to Reject Prefixes

Let $\mathcal{D}$ be a training set and $RL$ be a rule list. We use $W_{\mathcal{D}}^{RL}$ to denote the number of examples of dataset $\mathcal{D}$ well classified by $RL$:

$$W_{\mathcal{D}}^{RL} = TP_{\mathcal{D},p}^{RL} + TP_{\mathcal{D},u}^{RL} + TN_{\mathcal{D},p}^{RL} + TN_{\mathcal{D},u}^{RL} \tag{2.3}$$

$$= TP_{\mathcal{D},p}^{RL} + TP_{\mathcal{D},u}^{RL} + |\mathcal{D}^p \cap \mathcal{D}^-| - FP_{\mathcal{D},p}^{RL} + |\mathcal{D}^u \cap \mathcal{D}^-| - FP_{\mathcal{D},u}^{RL} \tag{2.4}$$

We slightly extend the notation introduced in Section 2.2.3. For a prefix $\delta$, among a group $\mathcal{D}^s$ with $s \in \{p, u\}$, we denote by $TP_{\mathcal{D},s}^{\delta}$ (respectively $TN_{\mathcal{D},s}^{\delta}$, $FP_{\mathcal{D},s}^{\delta}$ and $FN_{\mathcal{D},s}^{\delta}$) the number of true positives (respectively true negatives, false positives and false negatives) among the examples of $\mathcal{D}$ captured by $\delta$. Similarly, we define $W_{\mathcal{D}}^{\delta}$ as the number of examples well classified by $\delta$, among the examples of $\mathcal{D}$ that $\delta$ captures. Clearly, $W_{\mathcal{D}}^{\delta} = TP_{\mathcal{D},p}^{\delta} + TP_{\mathcal{D},u}^{\delta} + TN_{\mathcal{D},p}^{\delta} + TN_{\mathcal{D},u}^{\delta}$.

We define $\sigma(\delta)$ to be the set of all rule lists whose prefixes start with $\delta$: $\sigma(\delta) = \{(\delta_{RL}, q_0) \mid \delta_{RL} \text{ starts with } \delta\}$. Formally, we say that $\delta_{RL}$ starts with $\delta$ (a prefix of length $K$) if and only if the $K$ first rules of $\delta_{RL}$ are precisely those of $\delta$, appearing in the same order.

Consider $RL = (\delta_{RL}, q_0)$ such that $RL \in \sigma(\delta)$. On the one hand, some examples of $\mathcal{D}$ cannot be captured by $\delta$. On the other hand, all examples of $\mathcal{D}$ captured by $\delta$ are captured by $\delta_{RL}$ and have the same prediction as with $\delta$.

**Proposition 1.** *Given a prefix $\delta$, a rule list $RL \in \sigma(\delta)$ and $s \in \{p, u\}$, we have:*

$$TP_{\mathcal{D},s}^{\delta} \leq TP_{\mathcal{D},s}^{RL} \leq |\mathcal{D}^s \cap \mathcal{D}^+| - FN_{\mathcal{D},s}^{\delta}$$
$$FP_{\mathcal{D},s}^{\delta} \leq FP_{\mathcal{D},s}^{RL} \leq |\mathcal{D}^s \cap \mathcal{D}^-| - TN_{\mathcal{D},s}^{\delta}$$

*Proof.* The lower bounds are an immediate consequence of the fact that all examples captured by $\delta$ are captured by $RL$'s prefix and have the same predictions that in $\delta$. Concerning the upper bounds, we show the proof for the first inequality as the second can be proven using a similar argument. Define $T$ as the set of examples in $\mathcal{D}^s \cap \mathcal{D}^+$ that are not determined by $\delta$. When constructing $RL$ from $\delta$, the maximum possible augmentation of true positives within protected group $s$ is to predict all the examples in $T$ correctly. The size of the set containing true positives of $\delta$ and $T$ is equal to $|\mathcal{D}^s \cap \mathcal{D}^+| - FN_{\mathcal{D},s}^{\delta}$. Hence the upper bound. □

As a consequence of Proposition 1, $W_{\mathcal{D}}^{RL} \geq W_{\mathcal{D}}^{\delta}$. We now define four integer decision variables that are used in our Integer Linear Programming (ILP) models. These variables are used to model the confusion matrix of any rule list whose prefix starts with $\delta$ as well as to define constraints modeling accuracy and fairness requirements over such matrix.

$$x^{TP_{\mathcal{D},p}} \in [TP_{\mathcal{D},p}^{\delta}, |\mathcal{D}^p \cap \mathcal{D}^+| - FN_{\mathcal{D},p}^{\delta}],\ x^{TP_{\mathcal{D},u}} \in [TP_{\mathcal{D},u}^{\delta}, |\mathcal{D}^u \cap \mathcal{D}^+| - FN_{\mathcal{D},u}^{\delta}],$$
$$x^{FP_{\mathcal{D},p}} \in [FP_{\mathcal{D},p}^{\delta}, |\mathcal{D}^p \cap \mathcal{D}^-| - TN_{\mathcal{D},p}^{\delta}],\ x^{FP_{\mathcal{D},u}} \in [FP_{\mathcal{D},u}^{\delta}, |\mathcal{D}^u \cap \mathcal{D}^-| - TN_{\mathcal{D},u}^{\delta}].$$

Let $L$ and $U$ be two integers such that $0 \leq L \leq U \leq |\mathcal{D}|$. $L$ (respectively, $U$) is a lower bound (respectively, upper bound) on the number of examples correctly classified by the rule list, as stated by the following constraint:

$$L \leq x^{TP_{\mathcal{D},p}} + x^{TP_{\mathcal{D},u}} + |\mathcal{D}^p \cap \mathcal{D}^-| - x^{FP_{\mathcal{D},p}} + |\mathcal{D}^u \cap \mathcal{D}^-| - x^{FP_{\mathcal{D},u}} \leq U. \tag{2.5}$$

We define $ILP(\delta, \mathcal{D}, L, U)$ to be the ILP model defined by the four variables $x^{TP_{\mathcal{D},p}}, x^{FP_{\mathcal{D},p}}, x^{TP_{\mathcal{D},u}}, x^{FP_{\mathcal{D},u}}$ and Constraint (2.5). Intuitively, if $ILP(\delta, \mathcal{D}, L, U)$ has no feasible solution, then one can guarantee that no rule list extending prefix $\delta$ can satisfy the provided lower and upper bounds on the number of well classified examples. This is demonstrated in Proposition 2.

**Proposition 2.** *Given a prefix $\delta$ and $0 \leq L \leq U \leq |\mathcal{D}|$, if $ILP(\delta, \mathcal{D}, L, U)$ is unsatisfiable then we have:*

$$\nexists RL \in \sigma(\delta) \mid L \leq W_{\mathcal{D}}^{RL} \leq U$$

*Proof.* Assume that there exists some $RL \in \sigma(\delta)$ such that $L \leq W_{\mathcal{D}}^{RL} \leq U$. Then, $x^{TP_{\mathcal{D},p}} = TP_{\mathcal{D},p}^{RL}$, $x^{TP_{\mathcal{D},u}} = TP_{\mathcal{D},u}^{RL}$, $x^{FP_{\mathcal{D},p}} = FP_{\mathcal{D},p}^{RL}$ and $x^{FP_{\mathcal{D},u}} = FP_{\mathcal{D},u}^{RL}$ is a solution to $ILP(\delta, \mathcal{D}, L, U)$. Indeed, Constraint (2.5) is satisfied by hypothesis, and the bounds of the four variables are respected due to Proposition 1 and the fact that $RL$ is an extension of $\delta$. Finally, if $\exists RL \in \sigma(\delta) \mid L \leq W_{\mathcal{D}}^{RL} \leq U$, then $ILP(\delta, \mathcal{D}, L, U)$ is satisfiable, which completes the proof by contrapositive. □

In the following paragraph, we show how the $ILP(\delta, \mathcal{D}, L, U)$ model can be extended to include the different considered statistical fairness metrics (defined in Table 2.1). We detail the procedure for the Statistical Parity metric and provide the key elements for the three other metrics, as the reasoning is identical. In particular, propositions similar to Proposition 3 can be adapted and proved for the three other metrics, following the same methodology.

#### 2.3.1.1 Integrating Statistical Parity

The constraint associated to the statistical parity fairness metric is provided within Table 2.1. To get rid of the denominators, we simply multiply both sides of the

inequality by a constant $C_1 = \varepsilon \times |\mathcal{D}^p| \times |\mathcal{D}^u|$ to obtain the following constraint:

$$-C_1 \leq |\mathcal{D}^u| \times (x^{TP_{\mathcal{D},p}} + x^{FP_{\mathcal{D},p}}) - |\mathcal{D}^p| \times (x^{TP_{\mathcal{D},u}} + x^{FP_{\mathcal{D},u}}) \leq C_1. \tag{2.6}$$

Let $ILP_{SP}(\delta, \mathcal{D}, L, U, \varepsilon)$ be the Integer Linear Programming model defined by the four variables $x^{TP_{\mathcal{D},p}}, x^{FP_{\mathcal{D},p}}, x^{TP_{\mathcal{D},u}}, x^{FP_{\mathcal{D},u}}$ and Constraints (2.5) and (2.6). If $ILP_{SP}(\delta, \mathcal{D}, L, U, \varepsilon)$ has no feasible solution, then one can guarantee that no rule list extending prefix $\delta$ can simultaneously satisfy the provided bounds on the number of well classified examples and the statistical parity fairness constraint. This is formalized in Proposition 3.

**Proposition 3.** *Given a prefix $\delta$, an unfairness tolerance $\varepsilon \in [0, 1]$, and $0 \leq L \leq U \leq |\mathcal{D}|$, if $ILP_{SP}(\delta, \mathcal{D}, L, U, \varepsilon)$ is unsatisfiable then we have:*

$$\nexists RL \in \sigma(\delta) \mid L \leq W^{RL}_{\mathcal{D}} \leq U \text{ and } \mathsf{unf}_{SP}(RL, \mathcal{D}) \leq \varepsilon$$

*Proof.* Assume that there exists some $RL \in \sigma(\delta)$ such that $L \leq W^{RL}_{\mathcal{D}} \leq U$ and $\mathsf{unf}_{SP}(RL, \mathcal{D}) \leq \varepsilon$. First, observe that Constraint (2.6) is equivalent to the mathematical formulation of the Statistical Parity condition defined in Table 2.1. Indeed, $\mathsf{unf}_{SP}(RL, \mathcal{D}) \leq \varepsilon$ if and only if $-C_1 \leq |\mathcal{D}^u| \times (TP^{RL}_{\mathcal{D},p} + FP^{RL}_{\mathcal{D},p}) - |\mathcal{D}^p| \times (TP^{RL}_{\mathcal{D},u} + FP^{RL}_{\mathcal{D},u}) \leq C_1$. Then, $x^{TP_{\mathcal{D},p}} = TP^{RL}_{\mathcal{D},p}$, $x^{TP_{\mathcal{D},u}} = TP^{RL}_{\mathcal{D},u}$, $x^{FP_{\mathcal{D},p}} = FP^{RL}_{\mathcal{D},p}$ and $x^{FP_{\mathcal{D},u}} = FP^{RL}_{\mathcal{D},u}$ is a solution to $ILP_{SP}(\delta, \mathcal{D}, L, U, \varepsilon)$. Finally, if $\exists RL \in \sigma(\delta) \mid L \leq W^{RL}_{\mathcal{D}} \leq U$ and $\mathsf{unf}_{SP}(RL, \mathcal{D}) \leq \varepsilon$, then $ILP_{SP}(\delta, \mathcal{D}, L, U, \varepsilon)$ is satisfiable, which completes the proof by contrapositive. □

#### 2.3.1.2 Integrating Other Statistical Fairness Metrics

Consider a prefix $\delta$, an unfairness tolerance $\varepsilon \in [0, 1]$ and bounds on the number of well classified examples $0 \leq L \leq U \leq |\mathcal{D}|$. As for the statistical parity metric, we define the following constants that will be useful to get rid of the constraints' denominators: $C_2 = \varepsilon \times |\mathcal{D}^u \cap \mathcal{D}^-| \times |\mathcal{D}^p \cap \mathcal{D}^-|$, and $C_3 = \varepsilon \times |\mathcal{D}^p \cap \mathcal{D}^+| \times |\mathcal{D}^u \cap \mathcal{D}^+|$.

**Predictive Equality.** Consider the following constraint:

$$-C_2 \leq |\mathcal{D}^u \cap \mathcal{D}^-| \times x^{FP_{\mathcal{D},p}} - |\mathcal{D}^p \cap \mathcal{D}^-| \times x^{FP_{\mathcal{D},u}} \leq C_2. \tag{2.7}$$

Let $ILP_{PE}(\delta, \mathcal{D}, L, U, \varepsilon)$ be the ILP model defined by the four variables $x^{TP_{\mathcal{D},p}}$, $x^{FP_{\mathcal{D},p}}$, $x^{TP_{\mathcal{D},u}}$, $x^{FP_{\mathcal{D},u}}$ and Constraints (2.5) and (2.7). If $ILP_{PE}(\delta, \mathcal{D}, L, U, \varepsilon)$ is unsatisfiable, then: $\nexists RL \in \sigma(\delta) \mid L \leq W^{RL}_{\mathcal{D}} \leq U$ and $\mathsf{unf}_{PE}(RL, \mathcal{D}) \leq \varepsilon$.

**Equal Opportunity.** Consider the following constraint:

$$-C_3 \leq |\mathcal{D}^p \cap \mathcal{D}^+| \times x^{TP_{\mathcal{D},u}} - |\mathcal{D}^u \cap \mathcal{D}^+| \times x^{TP_{\mathcal{D},p}} \leq C_3. \tag{2.8}$$

Let $ILP_{EOpp}(\delta, \mathcal{D}, L, U, \varepsilon)$ be the ILP model defined by the four variables $x^{TP_{\mathcal{D},p}}$, $x^{FP_{\mathcal{D},p}}$, $x^{TP_{\mathcal{D},u}}$, $x^{FP_{\mathcal{D},u}}$ and Constraints (2.5) and (2.8). If $ILP_{EOpp}(\delta, \mathcal{D}, L, U, \varepsilon)$ is

unsatisfiable, then: $\nexists RL \in \sigma(\delta) \mid L \leq W_{\mathcal{D}}^{RL} \leq U$ and $\mathtt{unf}_{EOpp}(RL, \mathcal{D}) \leq \varepsilon$.

**Equalized Odds.** Since the Equalized Odds metric is the conjunction of Equal Opportunity and Predictive Equality, we simply use the conjunction of both Constraints (2.7) and (2.8) to integrate it.

Let $ILP_{EO}(\delta, \mathcal{D}, L, U, \varepsilon)$ be the ILP model defined by the four variables $x^{TP_{\mathcal{D},p}}$, $x^{FP_{\mathcal{D},p}}$, $x^{TP_{\mathcal{D},u}}$, $x^{FP_{\mathcal{D},u}}$ and Constraints (2.5), (2.7) and (2.8). If $ILP_{EO}(\delta, \mathcal{D}, L, U, \varepsilon)$ is unsatisfiable then: $\nexists RL \in \sigma(\delta) \mid L \leq W_{\mathcal{D}}^{RL} \leq U$ and $\mathtt{unf}_{EO}(RL, \mathcal{D}) \leq \varepsilon$.

### 2.3.2 Integration within `FairCORELS`

We have proposed a sufficient condition to reject prefixes that do not respect a given fairness metric within a requirement of well-classified examples. One can use this property to reject prefixes before they are treated in the main loop of `FairCORELS`. This pruning idea can be integrated using two approaches.

The first one called the *eager* approach, checks the sufficient condition before adding an extension of a prefix to the priority queue (before Line 12 with $\delta \cup r$ being the prefix given in the ILP). The second approach called the *lazy* approach, checks the sufficient condition when a prefix is removed from the priority queue and passed the branch and bound lower bound test at Line 5 with $\delta$ being the prefix tested. If the corresponding ILP (called with valid bounds) is unsatisfiable, then the prefix $\delta$ being tested can safely be discarded since no rule list whose prefix starts with $\delta$ can satisfy the conjunction of fairness and well-classified examples requirements. The difference between the two approaches can be seen as the trade-off between memory consumption and computational time. Indeed, given the same inputs and exploration strategies, the *eager* approach consumes less memory than the *lazy* approach as it prunes prefixes before adding them to the queue. However, it requires more calls to the ILP solver.

Finally, we also consider using the ILP models to guide exploration. To realize this, we add an objective to the previously defined ILP, maximizing $x^{TP_{\mathcal{D},p}} - x^{FP_{\mathcal{D},p}} + x^{TP_{\mathcal{D},u}} - x^{FP_{\mathcal{D},u}}$. The ILP is then called as in the *eager* approach, just before adding an extension of a prefix to the priority queue (before Line 12). Whenever it is unsatisfiable, the corresponding prefix is pruned. However, when it is satisfiable, we additionally get the best accuracy reachable (*e.g.*, a lower bound on the objective function value) while also meeting the fairness constraint and improving the objective function. We use this value to order the priority queue $Q$ and define the *ILP-Guided* search heuristic. Intuitively, it guides the exploration towards the prefixes whose fairness may conflict least with accuracy (those with highest ILP objective function).

When building the ILP models, we use tight computations for the lower and upper bounds on the number of well-classified examples $L$ and $U$ used in Constraint (2.5). We detail such bounds' computations hereafter.

**Lower Bound.** Within the main loop of `FairCORELS`, we can compute the minimum number of examples that any extension of a prefix of length $k$ must correctly classify in order to improve over the current best known solution $RL^c$. This value depends on $k$ (as any extension of a prefix is at least as long as the prefix itself), as well as on $RL^c$'s length and number of well classified examples. More precisely, let $L(k, RL, \mathcal{D}) = |\mathcal{D}| \cdot (1 - (\mathsf{misc}(RL, \mathcal{D}) + \lambda \cdot (K_{RL} - k)))$. We demonstrate in Proposition 4 that this value is the minimum number of examples that any extension of a prefix of length $k$ must correctly classify in order to improve over rule list $RL$.

**Proposition 4.** *Consider a rule list $RL_2$. A rule list $RL_1 = (\delta_{RL_1}, q_0)$ has better objective value on $\mathcal{D}$ than $RL_2$ if and only if $W_{\mathcal{D}}^{RL_1} > L(|\delta_{RL_1}|, RL_2, \mathcal{D})$, in which $|\delta_{RL_1}|$ is the length of $RL_1$'s prefix.*

*Proof.* $\mathsf{obj}_{\texttt{CORELS}}(RL_1, \mathcal{D}) < \mathsf{obj}_{\texttt{CORELS}}(RL_2, \mathcal{D}) \iff \mathsf{misc}(RL_1, \mathcal{D}) + \lambda \cdot K_{RL_1} < \mathsf{misc}(RL_2, \mathcal{D}) + \lambda \cdot K_{RL_2}$
$\iff |\mathcal{D}| \cdot (1 - \mathsf{misc}(RL_1, \mathcal{D})) > |\mathcal{D}| \cdot (1 - (\mathsf{misc}(RL_2, \mathcal{D}) + \lambda \cdot (K_{RL_2} - |\delta_{RL_1}|)))$
$\iff W_{\mathcal{D}}^{RL_1} > L(|\delta_{RL_1}|, RL_2, \mathcal{D})$ □

Consider the prefix $\delta$ and the current best solution $RL^c$ of the main loop. Let $RL = (\delta_{RL}, q_0) \in \sigma(\delta)$. Using Proposition 4, we have $RL$ has a better objective value than $RL^c$ if and only if $W_{\mathcal{D}}^{RL} > L(|\delta_{RL}|, RL^c, \mathcal{D}) \geq L(|\delta|, RL^c, \mathcal{D})$ because $|\delta_{RL}| \geq |\delta|$ (as $RL \in \sigma(\delta)$). Therefore $L(|\delta|, RL^c, \mathcal{D})$ is a valid lower bound for the ILP, ensuring that rule list $RL$ improves over the current best objective value.

**Upper Bound.** We leverage two observations to compute a tight value $U(\delta, \mathcal{D})$ such that $\forall RL \in \sigma(\delta), W_{\mathcal{D}}^{RL} \leq U(\delta, \mathcal{D})$. First, the examples captured and misclassified by $\delta$ will always be misclassified for any $RL \in \sigma(\delta)$. Second, among the examples not captured by $\delta$, some may conflict (*i.e.*, have the same features vector associated with different labels) and can never be simultaneously predicted correctly. This computation corresponds to the Equivalent Points Bound of `CORELS` (described in details in Section 3.14 of [Angelino *et al.* 2018]).

## 2.4 Experimental Study

The purpose of this section is two-fold. First, after describing our experimental setup, we show the efficiency of the proposed pruning approaches using two biased datasets and the four considered fairness metrics of Table 2.1. Afterwards, we demonstrate the scalability of our method as well as its complementarity with a new prefix permutation map, using a larger real-world dataset.

### 2.4.1 Setup

We implement and solve the ILP models in C++ using the `ILOG CPLEX 20.10` solver[3], with an efficient memoisation mechanism. Sensitive features are used for

[3]Source code of this enhanced version of the `FairCORELS` Python package is available on `https://github.com/ferryjul/fairCORELSV2`. The use of the `CPLEX` solver is possible but not

measuring and mitigating unfairness but are not used for the model's inference in order to prevent disparate treatment [Zafar *et al.* 2017]. For each dataset, we generate 100 different training sets by randomly selecting 90% of the dataset's instances, with reported values being averaged over the 100 instances. Test values are measured on the remaining 10% instances for each random split. All experiments are run on a computing grid over a set of homogeneous nodes using Intel Xeon E5-2683 v4 Broadwell @ 2.1GHz CPU.

We use three exploration heuristics: a best-first search *ILP-Guided*, a best-first search guided by `CORELS`'s objective and a Breadth-First-Search (BFS). The former inherently comes with an *eager* pruning. For the latter two, we compare the *original* `FairCORELS` (no ILP pruning), as well as *lazy* and *eager* integrations of our pruning approach. Then, we evaluate the seven exploration settings. However, results for the three best-first searches guided by `CORELS`'s objective are omitted in this section because they consistently provided worst performances (considering all evaluated criteria) than the BFS with equivalent pruning integration. This can be explained by the fact that this approach guides exploration towards accurate solutions first, which conflicts with fairness in practice. In the Appendix B, we provide detailed results including the three best-first searches guided by `CORELS`'s objective.

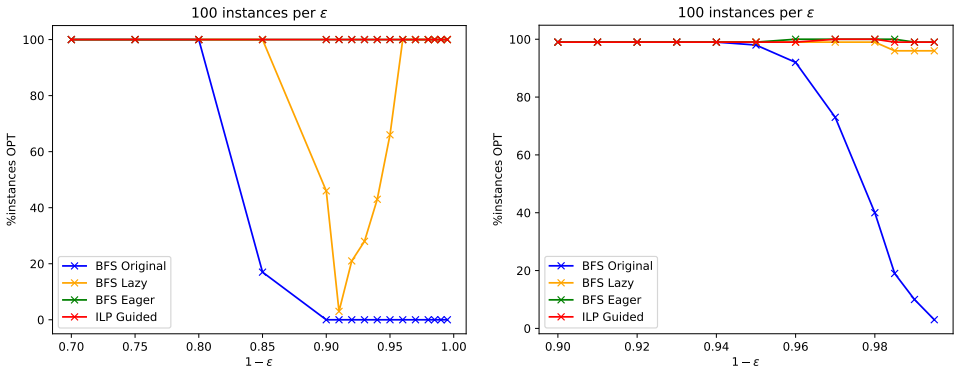
### 2.4.2 Evaluation of the Proposed ILP-based Pruning Approaches

To empirically assess the effectiveness of our proposed pruning on `FairCORELS`, we perform experiments for the four metrics of Table 2.1 using two well-known classification tasks of the literature with several fairness requirements. The first task consists in predicting which individuals from the COMPAS dataset [Angwin *et al.* 2016] will re-offend within two years. We consider race (African-American/Caucasian) as the sensitive feature. Features are binarized using one-hot encoding for categorical ones and quantiles (with 5 bins) for numerical ones. Rules are generated as single features without minimum support. The resulting preprocessed dataset contains 18 rules and 6150 examples.

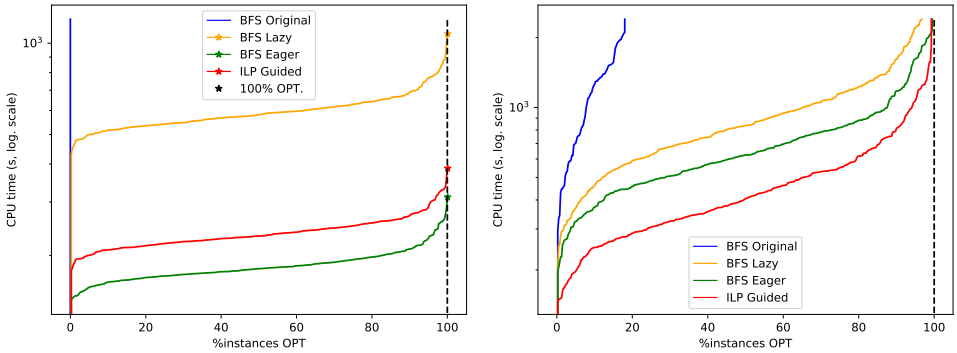
The second task consists in predicting whether individuals from the German Credit dataset [Dua & Graff 2017] have a good or bad credit score. We consider age (low/high) as the sensitive feature, with both groups separated by the median value. Features are binarized using one-hot encoding for categorical ones and quantiles (2 bins) for numerical ones. Rules are generated as single features with minimum support of 0.25 or conjunctions of two features with minimum support of 0.5. Gender-related features were excluded. The resulting preprocessed dataset contains 49 rules and 1000 examples. For experiments on the COMPAS (respectively German Credit) dataset, the maximum running time is set to 20 minutes (respectively 40 minutes). For each experiment, the maximum memory use is fixed to 4 Gb. We detail our evaluation for the Statistical Parity metric, as results for

mandatory, as our released code also embeds an open-source solver (whose configuration has been tuned to handle our pruning problem efficiently). This solver is Mistral-2.0 [Hebrard 2008, Hebrard & Siala 2017b], in its version used for the Minizinc Challenge 2020.

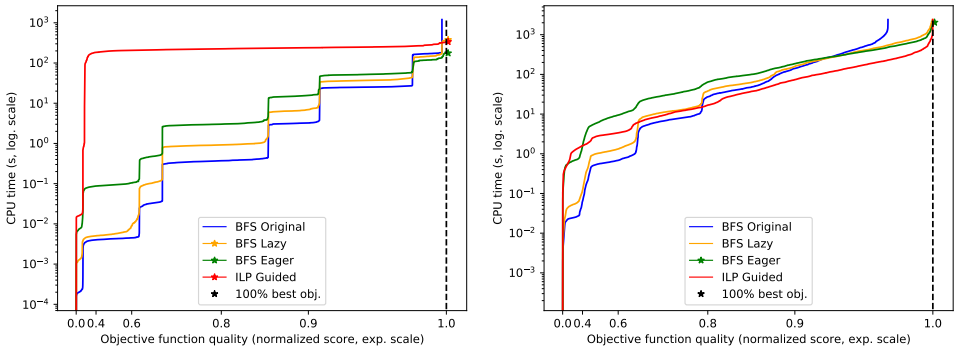
all other metrics show similar trends. Detailed results for all considered metrics are provided within the Appendix B.




(a) Proportion of instances solved to optimality as a function of $1 - \varepsilon$.




(b) CPU time as a function of the proportion of instances solved to optimality.




(c) Solving time as a function of the objective function quality normalized score.

Figure 2.3: Experimental evaluation of our pruning strategies for `FairCORELS` (left: COMPAS dataset, right: German Credit dataset) for the Statistical Parity metric.

Figure 2.3a displays the proportion of instances solved to optimality as a function of the fairness requirement (which gets harder as $1 - \varepsilon$ increases) to illustrate the joint action of `CORELS`' bounds and the proposed ILP-based pruning. For low fairness requirements, all evaluated methods reach optimality, thanks to the ac-

tion of `CORELS`' bounds. However, these bounds are less effective for strong fairness requirements, and without the ILP pruning, optimality can hardly be reached. Conversely, the higher the value of $1 - \varepsilon$, the larger the pruning of the search space. Hence, optimality is reached most of the time when performing an eager pruning (*eager* BFS or *ILP-Guided*). This joint effect is particularly visible with the *lazy* BFS approach on the COMPAS dataset. Interestingly, we observe on both datasets a threshold effect: when the fairness constraints become active and effectively modify the set of feasible solutions (*i.e.,* above a certain value for $1 - \varepsilon$), the original `FairCORELS` struggles to reach and prove optimality. As expected, this effect is mitigated by our pruning strategies and especially the *eager* ones (*eager* BFS and *ILP-Guided*).

Figures 2.3b and 2.3c are generated using high fairness requirements (unfairness tolerances ranging between 0.005 and 0.02). Indeed, this corresponds to a regime where the original `FairCORELS` struggles to reach and prove optimality for both datasets, as observed previously in Figure 2.3a. Hence, this is precisely in this regime that we aim at improving the exploration of the search space. Figure 2.3b presents the solving time as a function of the proportion of instances solved to optimality (lower is better). It shows a clear dominance of the proposed pruning approaches. For COMPAS, the *original* `FairCORELS` does not prove optimality to any of the instances, whereas all pruning methodologies prove optimality to all instances. For German Credit, similar trends are observed. Overall, the *eager* approach appears more suitable to prove optimality, as it keeps the size of the queue as small as possible. For experiments with German Credit, the *ILP-Guided* approach effectively speeds up convergence and proof of optimality by guiding exploration towards fair and accurate solutions. This is not the case when using COMPAS, but the approach is still able to reach the best solutions, thanks to the performed pruning. Figure 2.3c shows the learning time as a function of the objective function quality (normalized objective score proposed in [Hebrard & Siala 2017a]). The proposed pruning allows finding better solutions within the time and memory limits after a slow start. Indeed, the pruning slows the beginning of the exploration, but pays off, given enough time, by effectively limiting the growth of the priority queue. The *lazy* approach is faster than the *eager* one at the beginning of the exploration. However, this trend is inverted given sufficient time. Again, the *ILP-Guided* approach speeds up convergence on German Credit, but worsens it on COMPAS.

Finally, the reported results illustrate the efficiency of the proposed pruning approaches to speed up the exploration of the prefix tree. The *lazy* approach less slows exploration at the beginning, but the *eager* approach gives better results given sufficient time. The *ILP-Guided* strategy showed an ability to speed up convergence, but its performances depend on the problem at hand.

In Appendix B, we provide detailed results regarding these experiments. First, we provide in Figure B.1 a detailed version of Figure 2.3, additionally including the three approaches guided by `CORELS`'s original objective (which are omitted here because they consistently perform worst than the BFS-based ones). We also provide detailed results for all the other considered fairness metrics in Figures B.2, B.3, and

B.4.

Test results are reported in Table 2.2, and suggest that building optimal models does not result in worsening accuracy nor fairness generalization. More precisely, we report, for each dataset and each fairness metric, the relative number of runs (*i.e.*, for the different values of $\varepsilon$) for which each pruning approach led to the best training (respectively test) accuracy. We also report average violation of the fairness constraint at test time.

Table 2.2: Learning quality evaluation using different pruning strategies for `FairCORELS` (unfairness tolerances ranging between 0.005 and 0.05). We report the proportion of instances for which each method led to the best train (resp. test) accuracy, and the average violation of the fairness constraint at test time. For each experiment, best results are shown in **bold**.

| | **BFS Original** | | | **BFS Lazy** | | | **BFS Eager** | | | **ILP Guided** | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | %Best | | Test | %Best | | Test | %Best | | Test | %Best | | Test |
| **UNF** | Train Acc | Test Acc | Unf Viol. | Train Acc | Test Acc | Unf Viol. | Train Acc | Test Acc | Unf Viol. | Train Acc | Test Acc | Unf Viol. |
| **COMPAS dataset** | | | | | | | | | | | | |
| SP | .951 | .971 | **.009** | **1** | .98 | **.009** | **1** | **.981** | **.009** | **1** | .98 | **.009** |
| PE | .927 | .956 | **.033** | **1** | **.977** | .034 | **1** | **.977** | .034 | **1** | **.977** | .034 |
| EOpp | .941 | .961 | **.03** | **1** | .98 | .031 | **1** | **.983** | .031 | **1** | **.983** | .031 |
| EO | .897 | .934 | **.035** | .997 | .974 | .036 | **1** | **.976** | .036 | **1** | .974 | .036 |
| **German Credit dataset** | | | | | | | | | | | | |
| SP | .567 | **.799** | **.045** | .994 | .77 | **.045** | **.999** | .783 | **.045** | .996 | .779 | **.045** |
| PE | .967 | .914 | .138 | **1** | .914 | **.137** | **1** | .914 | .138 | .997 | **.927** | .138 |
| EOpp | .683 | .816 | .056 | .99 | .799 | .055 | **1** | .806 | .055 | .991 | **.829** | **.054** |
| EO | .52 | .759 | **.158** | .979 | .751 | .161 | .997 | .741 | .16 | **1** | **.771** | .159 |

### 2.4.3 Scalability and Complementarity with the Permutation Map

As discussed in Section 2.2.3, a prefix permutation map speeds up the `CORELS` algorithm by leveraging symmetries. This structure ensures that only the best permutation of a given set of rules implying the same antecedents is stored in the priority queue and further explored. By doing so, it avoids exploring provably sub-optimal parts of the search space. However, `CORELS`'s original permutation map cannot be used within `FairCORELS` as optimality would no longer be guaranteed. This is due to the fact that a given permutation of rules may allow for better objective function values than others but may not lead to solutions meeting the fairness requirement.

In this section, we modify `CORELS`'s prefix permutation map to enforce a weaker symmetry-breaking mechanism while maintaining the guarantee of optimality. More precisely, the proposed new prefix permutation map (PMAP) considers that two prefixes of equal length are equivalent if and only if they have exactly the same confusion matrix and their rules imply the same antecedents. Note that two prefixes may not correspond to the same classification function, but still meet this requirement. Because equivalent prefixes have the same confusion matrix and capture the same examples (because they use the same antecedents), we only need to keep one

Table 2.3: Learning quality evaluation using different pruning strategies for `FairCORELS` (Adult Income dataset, unfairness tolerances ranging between 0.005 and 0.1). We report the proportion of instances for which each method led to the best train (resp. test) accuracy, and the average violation of the fairness constraint at test time. For each experiment, best results are shown in **bold**.

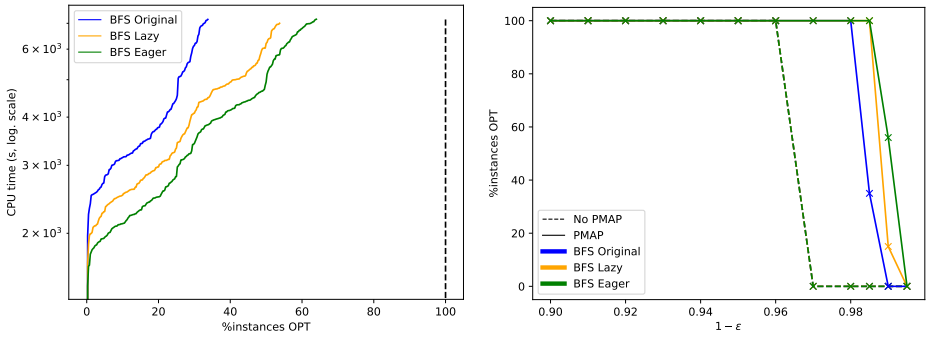
| | **BFS Original** | | | **BFS Lazy** | | | **BFS Eager** | | |
|---|---|---|---|---|---|---|---|---|---|
| | %Best | | Test | %Best | | Test | %Best | | Test |
| **Map type** | Train Acc | Test Acc | Unf Viol. | Train Acc | Test Acc | Unf Viol. | Train Acc | Test Acc | Unf Viol. |
| **All $\varepsilon$ values** | | | | | | | | | |
| No PMAP | .938 | .942 | **-.004** | .963 | .966 | **-.004** | .964 | .967 | **-.004** |
| PMAP | .966 | .97 | **-.004** | .998 | .987 | **-.004** | **1** | **.989** | **-.004** |
| $\varepsilon < 0.02$ | | | | | | | | | |
| No PMAP | .815 | .835 | **.0** | .89 | .907 | .001 | .892 | .91 | .001 |
| PMAP | .897 | .91 | .001 | .993 | .96 | .001 | **1** | **.968** | .001 |

such prefix in the priority queue. The new PMAP is then implemented modifying `CORELS`'s original one, and pushes a new prefix to the priority queue $Q$ (Line 12 of Algorithm 1) only if $Q$ contains no equivalent prefix.

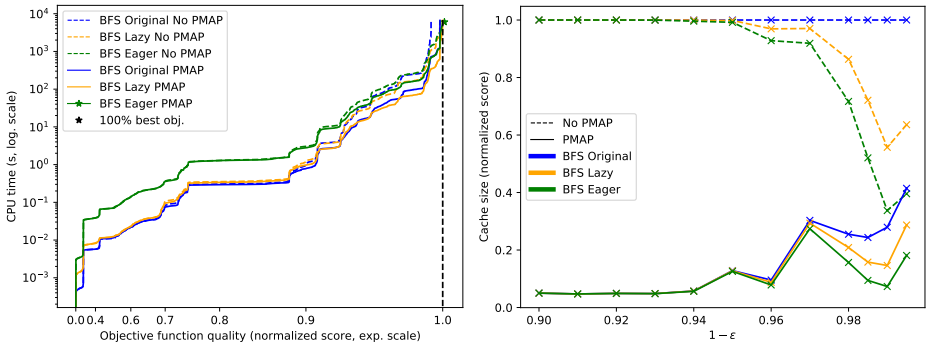
To evaluate the scalability of our pruning approaches, we consider Adult Income [Dua & Graff 2017], a larger dataset that gathers records of individuals from the 1994 U.S. census. We consider the task of predicting whether an individual earns more than $50,000$\$ per year, with gender (male/female) being the sensitive attribute. Categorical attributes are one-hot encoded and numerical ones are discretized using quantiles (3 bins). The resulting dataset contains $48,842$ examples and 47 rules (attributes or their negation), with a minimum support of 0.05[4]. We consider only the Statistical Parity metric, as the three others do not conflict strongly with accuracy in this setting as observed in Figure 1(a) of [Aïvodji *et al.* 2019b]. Experiments are performed with and without the new PMAP. The maximum running time is set to two hours, with a maximum memory use of 8 Gb. Results for the *ILP-Guided* approach are excluded here (but provided within Figure B.5 in the Appendix B) as they show no clear improvement over the *eager* pruning, suggesting that the guidance was not beneficial overall.

Results are summarized in Figure 2.4. The left plot of Figure 2.4a shows the proportion of instances solved to optimality, for strong fairness requirements (unfairness tolerances ranging between 0.005 and 0.02). For these strong fairness requirements, the approaches not using the new PMAP were never able to prove optimality (as can be seen in the right plot) and are not represented. The complementarity with our pruning approach is particularly visible, with the methods using

[4]While we could have considered a larger dataset (both in terms of features and examples), this version of Adult Income is empirically sufficient to challenge our algorithms. Indeed, as can be seen in Figure 2.4a (left plot), even the most efficient pruning techniques along with the new permutation map fail to prove optimality for roughly 30% of the instances. Getting closer to the limits of our methods using larger datasets (without necessarily requiring optimality guarantees) remains an interesting direction.




(a) Left: CPU time as a function of the proportion of instances solved to optimality. Right: proportion of instances solved to optimality as a function of $1 - \varepsilon$.




(b) Left: CPU time as a function of the objective function score. Right: relative cache size as a function of $1 - \varepsilon$.

Figure 2.4: Experimental evaluation of our pruning strategies for `FairCORELS` on the Adult Income dataset.

both the PMAP and the ILP pruning having the best performances, both in terms of objective function quality (Figure 2.4b, left plot) and proof of optimality. This is also observed in terms of memory use in Figure 2.4b (right plot). Indeed, the PMAP considerably reduces the size of the queue, leveraging the prefix tree symmetries. However, its effect is weakened for strong fairness constraints. The use of the ILP pruning mitigates this trend and for very strong fairness requirements, the *eager* pruning alone proposes lower memory consumption than the PMAP alone, to reach the same solutions. Finally, learning quality results are provided in Table 2.3 and confirm these observations. More precisely, they consistently show that the approaches improving train accuracy also improve test accuracy, without impacting fairness violation.

## 2.5 Improving Fairness Generalization

While experimenting with `FairCORELS` and our ILP-based pruning method, we observed, as can be seen in Tables 2.2 and 2.3, that a model meeting some fairness

constraint on its training set may still exhibit an unfairness violation greater than zero when applied on a separate test set. While this violation can remain relatively small as in the experiments using the UCI Adult Income dataset (Table 2.3), it may also reach non-negligible values, as observed in Table 2.2. Indeed, as already noted in Section 1.3.6, fairness generalization is an open issue. As a motivational example, we report in Figure 2.5 the training and test Pareto frontiers approximations (sets of trade-offs between error and unfairness) we obtained using `FairCORELS` on the Adult Income dataset for the Equal Opportunity fairness metric. As one can observe, unfairness does not generalize well, and while the learnt rule lists provide an interesting Pareto frontier approximation at training time, the error/unfairness trade-offs significantly degrade when applied to unseen data. This observation motivated our formulation of a sample-based robustness framework for fairness, which we briefly summarize in this section.

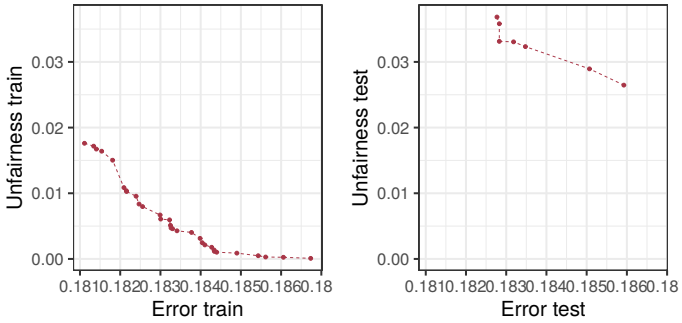



Figure 2.5: Pareto frontier approximations (trade-offs between error and unfairness on both the training and test sets, averaged using 5-folds cross-validation) obtained using `FairCORELS` on the Adult Income dataset, for the equal opportunity fairness metric (*cf.* Table 2.1).

We first introduce Distributionally Robust Optimization, which inspired our method. We then discuss the connection of our proposed approach with the literature. We present our sample-based robustness framework for statistical fairness, and derive a heuristic application of it. Finally, we provide some experimental results demonstrating the effectiveness of our method to improve fairness generalization.

### 2.5.1 Distributionally Robust Optimization

As stated in Section 1.1.1, an important challenge in machine learning is that we usually do not know the true underlying distribution $\mathcal{P}$. Instead, we often have access to a limited training set $\mathcal{D}$, whose distribution $\mathcal{P}'$ may differ from $\mathcal{P}$. To take into account this uncertainty, Distributionally Robust Optimization (DRO) techniques can be leveraged. Instead of minimizing an objective function $\mathsf{obj}$ for a given distribution $\mathcal{P}'$, DRO [Ben-Tal *et al.* 2013, Duchi *et al.* 2021, Rahimian & Mehrotra 2019] consists in minimizing $\mathsf{obj}$ for a worst-case distribu-

tion, among a set of perturbed versions of $\mathcal{P}'$ [Sagawa *et al.* 2020]. More precisely, the objective is to build a model $h$ minimizing $\mathsf{obj}$ for a set of neighbouring distributions of $\mathcal{P}'$. Such neighbouring distributions are contained in a *perturbation set* (also called *ambiguity set*) $\mathcal{B}(\mathcal{P}')$. In the DRO setting, the supervised machine learning problem becomes:

$$\arg\min_{h\in\mathcal{H}} \max_{\mathcal{Q}\in\mathcal{B}(\mathcal{P}')} \mathsf{obj}(h,\mathcal{Q}). \tag{2.9}$$

Distributionally Robust Optimization has been used in many different domains [Rahimian & Mehrotra 2019], and has been applied widely in machine learning [Kang 2017].

### 2.5.2 Related Works on Improving Fairness Generalization

Recent work on fairness generalization (reviewed in Section 1.3.6) targets integrating different techniques for improving robustness into existing fair learning algorithms. While such methods have been shown (theoretically and empirically) to improve fairness generalization, they often induce a considerable computational overhead (*e.g.*, solving an additional problem to determine a worst-case unfairness [Mandal *et al.* 2020]), and thus have limited scalability. Some methods do not suffer from this drawback but instead require additional splitting of the data [Cotter *et al.* 2018, Cotter *et al.* 2019], hence possibly penalizing utility, as the amount of data used to update the model is reduced. Finally, other approaches have limited applicability, as they are designed for a particular algorithm or hypothesis class [Taskesen *et al.* 2020, Wang *et al.* 2021], or require some special property of the underlying algorithm (*e.g.*, access to a cost-sensitive classification oracle [Mandal *et al.* 2020]).

To tackle these issues, we propose a new framework for statistical fairness robustness. Intuitively, our approach consists in ensuring fairness over a variety of samplings of the training set. The approach that is the more closely related to ours is that of [Mandal *et al.* 2020], which is based on a similar intuition, namely ensuring fairness on a set of neighbouring distributions of the training set, called re-weightings versions, can improve its generalization. However, we consider different definitions for such neighbouring distributions and in addition we propose a heuristic approach variant exhibiting practical advantages compared to the exact one. The work of [Huang & Vishnoi 2019] (which adds a stability-based regularizer to the objective function) is also related to ours, as we seek to improve fairness robustness on samplings of the training set (which can be viewed as a form of training fairness stability). We elaborate on our proposed theoretical framework in the next subsection.

### 2.5.3 Sample-based Robustness for Statistical Fairness

Following the principles of DRO, it has been shown that enforcing fairness over a set of distributions that are neighbours to the training one is an efficient way to improve its generalization [Mandal *et al.* 2020, Sagawa *et al.* 2020, Taskesen *et al.* 2020]. While DRO was formalized using distributions, practical machine learning applications usually deal with finite training sets that are sampled from an underlying distribution. Indeed, instead of considering fairness robustness over perturbed underlying distributions (which, in practice, are unknown), we enforce robustness with respect to the training set sampling. For this reason, we propose to use the Jaccard distance $J$ as the distance metric measuring similarity between sample sets.

**Definition 2.** ***(Jaccard distance)*** *Let $\mathcal{D}_1$ and $\mathcal{D}_2$ be two sample sets. The Jaccard distance between $\mathcal{D}_1$ and $\mathcal{D}_2$ is defined as follows:* $J(\mathcal{D}_1, \mathcal{D}_2) = 1 - \frac{|\mathcal{D}_1 \cap \mathcal{D}_2|}{|\mathcal{D}_1 \cup \mathcal{D}_2|}$

The Jaccard distance is a very popular measure, used to quantify (dis)similarity between sample sets in a wide range of applications. For example, it has been used in Machine Learning for feature ranking stability [Khoshgoftaar *et al.* 2013, Saeys *et al.* 2008] and feature selection [Zou *et al.* 2016]. Intuitively, two sample sets $\mathcal{D}_1$ and $\mathcal{D}_2$ that have a large intersection are close (*i.e.*, $J(\mathcal{D}_1, \mathcal{D}_2)$ is small and in particular $J(\mathcal{D}, \mathcal{D}) = 0$) while two sample sets $\mathcal{D}_3$ and $\mathcal{D}_4$ with empty intersection are far from each other (*i.e.*, $J(\mathcal{D}_3, \mathcal{D}_4)$ is 1). We use the Jaccard distance to define the perturbation sets of a given dataset $\mathcal{D}$.

**Definition 3.** ***(Perturbation sets)*** *Let $\tau \in [0, 1]$, we define a perturbation set $\mathcal{B}(\mathcal{D}, \tau)$ as the set of subsets of $\mathcal{D}$ whose Jaccard distance from $\mathcal{D}$ is less than or equal to $\tau$. That is, $\mathcal{B}(\mathcal{D}, \tau) = \{\mathcal{D}' \mid J(\mathcal{D}, \mathcal{D}') \leq \tau \wedge (\mathcal{D}' \subseteq \mathcal{D})\}$.*

Definition 3 states that $\mathcal{B}(\mathcal{D}, \tau)$ contains all subsets of $\mathcal{D}$ of size at least $|\mathcal{D}| \times (1 - \tau)$. A special case arises if $\tau = 0$, as $\mathcal{B}(\mathcal{D}, 0)$ simply contains $\mathcal{D}$ itself. In a nutshell, the subsets of $\mathcal{D}$ contained in $\mathcal{B}(\mathcal{D}, \tau)$ can be seen as points in a metric space equipped with the Jaccard distance[5], contained within a ball centered around $\mathcal{D}$ whose radius is $\tau$. This ball is itself contained within all sets $\mathcal{B}(\mathcal{D}, \tau')$ with $\tau' \geq \tau$. We illustrate this nested structure in Figure 2.6, on a toy dataset with two protected groups for the statistical fairness measure.

Similar to DRO, the proposed approach consists in ensuring a given property (*e.g.*, fairness) over a set of elements contained in a perturbation set. For DRO, such elements are distributions while we rather consider sample sets. By considering the perturbation set $\mathcal{B}(\mathcal{D}, \tau)$ as a set of samplings of the dataset $\mathcal{D}$, we aim at building a model that is fair on all sets of $\mathcal{B}(\mathcal{D}, \tau)$, including $\mathcal{D}$ itself. This leads to the formulation of our sample-robust version of Problem (1.3). More precisely, the sample-robust fair learning problem on a perturbation set $\mathcal{B}(\mathcal{D}, \tau)$ is provided hereafter in Problem (2.10). An optimal solution to this problem corresponds to

[5]The Jaccard distance satisfies all required properties to equip a metric space, and in particular the triangle inequality [Kosub 2019].

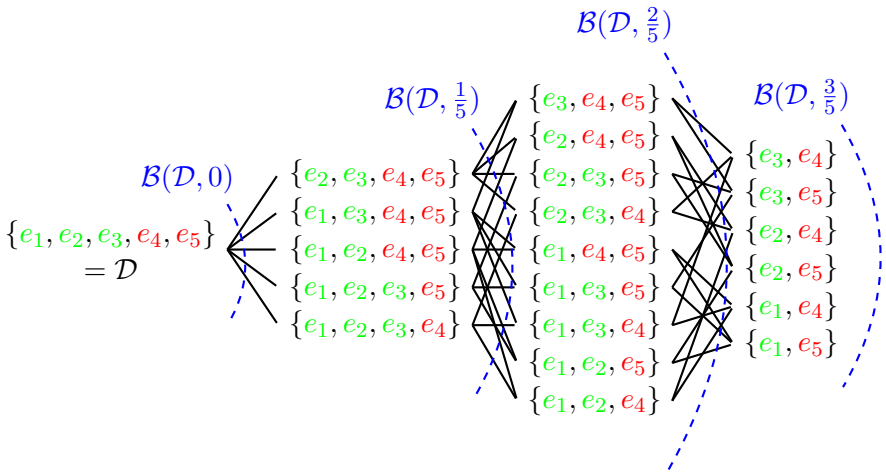



Figure 2.6: Example of perturbation sets for a dataset $\mathcal{D}$ with 5 examples and two protected groups p ($\{e_1, e_2, e_3\}$) and u ($\{e_4, e_5\}$). Subsets that can not be used to audit a model's fairness with respect to protected groups p and u (*i.e.,* datasets for which the unfairness metric is undefined because they do not contain at least one example from each protected group) are not represented.

a model $h$ that minimizes the objective function $\mathsf{obj}$ on $\mathcal{D}$, among those of $\mathcal{H}$ that exhibit unfairness at most $\varepsilon$ over all sets contained in $\mathcal{B}(\mathcal{D}, \tau)$, including $\mathcal{D}$ itself.

$$\begin{aligned} \underset{h \in \mathcal{H}}{\arg\min} \quad & \mathsf{obj}(h, \mathcal{D}) \\ \text{s.t.} \quad & \max_{\mathcal{D}' \in \mathcal{B}(\mathcal{D}, \tau)} \mathsf{unf}(h, \mathcal{D}') \leq \varepsilon \end{aligned} \tag{2.10}$$

This formulation is a particular instantiation of the general DRO formulation of Problem (2.9), in which robustness is applied only on the enforced fairness constraints rather than on the objective function. With the proposed perturbation sets definition, we observe that augmenting the distance $\tau$ increases the number of subsets being considered. As a consequence, considering higher values of $\tau$ can only raise the worst-case fairness violation, thus hardening the problem. Hence, the parameter $\tau$ directly controls the strength of the enforced robustness of the fairness constraint. We further study the structure of our proposed perturbation sets and the consequences in terms of fairness violation in our published work. Hereafter, we show how one can ensure that fairness is satisfied within all sample sets contained by $\mathcal{B}(\mathcal{D}, \tau)$ without enumerating them all, which could be costly. To this end, we introduce in the following definition the notion of fairness sample-robustness.

**Definition 4.** *(**Quantifying sample-robustness for fairness**) Consider a dataset $\mathcal{D}$, a classifier $h$ and an acceptable unfairness tolerance $\varepsilon$. The unfairness sample-robustness of $h$ on $\mathcal{D}$ for constraint $\varepsilon$, denoted by $\mathcal{SR}(h, \mathcal{D}, \varepsilon)$, is the Jaccard distance ($\mathcal{SR}(h, \mathcal{D}, \varepsilon) \in [0, 1]$) such that:*

1. $\forall \tau \geq \mathcal{SR}(h, \mathcal{D}, \varepsilon), \exists \mathcal{D}' \in \mathcal{B}(\mathcal{D}, \tau)$ *such that* $\mathsf{unf}(h, \mathcal{D}') > \varepsilon$.

2. $\forall \tau < \mathcal{SR}(h, \mathcal{D}, \varepsilon), \forall \mathcal{D}' \in \mathcal{B}(\mathcal{D}, \tau), \mathsf{unf}(h, \mathcal{D}') \leq \varepsilon$.

*In other words,* $\mathcal{SR}(h, \mathcal{D}, \varepsilon)$ *is the largest possible value of the Jaccard distance* $\tau$ *such that* $h$ *is fair over all sets in* $\mathcal{B}(\mathcal{D}, \tau'), \forall \tau' < \tau$.

Again, consider that $\mathcal{D}$ and all its subsets are points into a metric space equipped with the Jaccard distance. Intuitively, $\mathcal{SR}(h, \mathcal{D}, \varepsilon)$ is the radius of the largest ball centered around $\mathcal{D}$ such that $h$ if fair over all sample sets strictly contained within this ball. In simple words, $h$ is fair on $\mathcal{D}$ and on subsets of $\mathcal{D}$ up to a (Jaccard) distance of $\mathcal{SR}(h, \mathcal{D}, \varepsilon)$. The bigger $\mathcal{SR}(h, \mathcal{D}, \varepsilon)$, the more sample-robust $h$'s fairness is.

**Computing $\mathcal{SR}(h, \mathcal{D}, \varepsilon)$ in practice.** In our published work, we propose an integer programming model, coined $\mathcal{IPSR}(h, \mathcal{D}, \varepsilon)$, and demonstrate that it can be used to compute the exact value of $\mathcal{SR}(h, \mathcal{D}, \varepsilon)$. Furthermore, we also show that a simple, linear-time (with respect to the number of examples) greedy algorithm can be used to upper-bound $\mathcal{SR}(h, \mathcal{D}, \varepsilon)$. $\mathcal{IPSR}(h, \mathcal{D}, \varepsilon)$ is introduced and described within the Appendix C.

Finally, we can formulate the sample-robust fair learning problem as a multi-objective problem, using an $\varepsilon$-constraint method. In other words, considering the fair learning problem (1.3), we include our fairness sample-robustness term as a constraint:

$$\begin{aligned} \underset{h \in \mathcal{H}}{\arg\min} \quad & \mathsf{obj}(h, \mathcal{D}) \\ \text{s.t.} \quad & \mathsf{unf}(h, \mathcal{D}) \leq \varepsilon \\ & \mathcal{SR}(h, \mathcal{D}, \varepsilon) > \mu \end{aligned} \tag{2.11}$$

Note that Problem (2.11) is indeed equivalent to Problem (2.10), reformulated to use the fairness sample-robustness quantification notion introduced in Definition 4 (*i.e.*, $\mathcal{SR}(h, \mathcal{D}, \varepsilon)$ and the discussed tools to measure it). In particular, the $\mu$ parameter of Problem (2.11) corresponds to the $\tau$ parameter of Problem (2.10). One can observe that we have to enforce that $\mathcal{SR}(h, \mathcal{D}, \varepsilon)$ is strictly greater than $\mu$ to ensure that $h$ meets the fairness constraint over all subsets of $\mathcal{D}$ up to a Jaccard distance $\mu$ (including $\mu$ itself). Indeed, as stated in Definition 4, if $\mathcal{SR}(h, \mathcal{D}, \varepsilon) = \mu$, then $h$ is fair over all subsets of $\mathcal{D}$ up to Jaccard distance $\mu$, *excluding* $\mu$.

An important difficulty with Problem (2.11) is the calibration of the $\mu$ parameter. More precisely, as a meaningful value of $\mu$ depends on the dataset at hand, on the considered sensitive attributes, on the unfairness metric and on the unfairness constraint $\varepsilon$, determining a good value for $\mu$ is difficult. For this reason, we propose to build a Pareto frontier between utility ($\mathsf{obj}(h, \mathcal{D})$) and fairness sample-robustness ($\mathcal{SR}(h, \mathcal{D}, \varepsilon)$), for a fixed value of $\varepsilon$. To realize this, we first solve Problem (2.11) with no constraint on $\mathcal{SR}(h, \mathcal{D}, \varepsilon)$ (*i.e.,* $\mu < 0$) to obtain the baseline fair (non

robust) model $h_0$. Then, we measure $h_0$'s fairness sample-robustness $\mathcal{SR}(h_0, \mathcal{D}, \varepsilon)$ and solve Problem (2.11) again, using this value for the $\mu$ parameter in order to (strictly) strengthen the fairness sample-robustness constraint. We iterate this process until a given stopping criterion is met. In our experiments, we consider different stopping conditions, with some of them leveraging a separate validation set.

However, practical difficulties remain, such as an important computational overhead and practical integration challenges (solving a MIP within a learning algorithm). These challenges motivate a heuristic formulation of the problem.

### 2.5.4 Heuristic Formulation

We have showed that an exact application of our proposed formulation is possible, but challenging. Indeed, in practice, a heuristic application of our proposed principle can be beneficial, even if no formal guarantees hold. The approach we propose consists in computing $n$ random subsets of the training set using $n$ random binary masks. Each mask $\mathcal{M}$ is a vector of size $N$, in which each coordinate $\mathcal{M}_{j \in \{1..N\}} \in \{0, 1\}$ is a random binary value. We denote by $\mathcal{D}$ the subset associated with mask $\mathcal{M}$ as follows: $\mathcal{D} = \{e_j \in \mathcal{D} \mid \mathcal{M}_j = 1\}$. This is used in Definition 5 to define the heuristic perturbation set.

**Definition 5.** *(**Heuristic perturbation sets**) Consider a dataset $\mathcal{D}$ and a set of $n$ binary masks $\mathcal{M}_1 \dots \mathcal{M}_n$ of size $|\mathcal{D}|$. The heuristic perturbation set, denoted by $\mathcal{B}_\omega(\mathcal{D}, n)$, is defined as: $\mathcal{B}_\omega(\mathcal{D}, n) = \{\mathcal{D}, \mathcal{D}_1, \mathcal{D}_2, \dots \mathcal{D}_n\}$.*

In a nutshell, instead of considering the entire previously defined perturbation set $\mathcal{B}(\mathcal{D}, \tau)$, we only enforce fairness on some randomly generated subsets (belonging to $\mathcal{B}(\mathcal{D}, \tau)$ by construction). Intuitively, $\mathcal{B}(\mathcal{D}, \tau)$ considers all subsets of $\mathcal{D}$ whose Jaccard distance from $\mathcal{D}$ is at most $\tau$. In contrast, $\mathcal{B}_\omega(\mathcal{D}, n)$ only considers $n$ random subsets of $\mathcal{D}$ (along with $\mathcal{D}$ itself). In the graph representation of Figure 2.6, our heuristic perturbation sets contain randomly selected vertices.

By replacing $\mathcal{B}(\mathcal{D}, \tau)$ by $\mathcal{B}_\omega(\mathcal{D}, n)$ in Problem (2.10), we get the heuristic formulation of our sample-robust fair learning problem. The intuition behind this heuristic approach is that the randomly sampled subsets of $\mathcal{D}$ have slightly different distributions. Hence, enforcing fairness for such subsets effectively leads to a form of heuristic distributionally robust optimization. It is possible to draw a parallel with the Bagging (Bootstrap AGGregatING) ensemble learning method [Zhou 2012]. Indeed, the idea underlying bagging is that training different models using different samplings of the training set may improve robustness by reducing the variance. This happens because such samplings have slightly different distributions, neighbouring the original one. While bagging leverages the different samplings to learn a set of models that will reduce the variance of the accuracy, we use them to enforce fairness in a robust manner.

This heuristic formulation does not have the theoretical appeal of our exact sample-robustness quantification framework, but exhibits considerable practical advantages. Indeed, it does not require calibrating the $\mu$ parameter of Problem (2.11),

which may require a separate validation set. In addition, computing unfairness over a finite set of subsets defined with masks can be done in linear time with respect to the input size, which is considerably simpler than solving $\mathcal{IPSR}(h, \mathcal{D}, \varepsilon)$. It is also easier to integrate within existing algorithms (and in particular, gradient-based techniques).

### 2.5.5 Some Experimental Results

We conducted a large empirical evaluation of our exact (Section 2.5.3) and heuristic (Section 2.5.4) approaches in our published work. In this subsection, we only report part of the results to highlight the main empirical findings.

**Experiments using `FairCORELS` (exact and heuristic approaches).** We integrated both our exact and heuristic fairness sample-robustness approaches within `FairCORELS`. In a nutshell, before updating the current best solution (line 9 of Algorithm 1), we verify whether the fairness sample-robustness desideratum is achieved (using $\mathcal{IPSR}(RL, \mathcal{D}, \varepsilon)$ for the exact method and measuring unfairness over all the subsets defined by the $n$ masks for the heuristic one). Then, we only perform the current best solution update if the candidate rule list $RL$ satisfies the fairness sample-robustness criterion.

Hereafter, we focus on our experiments using the Statistical Parity fairness metric (*cf.* Table 2.1), and four different datasets widely used in the fairness literature. For an unfairness tolerance $\varepsilon = 0.01$, we learn a set of rule lists using our modified versions of `FairCORELS`, with either the exact or heuristic fairness sample-robustness method, varying the robustness parameters. For the heuristic approach introduced in Section 2.5.4, we set the number of masks $n$ to either 10 or 30. For the exact approach, we build a set of trade-offs using the strategy described at the end of Section 2.5.3, iteratively strengthening the fairness sample-robustness constraint. We consider three possible stopping criteria, two of them leveraging a separate validation set.

We report the error/unfairness trade-offs on the test sets (averaged using 5-folds cross validation) in Figure 2.7. One can see that the original `FairCORELS` did not meet the fairness constraint at test time on three out of the four considered datasets (*i.e.,* the test unfairness is greater than the considered unfairness tolerance $\varepsilon$). Importantly, all the proposed methods usually diminish fairness violation at test time (the associated points are either under $\varepsilon$ or closer to it). This improvement on fairness generalization induces a cost on the model's error. As a general trend, we see that the greater the fairness generalization improvement, the greater the error incurred. However, the generated solutions often propose interesting trade-offs between error and unfairness. In particular, in the Bank Marketing experiment the robustness enforced for fairness can sometimes benefit the error generalization as well. These partial results illustrate the usefulness of our proposed fairness robustness framework to generate models whose fairness generalize better.

Going back to our motivational example of Figure 2.5, we show in Figure 2.8

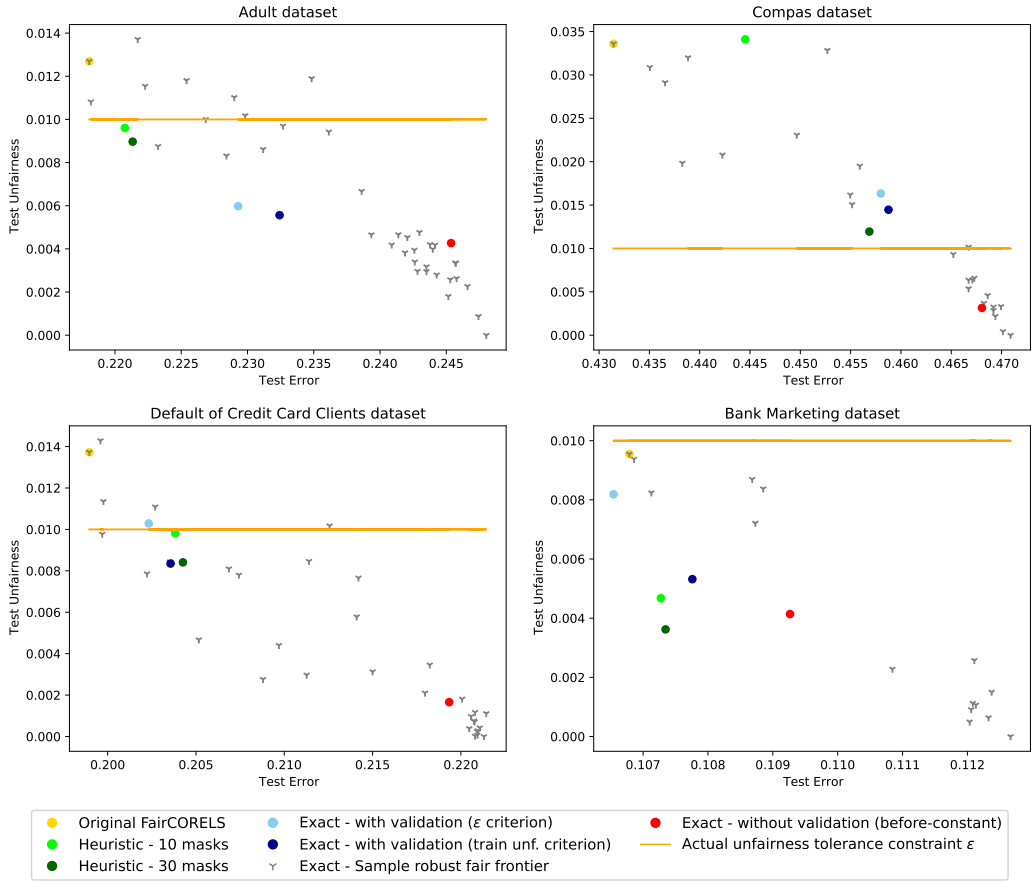



Figure 2.7: Test error and unfairness of models generated by `FairCORELS` using our exact and heuristic sample-robust fair methods (Statistical Parity metric, $\varepsilon = 0.01$).

that our heuristic sample-robust method (used here with either $n = 10$ or $n = 30$ masks) is suitable to improve fairness generalization (rightmost plot), resulting in significantly better trade-offs between error and unfairness at test time.

**Experiments using `TFCO` (heuristic approach and comparison with a state-of-the-art method).** We also integrated our heuristic fairness sample-robustness approach within a deep learning framework of the literature: `TensorFlow Constrained Optimization`[6] (`TFCO`). This Python library can be used to optimize inequity-constrained problems in TensorFlow and produce machine learning models (not restricted to the fair learning problem). To integrate our heuristic fairness sample-robustness approach within `TFCO`, we simply need to declare additional constraints, stating that the fairness constraint(s) must be satisfied over each subset specified by one of the $n$ masks.

We consider our heuristic sample-robust approach `Heur.n masks` with a number

[6] https://github.com/google-research/tensorflow_constrained_optimization

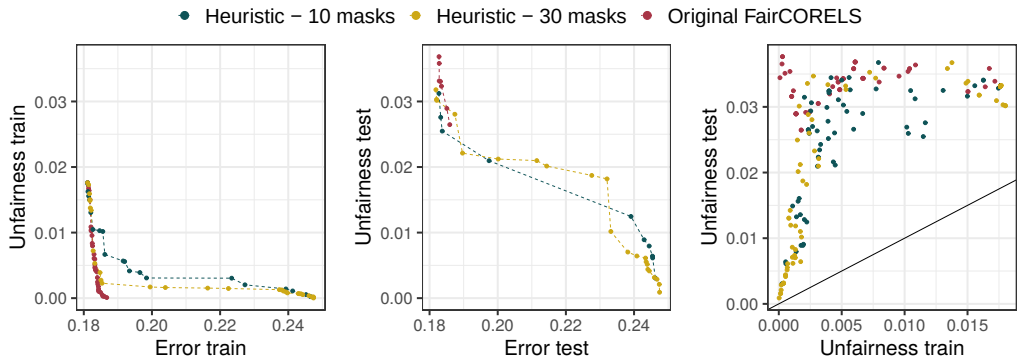



Figure 2.8: Pareto frontier approximations (trade-offs between error and unfairness on both the training and test sets) and unfairness generalization obtained using `FairCORELS` on the Adult Income dataset (with 5-folds cross-validation), for the equal opportunity fairness metric (*cf.* Table 2.1). We report results for the original `FairCORELS`, as well as for our heuristic sample-robust version of `FairCORELS` with $n = 10$ and $n = 30$ masks.

of masks $n$ set to 10, 30, or 50. We compare it with three different methods. The `unconstrained` approach simply trains a model without fairness considerations. The `baseline` method implements the traditional fair learning strategy, simply encoding the fairness constraints. Finally, `validation` is the approach described in [Cotter *et al.* 2018, Cotter *et al.* 2019], which is proposed to improve fairness generalization over the `baseline` approach. In a nutshell, to avoid *constraints overfitting*, it splits the dataset between two distinct sets: *train* and *validation*. The former is used to update the model's parameters, while the later is leveraged for auditing fairness violation. By measuring fairness violations on a separate validation set, this approach was shown to improve fairness generalization (more details regarding this method can be found in Section 1.3.6).

We compare all six methods on several experimentations, each using a different dataset, for a particular fairness setting and model architecture. We report the main insights hereafter, focusing on two particular experiments. The first one uses the Adult Income dataset, and considers the statistical parity fairness metric. The second one is based on the COMPAS dataset subject to Equal Opportunity fairness constraints. Results are summarized in Table 2.4, for the two algorithms provided within `TFCO`. They show that our heuristic sample-robust fair approach effectively improves fairness generalization while not penalizing accuracy significantly. Overall, it is competitive to the state-of-the-art `validation` method without requiring prior split of the data. Results of Experimentation 1 on Adult Income demonstrate that the fairness constraints violations on the test set are the smallest using our method. Furthermore, increasing the number of masks seems to improve the fairness generalization while penalizing accuracy, which suggest a fairness robustness / accuracy trade-off controlled by the number of masks $n$. While the `validation` method also

Table 2.4: Results of our experiments on fairness generalization using `TFCO`. We report error rates and maximum fairness constraints violations for all compared methods, for two of our experiments (all values are averaged over 100 runs). Best test results are shown in **bold**, second best in *italics*.

| | **Proxy Lagrangian** | | | | **Lagrangian** | | | |
|---|---|---|---|---|---|---|---|---|
| | Train | | Test | | Train | | Test | |
| **Method** | Error | Viol. | Error | Viol. | Error | Viol. | Error | Viol. |
| **Adult Income Dataset** | | | | | | | | |
| `unconstrained` | .122 | .072 | **.144** | .071 | .122 | .072 | **.144** | .071 |
| `baseline` | .141 | 0 | *.154* | .009 | .141 | 0 | *.155* | .006 |
| `validation` | .132 | -.002 | .158 | .004 | .134 | 0 | .157 | *.004* |
| `Heur.10 masks` | .14 | -.003 | .156 | *.003* | .143 | -.001 | *.155* | **-.003** |
| `Heur.30 masks` | .14 | -.004 | .157 | **-.001** | .148 | -.002 | .156 | **-.003** |
| `Heur.50 masks` | .14 | -.003 | .157 | **-.001** | .151 | -.002 | .157 | **-.003** |
| **COMPAS Dataset** | | | | | | | | |
| `unconstrained` | .265 | .043 | **.33** | .064 | .265 | .043 | .33 | .064 |
| `baseline` | .263 | -.004 | **.33** | .019 | .264 | -.003 | .328 | .025 |
| `validation` | .235 | .001 | .353 | **.005** | .235 | -.002 | .352 | .001 |
| `Heur.10 masks` | .261 | -.008 | *.336* | .014 | .295 | -.007 | *.326* | -.006 |
| `Heur.30 masks` | .261 | -.009 | .337 | .015 | .307 | -.009 | *.326* | *-.011* |
| `Heur.50 masks` | .262 | -.009 | .337 | *.013* | .31 | -.011 | **.322** | **-.012** |

proposes an important reduction of the test fairness violation, our `Heur.n masks` approaches give more interesting results on these experiments while less conflicting with accuracy (which was expected as we do not require prior splitting of the data). Results for Experimentation 2 (on the COMPAS dataset) suggest that in some situations the fact that our approach does not use a separate validation set (but subsets of the same training data) can limit its generalization improvement abilities. However, compared to `validation`, it has a considerably smaller impact on accuracy, and the resulting trade-offs appear competitive overall. Additionally, we observe that enforcing fairness constraints in a robust manner can improve error generalization due to the metric used (*i.e.*, in this case, Equal Opportunity) being aligned with accuracy. Hence, ensuring fairness robustness may also benefit to accuracy.

Finally, this overview of the experiments using our sample-robustness framework for fairness demonstrates the applicability of our approach and its ability to effectively improve fairness generalization.

## 2.6 Conclusion and Future Research

We propose effective ILP models leveraging accuracy and fairness jointly to prune the search space of `FairCORELS` and learn optimal fair rule lists. Our large ex-

perimental study shows clear benefits of our approach to speed-up the learning algorithm on well-known datasets from the literature. This gain is illustrated on three dimensions: achieving better training objective function values (without loss of the learning quality), using less memory footprint (*i.e.*, reduced cache size) and certifying optimality in limited amounts of time and memory. Combined with a proposed simple data structure, the ILP pruning approaches allow the learning of optimal rule lists under fairness constraints for datasets of realistic size.

Thanks to the declarative nature of our pruning approach, our framework is flexible and can simultaneously handle multiple fairness criteria for any number of sensitive groups. Indeed, each group's confusion matrix is modeled using two variables in our ILP. Considering more than two groups would require declaring additional variables, along with desired constraints using these variables.

Overall, our work illustrates the fact that statistical fairness and accuracy, when considered jointly, can be leveraged to reduce the scope of feasible solutions efficiently. In the future, it would be interesting to pursue this line of work by considering other learning algorithms and machine learning requirements. Guiding the exploration by leveraging on the ILP models (as attempted with the *ILP-Guided* approach) also seems to be a promising direction.

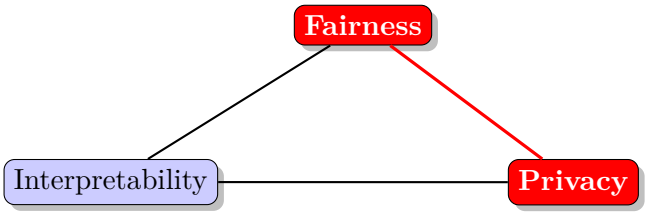
Motivated by our empirical observations, we proposed a novel formulation of robustness for fair learning aimed at enhancing statistical fairness generalization in machine learning. Our framework is metric-agnostic and based on the idea that one wants to learn a model whose fairness is verified, even if the training dataset sampling is somehow different. Our formulation is designed to be widely applicable, as many real-world machine learning applications consider finite training sets. In addition, the proposed method can be used both to audit any classifier's fairness robustness without any knowledge of the classifier's structure but also for robust fair learning, although it has some practical limitations. To deal with this issue, we proposed an effective and efficient heuristic method, exhibiting practical advantages while still improving fairness sample-robustness and fairness generalization.

A limitation of our framework is that it considers only subsets of the training set (and not all possible sample sets within a given Jaccard distance). This prevents the creation of unrealistic sample sets, which could result in over-constraining the problem. It also gives an interesting structure to our perturbation sets, allowing the derivation of several theoretical properties. Additionally, it leads to an important computational advantage. Indeed, fairness sample-robustness audit can be performed solving an integer programming model whose objective function is linear in the decision variables. However, in a more general formulation of sample-robustness, this would not be the case, as the denominator of the Jaccard distance would no longer be a constant. Formulating and solving this problem efficiently to ensure fairness over Jaccard-distance based perturbation sets in the general case is a promising direction.

Finally, automatically determining the best parameters for our heuristic method (*i.e.*, distribution and number of the binary masks, as well as the cardinalities of the defined subsets) is also a research avenue that we want to pursue in the future.

Chapter 3

# Exploiting Fairness to Reconstruct Sensitive Attributes




In this chapter, we first review the literature on the interplays between fairness and privacy in machine learning. Depending on the considered notions, several synergies and tensions arise. In particular, enforcing fairness constraints was often shown to cause privacy vulnerabilities. We elaborate on this tension and introduce declarative programming approaches aimed at reconstructing the sensitive attributes of a fair model's training set. By directly encoding information regarding the model's fairness, these methods illustrate the intrinsic tension between enforcing fairness with respect to some sensitive attributes, and protecting their privacy.

## Contents

In recent years, a growing body of work has emerged on how to learn machine learning models under fairness constraints, often expressed with respect to some *sensitive attributes* [Barocas *et al.* 2019, Caton & Haas 2023, Mehrabi *et al.* 2022]. These *sensitive attributes* correspond to characteristics such as gender, age or race [Ding *et al.* 2021], which should not be taken into account in decision-making processes impacting individuals [Barocas *et al.* 2019], for legal, ethical, social or philosophical reasons. While fair models usually do not use such sensitive attributes at inference time to avoid disparate treatment [Barocas & Selbst 2016], they still require access to them at training time [Zliobaite & Custers 2016]. The fact that these models are learnt with the objective to meet specific constraints regarding these sensitive attributes indicates that fair models intrinsically contain information about them.

Another fundamental aspect of responsible machine learning is the protection of privacy. Indeed, machine learning models are often trained on large amounts of personal data. Here, the main challenge is ensuring that these models learn useful generic patterns without leaking private information about individuals.

In many applications, privacy and fairness should not exist in isolation: both are actually needed for ethical use of machine learning, and it is therefore imperative to understand their interactions [Chang & Shokri 2021, Ekstrand *et al.* 2018]. For instance, in recidivism prediction applications, demographic groups (*e.g.,* black defendants and white defendants) should experience similar treatments (*e.g.,* similar accuracy, similar true positive rates, true negative rates...). Simultaneously, participation in the training data means that an individual once committed a crime, therefore requiring that privacy is enforced. Indeed, in this setting, membership reveals the criminal records of individuals, which is a sensitive information. In addition, there can be situations in which privacy leaks lead to future discrimination, or disparate impact may cause privacy issues.

In this context, *inference attacks* aim at leveraging the output of a computation (*e.g.*, a trained model) to retrieve information regarding its inputs (*e.g.*, a training dataset) [Dwork *et al.* 2017, Rigaki & Garcia 2023, Cristofaro 2020]. Our work belongs to the category of *dataset reconstruction attacks*, in which an adversary tries to recover part of a model's training data [Cristofaro 2020]. More precisely, we study the setting in which an adversary aims at retrieving the entire column of sensitive attributes of the training set. Depending on the available *auxiliary knowledge*, several strategies can be adopted by an adversary to reconstruct the sensitive attributes of the training set. The proposed approach is a post-processing method

that we coin as *reconstruction correction*, which takes as input an initial reconstruction performed by an adversary, optionally associated with confidence scores for each guess. The reconstruction correction method then minimally updates the adversary's initial guess to satisfy some user-defined constraints. Our work focuses on the scenario in which these are fairness constraints and the adversary leverages the fact that a model is known to be fair to improve his initial reconstruction. Such *fairness information* can for instance be the results of legal requirements, such as the "80 percent rule" for Statistical Parity [Feldman *et al.* 2015] stated by the US Equal Employment Opportunity Commission (EEOC) [EEOC. 1979]. Indeed, this text states that the difference between acceptance rates of two sub-population must not exceed 20%.

The tensions between fairness and privacy in machine learning have been studied in recent years, mainly through the theoretical [Cummings *et al.* 2019, Agarwal 2021b] and empirical [Bagdasaryan *et al.* 2019, Chang & Shokri 2021, Fioretto *et al.* 2022] conflicts existing between statistical fairness metrics and Differential Privacy (DP). Our work takes a different direction but also demonstrates that enforcing statistical fairness can endanger the privacy of sensitive attributes.

**Related Works.** Related work on reconstruction attacks can be found in Section 1.5.3. Few works are directly related to our contribution. [Hu & Lan 2020] propose a mechanism whose principle is related to ours: explicitly exploiting fairness by encoding it within declarative programming frameworks to enhance the reconstruction. They however consider a very particular framework where a *learner* can query an *auditor* (owning the training set sensitive attributes) to known whether some model's parameters satisfy the fairness constraint(s). This particular setup is more favorable to the attacker, as he possesses the fairness information for a whole set of models instead than only for the final trained one. While the intuition is similar, our work covers a more general setting, with no assumption on the underlying fairness-enhancing method and in a less favorable attack setup. [Aalmoes *et al.* 2022] can serve as a baseline work for our proposed attack, as they also consider the traditional learning pipeline but do not leverage the fairness information. More precisely, their proposed attacker possesses a separate auxiliary dataset for which the sensitive attributes are known. He uses this dataset, along with query access to the target black-box, to train an attack model to predict the sensitive attributes given an example's unsensitive ones, label and black-box prediction. The trained attack model can then be leveraged to predict the training examples' sensitive attributes.

**Outline of the Chapter.** In Section 3.1, we first provide a large literature review on the connections between fairness and privacy in machine learning. We summarize they key identified synergies, compatibilities and tensions, observing that, in particular, statistical fairness and privacy notions often conflict. In the remainder of the chapter, we build on this observation and propose an original contribution, also highlighting an intrinsic tension between fairness and training data privacy. More

precisely, we present our contribution on reconstructing a fair model's training set sensitive attributes leveraging information regarding its fairness. We first introduce our proposed attack framework in Section 3.2. We then empirically assess its effectiveness to reconstruct fair models' training set sensitive attributes in Section 3.3. Finally, in Section 3.4, we discuss and evaluate possible mitigation strategies.

## 3.1 Connections between Fairness and Privacy

In this bibliography section, we review the literature at the intersection between fairness and privacy in machine learning. Part of this intersection is covered by a recent survey [Fioretto *et al.* 2022] studying the interactions between fairness and differential privacy (DP), in both decision making and machine learning tasks. Hereafter, we first highlight the identified theoretical and empirical tensions between the two notions. We then review some synergies and compatibilities illustrating that the two desiderata can be conciliated.

### 3.1.1 Tensions

As discussed in Section 1.3, it is desirable and often legally required to ensure that sensitive attributes do not directly or indirectly influence the predictions of a machine learning model. However, while many popular fairness-enhancing approaches require the availability of such sensitive attributes, their collection and use may be prohibited by law. Some approaches propose to use an encrypted version of the sensitive attributes so that the users do not have to explicitly reveal this information. For instance, [Kilbertus *et al.* 2018] leverage secure Multi-Party Computation (MPC) to build a fair model. Nevertheless, as discussed in Section 1.5.1, processing encrypted information does not protect the computation's output from inference attacks. This illustrates a first, straightforward intrinsic conflict between fairness and privacy. Furthermore, when applied jointly, both notions can still conflict, as discussed with more details in the following paragraphs.

**Group fairness and differential privacy are theoretically incompatible.** It is provably impossible to build machine learning models strictly respecting a given fairness constraint while respecting DP. More precisely, [Cummings *et al.* 2019] show that $(\varepsilon_{DP}, 0)$-DP and fairness (Equal Opportunity) can not be simultaneously satisfied without reaching trivial accuracy. Authors note that this holds for pure DP $(\varepsilon_{DP}, 0)$-DP, but is also applicable for $(\varepsilon_{DP}$,$\delta_{DP})$-DP (as $\delta_{DP}$ is usually required to be cryptographically small). [Agarwal 2021b] also state an impossibility theorem, considering popular group fairness definitions: *if a learning algorithm* $\mathcal{L}$ *is* $(\varepsilon_{DP}, 0)$*-differentially private and is guaranteed to output an approximately fair classifier, then* $\mathcal{L}$ *is constrained to output a constant classifier.* The idea of their proof is essentially the same as [Cummings *et al.* 2019]. (i) Consider a learning algorithm $\mathcal{L}$ that is $(\varepsilon_{DP}, 0)$-DP. For any two datasets $\mathcal{D}$ and $\mathcal{D}'$, and for any classifier $h$, if $\mathcal{L}$ outputs $h$ for $\mathcal{D}$ with probability strictly greater than zero, then it must

output $h$ for $\mathcal{D}'$ with strictly positive probability too. This can be proved because, for any two datasets $\mathcal{D}$ and $\mathcal{D}'$, we can build a serie of datasets neighboring two-by-two, from $\mathcal{D}$ to $\mathcal{D}'$ (and the property must be verified for all pairs of neighbouring datasets by definition of pure DP). (ii) Recall that $\mathcal{L}$ can only output classifiers respecting a given (exact or approximate) fairness requirement: if a classifier $h$ does not meeet the fairness requirement on the training set $\mathcal{D}$, then $P(\mathcal{L}(\mathcal{D}) = h) = 0$. The conjunction of (i) and (ii) implies that $\mathcal{L}$ can only release constant classifiers (and hence pure DP and fairness can not be satisfied jointly - or we must accept some fairness or DP violations).

**Enforcing fairness increases privacy vulnerabilities.** [Kulynych *et al.* 2022] show that there exist disparities with respect to the vulnerability to Membership Inference Attacks (MIAs) between various subgroups of the population. They demonstrate that vulnerability to MIA is caused by *distributional overfitting*, which quantifies the distance between the distributions of outputs of the model on the training set and outside. Disparate vulnerability to MIAs arises if and only if distributional overfitting differs across subgroups. In practice, subgroups that are inherently more difficult to fit and/or that are less represented in the data are indeed more vulnerable to MIAs. Additionally, overfitting can increase these vulnerabilities, but also their disparities. They empirically show that enforcing fairness constraints may help under certain conditions, but can also exacerbate the observed disparities or even create new ones in real-world applications. Finally, they recall that DP upper-bounds the vulnerability of all individuals or subgroups, hence also upper-bounding their disparity. It however does not remove it completely, and to get an interesting mitigation, the privacy budget must often be really tight, hence resulting in significant utility drops.

Indeed, in a position paper, [Ekstrand *et al.* 2018] emphasize the importance for a privacy-preserving mechanism to protect individuals with equivalent effectiveness. However, while DP provides the same (worst-case) theoretical protection to all dataset examples, the actual privacy vulnerability is often not uniformly distributed in practice. [Chang & Shokri 2021] empirically study the privacy implications of fairness, quantifying the *data privacy risk* as the success of a black-box *membership inference attack*. They empirically show that enforcing fairness constraints disproportionately raises the privacy risk of the unprivileged subgroups: "fairness comes at the cost of privacy, and the privacy cost is not equal across subgroups". This is explained by the fact that the fairness requirements they use impose the model to equally fit the unprivileged subgroups. When such subgroups are smaller, each example then has a stronger impact over the resulting model and, in the worst case, is memorized. In addition, the more unfair the unconstrained model is, the higher the privacy vulnerability disparity will be, as there is more unfairness to be compensated. Finally, as discussed in the introduction, previous work [Hu & Lan 2020] demonstrated that information regarding a model's fairness can directly be leveraged to reconstruct the model's sensitive attributes. In the

remainder of this chapter, we also illustrate this intrinsic tension in a more general setting.

**DP disproportionately affects utility.** [Bagdasaryan *et al.* 2019] study the effects of enforcing differential privacy on a model's accuracy on different subgroups of the population, using the *accuracy parity* fairness notion (which equalizes the model's accuracy across the subgroups). Considering several image classification and natural language tasks, they use the popular `DP-SGD` [Abadi *et al.* 2016] framework for differentially private deep learning in both centralized and federated settings. This large empirical study shows that gradient clipping and random noise addition, the key mechanisms of `DP-SGD`, disproportionately affect underrepresented and complex classes and subgroups. Indeed, enforcing DP leads to higher accuracy drops for minorities and discriminated groups, such as darker-skinned people in the context of facial recognition, but also intersections of different sensitive subgroups. They note a "poor gets poorer effect" : the classes with low accuracy in the non-DP setting suffer the largest accuracy drops when applying DP. In a follow-up work, [Uniyal *et al.* 2021] empirically observe that the differentially private `PATE` [Papernot *et al.* 2017, Papernot *et al.* 2018] framework (introduced in Section 1.5.5) also has disparate impact on the built model's utility. They however report that `PATE` has smaller disparate impact compared to `DP-SGD` to reach similar privacy levels, and note that a *sweet spot* for the number of teachers exists, minimizing the induced disparities. [Farrand *et al.* 2020] observe that the accuracy disparity caused by DP still occurs even when the data is slightly imbalanced, and for loose privacy guarantees. Indeed, two main factors were identified in the literature to explain this effect: properties of the training data, and models' characteristics, which are summarized and analyzed with more details in a recent survey [Fioretto *et al.* 2022]. Concerning the former, input norms and distance to the decision boundary, are "key characteristics of the data connected with exacerbating the disparate impacts of private learning tasks" [Tran *et al.* 2021b].

It was also observed in health care applications (x-ray images classification and mortality prediction in time series) that small groups and samples at the tail of the data distribution suffer from a larger accuracy drop compared to majority groups and typical examples [Suriyakumar *et al.* 2021]. Furthermore, the characteristics of the DP learning mechanisms themselves are also directly related to the magnitude of the observed disparate impact. This encompasses the gradient clipping and noise addition mechanisms of `DP-SGD` (as aforementioned), as well as the size of the teacher ensemble and the confidence of the voting teachers in the context of `PATE` [Tran *et al.* 2021a]. Different technical solutions to mitigate the disparate impact of DP on a model's utility were proposed. Indeed, it was shown possible to modify `DP-SGD` to use different clipping bounds for the different identified subgroups [Xu *et al.* 2021]. Other work [Zhang *et al.* 2021] performs early stopping, leveraging a public validation set. When using `PATE` in low voting confidence regimes, small perturbations may significantly affect the result of the voting result.

To mitigate this phenomenon, [Tran *et al.* 2021a] propose to use soft labels and report confidence scores associated with each target label, rather than reporting solely the label with the largest confidence. While being heuristic in the sense that they do not guarantee any form of fairness, these approaches are empirically shown to reduce the disparate impact caused by traditional DP mechanisms. The disparate impact of DP mechanisms was also observed for decision tasks. [Pujol *et al.* 2020] study the setup where agencies release differentially private versions of their databases, that are then used for several allocation problems. The authors consider three real-life allocation problems using the differentially-private Census data: printing of election materials in minority languages, allocation of funds to school districts to assist disadvantaged children, and apportionment of legislative representatives. They show that the noise added by the DP mechanism causes errors in the computed allocations, compared to the true allocations (*i.e.,* the allocations that would be decided without the DP noise). The key point of their work is that this error affects the entities being allocated some resources in a disparate manner. For instance, it is empirically shown that small school districts often benefit an overestimated allocation. On the other side, larger district may get a smaller allocation, which harms their enrolled children. This effect was also observed in the literature, and two main causes were identified [Fioretto *et al.* 2022]. In a nutshell, the shape of the decision problem can disproportionately exacerbate the noise added by the DP data release if it involves non-linearities in its computation, such as thresholds for funds allocation. Additionally, post-processing steps can induce intrinsic biases. For instance, ensuring simple non-negativity constraints within the computed values can imply a positive bias. It was also shown that DP mechanisms adding data-dependent noise are responsible for a more important disparity, due to the fact that, contrary to standard DP mechanisms (such as the Laplace mechanism), the effect of DP differs between entities. Finally, other notions of privacy may also impact fairness. For instance, recent work [Koch & Soll 2023] show that training models to take into account potential future un-learning requests from the training set users (such requests are stated as a right by privacy regulations) disproportionately affects the utility for minority groups.

**Differential privacy disproportionately affects the quality of post-hoc explanations.** [Datta *et al.* 2016] propose differentially private post-hoc explanations, among which some aim at identifying proxy features that cause a *group disparity* (*i.e.,* a difference in the average prediction between several protected groups). Then, it is shown that, for small protected groups such as demographic minorities, the amount of noise required to make the explanations differentially private results in a significant loss in its utility, hence making more difficult the discovery of discriminatory proxy features. While proposing a framework to generate differentially private post-hoc explanations, [Patel *et al.* 2022] observe that sparse data regions, which often correspond to underrepresented groups (minorities) are associated to poorer performances, either in terms of required privacy budget or explanation

quality. In both cases, privacy disproportionately affects minority groups, which is consistent with previously mentioned works.

Overall, while DP and statistical fairness are theoretically incompatible, they also strongly conflict in practice. On the one side, to ensure fairness for protected, underrepresented groups, the corresponding examples shall yield a higher importance in the learning process, which exposes their information more than average. On the other side, to ensure DP, one must reduce more the influence of underrepresented groups, as learning an equivalent amount of information for them would result in an increased per-example privacy risk. In the next subsection, we nevertheless show that the two notions can be jointly applied, and that there can exist synergies between privacy and fairness for some particular notions.

### 3.1.2 Compatibilities & Synergies

**Differential privacy and approximate group fairness can be jointly enforced with some trade-offs.** As discussed in Section 3.1.1, it is not possible for a learning algorithm to satisfy DP while also producing a model strictly complying with fairness constraints. However, it is possible for a DP learning algorithm to output a model *approximately* satisfying a given fairness criteria [Cummings *et al.* 2019]. Then, a trade-off between the DP guarantees and the model's fairness is usually observed. Hereafter, we first introduce different methods of the literature jointly handling differential privacy and fairness. We then report a theoretical result which bounds the unfairness increase due to privacy.

***Differentially private and fair methods.*** [Cummings *et al.* 2019] propose the notion of Private and Approximately Fair Agnostic PAC Learning, stating that a learning algorithm satisfies differential privacy while returning an accurate and approximately fair classifier with high probability. They implement this notion using the Exponential Mechanism, with a utility function being the sum of a model's error and unfairness. The sensitivity of the utility function being data-dependent, the Laplace mechanism is used to upper-bound it in a Differentially Private manner. This approach achieves the desiderata of privacy, fairness and accuracy, but the running time of the Exponential Mechanism scales linearly with the hypothesis class size, which is exponential for common hypothesis classes. This motivates the need for an efficient, polynomial-time algorithm conciliating these desiderata. The authors build upon a polynomial-time algorithm from the literature, producing approximately fair and accurate randomized classifiers with high probability. In a nutshell, this algorithm formulates the fair learning problem as a two-player zero-sum game, between a Learner minimizing error while satisfying fairness constraints and an Auditor updating Lagragian multipliers to penalize the largest subgroup-wise fairness violations. This algorithm is modified to satisfy differential privacy by using a DP subroutine to privately compute the players' best responses in each round.

[Xu *et al.* 2019] propose two methods to achieve differential privacy and fairness jointly in logistic regression. They use the decision boundary fairness as a notion of fairness that provably minimizes statistical parity violation. A first approach coined `PFLR` consists in considering the fairness constraint as a penalty term to the objective function. Differential privacy is enforced using the functional mechanism [Zhang *et al.* 2012]. More precisely, the objective function is approximated through its polynomial representation based on Taylor expansion. The objective function is then perturbed by injecting Laplace noise into its polynomial coefficients. Minimizing the perturbed objective function leads to the computation of differentially private model parameters. A second approach, named `PFLR*` and based on the first one, takes advantage of the connection between ways of achieving differential privacy and fairness. More precisely, authors note that adding the fairness penalty is equivalent to shifting the value of some coefficients of the polynomial form of the objective function. Thus, they do not incorporate the fairness penalty term directly in the objective function and rather integrate it via mean-shifting the Laplace noise added to a subset of the coefficients. As such shift is dataset-dependent, a small part of the privacy budget is used to estimate it in a differentially private manner. Theoretical analysis, as well as empirical evaluation, shows that `PFLR*`, by separating privacy budgets on objective function and fairness constraint, offers a more flexible framework to find good trade-offs among privacy, fairness, and utility. In a follow-up work, [Ding *et al.* 2020] extend `PFLR` and propose to have two distinct privacy budgets in order to add Laplace noise with larger magnitude to the coefficients of the terms involving the sensitive attributes than to the others within the objective function. They also propose a second approach using the relaxed functional mechanism to enforce the relaxed version of differential privacy ($\varepsilon_{DP}$,$\delta_{DP}$)-DP in order to improve the utility. It utilizes the extended Gaussian mechanism to perturb the objective, adding random Gaussian noise to the coefficients of the polynomial form of the objective function. Empirical evaluation on real-world datasets confirms that the use of ($\varepsilon_{DP}$,$\delta_{DP}$)-DP allows an improved utility in all scenarios compared to pure DP. Furthermore, the use of two distinct privacy budgets can help enforcing stronger privacy guarantees while also reducing the correlations with the protected attribute, practically improving fairness.

[Tran *et al.* 2021c] propose a Differentially Private framework to train Deep Learning models that satisfy several popular group fairness notions. The approach considers the Lagrangian relaxation of the fairness-constrained learning problem, and leverages a Lagrangian dual approach to solve it: the fairness violation terms, weighted by Lagrangian multipliers, are directly added to the objective function. The training procedure then consists of iteratively repeating two successive steps: primal and dual. Primal update step optimizes the model parameters to minimize the objective function, given the current Lagrangian multipliers. Then, the dual update step updates the value of the Lagrangian multiplier to approximate the stronger Lagrangian relaxation. In order to enforce differential privacy for sensitive attribute information, differential privacy is achieved at both steps, when computing the fairness violation terms or their gradients. In the primal update step, clipped

and noisy gradients are used. The model parameters optimization is done on this noisy version of the objective function (where only the fairness violation term, accessing sensitive group membership which we want to protect, is impacted by the DP mechanism). A similar mechanism is done on the dual update step, where constraint violations are clipped and perturbed with carefully calibrated Gaussian noise. Extensive empirical evaluation shows that the fairness violation decreases as the privacy budget increases: enforcing DP leads to violating more fairness. This is explained by the fact that relaxing the DP constraint allows either to perform more iterations (hence propagating more fairness violation information) or to inject less noise for a fixed number of iterations (hence propagating more accurate fairness violation information). Another surprising trend is that the model accuracy slightly decreases as $\varepsilon_{DP}$ increases. This is due to the fact that enforcing weaker DP allows the fairness constraints to have more impact on the objective function, hence penalizing more the accuracy.

[Jagielski *et al.* 2019] adapt two fair learning algorithms in order to satisfy both fairness (Equalized Odds) and differential privacy (with respect to the sensitive attributes). They first consider the post-processing method of [Hardt *et al.* 2016], which we introduced in Section 1.3.3. Given a pre-trained and possibly unfair classifier, the approach first computes its per-group per-ground truth prediction proportions. It then solves a Linear Program to compute per-group per-class prediction probabilities defining a fair randomized classifier. To enforce $\varepsilon_{DP}$-DP in this setting, the authors simply add well-calibrated noise drawn from the Laplace distribution to the computed statistics before solving the LP with them. Theoretical analysis of how the introduced noise propagates to the solution of the LP leads to bounds on accuracy and fairness violation that are met with high probability. This quantifies a trade-off between accuracy, fairness and privacy: weaker DP guarantees lead to tighter bounds on accuracy and fairness, while stronger DP guarantees (satisfied by adding more noise) increase the bounds, and the possible loss on accuracy and fairness. Experimental evaluation shows that this simple method is able to give interesting trade-offs even with small datasets but is expected to perform worst than the second approach on large ones. The later builds upon an in-processing approach [Agarwal *et al.* 2018], which we also introduced in Section 1.3.3. It formulates the problem of learning a fair and accurate classifier as finding the equilibrium of a two-player min-max game. A Learner minimizes the objective function over the set of possible classifiers while an Auditor maximizes it by choosing the value of the multipliers penalizing fairness violations. To enforce (approximate) ($\varepsilon_{DP}$,$\delta_{DP}$)-DP, the authors add well-calibrated Laplace noise while computing the gradients of the Auditor, and use the exponential mechanism for the Learner's model selection. Similar to the first case, a stronger privacy guarantee (smaller $\varepsilon_{DP}$ and $\delta_{DP}$) leads to weaker accuracy and fairness guarantees. However, a new trade-off can be controlled through the maximum norm of the multipliers: larger values lead to tighter fairness bounds but looser error bounds, and vice-versa. For both approaches, we see that introducing noise to achieve DP leads to a reduction in the fairness guarantees (in a similar manner as for accuracy).

[Mozannar *et al.* 2020] consider the setup in which the sensitive attributes are released using local differential privacy, and propose a two-step approach. First, a classifier which is fair with respect to the noisy sensitive attributes is built, using a state-of-the-art in-processing fair learning algorithm [Agarwal *et al.* 2018]. Second, a modified version of a post-processing fairness-enhancing method [Hardt *et al.* 2016] is used to ensure with high probability that the model is also fair with respect to the (unknown) original sensitive attributes. For strong privacy regimes, this post-processing step is empirically shown to significantly decrease fairness violation. Interestingly, the set of trade-offs between accuracy, fairness and privacy is shown to differ in this local DP setup, compared to the previously mentioned central DP approaches.

***The fairness cost of differential privacy can be theoretically bounded.*** Recent work theoretically shows that the impact of DP on fairness is bounded and can be computed to obtain non-trivial guarantees regarding the private model's fairness [Mangold *et al.* 2023]. The underlying analysis relies on the fact that, just like a model's accuracy, common statistical fairness metrics are pointwise Lipschitz continuous with respect to the model parameters. Then, proving that the private model is sufficiently close to the optimal non-private one implies that their fairness are also close. Interestingly, the theoretical bound tightens linearly with respect to the size of the training set: the "loss of fairness" due to privacy vanishes when $N$ increases.

**Individual fairness and differential privacy are both robustness definitions.** As introduced in Section 1.3.2, individual fairness can be formulated as a Lipschitz condition: just like differential privacy, it is a robustness definition [Ignatiev *et al.* 2020]. More precisely, [Dwork *et al.* 2012] observe that individual fairness constitutes a generalization of differential privacy. The authors draw an analogy between individuals in the setting of fairness and databases in the setting of differential privacy. Indeed, as also noted by [Zemel *et al.* 2013], differential privacy requires that "algorithms behave similarly on similar databases", while individual fairness enforces that classifiers yield similar outcomes for similar instances. This allows the use, for fairness purposes, of mechanisms designed for differential privacy. For instance, [Dwork *et al.* 2012] propose an efficient individually fair learning algorithm based on the exponential mechanism [McSherry & Talwar 2007], coming with provable loss bounds. In [Jagielski *et al.* 2019], the proposed privacy-preserving approach (ensuring DP for the sensitive attributes) can be seen as a relaxation of the strict notion of individual fairness proposed in [Ignatiev *et al.* 2020]. Indeed, while the former enforces a ratio on the probabilities of different outcomes when a single sensitive attribute label is modified, the latter enforces that the sensitive attribute is never used. In this way, fairness through unawareness is a strict, simple but certifiable way to ensure sensitive attribute privacy.

**Privacy and statistical fairness can enhance each other for particular setups.** [Khalili *et al.* 2021] consider the particular setup where a pre-trained model generates qualification scores for a set of applicants. These scores are then used to determine a fixed number of candidates that will be selected by the process (*e.g.,* for a grant, a job...etc). The authors show that the exponential mechanism can be used to perform the selection given the qualification scores, in order to both enforce DP for the selection process and improve fairness (Equal Opportunity). Under some conditions regarding properties of the different protected groups, the proposed approach can make the selection procedure perfectly fair. Other notions of privacy can also have different interactions with fairness definitions. For instance, [Ruggieri 2013] studies the context of itemset mining: given a dataset, the objective is to mine frequent patterns. The author shows that $t$-closeness (a data anonymization technique, introduced in Section 1.5.1) with carefully chosen parameters implies popular group fairness notions. [Hajian *et al.* 2015] also consider frequent patterns discovery, and propose two-step algorithms to jointly address non-discrimination (fairness) and privacy. More precisely, they first apply a privacy preserving mechanism, before using data sanitization methods to enforce non-discrimination. Indeed, considering either $k$-anonymity or differential privacy, they theoretically prove that the privacy guarantees are not affected by the later fairness-enhancing stage. On the contrary, they observe that applying privacy-preserving mechanisms on a sanitized data could alter the resulting patterns' fairness, either increasing or decreasing discrimination depending on the considered scenario (in line with the aforementioned tensions). Importantly, they empirically note that the utility loss incurred by jointly enforcing fairness and privacy is only marginally higher than that of enforcing privacy only. This result highlights a form of synergy between the two desiderata, where the former privacy-enhancing step sometimes also improves fairness, overall leading to a more modest utility drop from the later discrimination sanitizing step. This trend is valid for both $k$-anonymity and differential privacy, although the later leads to a significantly higher utility cost.

As discussed throughout this section, there are multiple conflicts and synergies that can be highlighted between privacy and fairness notions. In particular, enforcing fairness was shown to increase privacy vulnerabilities of the resulting trained model. Consistent with this observation, we illustrate in the next subsections an intrinsic tension between enforcing fairness with respect to some sensitive attributes, and ensuring such attributes' privacy.

## 3.2 Leveraging Fairness to Improve Sensitive Attributes Reconstruction

In this section, we introduce our proposed framework to enhance the reconstruction of sensitive attributes by leveraging the information about the target model's fairness. Afterwards, we describe a general integer linear programming model that

can be used to correct any adversary's guess about the sensitive attributes vector, given some knowledge expressed as constraints over this vector. We show how this model can be reformulated leveraging tools from constraint programming to improve scalability in the case of statistical fairness metrics. Finally, we discuss how the proposed models can be generalized to handle other metrics and sensitive attribute values.

Table 3.1: Summary of the considered statistical fairness metrics for our reconstruction correction experiments.

| **Metric** | **Constraint Expression** |
|---|---|
| Statistical Parity (SP) | $\forall s, \left\lvert \frac{\sum_{e_j \in \mathcal{D}} \hat{y}_j}{\lvert\mathcal{D}\rvert} - \frac{\sum_{e_j \in \mathcal{D} \mid s_j = s} \hat{y}_j}{\lvert\{e_j \in \mathcal{D} \mid s_j = s\}\rvert} \right\rvert \leq \varepsilon$ |
| Predictive Equality (PE) | $\forall s, \left\lvert \frac{\sum_{e_j \in \mathcal{D} \mid y_j = 0} \hat{y}_j}{\lvert\{e_j \in \mathcal{D} \mid y_j = 0\}\rvert} - \frac{\sum_{e_j \in \mathcal{D} \mid y_j = 0, s_j = s} \hat{y}_j}{\lvert\{e_j \in \mathcal{D} \mid y_j = 0, s_j = s\}\rvert} \right\rvert \leq \varepsilon$ |
| Equal Opportunity (EOpp) | $\forall s, \left\lvert \frac{\sum_{e_j \in \mathcal{D} \mid y_j = 1} \hat{y}_j}{\lvert\{e_j \in \mathcal{D} \mid y_j = 1\}\rvert} - \frac{\sum_{e_j \in \mathcal{D} \mid y_j = 1, s_j = s} \hat{y}_j}{\lvert\{e_j \in \mathcal{D} \mid y_j = 1, s_j = s\}\rvert} \right\rvert \leq \varepsilon$ |
| Equalized Odds (EO) | Conjunction of PE and EOpp |

**Considered Fairness Metrics.** The different fairness metrics we consider are summarized in Table 3.1. As in Chapter 2, we consider the binary classification task, which is in line with the fairness literature. We use the popular *one-vs-all* formulation of group fairness notions, bounding the difference between each protected group and the overall dataset. This notion is in particular used in the fair learning frameworks that we later use in our experiments. Compared to the *one-vs-one* (pairwise) formulation (used in Chapter 2), it only requires a linear number of constraints (compared to a quadratic one) with respect to the number of protected groups $|\mathcal{S}|$ (*cf.* Table 1.2). Furthermore, in our proposed reconstruction correction models (described hereafter), it results in linear constraints (while the pairwise formulation leads to quadratic ones). This is due to the fact that the sensitive attributes are unknown in our tackled reconstruction problem (while they are usually known in the fair learning problem). As noted in Section 1.3.2 this formulation is equivalent to the pairwise one, with carefully chosen unfairness tolerance $\varepsilon$.

### 3.2.1 Attack Pipeline

Figure 3.1 illustrates the different components of the proposed framework. Given a training dataset $\mathcal{D} = (X, S, Y)$, a model $h$ is trained using a fair learning algorithm $\mathcal{L}$, which ensures that $h$ is fair on $\mathcal{D}$ according to some statistical fairness metric with respect to the sensitive attribute $S$. Note that $h$ does not use the sensitive attribute $S$ for inference to prevent disparate treatment [Barocas & Selbst 2016]. Thus once trained, $h$ can be used for inference based only on non-sensitive attributes $X$. Our approach does not make any assumption on the underlying fairness-enhancing technique $\mathcal{L}$ used. Indeed, the only requirement of our attack is the knowledge of the

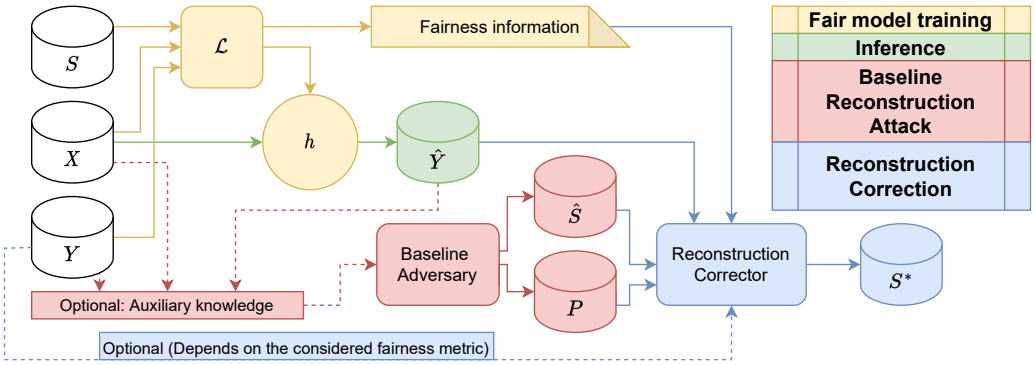



Figure 3.1: The proposed attack framework. A model $h$ is learnt by the fair learning procedure $\mathcal{L}$ and used for inference. Then, a *Baseline Adversary* tries to reconstruct the sensitive attributes $S$ of $h$'s training set. Our contribution lies in the *Reconstruction Corrector* component, which takes as input the Baseline Adversary's guess $\hat{S}$ and corrects it to comply with the fairness information by outputting $S^*$, the corrected sensitive attributes reconstruction.

fairness information. Interestingly, we will see in Section 3.4 than this information can easily be estimated by an adversary, hence relaxing this assumption.

The attack itself aims at retrieving the training set sensitive attributes vector $S$. Recall that in the considered pipeline, $S$ is only used by $\mathcal{L}$ to ensure $h$'s fairness (and never used again). In the first step of the attack, a *Baseline Adversary* makes a *guess* $\hat{S}$ on $S$, based on some auxiliary knowledge. The adversary also outputs a probability vector $P$, illustrating his confidence for each component of the guess vector $\hat{S}$. Our attack does not assume anything about the form of the auxiliary knowledge. If the adversary does not compute confidence scores, the confidence vector can simply be set to the identity vector. Importantly, our framework is agnostic to the used baseline adversary as the proposed reconstruction correction process only requires access to its outputs ($\hat{S}$ and $P$). In our experiments, we will use an adversary of the literature [Aalmoes *et al.* 2022], which leverages a separate dataset to learn an attack model predicting the sensitive attribute(s) of an example, given its non-sensitive ones, its label, and its prediction from the target model. Then, this attack model is used to compute $\hat{S}$ (and $P$) from $X$, $Y$ and $\hat{Y}$.

In the second step of the attack, a *Reconstruction Corrector* component takes as input the baseline adversary's guess and confidence vectors ($\hat{S}$ and $P$). It outputs a new reconstruction guess $S^*$ minimizing the (confidence-weighted) changes to the adversary's guess while satisfying some given properties, such as statistical fairness constraints. To ensure the respect of such constraints, the *Reconstruction Corrector* component also needs as input the fairness information, the target model's predictions on the training set $\hat{Y}$ as well as (depending on the particular statistical fairness metric at hand) the true labels $Y$. Importantly, if the actual fairness information is unknown, it can still be estimated as discussed later in Section 3.4.2. As stated previously, our attack does not make any assumptions about the target model $h$,

which can be seen as a black-box as it only requires access to its predictions $\hat{Y}$. Importantly, the attack is agnostic to the actual type of the model, the training algorithm and the fairness mitigation procedure.

The success of the attack pipeline can be evaluated as the *reconstruction accuracy* of $S^*$ (*i.e.*, proportion of elements of $S$ correctly predicted in $S^*$). The core contribution of our attack lies in the Reconstruction Corrector component, which, by incorporating solely the fairness information, is able to significantly improve the quality of the reconstruction of the sensitive attribute. Such improvement can be quantified by comparing the reconstruction accuracy of the initial adversary's guess $\hat{S}$ and that of the corrected one $S^*$.

### 3.2.2 General Reconstruction Correction Model

We now introduce $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$, a general Integer Linear Programming model implementing the Reconstruction Corrector component of Figure 3.1, for the binary sensitive attributes setting: $\mathcal{S} = \{0, 1\}$. Its objective is to modify the adversary's guess for the training set sensitive attributes to satisfy some constraints (here, the fairness information) while minimizing the (confidence-weighted) changes to the adversary's original guess.

**Inputs**

- $\hat{s}_j \in \{0, 1\},\ j = 1, \ldots, N$ (adversary's initial guesses).
- $p_j \in [0, 1],\ j = 1, \ldots, N$ (adversary's confidence for $\hat{s}_j$).
- $\hat{y}_j \in \{0, 1\},\ j = 1, \ldots, N$ (target model $h$'s predictions).
- Fairness information: $h$ satisfies fairness constraints for some metric (*e.g.*, SP) and some tolerance $\varepsilon$.

**Decision variables**

- $s^*_j \in \{0, 1\},\ j = 1, \ldots, N$ (corrected guess for the sensitive attributes vector).

**Integer linear programming model $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$**

$$\min \sum_{j=1}^{N} (p_j \cdot (1 - \hat{s}_j) \cdot s^*_j) + \sum_{j=1}^{N} (p_j \cdot \hat{s}_j \cdot (1 - s^*_j)) \tag{3.1}$$

$$s.t. : 0 < \sum_{j=1}^{N} s^*_j < N \tag{3.2}$$

$$-\varepsilon \leq \frac{\sum_{j=1}^{N} \hat{y}_j}{N} - \frac{\sum_{j=1}^{N} \hat{y}_j \cdot s^*_j}{\sum_{j=1}^{N} s^*_j} \leq \varepsilon \tag{3.3}$$

$$-\varepsilon \leq \frac{\sum_{j=1}^{N} \hat{y}_j}{N} - \frac{\sum_{j=1}^{N} \hat{y}_j \cdot (1 - s^*_j)}{\sum_{j=1}^{N} (1 - s^*_j)} \leq \varepsilon \tag{3.4}$$

The objective (3.1) aims at minimizing the confidence-weighted changes to the original adversary's guess $\hat{S}$. Each modification of a component $\hat{s}_j$ of the original adversary's guess is penalized with cost $p_j$ and the model minimizes the total cost. Constraint (3.2) simply ensures that the reconstruction contains at least one example from each protected group. Finally, constraints (3.3) and (3.4) encode the fairness constraint for the Statistical Parity metric. Note that considering any other statistical fairness metric would simply require plugging the adequate constraint within the model. Constraint (3.3) (respectively, constraint (3.4)) ensures that the Positive Prediction Rate (PPR) on group 1 (respectively, group 0) is no further than $\varepsilon$ from the PPR on the overall dataset. Indeed, as mentioned at the beginning of this section, we use the *one-vs-all* formulation, which bounds the difference of the given statistical measure between each protected group and the entire dataset. Because there are two protected groups, this results in two constraints, while the use of the *one-vs-one* formulation would have created a single one in this particular case. However, here, fairness is ensured by modifying the reconstruction of the sensitive attributes. This differs from the typical case of fair model training, in which the sensitive attributes are known and fairness is ensured by modifying the model's predictions $\hat{y}_j$ (which, in turn, are fixed here, and exploited to build the sensitive attributes $s_j^*$). Importantly, note that the two constraints can easily be linearized because the denominator of the first term ($N$) is a constant, which would not be the case using a *one-vs-one* formulation.

Finally, an optimal solution to our general reconstruction correction model $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$ is an assignment of the binary variables $s_j^*$ that minimizes (3.1) while satisfying constraints (3.2) to (3.4). This assignment $S^*$ corresponds to the minimum (confidence-weighted) changes to the original adversary guess $\hat{S}$ in order to meet the fairness requirement. If the performed changes are correct most of the time (which is to be expected if the adversary provides good confidence scores), then the overall reconstruction accuracy will be improved. In any case, the algorithm is guaranteed to find a solution satisfying the fairness constraint - which is not the case of the baseline adversary. Indeed, as it is able to modify the sensitive attributes guess of all training examples, the model could actually set any fairness value regarding the sensitive attributes corrected reconstruction. Thus, the knowledge of the exact training unfairness value (rather than a simple upper bound) could easily be used to reduce the set of acceptable reconstructions and enhance the performance of the reconstruction correction. Finally, because it explicitly encodes each training example's sensitive attribute, $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$ can be used to formulate *any* constraint using such attributes.

### 3.2.3 Efficient Model for Statistical Fairness

The search space of the reconstruction correction model $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$ grows exponentially with the number of training examples $N$. As each element of the sensitive attributes vector $S$ is considered independently from the others (and represented as a binary decision variable), the search space of this model is $O(2^N)$, which lim-

its its scalability. However, when considering statistical fairness metrics, one does not need such granularity. More precisely to satisfy the fairness constraint, the reconstruction corrector may consider exactly four different moves: (i) flipping an element of the reconstructed sensitive attributes $\hat{s}_j$ from 1 to 0, for an example with prediction $\hat{y}_j = 1$, (ii) flipping $\hat{s}_j$ from 0 to 1, for an example with prediction $\hat{y}_j = 1$, (iii) flipping $\hat{s}_j$ from 1 to 0, for an example with prediction $\hat{y}_j = 0$, or (iv) flipping $\hat{s}_j$ from 0 to 1, for an example with prediction $\hat{y}_j = 0$. Then, for the chosen move, the model will always select the example with the lowest confidence score (and then, eventually, the second lower and so on), which drastically reduces the size of the search space as we explain below.

Let $n_1^+$ be the number of training examples positively predicted by the target model and assigned to group 1 by the initial adversary's guess: $n_1^+ = \sum_{j=1}^{N} \hat{s}_j \cdot \hat{y}_j$. Similarly, let:

$$n_0^+ = \sum_{j=1}^{N} (1 - \hat{s}_j) \cdot \hat{y}_j, \quad n_1^- = \sum_{j=1}^{N} \hat{s}_j \cdot (1 - \hat{y}_j), \quad \text{and} \quad n_0^- = \sum_{j=1}^{N} (1 - \hat{s}_j) \cdot (1 - \hat{y}_j).$$

The four numbers $n_1^+$, $n_0^+$, $n_1^-$ and $n_0^-$ are the cardinalities of the four groups of examples defining the four possible moves (respectively, (i), (ii), (iii) and (iv)) from a fairness perspective. For each group, we sort and cumulate the confidence scores associated to its examples and obtain the following arrays: $T_{1^+}$, $T_{0^+}$, $T_{1^-}$ and $T_{0^-}$. For instance, $T_{1^+}$ contains the confidence scores associated to the $n_1^+$ training examples positively predicted by the target model and assigned to group 1 by the initial adversary's guess. $T_{1^+}[j]$ is the sum of the $j$ lowest confidence scores among this group. Indeed, $T_{1^+}[j]$ is the exact minimal cost of switching the final reconstruction guess from 1 to 0 for $j$ examples positively predicted by the target model. We use four positive integer decision variables, modeling the number of times each of the four moves is performed to correct the reconstruction. We now define our efficient model for sensitive attributes reconstruction correction: $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$.

**Inputs**

- Original guesses cardinalities $n_1^+$, $n_0^+$, $n_1^-$ and $n_0^-$.
- Arrays of sorted and cumulated adversary's probabilities for each original guess : $T_{1^+}$, $T_{0^+}$, $T_{1^-}$ and $T_{0^-}$.
- Fairness information: $h$ satisfies fairness constraints for some metric (*e.g.*, SP) and some tolerance $\varepsilon$.

**Decision variables**

- $s_{01}^+ \in \{0, \ldots, n_0^+\}$: number of changes of $\hat{s_j}$ from 0 to 1, for examples such that $\hat{y}_j = 1$.

- $s_{10}^{+} \in \{0, \ldots, n_1^{+}\}$: number of changes of $\hat{s_j}$ from 1 to 0, for examples such that $\hat{y}_j = 1$.
- $s_{01}^{-} \in \{0, \ldots, n_0^{-}\}$: number of changes of $\hat{s_j}$ from 0 to 1, for examples such that $\hat{y}_j = 0$.
- $s_{10}^{-} \in \{0, \ldots, n_1^{-}\}$: number of changes of $\hat{s_j}$ from 1 to 0, for examples such that $\hat{y}_j = 0$.

**Constraint programming model $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$**

$$\min T_{0^+}[s_{01}^{+}] + T_{1^+}[s_{10}^{+}] + T_{0^-}[s_{01}^{-}] + T_{1^-}[s_{10}^{-}] \tag{3.5}$$

$$s.t. : n_0^{+} + n_0^{-} - s_{01}^{+} - s_{01}^{-} + s_{10}^{+} + s_{10}^{-} > 0 \tag{3.6}$$

$$n_1^{+} + n_1^{-} - s_{10}^{+} - s_{10}^{-} + s_{01}^{+} + s_{01}^{-} > 0 \tag{3.7}$$

$$-\varepsilon \leq \frac{\sum_{j=1}^{N} \hat{y}_j}{N} - \frac{n_1^{+} - s_{10}^{+} + s_{01}^{+}}{n_1^{+} + n_1^{-} - s_{10}^{+} - s_{10}^{-} + s_{01}^{+} + s_{01}^{-}} \leq \varepsilon \tag{3.8}$$

$$-\varepsilon \leq \frac{\sum_{j=1}^{N} \hat{y}_j}{N} - \frac{n_0^{+} - s_{01}^{+} + s_{10}^{+}}{n_0^{+} + n_0^{-} - s_{01}^{+} - s_{01}^{-} + s_{10}^{+} + s_{10}^{-}} \leq \varepsilon \tag{3.9}$$

Similarly to the general model, the objective (3.5) minimizes the confidence-weighted sum of the changes. It can be efficiently implemented using `element` constraints within a Constraint Programming (CP) solver. Such constraints are used to access a data array at index given by the value of a variable: $T_{0^+}[s_{01}^{+}] =$ `element`$(T_{0^+}, s_{01}^{+})$. Furthermore, when minimizing only the number of changes, one could simply sum the four decision variables. The objective would then become linear as the whole model which could be solved using off-the-shelf Mixed Integer Linear Programming solvers.

Constraints (3.6) and (3.7) simply ensure that the reconstruction contains at least one example from each protected group. Finally, constraints (3.8) and (3.9) encode the fairness constraint for the Statistical Parity metric. As in $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$, the two constraints can easily be linearized because the denominator of the first term is a constant ($N$). More generally, $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$ could be used to encode any rate constraints on the target model's outputs (using the sensitive attributes), including (but not restricted to) all statistical fairness metrics.

Once the model is solved, optimal assignments of the four decision variables define the (confidence-weighted) minimal number of moves that must be done to ensure fairness. In a post-processing step, the associated moves are performed to the corresponding examples in an increasing order of the confidence scores (so that the overall cost is exactly the objective value (3.5) of the solved model). This results in the corrected reconstruction vector $S^*$. One can notice that $S^*$ is also an optimal solution to the general reconstruction correction model $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$. Indeed, as stated in Theorem 1, both models share the same set of optimal solutions, even though their encodings of such solutions differ. The difference is that some non-optimal solutions to the general model $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$ do not correspond to any

solution to our efficient model $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$ (*i.e.*, they are simply not part of its search space). Such solutions are all the assignments in which the corrector makes one of the four aforementioned moves but does not select the example with the lowest confidence score (which in this context does not make sense).

**Theorem 1** (Equivalence of models)**.** *In the context of statistical fairness constraints, the general reconstruction correction model $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$ and the efficient one $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$ share the same set of optimal solutions.*

*Proof. (a) Any optimal solution to $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$ corresponds to a solution to $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$.* Let $S^*$ be an optimal solution to $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$. Then, count the number of performed changes of each type between $S^*$ and $\hat{S}$ (*i.e.*, for an example $j$ with $\hat{y}_j = 0$ (or 1), switching $\hat{s}_j$ from 0 to 1 (or the contrary)). When performing such changes, the solver must have chosen the examples with the lowest confidence scores, or else another solution also satisfies the fairness constraint and has a better objective function value, which contradicts the optimality hypothesis. Afterwards, $S^*$ corresponds to a solution to $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$, represented by the counts for the four moves. Indeed, application of the aforementioned post-processing procedure then allows to retrieve $S^*$.

*(b) Any solution to $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$ corresponds to a solution to $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$.* Consider a solution to $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$ and then apply the post-processing step aforementioned. The obtained reconstruction vector is a solution to $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$.

*(c) The objective function value of any solution of $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$ is the same in $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$ and $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$.* Consider a solution to $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$ with objective value $o$ and apply the aforementioned post-processing step before plugging the resulting reconstruction vector into $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$. By construction, the objective value of this solution of $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$ will be exactly $o$.

Overall, by *(a)*, *(b)*, and *(c)*, each optimal solution to one of the models is also an optimal solution to the other. □

Model $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$ uses four variables whose total sum cannot exceed $N$. Its search space is then $O(N^4)$, which is polynomial in the training set cardinality. Our resolution method also requires some polynomial $O(N \cdot log(N))$ pre-processing and $O(N)$ post-processing computations, which does not modify the overall solving complexity. Overall, for statistical fairness constraints, solving our new model is equivalent to solving the general one, but with polynomial search space instead of exponential one. In practice, this will lead to running times smaller by several orders of magnitude.

**Remark 1.** *Designing an ad-hoc algorithm for reconstruction correction would also be possible. Because the size of the search space explored in this subsection is polynomial in the training set cardinality, such a dedicated algorithm would have at worst polynomial $O(N^4)$ complexity, which corresponds to an exhaustive enumeration. However, it would not yield the flexibility of the proposed declarative programming approach, which can easily handle additional or different constraints.*

*Furthermore, the potential running time improvements would be relatively modest, as $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$ is solved to optimality within fractions of seconds in all our experiments, even for the largest datasets - which is negligible compared to the running times of the baseline adversaries.*

### 3.2.4 Generalizing the Reconstruction Correction

The proposed models directly encode the Statistical Parity fairness constraints, but can also be used to correct sensitive attributes reconstructions from all the other metrics of Table 3.1. Recall that the Predictive Equality (PE) metric equalizes the False Positive rates (across the protected groups), which is equivalent to satisfying Statistical Parity over the negatively-labelled subset of the training set. Then, one can simply use the reconstruction correction model on the negatively-labelled subset of the training set. Indeed, PE gives no information on the positively-labelled subset of the training set. Similarly, Equal Opportunity (EOpp) equalizes the True Positive rates, and reconstruction can be achieved using the proposed model on the positively-labelled subset of the training set. Finally, dealing with the Equalized Odds (EO) metric can be done by successively applying Predictive Equality and Equal Opportunity reconstruction corrections. The overall corrected reconstruction is still guaranteed to be optimal because the two successive reconstructions are completely independent: they work on disjoint subsets of the training set, hence optimizing over disjoint subsets of variables (corrected sensitive attributes). Overall, the model proposed for the Statistical Parity metric can actually be used for any of the statistical fairness metrics of Table 3.1, by applying the reconstruction correction on the appropriate data slice(s).

An important remark is that while the true labels $Y$ are not required when dealing with the statistical parity metric (and are not used to build our proposed reconstruction correction models, as shown in Sections 3.2.2 and 3.2.3), they are necessary to handle the metrics discussed in the previous paragraph (PE, EOpp, and EO). Indeed, one must first select the appropriate subset of the entire training set based on the labels' values, before applying the reconstruction correction (either using $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$ or $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$) on this subset.

Observe that even though $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$ is proposed for the binary sensitive attributes setting, it could easily be generalized by adapting the domains of the $s_j^*$ variables and adding the appropriate cardinalities and fairness constraints for the additional groups. Extending $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$ can also be done by declaring additional variables and constraints. Appendix D depicts how both models can be extended to the general case of multi-valued sensitive attributes, along with a discussion regarding the resulting complexity.

## 3.3 Experimental Study

In this section, we present our large experimental study regarding the proposed reconstruction framework. More precisely, we consider a wide range of scenarios

using two fair learning algorithms intervening at different stages of the machine learning pipeline, three datasets of various sizes with diverse sensitive attributes, four fairness metrics and a variety of unfairness tolerances. First, we describe our baseline adversaries before detailing the experimental setup and the results obtained.

### 3.3.1 Baseline Adversaries Initial Reconstruction

We instantiate the framework described in Figure 3.1 with two different baseline adversaries, $\mathcal{A}$ and $\mathcal{A}'$, which are introduced separately hereafter. Knowledge of both adversaries is further summarized in Table 3.2. In line with the reconstruction literature [Dinur & Nissim 2003, Dwork *et al.* 2007, Gadotti *et al.* 2019], we consider that the dataset contains a *"large amount of nonprivate identifying information and a secret bit, one per individual"* [Dwork *et al.* 2017]. Here, the private bit of every individual $j$ is his sensitive attribute $s_j$. Both adversaries hence know the training set non-sensitive attributes vector $X$ and ground truth labels $Y$ (*i.e.*, all training set columns except the *secret* one, which is the sensitive attribute in our case). Furthermore, both adversaries have access to an auxiliary *attack set*, $\mathcal{D}_A = (X_A, S_A, Y_A)$ drawn from the same distribution as the actual training set. This attack set models the knowledge of an approximation of the distribution of the sensitive attribute with respect to the non-sensitive ones and the ground truth label. Indeed, the use of such *attack set* to train an *attack model* is in line with the literature [Aalmoes *et al.* 2022].

**Adversary $\mathcal{A}$.** Adversary $\mathcal{A}$ can be used to estimate to what extent general knowledge about the distribution (of the sensitive attributes with respect to the non-sensitive ones and the ground truth label) can be leveraged to reconstruct the sensitive attributes of the training set. Indeed, it does not have any knowledge about the sensitive attributes singularities of the training set, as $S$ is not used directly or indirectly for any of its inputs. As aforementioned, adversary $\mathcal{A}$ has access to the training set non-sensitive attributes vector $X$ and ground truth labels $Y$, and owns an auxiliary attack set $\mathcal{D}_A = (X_A, S_A, Y_A)$. It relies on such attack set to train a machine learning model (coined *attack model*) to predict $S_A$ from $(X_A, Y_A)$. Adversary $\mathcal{A}$ then uses his trained *attack model* to predict $\hat{S}$ from $(X, Y)$.

**Adversary $\mathcal{A}'$.** Adversary $\mathcal{A}'$ has access to all information that our reconstruction correction will later use, which constitutes the strongest baseline possible to compare against our reconstruction correction. Furthermore, it corresponds to the adversary proposed in [Aalmoes *et al.* 2022]. More precisely, $\mathcal{A}'$ also has access to the auxiliary attack set $\mathcal{D}_A = (X_A, S_A, Y_A)$, and to the training set non-sensitive attributes $X$ and ground truth labels $Y$ (just like $\mathcal{A}$). However, $\mathcal{A}'$ also knows the target model's predictions on the training set $\hat{Y} = h(X)$ and on the attack set $\hat{Y}_A = h(X_A)$. Adversary $\mathcal{A}'$ relies on the attack set to train an *attack model* to predict $S_A$ from $(X_A, Y_A, \hat{Y}_A)$. He then uses his trained *attack model* to predict $\hat{S}$ from $(X, Y, \hat{Y})$.

Table 3.2: Summary of the knowledge of the considered baseline adversaries, introduced in Section 3.3.1.

| | **Auxiliary attack set** $\mathcal{D}_A = (X_A, S_A, Y_A)$ | **Training set non-sensitive attributes vector and true labels** $(X, Y)$ | **Target model predictions** | |
|---|---|---|---|---|
| | | | **Training set** $\hat{Y} = h(X)$ | **Attack set** $\hat{Y}_A = h(X_A)$ |
| $\mathcal{A}$ | ✓ | ✓ | ✗ | ✗ |
| $\mathcal{A}'$ | ✓ | ✓ | ✓ | ✓ |

### 3.3.2 Confidence Scores Calibration

The attack models perform binary classification, hence their confidence scores lie between 0.5 and 1.0. Using these scores directly to weight our reconstruction correction problem would imply that modifying a prediction with confidence 1.0 (the attacker was certain about it) is better than modifying two predictions with confidence 0.51 (the attacker was unsure). To encourage the reconstruction correction to target the predictions with the lowest scores, we normalize all confidence scores and exponentiate them in order to enlarge their differences. In practice, all the normalized scores are set to the power of $k$, in which $k$ is chosen to maximize reconstruction correction accuracy on part of the attacker's data used as a validation set. However, other confidence scores processing techniques are possible and may improve the reconstruction correction step. For instance, an adversary could learn how to best discriminate the confidence scores between correct and incorrect predictions on his attack set. Preliminary experiments using such supervised confidence scores processing did not show significant improvements over our simple exponentiation technique. Overall, each adversary outputs a guess $\hat{S} = \{\hat{s}_{j\in\{1...N\}}\}$ for the sensitive attributes vector, along with a confidence vector $P = \{p_{j\in\{1...N\}}\}$.

### 3.3.3 Setup

Table 3.3: Summary of the datasets used in our sensitive attributes reconstruction correction experiments.

| **Dataset** | **Binary Prediction Task** | **#Examples** | **#Non-Sensitive Features** | **Sensitive Feature** |
|---|---|---|---|---|
| UCI Adult Income [Dua & Graff 2017] | Income above \$50K | 45,222 | 7 categorical, 6 numerical | Gender (Male/Female) |
| ACSPublicCoverage* [Ding *et al.* 2021] | Coverage from public health insurance | 98,928 | 17 categorical, 1 numerical | Age (First Quartile /Others) |
| ACSIncome* [Ding *et al.* 2021] | Income above \$50K | 135,924 | 7 categorical, 2 numerical | Race Code (White/Other) |

*(Texas State, 2018)

**Datasets.** To obtain sufficiently diverse scenarios, we consider three datasets of the fairness literature with different sizes, each with a different binary sensitive attribute. The first one is the UCI Adult Income dataset [Dua & Graff 2017], which gathers records about the 1994 US Census database, with the classification task being to predict whether individuals earn more than $\$50,000$ per year. The considered

sensitive attribute is gender (female/male). We also consider two datasets built from the American Community Survey (ACS) Public Use Microdata Sample (PUMS) of the US Census Bureau. More precisely, the datasets are built from data collected in the Texas state in 2018. The second dataset, ACSPublicCoverage [Ding *et al.* 2021], contains data about individuals under the age of 65, with an income of less than $\$30,000$, with the classification task being to predict whether they are covered by public health insurance. Here, age is used as the sensitive attribute (younger quartile/others). The third dataset, ACSIncome [Ding *et al.* 2021], gathers records about individuals above the age of 16, who reported usual working hours of at least 1 hour per week in the past year, and an income of at least $100. Similar to the original UCI Adult Income dataset, the classification task is to predict whether individuals earn more than $\$50,000$ per year. We rely on the binarized race (white/others) as the sensitive attribute.

Table 3.3 summarizes the datasets used in our experiments. For all experiments, each dataset is split between a training set $\left(\frac{1}{3}\right)$, a test set $\left(\frac{1}{3}\right)$ and an attack set $\left(\frac{1}{3}\right)$. The test set is only used to ensure that the fair target model is trained appropriately (in particular, to show that it does not overfit). The attack set is known by the baseline adversary (see Section 3.3.1).

**Target Fair Models.** To validate our approach, we have tested two off-the-shelf fair learning methods implemented in the `Fairlearn` library [Bird *et al.* 2020]: one in-processing method, ExponentiatedGradient [Agarwal *et al.* 2018], as well as a post-processing method, ThresholdOptimizer [Hardt *et al.* 2016]. Both approaches are introduced with more details in Section 1.3.3. In a nutshell, ExponentiatedGradient [Agarwal *et al.* 2018] formulates the fair classification problem as a sequence of cost-sensitive classification problems. Given a cost-sensitive *base learner*, it follows a two-player game structure in which one player trains the base learner while the other adapts the training examples weights. ThresholdOptimizer [Hardt *et al.* 2016] takes as input a trained (possibly unfair) classifier and computes group-specific thresholds on the outputs of the classifier to *adjust* its predictions. The thresholds are optimized to enforce some fairness constraints while having minimal impact on classification accuracy.

By using two fair learning techniques intervening at different steps of the machine learning pipeline, we want to emphasize that our method is completely agnostic to the type of fairness intervention. Indeed, the only information used by our reconstruction correction strategy is the final fairness information, along with the predictions of the model. Both methods require the use of a base learner: the ExponentiatedGradient method uses it to iteratively solve the cost-sensitive classification problems, while the ThresholdOptimizer post-processes its predictions after training. We use `scikit-learn` [Pedregosa *et al.* 2011] Decision Tree classifiers as base learners with the maximum depth being set to 8 and all other parameters left to their default values.

**Fairness Metrics.** We run experiments for the four fairness metrics presented in Table 3.1. Experiments using the ExponentiatedGradient method use 49 different values of the unfairness tolerance $\varepsilon$, ranging non-linearly from 0.0 (exact fairness) to 0.20 (loose constraint). The ThresholdOptimizer method modifies the initial model's predictions to approximate 0.0 unfairness, so we cannot vary the unfairness tolerance here.

**Attack Models.** The attack models used by our baseline adversaries are `scikit-learn` [Pedregosa *et al.* 2011] Random Forest classifiers, which are known to be resistant to overfitting and generalize well in many situations. This hypothesis class was chosen based on thorough preliminary experiments. To handle sensitive attributes imbalance [Aalmoes *et al.* 2022], we use a class-balanced loss. The Random Forest hyperparameters are optimized using the `HyperOpt-Sklearn` framework [Komer *et al.* 2014], with a maximum of 100 evaluations for its Tree of Parzen Estimators search algorithm. This setup ensures that the baseline adversary implements a strong baseline and is in line with the literature.

**Reconstruction Correction.** Our efficient reconstruction correction model $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$ (depicted in Section 3.2.3) is implemented and solved using the `IBM ILOG CP Optimizer Version 12.10`[1] via the `DOcplex`[2] Python Modeling API (version 2.21.207) and its default configuration. The number of threads used in CP Optimizer is set to 1 and the optimality tolerance (absolute and relative) is set to 0.0. Indeed, due to the probabilities exponentiation process presented in Section 3.3.1, some values can be very small and would lie below the solver's default optimality tolerance. Our reconstruction correction method is implemented as a Python class and is available on our repository[3].

**Experimental Parameters.** We set a one minute timeout for the reconstruction correction step (model creation and solving). It was never reached in practice, and all models were solved to optimality in less than a few seconds (less than one second in average). Each experiment is repeated 100 times, with different seeds for the data split process and the random state of the algorithms. The results are averaged over the 100 runs and the standard deviation is reported. All experiments are run on a computing cluster over a set of homogeneous nodes using Intel Xeon E5-2683 v4 Broadwell @ 2.1GHz CPU.

### 3.3.4 Results

#### 3.3.4.1 Experiments using the ExponentiatedGradient technique

In this section, we report the results for adversary $\mathcal{A}'$. Results for adversary $\mathcal{A}$, which are provided in our full paper [Ferry *et al.* 2023a], are almost perfectly iden-

[1] `https://www.ibm.com/analytics/cplex-cp-optimizer`
[2] `http://ibmdecisionoptimization.github.io/docplex-doc/`
[3] `https://github.com/ferryjul/SensitiveAttributesReconstructionCorrector/`

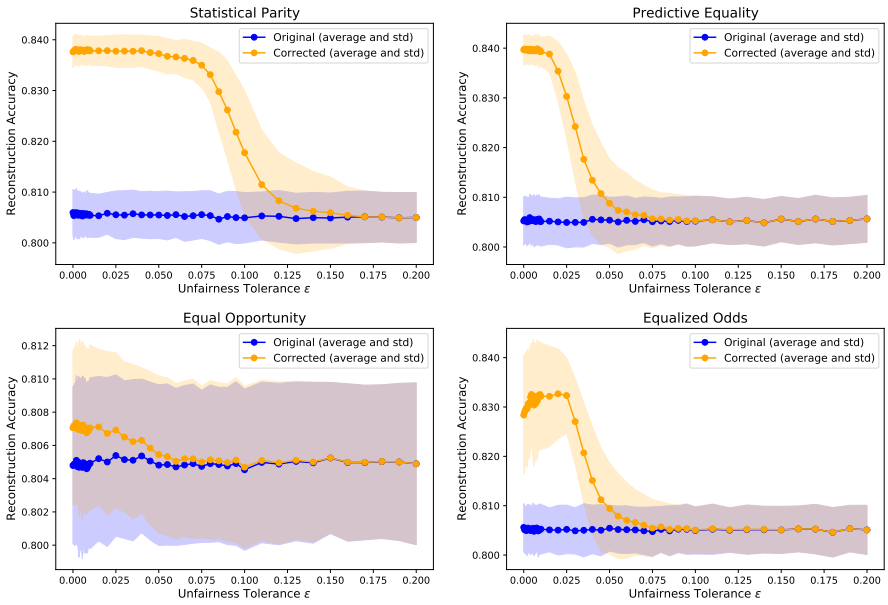



Figure 3.2: Corrected and original (adversary $\mathcal{A}'$) reconstruction quality, for our experiments using the ExponentiatedGradient in-processing fairness enhancing method with four fairness metrics, on the UCI Adult Income dataset.

tical and follow the same trends. This suggests that the adversary $\mathcal{A}'$ is not able to leverage the additional knowledge. One possible explanation is that contrary to our approach, its attack model does not explicitly encode the fairness information in a structured manner. The training and test performances of the target fair models are also reported in our published work, and consistently show that the learnt fair models do not overfit. As expected, training accuracy and unfairness both increase when the fairness constraint is relaxed (*i.e.*, $\varepsilon$ increases). Due to the models' relatively good generalization, test accuracy and unfairness follow the same trends.

Results of our experiments using the ExponentiatedGradient in-processing method [Agarwal *et al.* 2018] are displayed for the three considered datasets and the four fairness metrics in Figures 3.2, 3.3 and 3.4. The reconstruction accuracy results demonstrate the effectiveness of the proposed approach. As the adversary $\mathcal{A}'$ exploits all the information that our reconstruction correction uses, any further improvement in the reconstruction accuracy can only be explained by the semantics of the fairness constraint integrated in our Reconstruction Corrector model. Recall that the reconstruction accuracy is the proportion of training examples $e_j$ for which the sensitive attribute $s_j \in S$ was correctly reconstructed (in the baseline attacker original guess $\hat{s}_j \in \hat{S}$ or in the corrected one $s_j^* \in S^*$).

One can observe that the corrected reconstruction is always more accurate than the adversary's original guess, which means that the changes made by the reconstruction correction model are correct most of the time. Furthermore, the corrected reconstruction accuracy gets better as the fairness constraint becomes tighter (*i.e.*,

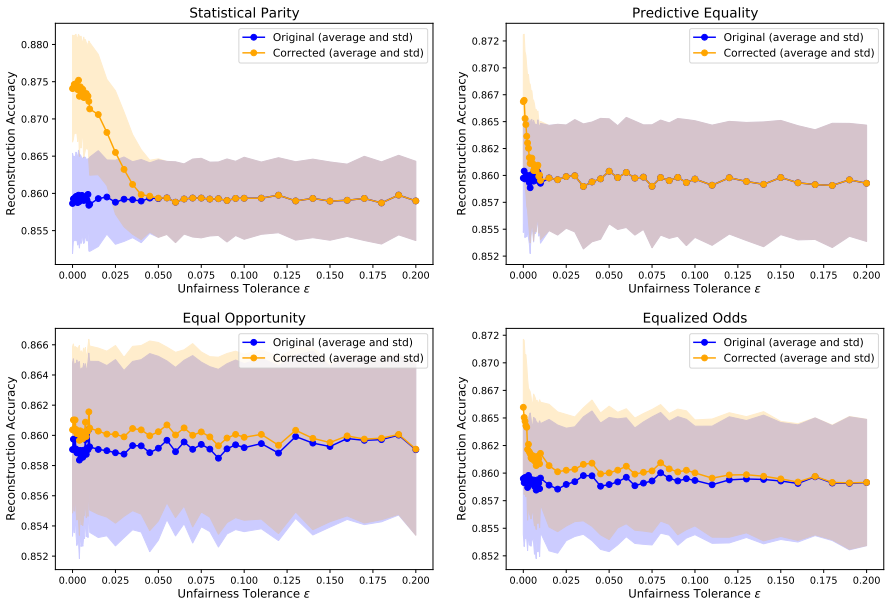



Figure 3.3: Corrected and original (adversary $\mathcal{A}'$) reconstruction quality, for our experiments using the ExponentiatedGradient in-processing fairness enhancing method with four fairness metrics, on the ACSPublicCoverage dataset.

lower values of the unfairness tolerance $\varepsilon$). Indeed, the reconstruction accuracy improvement is related to the amount of bias mitigated by the fair learning technique, which in turn depends on the considered fairness metric, the unfairness tolerance and the original data bias. For tight fairness constraints, we observe reconstruction accuracy absolute improvements up to 0.06, as in the experiments using the Statistical Parity metric on the ACSIncome dataset (Figure 3.4, top left). Such improvements are due to the fairness information, which is the only constraint of our correction models.

Recall that the Predictive Equality (respectively Equal Opportunity) metric only applies to the negatively-labelled (respectively positively-labelled) training examples. This means that such metrics can only help in partially correcting the adversary's guess (as described in Section 3.2.4). Because the datasets used are imbalanced, with the majority of training examples belonging to the negative class, the Equal Opportunity metric relates only to a minority of training examples. As a result, the reconstruction accuracy improvement is more modest than for the remaining metrics. Indeed, even with a close rate of correct modifications, the number of corrections applied (and thus the overall improvement) is smaller.

When varying the unfairness tolerance $\varepsilon$, the only input of the reconstruction methods that is modified is the fair model's predictions $\hat{Y}$ (and the fairness information). The fact that the reconstruction accuracy of the baseline adversary $\mathcal{A}'$ is rather constant across variations of $\varepsilon$ shows that the fair model's predictions $\hat{Y}$ are not used a lot by the learnt attack models. In contrast, as our method knows

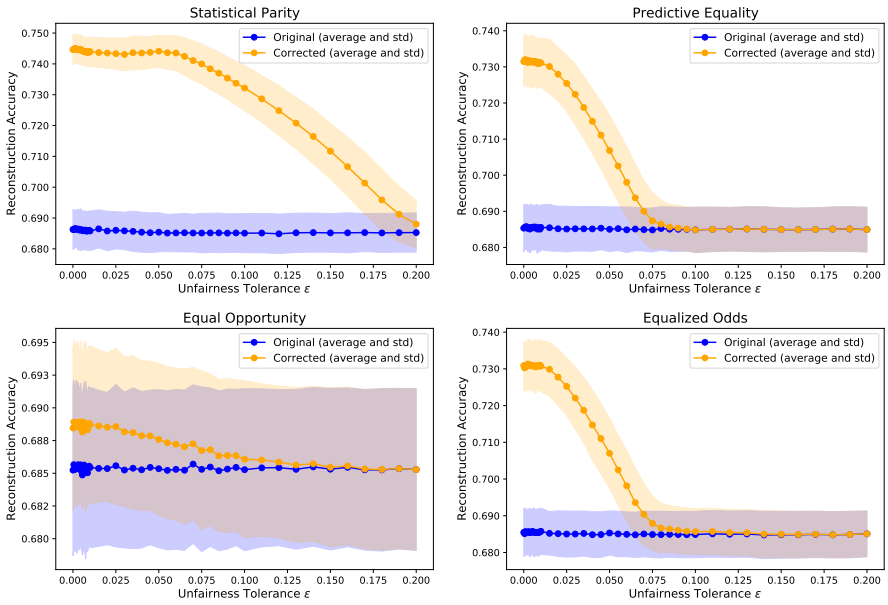



Figure 3.4: Corrected and original (adversary $\mathcal{A}'$) reconstruction quality, for our experiments using the ExponentiatedGradient in-processing fairness enhancing method with four fairness metrics, on the ACSIncome dataset.

exactly how to interpret the fairness information with respect to $\hat{Y}$, it is able to exploit it to significantly improve the final reconstruction accuracy.

Finally, the empirical results show that our reconstruction correction method is able to considerably improve the reconstruction accuracy of the training set sensitive attributes, even when the original adversary is as informed as our method.

#### 3.3.4.2 Experiments using the ThresholdOptimizer technique

Results of our experiments using the ThresholdOptimizer [Hardt *et al.* 2016] fair post-processing method are displayed in Table 3.4. The observed trends are similar to that of the previous subsection, which demonstrates that the type of fairness intervention does not impact our framework. One can observe that the performances of both baseline adversaries are very close. As he possesses more information than $\mathcal{A}$, $\mathcal{A}'$ always performs better on the attack set (used to train the attack models). However, his generalization is sometimes poorer, resulting in worse reconstruction performances when used on the target model training set. This may be due to the distribution of the target fair model's predictions on its own training set $\hat{Y}$ being different from that on the adversary's attack set $\hat{Y}_A$.

Importantly, we observe that the reconstruction correction step always improves the reconstruction accuracy. Indeed, the improvement obtained depends on the considered fairness metric and on the original bias of the reconstruction (which is related to the inherent bias of the original training set). The reconstruction accuracy

Table 3.4: Summary of the results of our sensitive attributes reconstruction correction experiments using a post-processing method for fairness.

| Metric | Target model $h$ (under attack) | | | | Reconstruction Accuracy | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | | Baseline | | Corrected | |
| | *Train Acc.* | *Test Acc.* | *Train Unf.* | *Test Unf.* | $\mathcal{A}$ | $\mathcal{A}'$ | $\mathcal{A}$ | $\mathcal{A}'$ |
| **UCI Adult Income dataset** | | | | | | | | |
| SP | 0.820 ±0.008 | 0.808 ±0.009 | 0.003 ±0.002 | 0.005 ±0.003 | 0.808 ±0.005 | 0.814 ±0.006 | 0.851 ±0.003 | **0.858 ±0.005** |
| PE | 0.849 ±0.005 | 0.836 ±0.006 | 0.002 ±0.001 | 0.003 ±0.003 | 0.808 ±0.005 | 0.807 ±0.005 | 0.843 ±0.003 | **0.844 ±0.004** |
| EOpp | 0.857 ±0.005 | 0.845 ±0.005 | 0.005 ±0.005 | 0.041 ±0.023 | 0.808 ±0.005 | 0.805 ±0.005 | **0.810 ±0.005** | 0.807 ±0.005 |
| EO | 0.846 ±0.006 | 0.834 ±0.007 | 0.007 ±0.006 | 0.037 ±0.021 | 0.808 ±0.005 | 0.807 ±0.004 | 0.839 ±0.008 | **0.840 ±0.009** |
| **ACSPublicCoverage dataset** | | | | | | | | |
| SP | 0.861 ±0.003 | 0.851 ±0.003 | 0.001 ±0.001 | 0.003 ±0.002 | 0.861 ±0.005 | 0.860 ±0.006 | 0.874 ±0.005 | **0.875 ±0.007** |
| PE | 0.861 ±0.002 | 0.853 ±0.002 | 0.001 ±0.000 | 0.003 ±0.002 | 0.861 ±0.005 | 0.860 ±0.005 | 0.864 ±0.005 | **0.870 ±0.007** |
| EOpp | 0.851 ±0.005 | 0.843 ±0.004 | 0.002 ±0.002 | 0.022 ±0.011 | 0.861 ±0.005 | 0.859 ±0.006 | **0.862 ±0.004** | 0.861 ±0.006 |
| EO | 0.841 ±0.004 | 0.833 ±0.004 | 0.003 ±0.002 | 0.023 ±0.011 | 0.861 ±0.005 | 0.860 ±0.005 | **0.862 ±0.004** | 0.861 ±0.005 |
| **ACSIncome dataset** | | | | | | | | |
| SP | 0.788 ±0.003 | 0.776 ±0.003 | 0.002 ±0.001 | 0.005 ±0.004 | 0.690 ±0.007 | 0.715 ±0.010 | 0.756 ±0.005 | **0.764 ±0.006** |
| PE | 0.797 ±0.002 | 0.785 ±0.002 | 0.001 ±0.001 | 0.004 ±0.003 | 0.690 ±0.007 | 0.688 ±0.007 | **0.736 ±0.007** | 0.735 ±0.006 |
| EOpp | 0.796 ±0.003 | 0.784 ±0.003 | 0.001 ±0.001 | 0.010 ±0.007 | 0.690 ±0.007 | 0.685 ±0.006 | **0.693 ±0.007** | 0.689 ±0.006 |
| EO | 0.795 ±0.003 | 0.783 ±0.003 | 0.002 ±0.001 | 0.010 ±0.006 | 0.690 ±0.007 | 0.688 ±0.007 | **0.737 ±0.007** | 0.735 ±0.006 |

improvements over the two baseline adversaries are of the same magnitude than with the ExponentiatedGradient method. Again, reconstruction correction using the Equal Opportunity metric offers modest improvements due to the fact that it applies to a minority of training examples.

## 3.4 Discussion on Countermeasures

Previously, we have seen that the proposed reconstruction correction method is able to exploit the fairness information to significantly improve the reconstruction accuracy, even with an informed adversary. In this section, we discuss possible countermeasures to limit the effectiveness of the reconstruction correction step.

### 3.4.1 Differential Privacy

Differential Privacy (DP) [Dwork *et al.* 2006, Dwork & Roth 2014] (introduced in Section 1.5.4) is considered to be one of the state-of-the-art methods for preventing

inference attacks against machine learning models. While it may affect the performances of a baseline adversary, DP cannot be an effective countermeasure to our proposed reconstruction correction step. Indeed, it is designed to ensure that the output of a mechanism does not rely too much on any single example, but rather on general patterns. However, statistical fairness metrics are measured over an entire dataset and do not specifically rely on individual examples. Thus, as our reconstruction correction method only relies on group-level statistics, DP cannot effectively affect its performances [Cormode 2010].

Additionally, DP is incompatible with the strict respect of any statistical fairness measure [Cummings *et al.* 2019, Agarwal 2021b]. Indeed, releasing a model along with information regarding its strict respect of any statistical fairness constraint is intrinsically non-DP compliant.

### 3.4.2 Hiding the Fairness Information

Intuitive countermeasures consist in perturbing the fairness information (type of fairness metric used or unfairness tolerance parameter $\varepsilon$). Note that this may not be possible when a particular fairness requirement is also a legal requirement, as for the "80 percent rule" for Statistical Parity [Feldman *et al.* 2015] stated by the US Equal Employment Opportunity Commission (EEOC) [EEOC. 1979]. When possible, releasing noisy or empty fairness information may be a reasonable defense mechanism. However, adversaries may still use diverse strategies to infer both the fairness metric that was optimized and the unfairness tolerance parameter. Depending on the adversarial knowledge, such property inference attacks [Cristofaro 2020] might give a good estimation to the adversary, which we can expect would still allow reasonable reconstruction correction performances from our approach (for which the fairness information is simply an input). Using our baseline adversaries $\mathcal{A}$ or $\mathcal{A}'$, a simple strategy would be to quantify the target model unfairness on the attack set $\mathcal{D}_A$ for different metrics. Then, one can select the metric with the smallest measured unfairness, and consider that the model is fair for this metric with unfairness tolerance $\varepsilon$ equal to the measured unfairness. To assess its effectiveness, we implemented this fairness information estimation strategy and performed our experiments again.

Results for the experiments using the ThresholdOptimizer [Hardt *et al.* 2016] method are reported in Table 3.5. We report the performances of the fairness constraint estimation process, namely the rate of correct metric identification, and the average unfairness tolerance inferred. Due to the simple estimation process, the Equalized Odds metric can never be identified as its violation is the maximum of the Predictive Equality and Equal Opportunity violations (hence it can never be the smallest value). However, for the other metrics we observe that even this simple estimation process is often able to correctly identify the optimized metric. Several trends can be noted when comparing the reconstruction results with those of Table 3.4, in which the reconstruction correction is done using the actual fairness constraint. A first situation occurs when the fairness constraint is correctly

Table 3.5: Summary of the results of our sensitive attributes reconstruction correction experiments using a post-processing method for fairness, for the simple countermeasure of not revealing the fairness information. *Reconstruction results have to be compared with those of Table 3.4.*

| **Metric** | **Estimated Constraint** | | **Corrected Reconstruction Accuracy (Estimated Constraint)** | |
|---|---|---|---|---|
| | ***%Correct Metric*** | ***Average Tolerance*** | $\mathcal{A}$ | $\mathcal{A}'$ |
| **UCI Adult Income dataset** | | | | |
| SP | 0.95 | $0.004 \pm 0.003$ | $0.848 \pm 0.009$ | $0.856 \pm 0.011$ |
| PE | 0.97 | $0.003 \pm 0.002$ | $0.841 \pm 0.006$ | $0.843 \pm 0.007$ |
| EOpp | 0.26 | $0.018 \pm 0.010$ | $0.829 \pm 0.012$ | $0.828 \pm 0.013$ |
| EO | 0.00 | $0.005 \pm 0.005$ | $0.841 \pm 0.006$ | $0.843 \pm 0.007$ |
| **ACSPublicCoverage dataset** | | | | |
| SP | 1.00 | $0.002 \pm 0.002$ | $0.873 \pm 0.005$ | $0.873 \pm 0.009$ |
| PE | 1.00 | $0.003 \pm 0.002$ | $0.863 \pm 0.005$ | $0.865 \pm 0.007$ |
| EOpp | 0.28 | $0.008 \pm 0.005$ | $0.862 \pm 0.005$ | $0.862 \pm 0.005$ |
| EO | 0.00 | $0.002 \pm 0.002$ | $0.868 \pm 0.006$ | $0.869 \pm 0.007$ |
| **ACSIncome dataset** | | | | |
| SP | 0.80 | $0.003 \pm 0.003$ | $0.743 \pm 0.026$ | $0.754 \pm 0.020$ |
| PE | 0.86 | $0.003 \pm 0.003$ | $0.729 \pm 0.016$ | $0.728 \pm 0.016$ |
| EOpp | 0.73 | $0.008 \pm 0.006$ | $0.704 \pm 0.019$ | $0.700 \pm 0.020$ |
| EO | 0.00 | $0.002 \pm 0.002$ | $0.723 \pm 0.021$ | $0.721 \pm 0.022$ |

inferred, which is the case in all experiments using the Statistical Parity or Predictive Equality metrics with the ACSPublicCoverage dataset. In this scenario, the reconstruction correction still brings important improvement - slightly weakened by the fact that the estimated tolerance is usually not as tight as the actual one. A second interesting situation is when the fairness metric is not correctly identified, which is the case for all experiments using the Equalized Odds metric. Nonetheless, the fairness information estimation process can still come with a valid fairness constraint (even if it is not the one that was optimized during training), which can effectively be leveraged by the reconstruction correction step. When the fairness estimation proposes a metric more informative (in terms of number of involved examples) than the actual one, the reconstruction improvement can sometimes be better. For instance, consider the experiment using the UCI Adult Income dataset with the Equal Opportunity metric. In 74% of the runs, the fairness constraint estimation process came up with a Predictive Equality constraint. Even though this is not the actual constraint that was optimized during training, this constraint is approximately valid and the corresponding metric relates to a greater number of examples. As a consequence and somewhat counter-intuitively, the final reconstruction is better than with the actual constraint (see Table 3.4). Finally, one important drawback of the fairness estimation process is that the performances of the reconstruction correction step are more variable as shown by greater standard deviation values.

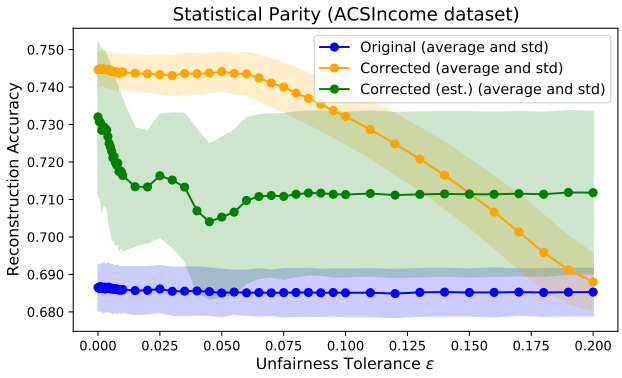



Figure 3.5: Original (adversary $\mathcal{A}'$), corrected (from actual fairness constraint, and from estimated one (`est.`)) reconstruction quality.

Results using the ExponentiatedGradient method [Agarwal *et al.* 2018] are provided in Figure 3.5 for the experiment using the ACSIncome dataset with the Statistical Parity metric and baseline adversary $\mathcal{A}'$. Results for all four metrics, three datasets and two adversaries are provided in our full paper [Ferry *et al.* 2023a]. They show similar trends as those using the ThresholdOptimizer method: estimating the fairness constraint still allows for good reconstruction correction performances but leads to a greater variability in the final reconstruction accuracy. Here also, inferring a fairness constraint different from the actual one can improve reconstruction correction, especially when the original tolerance is larger than the actual bias contained in the data (*i.e.*, large values of $\varepsilon$). In such cases, the adversary's baseline reconstruction already meets the actual fairness requirement and the reconstruction correction process cannot improve it. In contrast, the fairness constraint estimation process can infer a tighter value, allowing some reconstruction improvement.

Overall, we see that the knowledge of the actual fairness constraint is not necessary as estimations can provide comparable-quality reconstruction correction performances. Using the proposed fairness constraint estimation process, we provide in Appendix E additional reconstruction experiments using a pre-processing method for enhancing fairness. Results demonstrate the effectiveness of the proposed reconstruction correction approach, even when fairness metrics are not directly optimized and no fairness information is available. They also demonstrate empirically that the proposed framework is agnostic to the type of fairness intervention, as it was shown effective against pre-, in-, and post-processing fairness-enhancing methods.

## 3.5 Conclusion and Future Research

We have proposed a novel approach using declarative programming (either integer linear programming or constraint programming) to improve the reconstruction performances of any baseline adversary by incorporating user-defined constraints.

While the general problem may be computationally challenging, we have demonstrated that in the case of statistical fairness metrics (and, more generally, group-level constraints), it can be reformulated and solved efficiently.

In addition, our thorough experimental study shows that due to the use of the sensitive attribute information to ensure fairness of the built model, fairness-enhancing learning techniques inherently leak information about it. Indeed, the fairness constraints provide information regarding the distribution of a fair model's (training set) predictions with respect to the (training set) sensitive attributes. Even if such information is at the group level, it can be leveraged by an adversary to improve baseline reconstructions of the sensitive attributes. Furthermore, the tighter the fairness requirement, the more significant the reconstruction improvement.

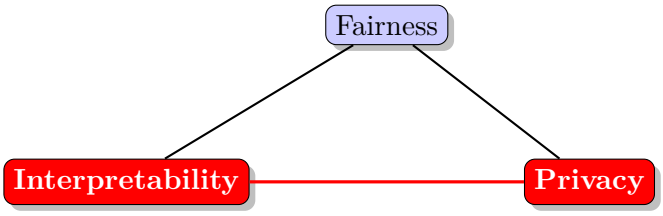
We additionally observed that, even if the fairness information is not available, an adversary can still try to infer it and obtain good (and sometimes, even better) reconstruction correction performances. While the fairness information is simply an input of our proposed reconstruction correction component, this finding demonstrates the applicability of our approach. It also illustrates the fact that due to their use of the sensitive attributes information, statistical fairness metrics intrinsically conflict with protecting the privacy of such attributes.

Future work includes combining our reconstruction correction component with different baseline adversaries, optimizing the adversary confidence vector $P$ processing as well as applying our framework in the context of non-binary sensitive attributes. The declarative nature of the reconstruction correction step allows considering a wide range of constraints. Hence, extending our proposed pipeline to improve baseline reconstruction attacks by enforcing other constraints (*e.g.*, proportion constraints, rate constraints, ...) is also an interesting research direction. Leveraging recent advances in end-to-end predict-then-optimize approaches (*e.g.,* the methods of [Berthet *et al.* 2020, Elmachtoub & Grigas 2022]), which train a ML model embedding a combinatorial optimization solver, to directly integrate constraints over the training set sensitive attributes within the attack model (rather than ensuring these constraints in a post-processing correction step) could also improve the attack's success.

Finally, the proposed reconstruction correction step proposes a reconstruction of the training set sensitive attributes by computing an optimal solution of either $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$ or $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$. However, this reconstruction is not necessarily unique, as there can be several different instantiations of the training set sensitive attributes with the same objective function value ((3.1) or (3.5)) and satisfying the fairness constraints. In our experiments, we simply used the optimal solution returned by the solver. Enumerating the other solutions to assess their number and evaluate them is an interesting direction, which is in line with the following chapter.

Chapter 4

# Interpretable Models Intrinsic Privacy Vulnerabilities




In this chapter, we first review the literature on the connections between privacy and interpretability in machine learning. Even if the two desiderata can be conciliated with some trade-offs, they often conflict. In particular, interpretability, by providing additional information to the users of the model, intrinsically constitutes a new attack surface. Consequently, we show how to measure the amount of information an interpretable model carries regarding its training data. This precise measure quantifies an inherent tension between releasing trained interpretable models and preserving the model's training data privacy.

## Contents

Interpretability and privacy are key requirements for trustworthy machine learning. On the one hand, the use of machine learning models for high-stakes decision-making necessitates some form of understanding of the model behavior from its users. Learning models that are inherently interpretable then appears as an appealing solution to avoid the drawbacks of post-hoc explainability frameworks (as discussed in Section 1.4.4). The field of medicine constitutes a good illustration, as scoring systems are popular models to assist practitioners in their analysis and diagnosis [Rudin *et al.* 2022]. On the other hand, machine learning techniques often require large amounts of data, which can be highly sensitive, as in the medicine example use case. Training useful models while preserving the privacy of the people whose data is used is then of particular importance.

Releasing interpretable models is desirable from a transparency perspective, however, it intrinsically leaks information regarding the model's training data. For instance, previous work [Gambs *et al.* 2012] exploited this information to build a *probabilistic* reconstruction of a decision tree's training set - effectively implementing a form of *reconstruction attack*. It is then possible to quantify the amount of information leaked by the model by measuring the uncertainty remaining within the reconstructed probabilistic dataset. Interestingly, this approach tackles one limitation mentioned at the end of Chapter 3: traditional reconstructions are often not unique, and probabilistic datasets are able to represent a whole set of possible ones. However, the proposed method relies on strong assumptions, such as statistical independence and uniform distribution of the random variables modeling the probabilistic dataset. While it allows probabilistic reconstructions from decision trees, it is not generic enough to encode more general types of knowledge, and cannot be used with other types of interpretable models, such as rule lists. In this chapter, we generalize the notion of probabilistic dataset by relaxing the aforementioned assumptions. In particular, we show how the success of such generalized probabilistic reconstructions can be assessed. We illustrate this point theoretically and empirically on several forms of interpretable models.

**Related Works.** Related work on reconstruction attacks can be found in Section 1.5.3. The only work directly related to this chapter is [Gambs *et al.* 2012], which also considers the setup where an attacker has white-box access to a trained interpretable machine learning model. The attacker then leverages this knowledge to reconstruct a probabilistic (uncertain) version of the model's training set. In Section 4.2, we provide more details about this baseline work, which we aim to extend hereafter.

**Outline of the Chapter.** We first provide a large literature review on the connections between interpretability and privacy in machine learning in Section 4.1. We synthesize the main identified synergies, compatibilities and tensions, before focusing on one particular aspect in the remainder of the chapter: the inherent information leak of interpretable models regarding their training data. More precisely, we introduce in Section 4.2 key notions from previous work regarding probabilistic dataset reconstructions from decision tree models. We generalize these notions to handle more generic types of knowledge in Section 4.3. In Section 4.4 we show how the success of such generalized probabilistic reconstructions can be assessed provided some assumptions regarding the structure of the interpretable model at hand. Finally, we illustrate in Section 4.5 the applicability of the approach through an example use case: comparing the amount of information optimal models carry compared to greedily-built ones.

## 4.1 Connections between Interpretability and Privacy

In this section, we survey the literature at the intersection between explainable AI in the broad sense (*i.e.,* either post-hoc explainability or interpretability, as introduced in Section 1.4.3) and privacy in machine learning. We first discuss synergies and compatibilities between the two fields, and summarize existing frameworks jointly addressing both of them. Motivated by the observed trade-offs, we then discuss their inherent tensions.

### 4.1.1 Compatibilities & Synergies

**Interpretability eases model audit and can be leveraged for privacy purposes.** [Doshi-Velez & Kim 2017] argue that interpretability can be used to confirm other desiderata of ML systems, such as privacy. [Rudin 2019] also states that it is easier to detect possible privacy issues when building interpretable models. Furthermore, this auditable nature is particularly appreciated in the area of machine learning based cybersecurity systems [Srivastava *et al.* 2022]. Indeed, machine learning models have shown great abilities to detect abnormal behaviors or intrusions. However, their black-box nature and lack of certification can be problematic as it possibly introduces weaknesses inside the security system. By providing an understanding of the underlying mechanisms and reasoning of the model, interpretability techniques can be helpful to detect overfitting, or in cases where the model captures noise or inaccurate values in the data. This allows deploying more trustworthy models, but also helps the administrators identify potential breaches.

**Interpretability can be conciliated with privacy with some trade-offs.** Learning interpretable models while satisfying differential privacy is possible, as pointed out in a recent survey [Gong *et al.* 2020] for decision trees or linear regression models. For instance, [Friedman & Schuster 2010] study data mining with differential privacy guarantees, considering decision tree induction as an illustrative

task. They demonstrate that the design of the privacy preserving mechanism is crucial, and that there is a huge difference in terms of model utility and required sample size between a naive implementation using a general purpose privacy preserving data interface and a task-specific differentially private learning algorithm. Their empirical study demonstrates the ability of their proposed algorithm to learn differentially private decision trees with reasonable cost in terms of accuracy. More precisely, they build a set of trade-offs between accuracy and the differential privacy parameter. Several other works also tackled differentially private decision tree building, as summarized in [Fletcher & Islam 2019]. An important finding it that the choice of the splitting criterion (*e.g., gini impurity, entropy ...*) highly influences the resulting model's utility. Indeed, different measures of the information gain yield different sensitivity - which can additionally be difficult to quantify or upper-bound. This implies that, for similar privacy guarantees (*i.e.,* values of $\varepsilon_{DP}$ and $\delta_{DP}$), different magnitudes of noise have to be added, resulting in different trade-offs with utility. Interestingly, the splitting criteria leading to the most accurate differentially private trees are not the ones exhibiting the best performances in the non-private setting. [Harder *et al.* 2020] propose to learn Locally Linear Maps (LLMs), that consist in a linear combination of logistic regressions for each possible class. Such interpretable models are suitable to provide local explanations (using the appropriate LLM) but also global ones, as the coefficients of each class's LLMs provide insights regarding which features really matter to it. The authors propose a procedure to learn LLMs under differential privacy, leveraging mechanisms from the `DP-SGD` framework [Abadi *et al.* 2016]. They finally observe an empirical trade-off between the privacy guarantee and the model's accuracy and interpretability.

**Post-Hoc Explainability can be conciliated with privacy with some trade-offs.** [Datta *et al.* 2016] propose a framework coined Quantitative Input Influence (QII), leveraging Shapley values to provide feature-based explanations quantifying the influence of input features over the model's predictions. As such measures may leak information regarding individual users, the authors introduce a mechanism to generate differentially private explanations to the so-called transparency queries. Providing pure differential privacy guarantees, it consists in adding Laplace noise to the query answers, scaled to the query function sensitivity. As the proposed measures generally have low sensitivity, the amount of added noise remains reasonable which results in relatively small average utility losses. Nonetheless, for some types of explanations with exceptionally high sensitivity, the amount of noise added may significantly harm their utility. [Patel *et al.* 2022] introduce a method to generate differentially-private feature-based explanations (local linear surrogates) of a black-box model. In their framework, the explanations are computed using a differentially private gradient descent leveraging the Gaussian mechanism. They further propose an adaptive mechanism, reducing the spending of the privacy budget by leveraging the explanations to previous queries when computing a new one. Using tabular, text and image data, they empirically observe that the explanations' qual-

ity degrade while the privacy guarantees tighten. [Naidu *et al.* 2021] investigate the impact of a model's differential privacy on the quality of post-hoc explanations (saliency maps generated with `Grad-CAM` [Selvaraju *et al.* 2017]) of this model and on its utility, considering either local DP (classical learning algorithm applied on DP data) or global DP (differentially private training algorithm). In both cases, one can note that the explanations are also differentially private by post-processing property (*cf.* Section 1.5.4). Handling either general or medical imaging applications, they learn neural networks under different differential privacy budgets and evaluate the quality of post-hoc explanations of their predictions using two metrics from the literature. In a nutshell, these metrics aim at quantifying how much the regions highlighted by explanation maps actually account for the explained decisions. The experimental results show that these metrics degrade while the privacy budget is tightened. Furthermore, they suggest the existence of a three dimensional trade-off space between privacy, explanation quality and model accuracy. To face the explanation-guided backdoor poisoning attack studied in [Severi *et al.* 2021] (and discussed in Section 4.1.2), [Nguyen *et al.* 2023] propose to generate Locally Differentially Private explanations. By randomly perturbating the top-$k$ features in the generated feature-based explanations, the mechanism is shown to mitigate the success of the attack. [Mochaourab *et al.* 2021] design an approach to generate robust counterfactual explanations for differentially private Support Vector Machines (SVMs). More precisely, privacy is achieved by adding Laplace noise to the SVMs' weights, and classical counterfactual explanation frameworks may generate counterfactuals that allow to cross the classifier's noisy boundaries, but not to actually change the example's class in real-life. To address this issue, they instead generate robust counterfactual explanations by solving an optimization problem with probabilistic constraints. In practice, the generated counterfactuals require more and more changes to the example as the privacy level tightens, in order to ensure that its classification changes with respect to the (unknown) non-private classifier. Again, this illustrates the tension between explanations quality and privacy protection. In the context of federated learning, recent work [Li *et al.* 2023] also noticed that differential privacy can alter the meaningfulness of gradient-based explanations. They propose an adaptive mechanism still providing differential privacy guarantees but injecting noise within the model's parameters in a manner aimed at preserving the quality of gradient-based explanations. Finally, recent work also studied differential privacy for counterfactual explanations [Yang *et al.* 2022]. The approach consists in using an autoencoder trained in a differentially private manner (through the functional mechanism) to build noisy class prototypes, which can then be leveraged to generate the counterfactuals.

As aforementioned, applying explainability techniques while preserving formal privacy guarantees is possible but implies some cost on either one aspect or the other. We further discuss the tensions between the two fields in the next subsection.

### 4.1.2 Tensions

**Interpretability/Explainability and Privacy conceptually have opposite goals.** While interpretability and privacy protection are both important requirements for trustworthy machine learning, they intrinsically pursue contrasting objectives [Datta *et al.* 2023]. Indeed, on the one side, interpretability tools aim at providing more information to enhance users' understanding of a model's behavior. On the other side, privacy protection requires a tight control of the leaked information, often obfuscating part of it to protect individuals' data. Jointly addressing both desiderata hence necessitates some form of arbitration [Banisar 2011].

**Explainability tools can be used with the purpose of designing attacks against machine learning models.** Tools from explainable AI can be leveraged by malicious entities to perform more effective attacks against machine learning based systems. For instance, [Severi *et al.* 2021] study malware detection models, that are usually trained on crowd-sourced data to distinguish between malicious softwares (malwares) and legitimate ones. Their objective is to perform backdoor poisoning attacks, where an attacker injects carefully chosen datapoints to the crowd-sourced training set, resulting in its chosen malware being wrongly classified as legitimate by the detection model. In this context, they leverage Shapley values to identify highly effective features and their values, and efficiently craft the poisoned examples. Explainable AI techniques were also leveraged to fool ML-based authentication systems. Such systems take as input a user ID along with some fingerprinting authenticating the user uniquely. Then, it was shown [Garcia *et al.* 2018] that an attacker can leverage perturbation-based feature explanation techniques on a local surrogate model to efficiently craft a fingerprint authenticating a desired user given its ID. Again, the feature importance explanations help guiding the malicious crafting process by indicating which features most influence the decision. [Kuppa & Le-Khac 2021] modify a counterfactual explanation framework to generate adversarial examples. They also use counterfactual explanations of a black-box model to identify the features that influence the model's decision boundaries and generate examples that can be used to conduct backdoor poisoning attacks.

**Post-Hoc explanations can be exploited to perform or improve inference attacks.** Inference attacks traditionally query a model (*e.g.,* via a prediction API) and use its outputs to achieve their goal, for instance determining an individual's membership in the training data, reconstructing part of the training dataset, extracting the model itself, or inferring an individual's missing attributes [Dwork *et al.* 2017, Cristofaro 2020]. Post-hoc explainability techniques, by offering explanations as additional outputs, expose a new attack surface. Several works showed that such explanations, whatever form they take (*e.g.,* example-based, feature-based . . . ), can be leveraged to enhance the different types of privacy attacks (introduced in Section 1.5.2):

***Model extraction attacks.*** [Milli *et al.* 2019] show that *gradient-based (a class of feature-based) explanations* of a black-box model can be exploited by an adversary to reconstruct the underlying model. In the considered setup, the adversary owns an auxiliary dataset and can query the black-box model to obtain the model's gradients as explanations for given input points. They design a near-optimal algorithm which provably extracts the entire underlying model within a bounded number of queries, in the particular case where it is a two-layer neural network with ReLU activations. For the general case, they design an effective heuristic inspired by previous works on standard reconstruction attacks against prediction APIs. More precisely, the attacker trains a surrogate model mimicking the black-box behavior and optimized to match its gradients thanks to the provided explanations. Importantly, the results show that model extraction from gradient explanations requires orders of magnitude less queries than from the sole predictions. [Miura *et al.* 2021] also consider gradient-based explanations, but assume no auxiliary dataset. In such case, the data used to query the black-box and train the surrogate model is outputted by a generative model, which, in turn, tries to generate examples for which the surrogate disagrees with the black-box. Importantly, the generative model is updated leveraging the provided gradient explanations - which dramatically reduces the required number of iterations (and queries to the black-box). Furthermore, [Aïvodji *et al.* 2020] show that providing *counterfactual (a class of example-based) explanations* (CFs) can help an attacker achieve model extraction attacks with better precision and limited number of requests. More precisely, the attacker queries the black-box model with a given attack set, and trains a surrogate using the predictions of both the attack set instances and the provided CFs. The authors empirically show that the use of the provided CFs improves the attack by both increasing the built surrogate's fidelity with respect to the black-box model, and dramatically decreasing the required number of queries. [Kuppa & Le-Khac 2021] propose a similar approach but leverage knowledge distillation techniques to train the surrogate model, which may mitigate the potential performance harm of an architecture mismatch between the actual black-box model and the reconstructed surrogate. [Wang *et al.* 2022b] also leverage CFs provided by Machine-Learning-as-a-Service (MLaaS) platforms and propose an efficient querying strategy to steal the underlying classification model. Their strategy is based on the following observation: the generated CFs usually lie close to the decision boundary, while the attack set examples do not necessarily. This leads to a "decision boundary shift issue", in which the surrogate model's decision boundary is shifted compared to that of the actual black-box. To circumvent this issue, the authors propose to generate counterfactuals for the CFs themselves, and to use them all for training the surrogate.

***Membership inference attacks.*** [Shokri *et al.* 2021] leverage *feature-based explanations* to perform membership inference attacks. More precisely, they consider both backpropagation-based (gradient-based) and perturbation-based expla-

nations. On the one side, they demonstrate that the former leak information regarding membership, and can effectively be leveraged to perform membership inference attacks. In particular, the explanations' variance is very informative: explanations of training examples usually exhibit a low variance, while, for unseen examples, this value can be considerably higher. This is due to the fact that for training examples, the model is usually very confident, as it was optimized on them, and small perturbations are likely to not change its predictions. On the contrary, unseen samples can be closer to the decision boundary, which results in some features having a great impact on the model's predictions (hence high gradients norms), and the resulting explanation having high variance. On the other side, they further show using two popular perturbation-based frameworks (`LIME` [Ribeiro *et al.* 2016] and `Smoothgrad` [Smilkov *et al.* 2017]) that the later are more resistant to membership inference. This may be explained by the fact that perturbation-based frameworks often generate perturbed examples that lie out of the data distribution [Kumar *et al.* 2020]. The black-box model behavior on such examples is unspecified, and so querying it with them does not provide insightful information to perform inference attacks. This also suggests that the resulting explanations may qualitatively by poorer: "privacy comes at the cost of explanation quality". [Kuppa & Le-Khac 2021] leverage *counterfactual explanations* to conduct membership inference attacks. More precisely, they query the black-box model with an auxiliary dataset and use the model's outputs and generated counterfactual examples to train a shadow model. Membership of a given example is then established by comparing the difference in prediction probabilities between the shadow model and the actual black-box to a threshold.

***Dataset reconstruction (and membership inference) attacks.*** [Shokri *et al.* 2021] consider an *example-based explainability* framework based on influence functions [Koh & Liang 2017], which returns influential training examples that most contribute to an example's prediction. Because they explicitly reveal training points, and a training point is likely to be used to explain itself, such explanations are highly vulnerable to membership inference attacks. Indeed, this class of explanations allows for stronger attacks, such that dataset reconstruction attacks. The authors propose two algorithms that leverage the provided example-based explanations to reconstruct (part of) the model's training set. The first algorithm is based on subspace reduction and comes with a certifiable lower bound on the number of points it discovers. Empirical evaluation shows that it can be used to retrieve most of the training dataset for high dimensional data. The second one is heuristic and offers no theoretical guarantees, but works well in practice for low dimensional data. It simply consists in using previously revealed points to reveal new points. Influence functions naturally define an influence graph structure over the training set, where an edge between two training examples means that one is provided as an explanation for the other. The proposed algorithm can then be used to explore entire Strongly Connected Components within this graph.

***Model inversion attacks.*** [Zhao *et al.* 2021] propose model inversion attacks that aim at reconstructing the black-box model's inputs given its outputs (here, its prediction along with some *feature-based explanation*), hence harming the privacy of test instances[1] (*i.e.,* active users of the model). In the context of image-based tasks, they focus on different types of saliency map explanations to reconstruct the target model's input images, namely gradient-based explanations [Simonyan *et al.* 2014], influence-based explanations [Ramaswamy *et al.* 2020] (obtained by multiplying each input feature by its associated gradient), activation-based explanations [Selvaraju *et al.* 2017], and layer-wise relevance propagation [Bach *et al.* 2015] (attributing pixels' importance by backpropagating neurons' relevance). The proposed attack uses an attack model, trained on an independent auxiliary dataset to predict images (given as input to the target model) given predictions and explanations (outputted by the target model). As expected, the frameworks directly using the input within the explanation computation (*i.e.,* influence-based ones) leak more information regarding the model's inputs, hence allowing better attack results. Importantly, the paper shows that even non-explainable models can be attacked, leveraging attention transfer to build an explainable surrogate whose explanations are used to conduct the attack. With a same attack goal, [Luo *et al.* 2022] show that Shapley value-based explanations provided by popular Machine Learning as a Service (MLaaS) providers can be exploited to reconstruct the private model inputs. They provide an information-theoretical analysis of the relationship between an example and its associated Shapley values, and demonstrate that an adversary can always infer useful information about the former using the later. This analysis also holds for sampling-based Shapley-values, which are commonly computed as an efficient approximation of the exact Shapley values. They then study two distinct adversarial settings, and show that even an adversary with no background knowledge can reconstruct most of the private model's input examples given only its outputs and explanations.

***(Sensitive) attribute inference attacks.*** [Duddu & Boutet 2022] study sensitive attribute inference attack leveraging *feature-based model explanations*, computed either with backpropagation-based or perturbation-based methods. They consider the two realistic scenarios where the sensitive attribute is (or not) used for training the model and for inference. In both studied scenarios, the attacker leverages an auxiliary dataset to train an attack model to predict an example's sensitive attribute given only the outputs of the target model (prediction and explanation) for this example. They empirically show that their attack is able to leverage such explanations to perform attribute inference attack. Furthermore, they suggest that model explanations lead to higher attack success compared to model predictions, hence constituting a stronger attack surface to exploit.

[1] This differs from the previously mentioned reconstruction attacks. Indeed, in reconstruction attacks, the goal of the adversary is to infer information regarding the model's training data. In the discussed model inversion attacks, the objective is to gain information about the examples provided to the model at inference time, by only observing the model's outputs.

**Interpretable models inherently leak information regarding their training data.** [Gambs *et al.* 2012] show that the structure of a trained decision tree can be leveraged to reconstruct a probabilistic version of its training set. We describe this work and summarize its key concepts in Section 4.2, and generalize it in the remainder of this chapter. More precisely, we show how the knowledge from a given interpretable model can be encoded to build a probabilistic reconstruction of its training data and quantify the associated information leak.

**Providing useful yet privacy-protective explanations remains an open challenge.** As we saw in Section 4.1.1, differentially private explainability tools have been proposed, but always imply some trade-off between the explanation quality, the privacy guarantee and the model utility. Furthermore, [Milli *et al.* 2019] recall that DP can help guard against attacks from prediction APIs, but it is not clear if this is a viable approach for preventing reconstruction from explanations. On the same line, [Shokri *et al.* 2021] state that "the effect of DP techniques (notably the randomness they induce) on model transparency is unknown." Furthermore, the effect of DP on the explanations' robustness and user trust are still to be investigated [Aïvodji *et al.* 2020].

As aforementioned, interpretability and explainability tools, by providing more information to the user of a model, intrinsically expose new attack surfaces which may be exploited by an adversary to infer information regarding the model's training set. In particular, in the remainder of this chapter, we show how the structure of a given interpretable model can be used to reconstruct a probabilistic version of its training set. Importantly, we precisely quantify the amount of information the model carries regarding its training data.

## 4.2 Probabilistic Dataset Reconstruction from Interpretable Models

A trained interpretable machine learning model, such as the decision tree presented in Figure 4.1, inherently encodes information regarding its training set. In DBLP:conf/dbsec/GambsGH12, this information is extracted and used to build a probabilistic reconstruction of the training dataset, in the form of a *probabilistic dataset*, as introduced in Definition 6.

**Definition 6.** ***(Probabilistic Dataset) [Gambs et al. 2012].*** *A probabilistic dataset $\mathcal{V}$ is composed of $N$ data points (also called* examples*) $\{e_1, \ldots, e_N\}$ and $M$ attributes $\{X_1, \ldots, X_M\}$. Each attribute $X_m$ has a domain of definition $\mathcal{X}_m$ that includes all the possible values of this attribute. The knowledge about attribute $X_m$ of example $e_j$ is modeled by a probability distribution over all the possible values of this attribute, using random variable $\mathcal{V}_{j,m}$. Importantly, variables $\{\mathcal{V}_{j\in[1..N],m\in[1...M]}\}$ are assumed to be statistically independent from each other and their probability distribution to be uniform.*

In practice, if a particular value $v_{j,m} \in \mathcal{X}_m$ of an attribute gathers all the probability mass (*i.e.,* it is perfectly determined: $\mathbb{P}(\mathcal{V}_{j,m} = v_{j,m}) = 1$), then the attribute is said to be deterministic. By extension, a probabilistic dataset whose attributes are all deterministic (*i.e.*, the knowledge about the dataset is perfect) is called a *deterministic dataset*.

Previous work [Gambs *et al.* 2012] propose a procedure to build a probabilistic dataset $\mathcal{V}^{DT}$ given the structure of a trained decision tree $DT$. Such probabilistic dataset gathers the knowledge that the decision tree inherently encodes about its (deterministic) training dataset $\mathcal{V}^{Orig}$. The construction of this probabilistic dataset can then be coined as a *probabilistic reconstruction attack*. By construction, $\mathcal{V}^{DT}$ is *compatible* with $\mathcal{V}^{Orig}$: the true value $v_{j,m}^{Orig}$ of any attribute $X_m$ for any example $e_j$ is always contained within the set of possible values for this attribute and this example in the probabilistic reconstruction: $\mathbb{P}\left(\mathcal{V}_{j,m}^{DT} = v_{j,m}^{Orig}\right) > 0$. A natural way to quantify the *success of the probabilistic reconstruction attack* is in terms of the average amount of uncertainty that remains in the built probabilistic dataset $\mathcal{V}^{DT}$, as stated in the following definition.

**Definition 7.** ***(Measure of success of a probabilistic reconstruction attack) [Gambs et al. 2012].*** *Let $\mathcal{V}^{Orig}$ be a deterministic dataset composed of $N$ data points and $M$ attributes, used to train a machine learning model $IM$. Let $\mathcal{V}^{IM}$ be a probabilistic dataset reconstructed from $IM$. By construction, $\mathcal{V}^{IM}$ is compatible with $\mathcal{V}^{Orig}$. The success of the reconstruction is quantified as the average uncertainty reduction over all attributes of all examples in the dataset:*

$$\mathsf{Dist}(\mathcal{V}^{IM}, \mathcal{V}^{Orig}) = \frac{1}{N \cdot M} \sum_{j=1}^{N} \sum_{m=1}^{M} \frac{H(\mathcal{V}_{j,m}^{IM})}{H(\mathcal{V}_{j,m})} \tag{4.1}$$

*in which random variable $\mathcal{V}_{j,m}$ corresponds to an* uninformed *reconstruction, uniformly distributed over all possible values $\mathcal{X}_m$ of attribute $X_m$, and $H$ denotes the Shannon entropy.*

Smaller values of $\mathsf{Dist}(\mathcal{V}^{IM}, \mathcal{V}^{Orig})$ indicate better reconstruction performance (*i.e.*, a more successful attack). In particular, if $\mathcal{V}^{IM} = \mathcal{V}^{Orig}$, $\mathsf{Dist}(\mathcal{V}^{IM}, \mathcal{V}^{Orig}) = 0$: the reconstruction is perfect and there is no uncertainty at all. In contrast, in other extreme in which $\mathcal{V}^{IM}$ contains no knowledge at all, $\mathsf{Dist}(\mathcal{V}^{IM}, \mathcal{V}^{Orig}) = 1$.

**Reminder: the Shannon Entropy.** Recall that the Shannon entropy of a random variable $\mathcal{V}_{j,m}^{IM}$ quantifies the average level of information inherent to the variable's possible outcomes, computed as:

$$H(\mathcal{V}_{j,m}^{IM}) = -\sum_{v_{j,m} \in \mathcal{X}_m} \mathbb{P}\left(\mathcal{V}_{j,m}^{IM} = v_{j,m}\right) \cdot log2\left(\mathbb{P}\left(\mathcal{V}_{j,m}^{IM} = v_{j,m}\right)\right)$$

which simplifies to $H(\mathcal{V}_{j,m}^{IM}) = -log2\left(\frac{1}{|\mathcal{X}_m|}\right)$ if all the values within $\mathcal{X}_m$ are equally probable for the realization of $\mathcal{V}_{j,m}^{IM}$ (*i.e.*, $\forall v_{j,m} \in \mathcal{X}_m, \mathbb{P}\left(\mathcal{V}_{j,m}^{IM} = v_{j,m}\right) = \frac{1}{|\mathcal{X}_m|}$).

**Remark 2.** *Definitions 6 and 7 are slightly more general than in [Gambs et al. 2012]. Indeed, both use actual random variables while in the original formulation each attribute of each example is simply modeled via a set of possible values, which is only suitable under the assumed hypothesis of statistical independence and uniform distribution of the random variables. Thus, our extended formulation eases the generalization we further provide in Section 4.3 while encompassing this particular case.*

Table 4.1: Example deterministic dataset $\mathcal{V}^{Orig}$.

| | $X_1$ | $X_2$ | $X_3$ | Label |
|---|---|---|---|---|
| $e_1$ | 12 | 0 | 3 | 0 |
| $e_2$ | 14 | 1 | 2 | 0 |
| $e_3$ | 11 | 1 | 2 | 1 |
| $e_4$ | 14 | 0 | 1 | 1 |

Table 4.2: Example probabilistic dataset $\mathcal{V}^{DT}$ reconstructed from a Decision Tree (Figure 4.1).

| | $X_1$ | $X_2$ | $X_3$ | Label |
|---|---|---|---|---|
| $e_1$ | $\in \{12, 13, 14, 15\}$ | $\in \{0, 1\}$ | $\in \{2, 3\}$ | 0 |
| $e_2$ | $\in \{12, 13, 14, 15\}$ | $\in \{0, 1\}$ | $\in \{2, 3\}$ | 0 |
| $e_3$ | $\in \{10, 11\}$ | $\in \{0, 1\}$ | $\in \{2, 3\}$ | 1 |
| $e_4$ | $\in \{10, 11, 12, 13, 14, 15\}$ | $\in \{0, 1\}$ | $\in \{1\}$ | 1 |

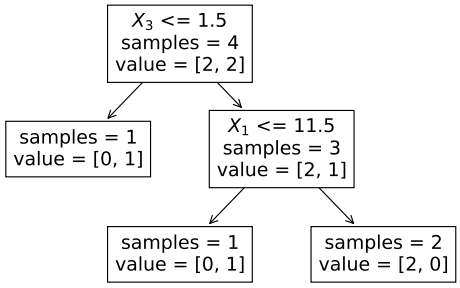



Figure 4.1: Example Decision Tree $DT$ trained using `scikit-learn` [Pedregosa *et al.* 2011], with 1.0 accuracy on $\mathcal{V}^{Orig}$ (Table 4.1).

We illustrate the reconstruction process proposed in [Gambs *et al.* 2012] with a toy example. A deterministic dataset $\mathcal{V}^{Orig}$, provided in Table 4.1 is used to train a decision tree classifier $DT$ depicted in Figure 4.1. This dataset includes four examples $e_{j\in\{1,2,3,4\}}$ with three attributes $X_{m\in\{1,2,3\}}$ with domains $\mathcal{X}_1 = \{10, 11, 12, 13, 14, 15\}$, $\mathcal{X}_2 = \{0, 1\}$ and $\mathcal{X}_3 = \{1, 2, 3\}$. This decision tree, learnt using the `scikit-learn` python library [Pedregosa *et al.* 2011], provides the per-label number of training examples in each internal node and each leaf. Intuitively, its structure can then be used to reconstruct a probabilistic version of its training dataset $\mathcal{V}^{DT}$, given in Table 4.2. The algorithm used to build $\mathcal{V}^{DT}$ simply follows each branch and performs the domains' reductions associated to each split along the branch. Using Definition 7, we can compute the *success of the reconstruction* as the average amount of uncertainty contained within $\mathcal{V}^{DT}$. For instance, we have $\mathcal{X}_1 = \{10, 11, 12, 13, 14, 15\}$ and $\mathcal{V}^{DT}_{3,1}$ takes values in $\{10, 11\}$. Then, considering that all the possible values are equally probable, the uncertainty reduction for attribute $X_1$ of example $e_3$ is: $\frac{H(\mathcal{V}^{DT}_{3,1})}{H(\mathcal{V}_{3,1})} = \frac{-log2(\frac{1}{2})}{-log2(\frac{1}{6})} \approx 0.387$. By averaging such

computation over the entire dataset (*i.e.*, over all attributes of all examples), we obtain $\mathsf{Dist}(\mathcal{V}^{DT}, \mathcal{V}^{Orig}) \approx 0.736$.

To facilitate reading, we aligned $\mathcal{V}^{Orig}$ and $\mathcal{V}^{DT}$. In practice, such alignment can be performed using the Hungarian algorithm [Kuhn 1955, Munkres 1957] as is done in [Gambs *et al.* 2012]. In a nutshell, it consists in performing a minimum cost matching between the examples of $\mathcal{V}^{Orig}$ and those of $\mathcal{V}^{DT}$, where the assignment cost is computed as the sum of the distances between the paired examples. Intuitively, the objective is to determine *to which example within* $\mathcal{V}^{Orig}$ *corresponds each reconstructed example in* $\mathcal{V}^{DT}$. However, this would not be needed in scenarios in which $\mathcal{V}^{Orig}$ is unknown, as $\mathcal{V}^{DT}$ is compatible with $\mathcal{V}^{Orig}$ by construction, and $\mathsf{Dist}$ (4.1) does not require further information regarding $\mathcal{V}^{Orig}$.

In the remainder of this chapter, we generalize the notions introduced in this section to be able to handle more general type of knowledge, coming from other types of interpretable models.

## 4.3 Generalizing Probabilistic Datasets Reconstruction

In this section, we first illustrate the limits of probabilistic datasets and motivate the need to relax some of their underlying assumptions. Consequently, we introduce *generalized probabilistic datasets*, which can be used to encode any arbitrary knowledge regarding a dataset. Finally, we define a generalized metric $\mathsf{Dist}_G$ which can be used to quantify uncertainty reduction within such datasets.

### 4.3.1 Motivation

The concept of probabilistic dataset as described in Definition 6 is suitable to encode knowledge regarding a dataset, as long as this knowledge involves each *cell* (*i.e.*, which corresponds to one attribute for one example) individually. For instance, this is appropriate for decision trees in which an example is classified *exactly* by one branch. Furthermore, each branch corresponds to a *conjunction* (*i.e.*, logical AND) of conditions (*splits*) over features, which all have to be satisfied. These conditions allow for the reduction of each such feature's domains *individually*. However, for other representations of interpretable classifiers, such as rule lists or rule sets, this condition will not be valid. Again, we illustrate this observation using a toy example.

More precisely, Rule List 4.1 was trained on (deterministic) dataset $\mathcal{V}^{Orig'}$, shown in Table 4.3. It gathers five examples $e'_{j\in\{1,2,3,4,5\}}$ described by three binary attributes, named $X'_{m\in\{1,2,3\}}$ (with domains $\mathcal{X}'_{m\in\{1,2,3\}} = \{0,1\}$). For each rule (including the default rule), $RL$ indicates the number of training examples it captures, for each class. For example, the second rule captures two training examples belonging to class 0 (here, $e'_3$ and $e'_4$).

The algorithm reconstructing a probabilistic version of a rule list $RL$'s training set from $RL$ itself simply follows the path of each example. For an example classified by the $i$th rule, it reduces the domains of the attributes involved in the $i$th rule accordingly. It also eliminates all attributes' conjunctions contradicting the fact

Table 4.3: Example deterministic dataset $\mathcal{V}^{Orig'}$.

| | $X'_1$ | $X'_2$ | $X'_3$ | Label |
|---|---|---|---|---|
| $e'_1$ | 1 | 1 | 1 | 1 |
| $e'_2$ | 1 | 1 | 0 | 1 |
| $e'_3$ | 0 | 1 | 1 | 0 |
| $e'_4$ | 1 | 0 | 1 | 0 |
| $e'_5$ | 1 | 0 | 0 | 1 |

Table 4.4: Example (generalized) probabilistic dataset $\mathcal{W}^{RL}$ reconstructed from Rule List 4.1.

| | $X'_1$ | $X'_2$ | $X'_3$ | Label |
|---|---|---|---|---|
| $e'_1$ | 1 | 1 | $\in \{0,1\}$ | 1 |
| $e'_2$ | 1 | 1 | $\in \{0,1\}$ | 1 |
| $e'_3$ | $\in \{(0,0),(0,1),(1,0)\}$ | | 1 | 0 |
| $e'_4$ | $\in \{(0,0),(0,1),(1,0)\}$ | | 1 | 0 |
| $e'_5$ | $\in \{(0,0),(0,1),(1,0)\}$ | | 0 | 1 |

```
if [X'1] and [X'2] then [true] ([0 ; 2] examples)
else if [X'3] then [false] ([2 ; 0] example)
else [true] ([0 ; 1] example)
```

Rule list 4.1: Example rule list $RL$ trained using `CORELS` [Angelino *et al.* 2017, Angelino *et al.* 2018], with 1.0 accuracy on $\mathcal{V}^{Orig'}$ (Table 4.3).

that the example did not match the previous rules within $RL$. For instance, the following knowledge can be extracted from Rule List 4.1:

- The **first rule** indicates that for 2 (positive) examples, the two Boolean attributes $X'_1$ and $X'_2$ are true.
- Using the **second rule**, we know that the Boolean attribute $X'_3$ is true for 2 (negatively-labelled) examples. *Furthermore, we know that $X'_1$ and $X'_2$ can not be simultaneously true for these examples (or else they would have been captured by the first rule).*
- Finally, the **default rule** states that for 1 (positively-labelled) example, $X'_3$ is false, and $X'_1$ *and* $X'_2$ *can not be simultaneously true.*

Using such knowledge, one can build a (generalized) probabilistic dataset as shown in Table 4.4. In this example, part of the model's knowledge directly reduces the individual domains of some attributes for the concerned examples. As such, the information it brings will successfully be quantified by Dist and encoded in a probabilistic dataset. However, other information (specified *in italic*) does not reduce any attribute's domain individually. For instance, as shown in Table 4.4, one knows that for examples $e'_3$ and $e'_4$, $X'_1$ and $X'_2$ can not simultaneously be true. Nevertheless, taken apart, their respective domains would be unchanged as both binary attributes can still take values in $\{0,1\}$. While such knowledge brings information for reconstruction, this cannot be quantified using Dist nor represented using a probabilistic dataset as formalized in Definition 6.

Indeed, one key assumption with Definition 6 is that the random variables representing each attribute for each example are independent from each other. This is leveraged by Dist, which computes the reductions of the individual entropies.

However, this representation cannot handle more generic knowledge, in which uncertainty can be spread jointly across multiple random variables. This limitation is also pointed out in the theory of probabilistic databases. More precisely, quoting [Suciu *et al.* 2011], this representation (talking about a scheme similar to probabilistic datasets as formalized in Definition 6 and illustrated in Figure 4.2) is "more compact", as we do not need to expand all possible combinations of the different variables' values explicitly. However, "it cannot account for correlations across possible readings of different fields, such as when we know that no two persons can have the same social security number". In this particular case, this corresponds to a correlation across examples, while in the aforementioned example of Rule List 4.1 we observed correlations between attributes within the same example. For instance, to encode the knowledge regarding attributes $X'_1$ and $X'_2$ of examples $e'_3$ and $e'_4$, we had to enumerate all the possible combinations of these two attributes' values (Table 4.4).

In the next subsection, we generalize probabilistic datasets to handle any form of knowledge, such as the italicized one from the above example.

### 4.3.2 Generalized Probabilistic Datasets

As illustrated in the previous subsection, the assumptions underlying probabilistic datasets (Definition 6) - namely statistical independence and uniform distribution of their random variables - make them inappropriate in the general case. Generalized probabilistic datasets remove these assumptions as stated in Definition 8.

**Definition 8.** ***(Generalized probabilistic dataset).*** *A generalized probabilistic dataset $\mathcal{W}$ is composed of $N$ data points (also called* examples*) $\{e_1, \ldots, e_N\}$ and $M$ attributes $\{X_1, \ldots, X_M\}$. The knowledge about attribute $X_m$ of example $e_j$ is modeled by a probability distribution over all the possible values of this attribute, using random variable $\mathcal{W}_{j,m}$. Importantly, variables $\{\mathcal{W}_{j\in[1..N],m\in[1..M]}\}$ are not necessarily statistically independent from each other and can follow any arbitrary distribution. Each possible instantiation $w = \{w_{j\in[1..N],m\in[1..M]}\}$ of the $\mathcal{W}_{j\in[1..N],m\in[1..M]}$ variables (*i.e.*, each deterministic dataset compatible with $\mathcal{W}$) is named a* possible world*. We let $\Pi(\mathcal{W})$ denote the set of possible worlds within $\mathcal{W}$: $\Pi(\mathcal{W}) = \{w \mid \mathbb{P}(\mathcal{W}_{j\in[1..N],m\in[1..M]} = w_{j\in[1..N],m\in[1..M]}) > 0\}$.*

Again, if all its variables are determined, a generalized probabilistic dataset is said to deterministic. A key difference between probabilistic datasets and their generalized counterparts is that the set of possible worlds of a probabilistic dataset simply consists in all combinations of the possible variables' values, all random variables being statistically independent. For generalized probabilistic datasets, it is not the case as there can exist complex inter-dependencies between the random variables that directly influence $\Pi(\mathcal{W})$ (as illustrated in Section 4.3.1).

Our generalized probabilistic dataset definition matches the notions of *probabilistic* or *incomplete databases* that are used in the theory of probabilistic

databases [Suciu *et al.* 2011]. Indeed, an *incomplete database* defines a set of *possible worlds*, denoting the possible states of the database (*i.e.*, set of values for the different relations). If one can associate a probability to each possible world, then the database is called a *probabilistic database* - which generalizes incomplete databases. In the context of this work, one could leverage external knowledge (*e.g.*, demographic information about the data distribution) to associate probabilities to the possible worlds in $\Pi(\mathcal{W})$. This would lead to a reduction of the uncertainty of the dataset (thus lowering its joint entropy and raising the reconstruction success).

Both incomplete and probabilistic databases are semantic definitions for which designing a practical representation is challenging [Suciu *et al.* 2011]. To circumvent this issue, some *compact* representations have been proposed. For instance, in *conditional tables* (or *c-tables*), the different values of the database cells are associated to a propositional formula, called condition, over some random variables. The different assignments of the random variables define the different states of the database (*i.e.*, possible worlds). *Probabilistic conditional tables* (or *pc-tables*) extend this concept by assigning probabilities to the conditional variables assignments. While (p)c-tables may be an interesting representation for generalized probabilistic datasets, we do not assume any specific representation for our generalized probabilistic datasets in this work. Rather, we demonstrate in Section 4.4 that in the context of training set reconstruction from an interpretable model, we can quantify the amount of uncertainty that remains in the resulting generalized probabilistic dataset without building it explicitly (which in practice may be infeasible even with efficient structures such as c-tables).

### 4.3.3 Generalized Measure of the Attack Success

We now generalize the metric introduced in Definition 7 to quantify the success of a probabilistic reconstruction attack. As stated in Definition 9, our new metric $\mathsf{Dist}_G$ is more general as it quantifies the uncertainty reduction on the entire dataset using the joint entropy of the underlying random variables.

**Definition 9.** ***(Generalized measure of success of a probabilistic reconstruction attack).*** *Let $\mathcal{W}^{Orig}$ be a deterministic dataset composed of $N$ data points and $M$ attributes, used to train a machine learning model $IM$. Let $\mathcal{W}^{IM}$ be a generalized probabilistic dataset reconstructed from $IM$. By construction, $\mathcal{W}^{IM}$ is compatible with $\mathcal{W}^{Orig}$* (i.e., *$\mathcal{W}^{Orig} \in \Pi(\mathcal{W}^{IM})$). The success of the performed reconstruction is quantified as the overall uncertainty reduction in the dataset:*

$$\mathsf{Dist}_G(\mathcal{W}^{IM}, \mathcal{W}^{Orig}) = \frac{H\left(\{\mathcal{W}^{IM}_{j,m} \mid j \in [1..N], m \in [1..M]\}\right)}{H\left(\{\mathcal{W}_{j,m} \mid j \in [1..N], m \in [1..M]\}\right)} \tag{4.2}$$

$$= \frac{\sum_{w \in \Pi(\mathcal{W}^{IM})} -\mathbb{P}(w) \cdot log2(\mathbb{P}(w))}{\sum_{j=1}^{N} \sum_{m=1}^{M} H(\mathcal{W}_{j,m})} \tag{4.3}$$

*in which $H$ denotes the Shannon entropy (or joint entropy, when applied to a set*

*of variables, as in (4.2)), and random variable $\mathcal{W}_{j,m}$ corresponds to an* uninformed *reconstruction, uniformly distributed over all possible values of attribute $X_m$.*

The denominator in Equation (4.2) can be decomposed as a sum in Equation (4.3) because the random variables $\mathcal{W}_{j\in[1..N],m\in[1..M]}$ are independent from each other, and the joint entropy of a set of variables is equal to the sum of the individual entropies of the variables in the set if and only if the variables are statistically independent. This is not the case for variables $\mathcal{W}^{IM}_{j\in[1..N],m\in[1..M]}$, and thus the generalized probabilistic dataset has to be considered as a whole through its set of possible worlds $\Pi(\mathcal{W}^{IM})$.

The key properties of $\mathsf{Dist}$ also hold for $\mathsf{Dist}_G$. In particular, for any deterministic dataset $\mathcal{W}^{Orig}$, we have $\mathsf{Dist}_G(\mathcal{W}^{Orig},\mathcal{W}^{Orig}) = 0$. Furthermore, if $\mathcal{W}^{IM}$ contains no knowledge at all, we have that $\mathsf{Dist}_G(\mathcal{W}^{IM},\mathcal{W}^{Orig}) = 1$ for any deterministic dataset $\mathcal{W}^{Orig}$.

One important difference between $\mathsf{Dist}$ and $\mathsf{Dist}_G$ is the fact, that due to its averaging over the per-example-per-attribute individual uncertainty reductions, $\mathsf{Dist}$ considers all features equal (in terms of contribution to the overall uncertainty) while it is not the case for $\mathsf{Dist}_G$. To illustrate this, let us assume a toy scenario with a (deterministic) dataset $\mathcal{V}^{Orig}$ with a single record $e_1 = (1, 1)$ and two attributes $X_1$ and $X_2$ with domains $\mathcal{X}_1 = \{0, 1\}$ and $\mathcal{X}_2 = \{1, 2, 3\}$. Consider the two probabilistic datasets $\mathcal{V}^{rec1}$, in which we know that for $e_1$, $X_1 = 1$, and $\mathcal{V}^{rec2}$, in which we know that for $e_1$, $X_2 = 1$. These datasets are summarized in Tables 4.5, 4.6 and 4.7.

Table 4.5: $\mathcal{V}^{Orig}$

| | $X_1$ | $X_2$ |
|---|---|---|
| $e_1$ | 1 | 1 |

Table 4.6: $\mathcal{V}^{rec1}$

| | $X_1$ | $X_2$ |
|---|---|---|
| $e_1$ | 1 | $\in \{1, 2, 3\}$ |

Table 4.7: $\mathcal{V}^{rec2}$

| | $X_1$ | $X_2$ |
|---|---|---|
| $e_1$ | $\in \{0, 1\}$ | 1 |

Using Definition 7, we have $\mathsf{Dist}(\mathcal{V}^{rec1},\mathcal{V}^{Orig}) = 0.5$, as:

$$\frac{H(\mathcal{V}^{rec1}_{1,1})}{H(\mathcal{V}_{1,1})} = \frac{-log2(1)}{-log2(\frac{1}{2})} = 0 \quad \text{and} \quad \frac{H(\mathcal{V}^{rec1}_{1,2})}{H(\mathcal{V}_{1,2})} = \frac{-log2(\frac{1}{3})}{-log2(\frac{1}{3})} = 1$$

Conversely, we also have $\mathsf{Dist}(\mathcal{V}^{rec2},\mathcal{V}^{Orig}) = 0.5$ because:

$$\frac{H(\mathcal{V}^{rec2}_{1,1})}{H(\mathcal{V}_{1,1})} = \frac{-log2(\frac{1}{2})}{-log2(\frac{1}{2})} = 1 \quad \text{and} \quad \frac{H(\mathcal{V}^{rec2}_{1,2})}{H(\mathcal{V}_{1,2})} = \frac{-log2(1)}{-log2(\frac{1}{3})} = 0$$

However, out of 6 possible reconstructions for $e_1$ (without any knowledge), 3 are possible within $\mathcal{V}^{rec1}$ while only 2 are possible with $\mathcal{V}^{rec2}$. Intuitively, $\mathcal{V}^{rec2}$ yields more information (or, conversely, less uncertainty) than $\mathcal{V}^{rec1}$, but $\mathsf{Dist}$ cannot account for this difference due to normalization and individual measure of entropy across examples' attributes. For notation consistency, we associate to these datasets their generalized counterparts $\mathcal{W}^{Orig}$, $\mathcal{W}^{rec1}$ and $\mathcal{W}^{rec2}$, containing the exact same information (recall that probabilistic datasets are simply a particular case of generalized probabilistic datasets, in which the dataset's variables are statistically

independent and uniformly distributed). Using our generalized metric introduced in Definition 9, we have:

$$\mathsf{Dist}_G(\mathcal{W}^{rec1}, \mathcal{W}^{Orig}) = \frac{H\left(\{\mathcal{W}^{rec1}_{1,1}, \mathcal{W}^{rec1}_{1,2}\}\right)}{H\left(\{\mathcal{W}_{1,1}, \mathcal{W}_{1,2}\}\right)} = \frac{-log2(\frac{1}{3})}{-log2(\frac{1}{6})} \approx 0.613$$

and

$$\mathsf{Dist}_G(\mathcal{W}^{rec2}, \mathcal{W}^{Orig}) = \frac{H\left(\{\mathcal{W}^{rec2}_{1,1}, \mathcal{W}^{rec2}_{1,2}\}\right)}{H\left(\{\mathcal{W}_{1,1}, \mathcal{W}_{1,2}\}\right)} = \frac{-log2(\frac{1}{2})}{-log2(\frac{1}{6})} \approx 0.387$$

As lower values indicate less uncertainty (*i.e.*, better reconstruction performances), we observe that $\mathsf{Dist}_G$ successfully distinguishes between $\mathcal{W}^{rec1}$ and $\mathcal{W}^{rec2}$. Thus by avoiding the drawbacks of the normalization across dataset cells, the new metric $\mathsf{Dist}_G$ successfully takes into account the specificities of the two probabilistic datasets.

## 4.4 Quantifying the Success of Generalized Probabilistic Reconstructions in Practice

We now investigate how to quantify the success of a probabilistic reconstruction attack in practice. First, we discuss how the attack success computation can be decomposed under reasonable assumptions regarding the structure of the interpretable model considered. Then, we show how it can be computed without explicitly building the entire set of possible worlds, as long as one is able to count them. Finally, we demonstrate that such simplification is possible for decision trees as well as rule lists models, and theoretically compare the reconstruction quality from these two hypothesis classes.

### 4.4.1 General Case

Let $\mathcal{W}^{IM}$ be a generalized probabilistic dataset reconstructed from an interpretable model $IM$. As stated in Definition 9, the success of the probabilistic reconstruction attack can be quantified using $\mathsf{Dist}_G$. One can observe that the denominator $\left(\sum_{j=1}^{N}\sum_{m=1}^{M} H(\mathcal{W}_{j,m})\right)$ is a constant, only depending on the attributes' domains $\mathcal{X}_{m\in[1..M]}$. Indeed, variables $\mathcal{W}_{j\in[1..N],m}$ are uniformly distributed over $\mathcal{X}_m$ (the domain of attribute $X_m$) and so $H(\mathcal{W}_{j,m}) = -log2\left(\frac{1}{|\mathcal{X}_m|}\right)$. Thus:

$$\sum_{j=1}^{N}\sum_{m=1}^{M} H(\mathcal{W}_{j,m}) = N \cdot \sum_{m=1}^{M} -log2\left(\frac{1}{|\mathcal{X}_m|}\right). \tag{4.4}$$

As the denominator in Equation (4.2) is a constant that can be easily computed, we will focus only on the numerator in the remaining of this section, using the

following notation:

$$\mathsf{Dist}_G(\mathcal{W}^{IM}, \mathcal{W}^{Orig}) \propto H\left(\{\mathcal{W}^{IM}_{j\in[1..N],m\in[1..M]}\}\right). \tag{4.5}$$

#### 4.4.1.1 Independence assumptions: decomposing the attack success computation

In the general case, the computation of the joint entropy of the generalized probabilistic dataset's cells must be done through its set of possible worlds $\Pi(\mathcal{W}^{IM})$, as shown in Equation (4.3). However, if one can establish the statistical independence of some of the $\mathcal{W}^{IM}_{j,m}$ variables, this computation can be further decomposed. Indeed, the joint entropy of a set of statistically independent variables is equal to the sum of their individual entropies. For instance, if the knowledge of model $IM$ applies to each data point $e_{j\in[1..N]}$ independently, the sets of variables $\{\mathcal{W}^{IM}_{j,m\in[1..M]}\}_{j\in[1..N]}$ are independent from each other. This condition is satisfied if $IM$ is a decision tree or a rule list, because each example is captured by exactly one "decision path" (*i.e.*, branch or rule). Indeed, this decision path reduces the set of possible reconstructions for each example $e_j$ independently from the other examples. By a slight abuse of notation, we let $\Pi_j(\mathcal{W}^{IM})$ denote the set of possible worlds (*i.e.*, reconstructions) for example (row) $e_j$. As a consequence, we have:

$$\mathsf{Dist}_G(\mathcal{W}^{IM}, \mathcal{W}^{Orig}) \propto \sum_{j=1}^{N} H\left(\{\mathcal{W}^{IM}_{j,m\in[1..M]}\}\right) \tag{4.6}$$

$$\propto \sum_{j=1}^{N}\left(\sum_{w_j\in\Pi_j(\mathcal{W}^{IM})} -\mathbb{P}(w_j)\cdot log2(\mathbb{P}(w_j))\right). \tag{4.7}$$

While Equation (4.7) holds for both rule lists and decision trees, its computation can be further decomposed for the later. Indeed, in a decision tree each example is classified by exactly one branch, and such branch defines a conjunction of Boolean conditions over attributes' values, called *splits*. Such conditions must all be satisfied for the example to be captured by the branch - hence all the concerned attributes' domains can be reduced individually. As a consequence, this implies that all variables $\mathcal{W}^{IM}_{j\in[1..N],m\in[1..M]}$ are actually statistically independent resulting in:

$$\mathsf{Dist}_G(\mathcal{W}^{IM}, \mathcal{W}^{Orig}) \propto \sum_{j=1}^{N}\sum_{m=1}^{M} H\left(\mathcal{W}^{IM}_{j,m}\right). \tag{4.8}$$

Note that Equation (4.8) corresponds to the particular case studied in [Gambs *et al.* 2012], with the computation being exactly as for their proposed $\mathsf{Dist}$ metric (Definition 7), with the only difference being the absence of normalization. Observe that Equation (4.8) does not hold in general for rule list models due to the fact that for a given example, the information that it did not match previous rules within the rule list corresponds to negating a conjunction, hence producing

a disjunction. As a result, this potentially breaks the statistical independence between some of the $\{\mathcal{W}^{IM}_{j,m\in[1..M]}\}$ variables.

#### 4.4.1.2 Uniform distribution assumptions: efficient attack success computation

The explicit enumeration of the possible worlds $\Pi(\mathcal{W}^{IM})$ is not practically conceivable for real-size datasets. However, quantifying a probabilistic reconstruction attack success can sometimes be done only by computing their number $|\Pi(\mathcal{W}^{IM})|$. Indeed, assuming a uniform probability distribution between them, one can then easily quantify the amount of uncertainty using $\mathsf{Dist}_G$ (Definition 9), as $\forall w \in \Pi(\mathcal{W}^{IM}), \mathbb{P}(w) = \frac{1}{|\Pi(\mathcal{W}^{IM})|}$, resulting in:

$$\sum_{w\in\Pi(\mathcal{W}^{IM})} -\mathbb{P}(w)\cdot log2(\mathbb{P}(w)) = -log2\left(\frac{1}{|\Pi(\mathcal{W}^{IM})|}\right). \tag{4.9}$$

Remark that only the number of possible worlds $|\Pi(\mathcal{W}^{IM})|$ is needed to compute Equation (4.9). In the general case, this number cannot be retrieved without building $\Pi(\mathcal{W}^{IM})$ explicitly. However, several types of interpretable models enable to compute $|\Pi(\mathcal{W}^{IM})|$ efficiently (*i.e.*, without building $\Pi(\mathcal{W}^{IM})$). For instance, this is the case when reconstructing generalized probabilistic datasets from decision tree or rule list models. Plugging together Equations (4.7) and (4.9), we have:

$$\mathsf{Dist}_G(\mathcal{W}^{IM}, \mathcal{W}^{Orig}) \propto \sum_{j=1}^{N} -log2\left(\frac{1}{|\Pi_j(\mathcal{W}^{IM})|}\right). \tag{4.10}$$

In the next subsections, we demonstrate how the number of possible reconstructions for each example $|\Pi_j(\mathcal{W}^{IM})|_{j\in[1..N]}$ can be computed in polynomial time (with respect to the model's size) for decision trees and rule lists.

### 4.4.2 Decision Trees

Let $DT$ be a decision tree with $K_{DT}$ branches, in which each branch $b_{i\in[1..K_{DT}]}$ is a conjunction of Boolean assertions over attributes' values ending with a leaf prediction. The value $\mathsf{num}(b_i)$ represents the number of different examples (*i.e.*, number of different combinations of attributes values) that satisfy $b_i$. It can be computed by multiplying the cardinalities of the reduced domains. Thus, for each example $e_j$ classified by branch $b_i$, we have $|\Pi_j(\mathcal{W}^{DT})| = \mathsf{num}(b_i)$. Additionally, $C_{i\in[1..K_{DT}]}$ is defined as the support of each leaf (*i.e.*, the number of training examples going through the branch $b_{i\in[1..K_{DT}]}$ ending by this leaf, as indicated in the decision tree of Figure 4.1). Importantly, the tree branches partition the set of examples (as the leaves' supports are all disjoints), so we have $\sum_{i\in[1..K_{DT}]} C_i = N$. Furthermore, the sum of Equation (4.10) which was performed over all $N$ examples can be replaced with a sum over the $K_{DT}$ branches, with the entropy of each branch $b_i$ being weighted by its support $C_i$. Plugging these new notions into Equation (4.10), we

obtain that the overall joint entropy of the reconstructed probabilistic version of $DT$'s training set is:

$$\mathsf{Dist}_G(\mathcal{W}^{DT}, \mathcal{W}^{Orig}) \propto \sum_{i=1}^{K_{DT}} -C_i \cdot log_2\left(\frac{1}{\mathsf{num}(b_i)}\right). \tag{4.11}$$

### 4.4.3 Rule Lists

Let $RL = (a_1, q_1)\ldots(a_{K_{RL}}, q_{K_{RL}})$ be a rule list, following the notation introduced in [Rivest 1987]. Each term $a_{i\in[1..K_{RL}]}$ is a conjunction of Boolean assertions over attributes' values and $q_{i\in[1..K_{RL}]}$ is a prediction. Rule $K_{RL}$ is the default decision, with $a_{K_{RL}}$ being the constant value $\mathsf{True}$[2]. Similarly to the leaves of a decision tree, each rule $i$ is associated with its support $C_i$. Again, let $\mathsf{num}(a_i)$ denote the number of different examples (*i.e.*, number of different combinations of attributes values) that satisfy $a_i$. As a branch, a rule corresponds to a conjunction, hence $\mathsf{num}(a_i)$ can be computed easily by simply multiplying the cardinalities of the attributes' reduced domains.

Finally, we define $\forall 1 \leq i \leq K_{RL}$, $\mathsf{Capt}_{RL}(a_i)$ as the number of possible different examples (*i.e.*, number of different combinations of attributes values) that $a_i$ captures *within* $RL$ (*i.e.*, examples satisfying $a_i$ while not matching the antecedents of the previous rules within $RL$). As a particular case, note that we always have $\mathsf{Capt}_{RL}(a_1) = \mathsf{num}(a_1)$ as the first rule of any rule list is always applied first. For $1 \leq i \leq K_{RL}$, a straightforward general formulation is:

$$\mathsf{Capt}_{RL}(a_i) = \mathsf{num}(a_i \wedge \bigwedge_{l=1}^{i-1} \neg a_l). \tag{4.12}$$

The main challenge is that $\mathsf{num}(\bigwedge_{l=1}^{i-1} \neg a_l)$, in which $\bigwedge_{l=1}^{i-1} \neg a_l$ is the conjunction of the negations of the previous rules' antecedents, cannot be computed directly as $a_{l\in[1..i-1]}$ may overlap. Indeed, each antecedent $a_l$ is a conjunction - hence its negation is a disjunction. More precisely, overall we get a conjunction of disjunctions, which means that the number of possible examples it characterizes cannot be computed by simply multiplying attributes' cardinalities as the different disjunctions may overlap. By a slight abuse of notation, we define for $1 \leq l \leq i \leq K_{RL}$, $\mathsf{Capt}_{RL}(a_l, a_i)$ as the number of possible different examples (*i.e.*, number of different combinations of features values) that $a_i$ could capture but that are actually captured by $a_l$ in $RL$:

$$\mathsf{Capt}_{RL}(a_l, a_i) = \mathsf{num}(a_l \wedge a_i) - \sum_{k=1}^{l-1} \mathsf{Capt}_{RL}(a_k, (a_l \wedge a_i)). \tag{4.13}$$

[2]This is consistent with the notation introduced in Chapter 2. For the sake of notation conciseness, we defined a rule list as $RL = (\delta_{RL}, q_0)$ in which $\delta_{RL} = (r_1, r_2, \ldots, r_K)$ with $r_i = a_i \rightarrow q_i$ is $RL$'s *prefix*, and $q_0 \in \{0, 1\}$ is a *default prediction*. Here, the first $K_{RL} - 1$ rules define $RL$'s prefix $\delta_{RL}$, while $q_{K_{RL}}$ is the default prediction.

The first term corresponds to the overlap between $a_l$ and $a_i$, while the second one subtracts the unique examples within this overlap that are actually captured by rules placed before $a_l$ in $RL$. Then:

$$\mathsf{Capt}_{RL}(a_i) = \mathsf{Capt}_{RL}(a_i, a_i) \tag{4.14}$$

$$= \mathsf{num}(a_i) - \sum_{l=1}^{i-1} \mathsf{Capt}_{RL}(a_l, a_i) \tag{4.15}$$

Just like the branches of a decision tree, the rules within a rule list partition the set of examples (as each example is captured by exactly one rule in the rule list). Then, the sum over all $N$ examples in Equation (4.10) can be reformulated using a sum over the $K_{RL}$ rules, with each rule's entropy being weighted by its support. Then, plugging (4.15) into (4.10), we obtain:

$$\mathsf{Dist}_G(\mathcal{W}^{RL}, \mathcal{W}^{Orig}) \propto \sum_{i=1}^{K_{RL}} -C_i \cdot log2 \left( \frac{1}{\mathsf{num}(a_i) - \sum_{l=1}^{i-1} \mathsf{Capt}_{RL}(a_l, a_i)} \right) \tag{4.16}$$

**Comparing Decision Trees and Rule Lists.** Comparing (4.16) to (4.11), we observe that an additional term is subtracted to the denominator of (4.16). This term corresponds to the information that the examples captured by rule $i$ did not match any of the previous rules $l < i$ within $RL$. By lowering the denominator, it raises the overall success of the probabilistic reconstruction attack. There is no such term in (4.11) because there can be no overlap between a decision tree's leaves' supports. On the contrary, the rules within a rule list can overlap because they are ordered. Overall, these theoretical results confirm that rule lists are more expressive than decision trees, encoding more information than a decision tree of equivalent size [Rivest 1987].

## 4.5 Experiments

While our proposed metric quantifies precisely and theoretically the amount of information an interpretable model carries regarding its training dataset, the aim of this section is to illustrate its practical usefulness through an example use. More precisely, we will investigate the differences between optimal and heuristically-built models, for both rule lists and decision trees.

### 4.5.1 Setup

In these experiments, we use both optimal and heuristic learning algorithms to compute decision trees and rule lists of varied sizes. Furthermore, optimal models are learnt optimizing solely accuracy, to avoid interference with other regularization terms. All details regarding the considered experimental setup are provided

hereafter.

**Learning algorithms.** We use the following learning algorithms:

- **Optimal decision trees.** We use the `DL8.5` algorithm [Aglin *et al.* 2020a, Aglin *et al.* 2020b] through its Python binding[3].
- **Heuristic decision trees.** We use an optimized version of the `CART` greedy algorithm [Breiman *et al.* 1984], as implemented within the `scikit-learn`[4] Python library [Pedregosa *et al.* 2011] with its `DecisionTreeClassifier` object. We coin this method **`sklearn_DT`**.
- **Optimal rule lists.** We use the `CORELS` algorithm [Angelino *et al.* 2017, Angelino *et al.* 2018] through its Python binding[5]. Note that `CORELS` was presented in details in Section 2.2.2.
- **Heuristic rule lists.** While some implementations exist in the literature for building *heuristic rule lists* (for example, one is provided within the `imodels`[6] library[7] [Singh *et al.* 2021]), they do not offer precise control over the desired rule support and/or maximum rule list depth. For this reason, we implemented a `CART`-like greedy algorithm (close to the `imodels`' implementation), that we coin **`GreedyRL`**. In a nutshell, this algorithm selects the rule yielding to the best Gini impurity improvement at each level of the rule list, in a top-down manner.

**Datasets.** We use two datasets (binarized, as required by `CORELS`) which are very popular in the trustworthy machine learning literature. First, the UCI Adult Income dataset[8] [Dua & Graff 2017] contains data regarding the 1994 U.S. census, with the objective of predicting whether a person earns more than \$50K/year. Numerical features are discretized using quantiles and categorical features are one-hot encoded. The resulting dataset includes $48,842$ examples and 24 binary features. As `DL8.5` was unable to learn optimal models within the specified time and memory limits for the largest size constraints, we randomly sub-sample 10% of the whole dataset. Second, the COMPAS dataset (analyzed by [Angwin *et al.* 2016]) gathers records about criminal offenders in the Broward County of Florida collected from 2013 and 2014, with the task being recidivism prediction. We consider its discretized version used to evaluate `CORELS` in [Angelino *et al.* 2017], consisting in $7,214$ examples characterized with 27 binary features[9].

[3] `https://github.com/aia-uclouvain/pydl8.5`
[4] `https://scikit-learn.org/`
[5] `https://github.com/corels/pycorels`
[6] `https://github.com/csinva/imodels`
[7] `imodels` is a Python library gathering tools to learn different types of popular interpretable machine learning models, such as decision trees, rule lists, rule sets, or scoring systems.
[8] `https://archive.ics.uci.edu/ml/datasets/adult`
[9] `https://github.com/corels/pycorels/blob/master/examples/data/compas.csv`

**Experimental Parameters.** For each experiment, we randomly select 80% of the dataset to form a training set, and use the remaining 20% as a test set to ensure that models generalize well. We repeat the experiment five times using different seeds for the random train/test split, and report results averaged across the five runs. All experiments are run on a computing cluster over a set of homogeneous nodes using Intel Platinum 8260 Cascade Lake @ 2.4Ghz CPU. Each training phase is limited to one hour of CPU time and 12 GB of RAM. Within the proposed experimental setup, all models produced by the optimal learning algorithms (`DL8.5` for decision trees or `CORELS` for rule lists) are certifiably optimal.

**Models Learning.** We set various size limits to the decision tree building algorithms, using maximum tree depths between 1 and 10 (ranging linearly by steps of 1) and (relative) minimum leaf supports between 0.01 and 0.05 (ranging linearly by steps of 0.01). For the rule list learning algorithms, we proceed identically and generate rule lists with various size constraints, using maximum depths *(number of rules within the rule list)* between 1 and 10 (ranging linearly by steps of 1) and (relative) minimum rule supports between 0.01 and 0.05 (ranging linearly by steps of 0.01). As we are interested in the optimality guarantee, we consider rules consisting in a single binary attribute (or its negation). Indeed, in our experiments, `CORELS` was unable to reach and certify optimality while also considering conjunction of features, as it dramatically increases the number of rules - and consequently, the algorithm search space. Finally, we set `CORELS`'s sparsity regularization coefficient to a value small enough (*i.e.,* smaller than $\frac{1}{N}$) to ensure that only accuracy is optimized. All methods' parameters are left to their default value.

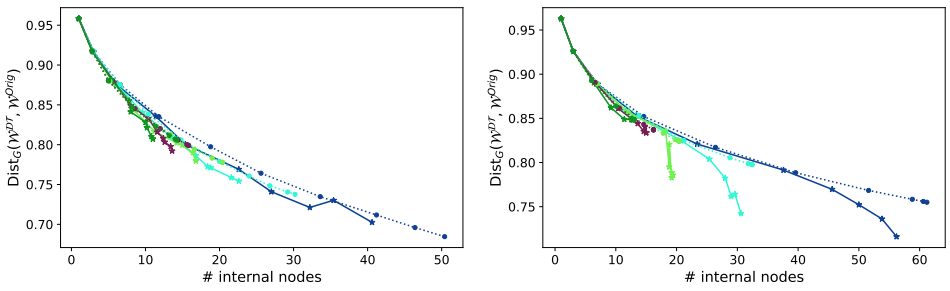
**Resources.** Source code for our implementation of the `CART`-like greedy rule list learning algorithm `GreedyRL` is provided on our repository[10]. We also provide the binarized datasets, and all scripts needed to reproduce our experiments, along with the results and plots themselves.

### 4.5.2 Results

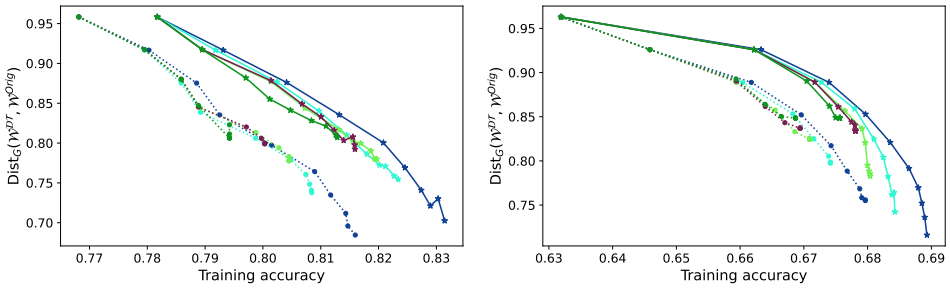
After having learnt optimal and heuristic decision trees and rule lists under various constraints, we compute the amount of information they contain regarding their training sets using $\mathsf{Dist}_G$, leveraging the computational tricks presented in Equations (4.11) and (4.16). Recall that lower uncertainty values indicate better reconstruction performances. We relate this value to two dimensions: the sizes of the models and their training accuracy. The former corresponds respectively to the number of splits performed in a decision tree or to the number of rules for a width-1 rule list. The later indicates the model's performance on its training set - *i.e.*, exactly what we aim at optimizing.

Results are provided for our experiments comparing exact and greedily-built decision trees and rule lists respectively in Figures 4.2 and 4.3. We observe the

[10] https://github.com/ferryjul/ProbabilisticDatasetsReconstruction




(a) Entropy reduction as a function of the tree size (number of splits/internal nodes).

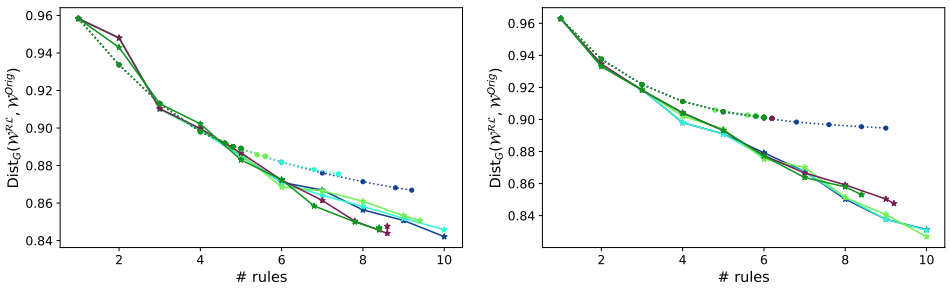



(b) Entropy reduction as a function of training accuracy.

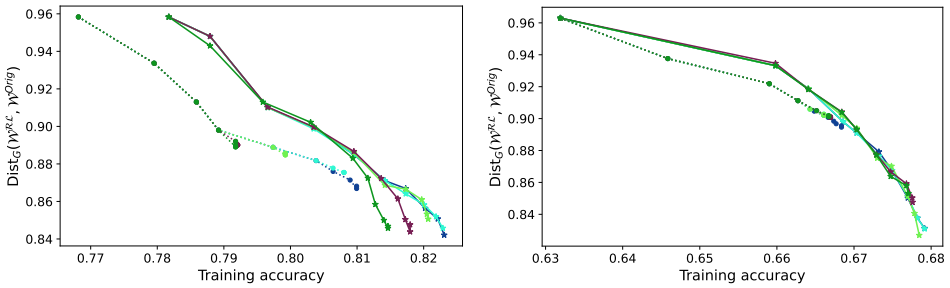

Figure 4.2: Results of our experiments comparing optimal and greedily-built decision trees (learnt respectively with `DL8.5` and `sklearn_DT`), for different (relative) minimum leaf support values. Left: Adult Income dataset, Right: COMPAS dataset.

same trends for the two types of models. First, one can observe in Figures 4.2a and 4.3a that optimal models usually represent more information in a more compact way: the reconstruction uncertainty decreases faster for optimal models than with greedily-built ones. However, while for a given size optimal models contain more information regarding their training data, they are also way more accurate. This dimension is observed in Figures 4.2b and 4.3b. More precisely, we consistently observe that for a given accuracy level, optimal models always leak less information regarding their training data. These observations can be explained by the nature of the learning algorithms. On the one side, greedy algorithms make heuristic choices iteratively. These choices are usually sub-optimal, and thus while leading to sub-optimal models (in terms of accuracy), they can also cause unnecessary leaks regarding their training data. On the other side, because they perform global optimization, optimal learning algorithms encode exactly the information needed in the most effective way.

For both datasets and types of models, the entropy reduction is not uniformly



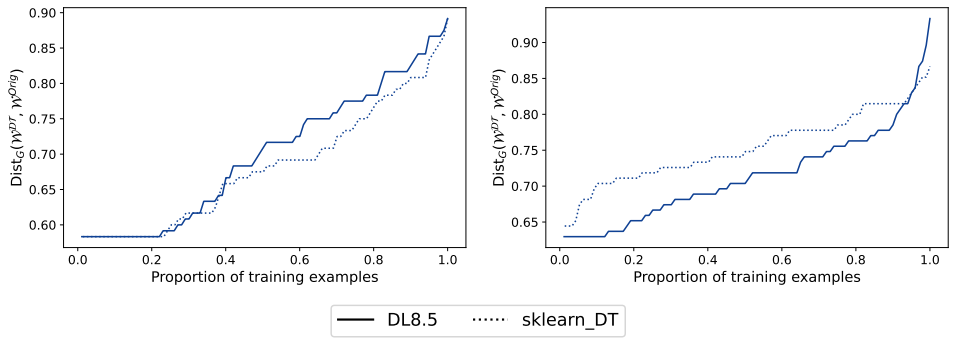

(a) Entropy reduction as a function of the rule list size (number of rules).



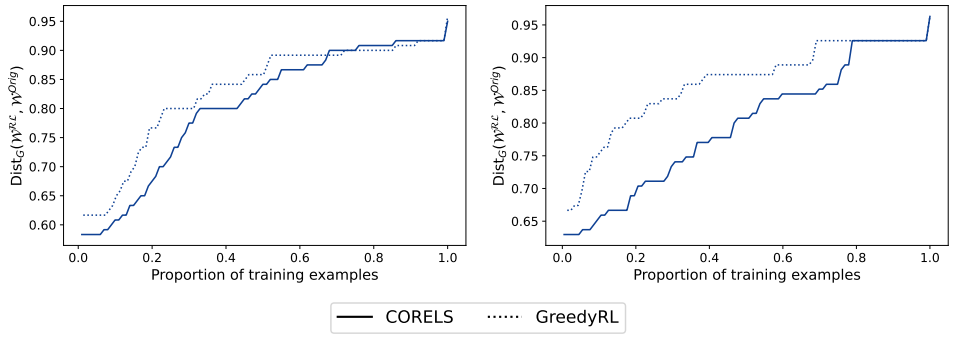

(b) Entropy reduction as a function of training accuracy.


Figure 4.3: Results of our experiments comparing optimal and greedily-built rule lists (learnt respectively with the `CORELS` and `GreedyRL` algorithms), for different (relative) minimum rule support values. Left: Adult Income dataset, Right: COMPAS dataset.

distributed across all training examples. Indeed, we plot in Figure 4.4 the minimum entropy reduction ratio as a function of the proportion of concerned training examples. One can observe that the amount of information contained by the learnt models varies significantly between different training examples. For instance, in the experiments using optimal rule lists with maximum size 10 and minimum support 0.01 on the Adult Income dataset (Figure 4.4b (left)), the less exposed training examples have an entropy reduction ratio above 0.95: knowledge of the rule list removes only 5% of the uncertainty regarding such samples. For the most exposed examples, this number becomes smaller than 0.60: knowledge of the rule list removes more then 40% of the uncertainty regarding such samples. This *disparate information leak* is intuitive: an example classified by a very long branch of a tree goes through many nodes, which gives more information regarding its features. This phenomenon is observed in all our experiments, with roughly identical distribution of the uncertainty reduction over the training datasets. It suggests that, beyond average-case uncertainty reduction as reported in Figures 4.2 and 4.3, investigating




(a) Optimal and greedily-built decision trees, learned respectively with the `DL8.5` and `sklearn_DT` algorithms.




(b) Optimal and greedily-built rule lists, learned respectively with the `CORELS` and `GreedyRL` algorithms.

Figure 4.4: Illustration of the *disparate information leak* phenomenon, for both optimal and greedily-built decision trees and rule lists, learned with the largest considered size constraints, *i.e.,* maximum depth 10 and minimum (relative) support 0.01. More precisely, we report the proportion of training examples for which the entropy reduction ratio is at most at a given value. Left: Adult Income dataset, Right: COMPAS dataset.

per-example uncertainty reductions can also be insightful.

One can note that averaging the curves of Figure 4.4 leads to the computation of dataset-wide metrics as shown in Figures 4.2 and 4.3. For instance, we observe in Figure 4.4b that for most proportions of training samples, rule lists learnt using `CORELS` exhibit a lower entropy reduction ratio than those produced by `GreedyRL`. As aforementioned, these experiments use the largest considered rule lists (learned with maximum depth 10 and minimum support 0.01) for both methods, corresponding to the rightmost points on Figure 4.3a. Observing these particular points, one can see that rule lists built with `CORELS` indeed exhibit lower entropy reduction

ratios than those built by `GreedyRL`, which is consistent with Figure 4.4b.

We observe a different trend for the decision trees learnt on the Adult Income dataset (Figure 4.4a (left)): the models built with the greedy `sklearn_DT` algorithm exhibit lower entropy reduction ratios than the optimal ones produced by `DL8.5`, hence containing more information. Again, these models correspond to the rightmost points on Figure 4.2a (left). For these experiments, the optimal models learnt with the `DL8.5` algorithm with the largest size constraint are indeed more compact than those produced by `sklearn_DT` and contain less information overall. As aforementioned, this illustrates a drawback of greedy learning algorithms: by performing local (possibly sub-optimal) choices, they can produce models performing non-necessary or redundant operations, leaking additional information regarding their training data. This dimension is further explored in the Appendix F, where we relate the actual models' sizes and entropy reduction ratios to the constraints enforced during learning.

Finally, comparing decision trees and rule lists empirically as was done theoretically in Section 4.4.3 could also be insightful. In particular, one could assess whether the rules' ordering, which allows the rules within a rule list to overlap (while the branches of a decision tree are all disjoint), empirically provides more information regarding the training data as was expected theoretically. However, such an experiment requires learning optimal rule lists whose rules' widths (*i.e.*, number of attributes involved in a rule's conjunction) match the depth of the tree's branches, which is computationally challenging. Indeed, considering sub-optimal models would bias the comparison as the results would depend on the performances of the learning algorithms rather than those of the models themselves.

## 4.6 Conclusion and Future Work

We extended previous work and proposed generic tools to represent and precisely quantify the amount of information an interpretable model encodes regarding its training data. While the practical use of such tools may be computationally challenging in the general case, we demonstrated theoretically that they can be employed efficiently under reasonable assumptions. We empirically illustrated their usefulness through an example use case: assessing the effect of optimality in training machine learning models.

A promising extension of our study consists in leveraging the knowledge of the learning algorithm's internals to lower the reconstructed generalized probabilistic dataset entropy. For instance, if a greedy algorithm uses the Gini impurity as a splitting criterion, we know that at a given node no feature other than the chosen one can yield a better Gini impurity value in the training set. Additionally, optimality itself gives information: some combinations of the attributes not used within an optimal decision tree can be discarded if they could allow the building of a better decision tree.

We observed in our experiments that the entropy reduction brought by the

knowledge of some interpretable model is not uniform across all examples of the probabilistic dataset. Investigating whether it disproportionately affects some subgroup of the population is an interesting direction. Another promising future work consists in combining the knowledge of different generalized probabilistic datasets, as was proposed in [Gambs *et al.* 2012]. This would require aligning them, as well as merging several probability distributions, while in the original setup it simply consisted in union of sets. Finally, investigating the effect of privacy-preserving methods such as the widely used *differential privacy* [Dwork *et al.* 2006, Dwork & Roth 2014] on the quality of the built probabilistic datasets (such as the differentially private decision trees proposed in [Friedman & Schuster 2010]) is also an insightful research avenue.

# Conclusions and Future Directions

## Conclusions

We have seen throughout this manuscript that while fairness, interpretability and privacy are three important dimensions of trustworthy machine learning, they often conflict, both theoretically and empirically. We then focused on using combinatorial optimization approaches to conciliate or highlight such tensions.

*Conciliating the observed conflicts.* The integer linear programming based pruning approach presented in Chapter 2 is able to prune part of an optimal fair learning algorithm's search space, effectively conciliating accuracy, fairness and interpretability. It is interesting to note that this pruning mechanism is effective *precisely* because the objective function and the fairness constraints conflict: our empirical study further shows that the stronger the conflict, the more effective the pruning mechanism. We also demonstrated how integer programming can help enhancing the generalization of statistical fairness metrics, by being able to quantify a form of fairness stability *(sample-robustness)* over a given dataset.

*Highlighting the identified tensions.* Learning a model to be fair with respect to some sensitive attributes necessarily influences the model building - as long as the fairness constraint is active, which is the case if there is some bias to be corrected. Our sensitive attributes reconstruction correction method, introduced in Chapter 3, unsurprisingly shows that this influence can be exploited to infer information regarding these sensitive attributes. It relies on either integer linear programming or constraint programming to exactly encode the considered constraints and enforce them within the computed reconstruction.
Finally, we used in Chapter 4 tools from information theory to precisely quantify the amount of information an interpretable model inherently leaks, through its structure, regarding its training data. This illustrates a tension between releasing interpretable models and keeping their training data private. Our experiments additionally show that optimal interpretable models usually represent information in a more compact way, hence leaking more information than sub-optimal models of the same size, but less information than sub-optimal models of the same utility.

Overall, applying machine learning techniques to a real-world high-stakes decision-making problem necessarily raises several challenges. First, one has to guarantee that the training data is properly protected and that the built model (or its predictions) can not be used to retrieve it. Second, it is important and legally required to ensure that the predictions do not discriminate individuals or subgroups based on protected characteristics. Third, trustworthiness, audibility and recent legal texts require that the model's logic can be understood by humans. Last and not least, while we showed throughout the thesis that many conceptual, technical and empir-

ical tensions exist between pairs of these three desiderata, one should enforce all three simultaneously, while also ensuring that the built model has a good utility. We identified synergies, or at least compatibilities, which suggests that this task is feasible but that compromises have to made. Nevertheless, this considerably increases the complexity of the learning process while requiring a thorough analysis of the used techniques. A *reasonable takeaway* could hence be that a preliminary step *before* considering the use of data-based approaches is to ensure that they are the only applicable strategy. In particular, if other approaches can be used that do not require the use of data, it may be worth it to try them at first. Finally if machine learning is the only possible method, learning a model with non-trivial utility and satisfying our three identified desiderata then requires a thorough theoretical design, being aware of the different existing tensions and of common techniques to enforce their compatibility.

Finally, it is crucial to promote an interdisciplinary approach, for computer scientists to ensure that the metrics they optimize for actually match legal and ethical requirements. This is a particularly challenging aspect: ethical analysis are often strongly context-dependent while genericity is a common practice in computer science, and not all legal and philosophical notions can easily be implemented and quantified using mathematical formulas. It is hence necessary to verify the alignment of the notions we use with the concepts we target, for the development of machine learning systems that can be trusted and that do not harm the society.

## Future Directions

As discussed throughout the thesis, many different techniques were proposed to ensure fairness, interpretability or privacy, and the three desiderata have a variety of pairwise interplays. Consequently, one can identify a great number of interesting future works to further characterize these interactions and address the observed tensions. Furthermore, we used combinatorial optimization techniques for several applications of trustworthy machine learning, and plenty other ones are possible. Hereafter, we summarize and briefly explain some future research directions.

**Fairness & Interpretability.**

*Studying the effect of regularization on fairness and other trustworthiness desiderata.* We saw that several learning algorithms consider regularization terms for sparsity (as was the case for `CORELS` as described in Section 2.2.2). It is then important to characterize the empirical effects of such sparsity regularization, and in particular see how they trade-off simplicity (as a proxy for interpretability), fairness and accuracy. Comparing the obtained trade-offs with other integrations of sparsity, such as hard constraints on the models' size, could help design proxies of interpretability that impact as least as possible the other desiderata.

*Characterizing (and improving) the fairness of hybrid interpretable models.* Hybrid interpretable models (discussed in Section 1.4.5) raise several interesting ques-

tions regarding fairness, because they inherently partition the input examples within two sets: these classified by an interpretable component, and those classified by a black-box. First, this directly results in a disparity regarding the right to explanation and the access to interpretability. Second, this may also cause disparities regarding privacy vulnerabilities. Empirically quantifying such disparities would be insightful, as well as designing privacy attacks specifically targeting hybrid interpretable models. The identified disparities could then be leveraged to propose mitigation mechanisms.

**Fairness & Privacy.**

*Characterizing the relationship between fairness sample-robustness and sensitive attributes privacy.* An interesting direction, connecting Chapter 3 to the end of Chapter 2, is to investigate the theoretical implications between our proposed sample-based robustness notion for fairness and sensitive attributes Differential Privacy. It could draw a connection between stability with respect to the training set sensitive attributes, and privacy protection of such attributes. Empirically assessing the effectiveness of fairness sample-robustness to prevent sensitive attributes inference (*e.g.,* using our own attack described in Chapter 3) would also be insightful.

**Privacy & Interpretability.**

*Optimizing the information leak while learning an interpretable model.* We showed in Chapter 4 how to quantify the amount of information an interpretable model encodes regarding its training data. An interesting future work consists in taking this value into account while training the model, to either limit its total theoretical information leak or ensure that it is homogeneously shared among the training set examples or demographic groups. Such measure could be integrated within branch-and-bound algorithms building interpretable models, such as `GOSDT` [Lin *et al.* 2020] for decision trees or `CORELS` [Angelino *et al.* 2017] for rule lists.

*Learning differentially private interpretable models.* Building interpretable and differentially private models can be done by leveraging recent advances in both interpretable machine learning algorithms and differentially private mechanisms. Furthermore, comparing the effects of different differential privacy mechanisms on the resulting training overhead and model's utility would be insightful. However, designing ad-hoc differential privacy mechanisms may lead to more interesting trade-offs between unfairness and the privacy budget. Finally, carefully integrating fairness constraints within the framework would be an ultimate step towards the holy grail: building accurate, fair, interpretable and differentially private models.

**Others.**

*Learning interpretable models from non-interpretable features.* As mentioned in Section 1.4.4, one important challenge for the development of interpretable machine learning is its use in contexts in which the examples' attributes are not in-

terpretable values. One then has to first learn an interpretable representation that can be used in a second step to compute an interpretable model. This may be done using disentanglement techniques, which consists in learning a latent representation whose dimensions encode semantically separated concepts. In addition, this technique was shown to have a positive effect on the resulting models' fairness [Locatello *et al.* 2019].

When the interpretable model training is done using declarative programming approaches, it may be directly integrated within a deep learning pipeline extracting the disentangled representation, thanks to the recent advances in end-to-end predict-and-optimize approaches [Berthet *et al.* 2020, Elmachtoub & Grigas 2022]. More precisely, these methods include the optimization models directly as a neural network layer and use different techniques to derive useful gradients from them (although these layers have no parameters to be updated). This yields the advantage that the upstream layers can be updated with respect to the overall pipeline decision, rather than only on intermediate representations accuracy.

*Personalizing explanations while preserving fairness and privacy of the user.* As discussed in Section 1.4.2, one important challenge for explainability is that it needs to be assessed with respect to the recipient of the explanations (*e.g.,* depending on its level of expertise). One possible solution is to ask information to the user to produce an appropriate explanation: the concept of *personalized explanations* was proposed [Schneider & Handali 2019] in recent years. However, while it can be useful to adapt the produced explanation, the collected additional information may also endanger the privacy of the user. Furthermore, it could be used in a discriminatory manner and result in unfair explanation mechanisms. Producing personalized explanations while tackling these two challenges is then an interesting direction.

*Leveraging declarative programming to search among a set of good interpretable models without explicit enumeration.* Recent works propose to build Rashomon sets, which are sets of *good models*. For instance, this was done for rule list models [Mata *et al.* 2022] as well as for decision trees [Xin *et al.* 2022]. It was also shown possible in the context of fair learning [Coston *et al.* 2021]. Such tools can then be leveraged to compute a set of near-optimal models and pick the preferred one according to some trustworthiness desiderata. For instance, one could select the good model with the lowest theoretical information leak (the later being quantified as described in Chapter 4).

Furthermore, other notions, such as fairness, can be characterized over an entire Rashomon set through declarative programming approaches when the learning task is formulated using them (*e.g.*, the ILP model described in Section 1.3.5). This yields two main advantages. First, the approach is generic enough to consider any metric that can be encoded within the considered framework. Second, one does not need to actually enumerate all good models within the Rashomon set to find extreme values for the chosen metric (as the solver handles the search and can guarantee optimality under some conditions).

# Appendices

Appendix A

# Summary of the Identified Interplays

In this appendix section, we provide a graphical summary of the key interplays identified between fairness, interpretability and privacy in machine learning. More precisely, we report compatibilities and synergies in Figure A.1, while we overview tensions in Figure A.2.

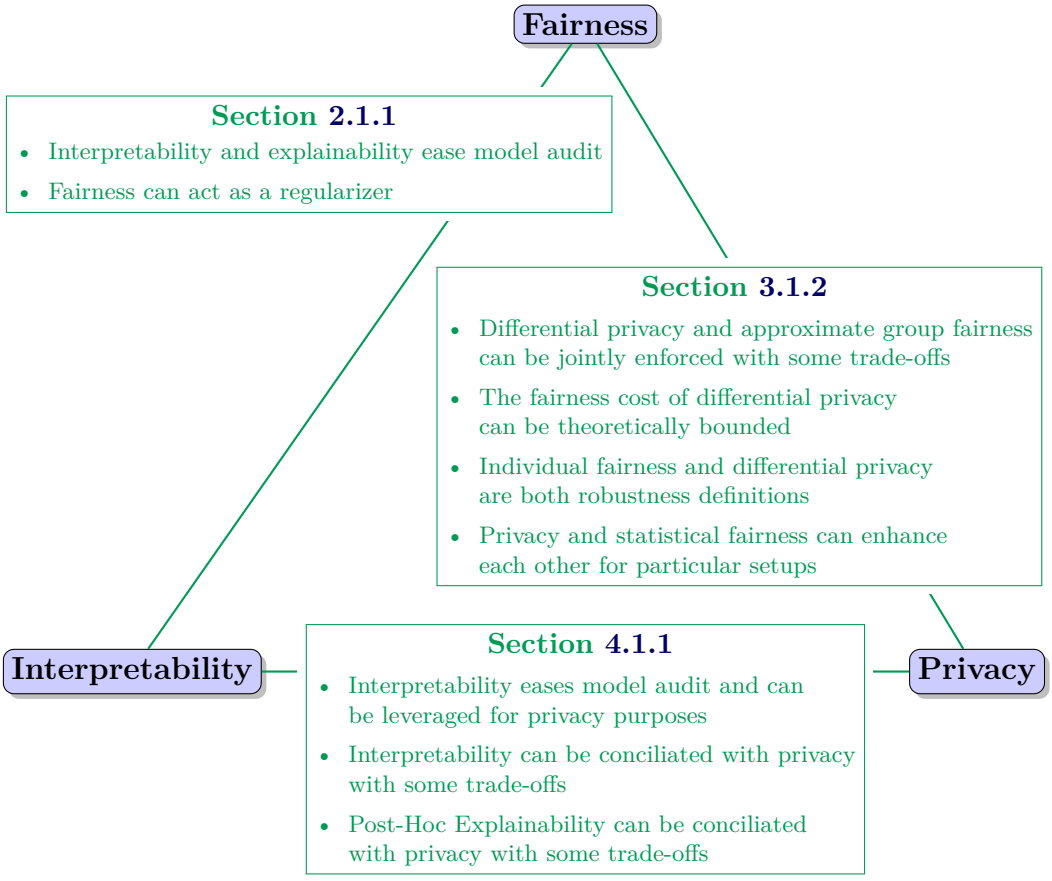



Figure A.1: Summary of the identified compatibilities and synergies between fairness, interpretability and privacy in machine learning.

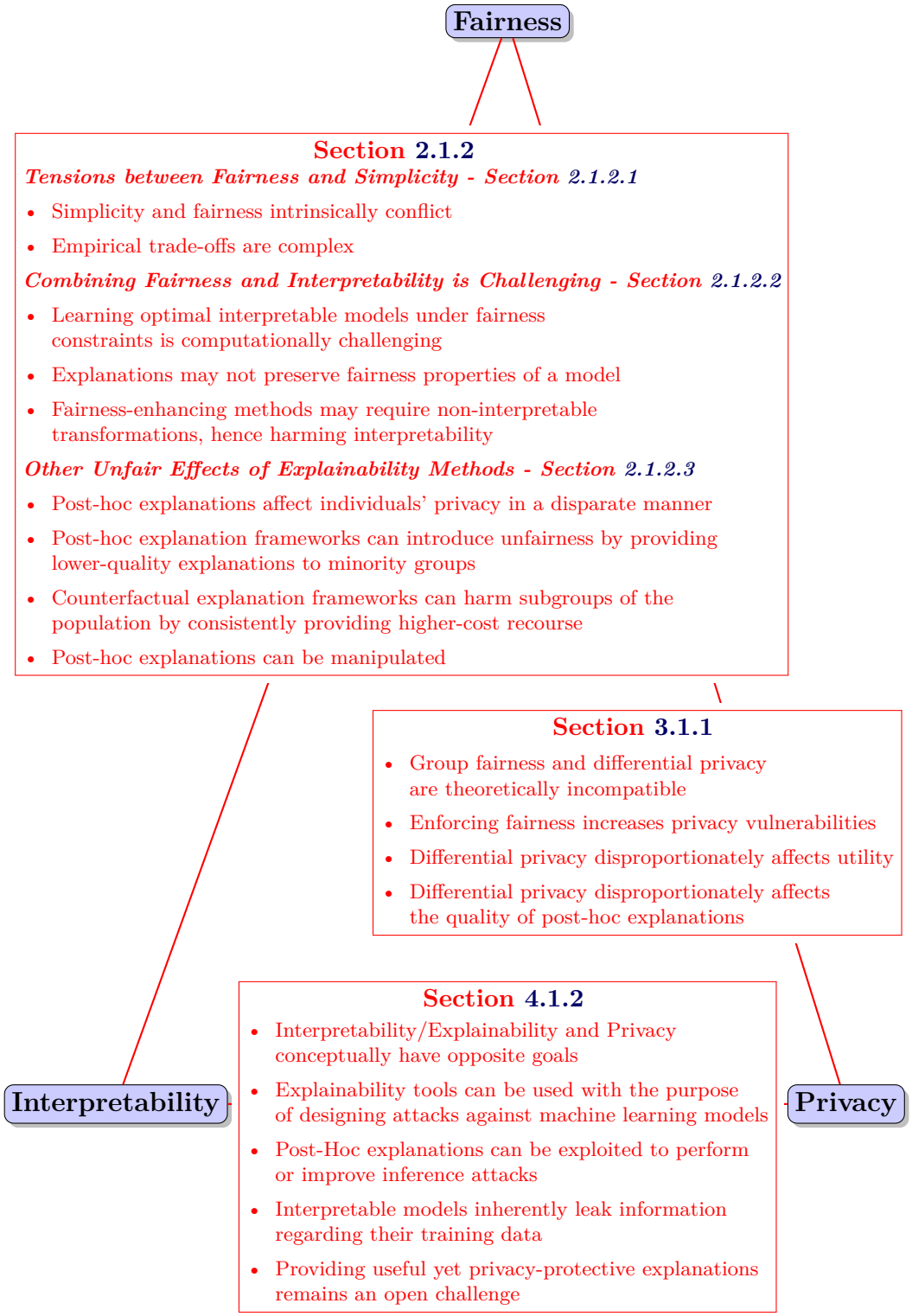



Figure A.2: Summary of the identified tensions between fairness, interpretability and privacy in machine learning.

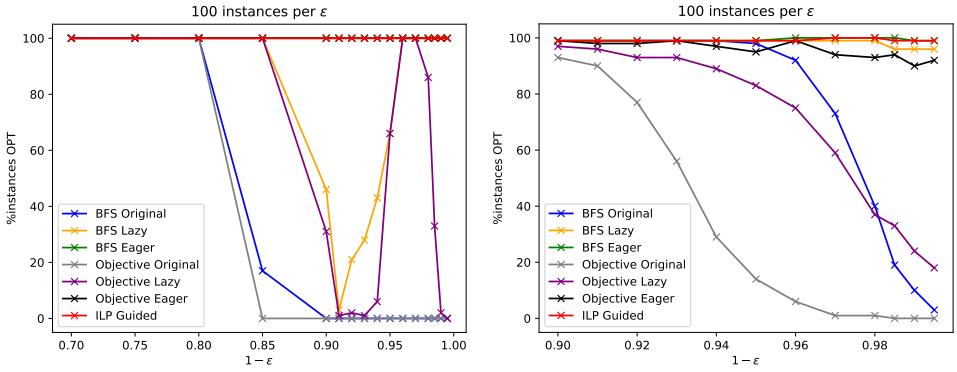
Appendix B

# ILP-Based Pruning for `FairCORELS`: Additional Results

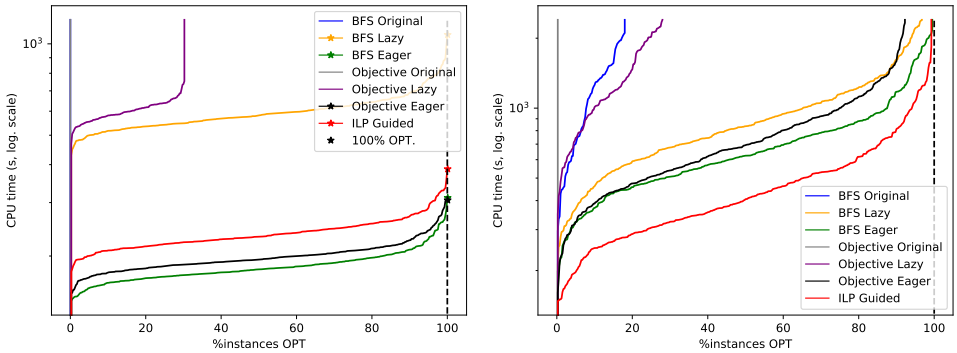
In this appendix section, we provide additional results regarding the experiments described in Section 2.4, on the use of Integer Linear Programming based pruning techniques to enhance the exploration of `FairCORELS`'s search space.

In Figure B.1, we first provide a detailed version of Figure 2.3, illustrating the results of the evaluation of our different pruning strategies using the Statistical Parity fairness metric. More precisely, Figure B.1 additionally includes the three best-first searches guided by `CORELS`'s objective, which consistently provide worst results than the Breadth First Searches with equivalent pruning strategies, as mentioned in Section 2.4.1. We then report detailed results for all other considered fairness metrics (*cf.* Table 2.1) in Figures B.2, B.3, and B.4. Interestingly, we see that the effect of the fairness constraints on the exploration is related to the amount of bias to be mitigated, which depends on the considered datasets and fairness metrics. For instance, when dealing with the statistical parity metric with the COMPAS dataset (Figure B.1a, left), an unfairness tolerance $\varepsilon = 0.10$ already prevents the original `FairCORELS` from reaching and proving optimality, whatever search heuristic is used. On the contrary, for the experiments using the predictive equality metric on the same dataset (Figure B.2a, left), the BFS-guided original `FairCORELS` is able to reach and prove optimality for all runs with unfairness tolerance $\varepsilon = 0.10$. Indeed, this metric, which takes into account the true labels for its computation (hence correcting a bias of the learning algorithm rather than a bias of the data as for the statistical parity), conflicts less with accuracy.

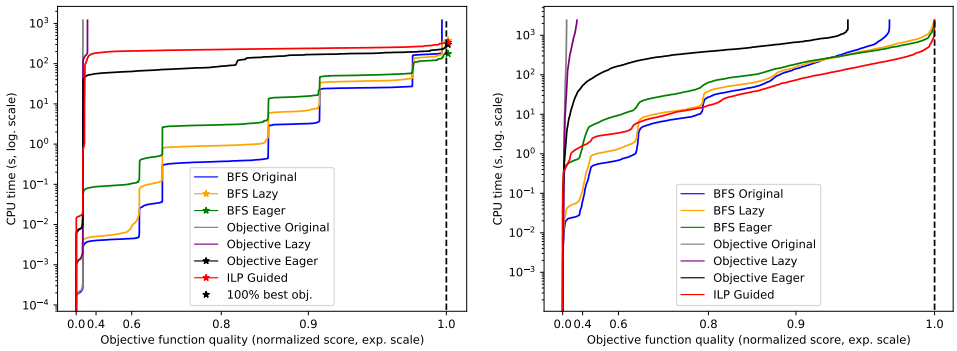
Finally, we report in Figure B.5 a summary of our experiments using the Adult Income dataset, including the *ILP-Guided* approach. As for Figure 2.4, the left plot (CPU time as a function of the proportion of instances solved to optimality) omits the experiments not using our proposed permutation map, as they never reach optimality. As mentioned in Section 2.4.3, the *ILP-Guided* strategy consistently provides the worst results among the proposed pruning methods.




(a) Proportion of instances solved to optimality as a function of $1 - \varepsilon$.

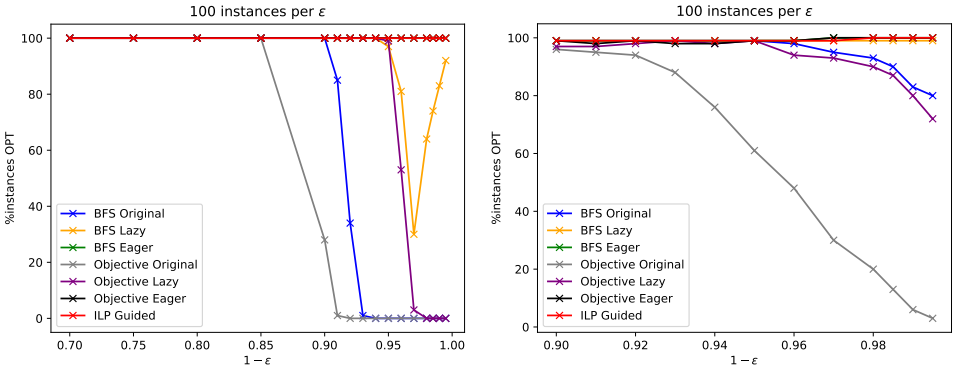



(b) CPU time as a function of the proportion of instances solved to optimality.

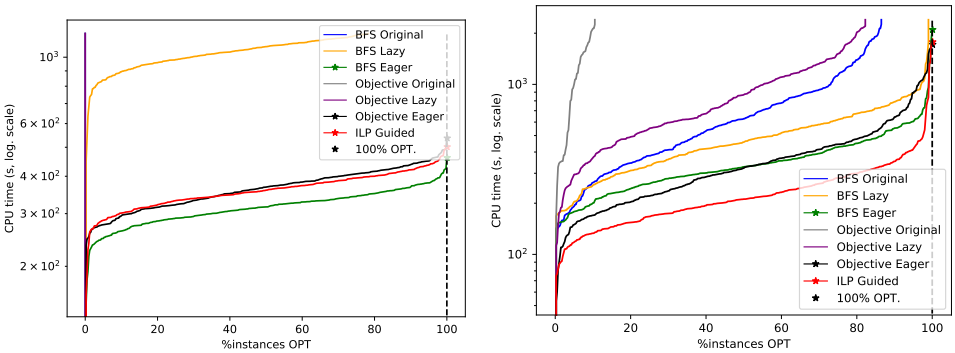



(c) Solving time as a function of the objective function quality normalized score.

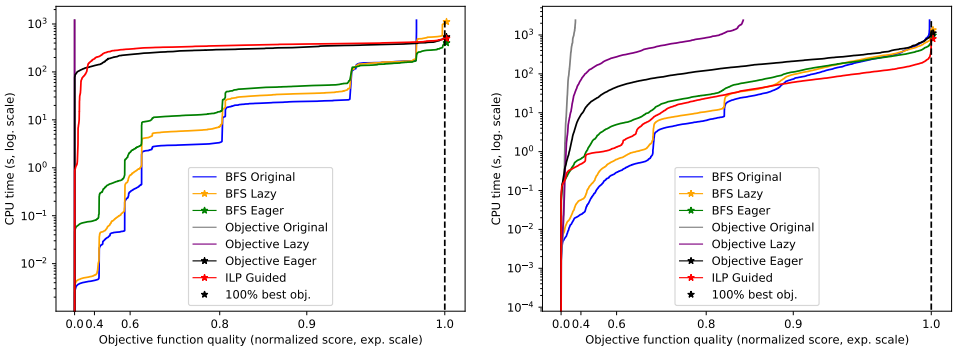
Figure B.1: Experimental evaluation of our pruning strategies for FairCORELS (left: COMPAS dataset, right: German Credit dataset) for the Statistical Parity metric.




(a) Proportion of instances solved to optimality as a function of $1 - \varepsilon$.

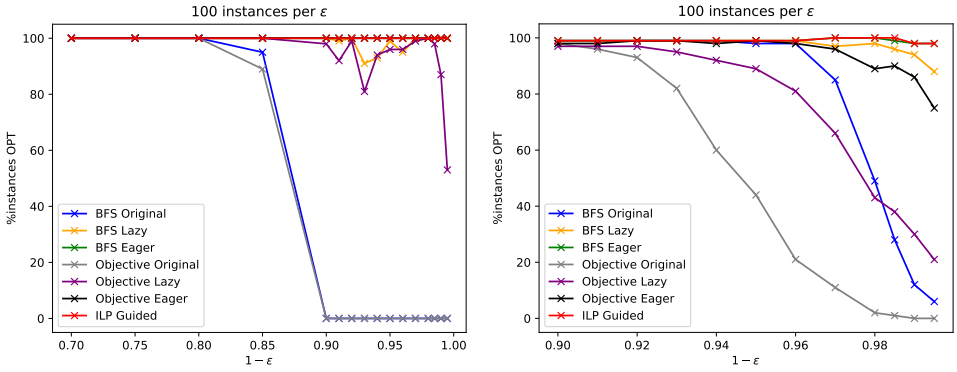



(b) CPU time as a function of the proportion of instances solved to optimality.

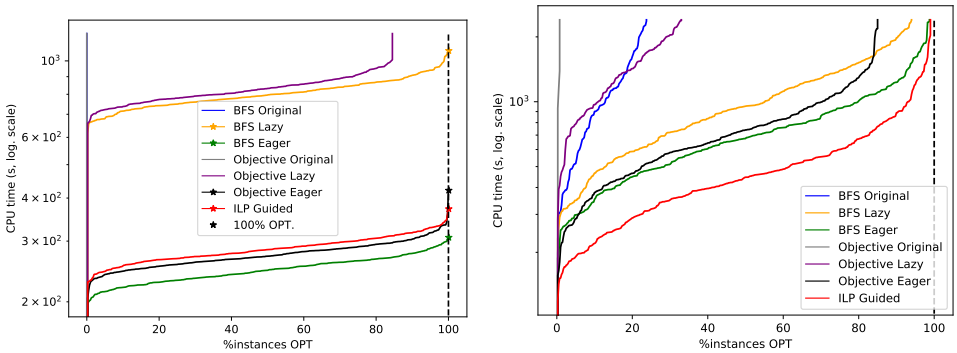



(c) Solving time as a function of the objective function quality normalized score.

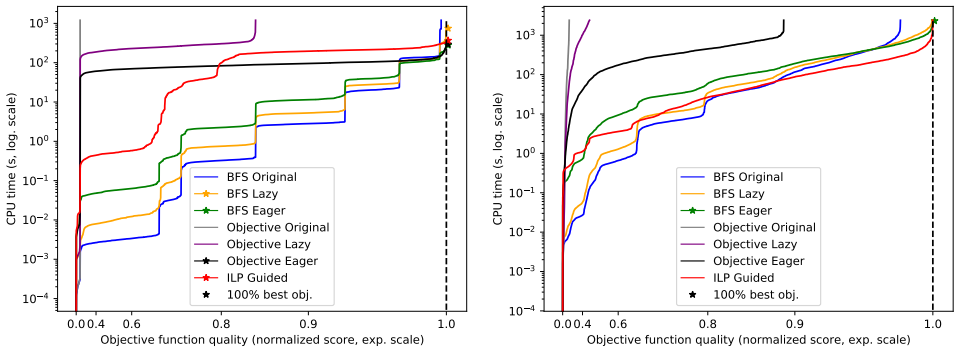
Figure B.2: Experimental evaluation of our pruning strategies for `FairCORELS` (left: COMPAS dataset, right: German Credit dataset) for the Predictive Equality metric.




(a) Proportion of instances solved to optimality as a function of $1 - \varepsilon$.

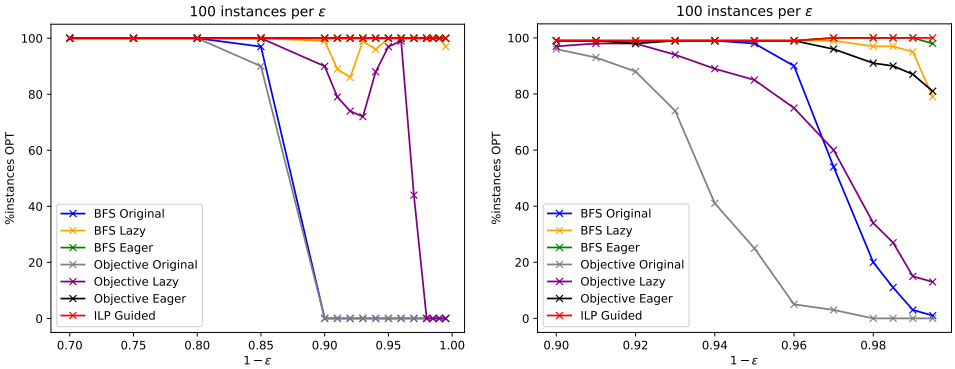



(b) CPU time as a function of the proportion of instances solved to optimality.

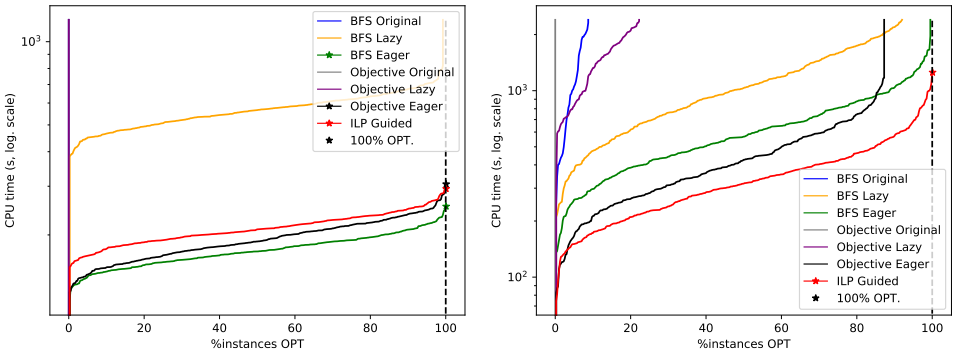



(c) Solving time as a function of the objective function quality normalized score.

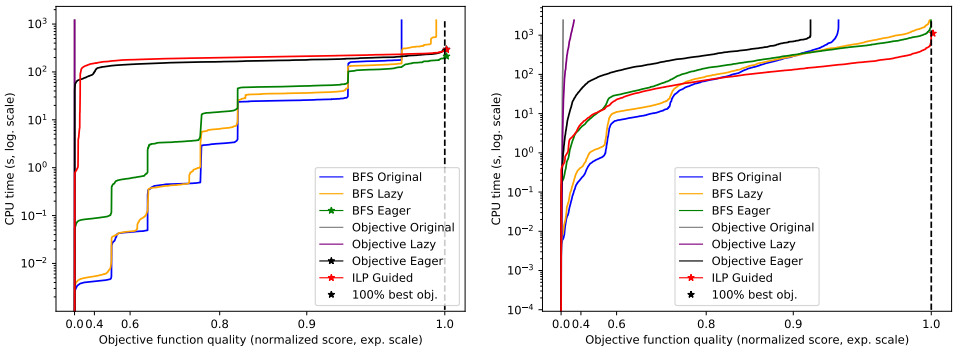
Figure B.3: Experimental evaluation of our pruning strategies for FairCORELS (left: COMPAS dataset, right: German Credit dataset) for the Equal Opportunity metric.




(a) Proportion of instances solved to optimality as a function of $1 - \varepsilon$.




(b) CPU time as a function of the proportion of instances solved to optimality.




(c) Solving time as a function of the objective function quality normalized score.

Figure B.4: Experimental evaluation of our pruning strategies for `FairCORELS` (left: COMPAS dataset, right: German Credit dataset) for the Equalized Odds metric.

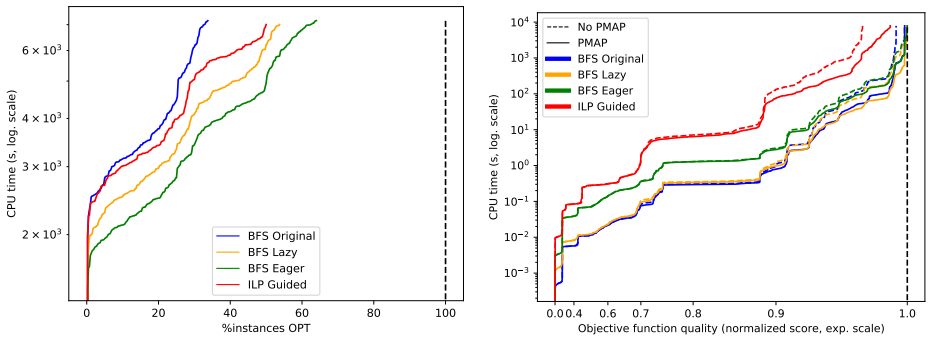



Figure B.5: Experimental evaluation of our pruning strategies for FairCORELS on the Adult Income dataset (left: CPU time as a function of the proportion of instances solved to optimality, right: CPU time as a function of the objective function score).

Appendix C

# Computing Fairness Sample-Robustness

We introduced in Section 2.5.3 our sample-based robustness notion for statistical fairness. More precisely, we defined $\mathcal{SR}(h, \mathcal{D}, \varepsilon)$ as the largest Jaccard distance around the training dataset $\mathcal{D}$ such that a classifier $h$ satisfies the fairness constraints over all subsets of $\mathcal{D}$ within this distance. In this appendix section, we show how $\mathcal{SR}(h, \mathcal{D}, \varepsilon)$ can be exactly computed by solving a simple integer programming model.

As mentioned in Section 2.2, we assume throughout Chapter 2 that $\mathcal{D}$ is partitioned into two groups: a protected group $\mathcal{D}^p$ and an unprotected group $\mathcal{D}^u$, based on the value of the sensitive feature(s). All considered fairness metrics are summarized in Table 2.1. One can note that they are all of the form:

$$\left| \frac{F_p^{\mathcal{D}}}{G_p^{\mathcal{D}}} - \frac{F_u^{\mathcal{D}}}{G_u^{\mathcal{D}}} \right| \leq \varepsilon$$

where for $s \in \{p, u\}$, $F_s^{\mathcal{D}}$ counts the number of examples satisfying some criterion among a subset of group $s$ whose cardinality is $G_s^{\mathcal{D}}$. For instance, consider the Equal Opportunity fairness metric. Recall that its expression, provided in Table 2.1, is:

$$\left| \frac{TP_{\mathcal{D},p}^h}{|\mathcal{D}^p \cap \mathcal{D}^+|} - \frac{TP_{\mathcal{D},u}^h}{|\mathcal{D}^u \cap \mathcal{D}^+|} \right| \leq \varepsilon$$

Equal Opportunity aims at equalizing the true positive rates accross the different protected groups. Then, the subset it considers for group $s \in \{p, u\}$ gathers the positive examples, and so we have: $G_s^{\mathcal{D}} = |\mathcal{D}^s \cap \mathcal{D}^+|$. Among such subset, $F_s^{\mathcal{D}}$ counts the number of positively predicted examples: $F_s^{\mathcal{D}} = TP_{\mathcal{D},s}^h$.

Recall that the Jaccard distance between two sets is computed as the ratio between the cardinalities of their intersection and their union (Definition 2). Then, the Jaccard distance between a set and any of its subsets only depends on the subset's cardinality. In order to estimate $\mathcal{SR}(h, \mathcal{D}, \varepsilon)$, we will try to find the closest (in terms of Jaccard distance) subset of $\mathcal{D}$ on which the fairness constraint is violated. We then propose to consider a simple constrained optimization problem, denoted by $\mathcal{IPSR}(h, \mathcal{D}, \varepsilon)$, to compute the minimal number of examples that need to be removed from $\mathcal{D}$ to build a subset of examples such that $h$ is not $\varepsilon$-fair over it.

**Definition 10.** ***(The integer program for quantifying sample-robustness for fairness)*** *A solution of $\mathcal{IPSR}(h, \mathcal{D}, \varepsilon)$ is a tuple $(f_p, f_u, g_p, g_u)$, in which these*

*four decision variables represent the number of examples to be removed from $\mathcal{D}$ to form a subset on which the unfairness constraint is violated.*

*More precisely, $f_s$ represents the number of examples of group $s$ satisfying the given criterion (hence counted within both $F_s^{\mathcal{D}}$ and $G_s^{\mathcal{D}}$), while $g_s$ represents the number of examples of group $s$ not satisfying the given criterion (hence counted only within $G_s^{\mathcal{D}}$). The optimal solution of $\mathcal{IPSR}(h, \mathcal{D}, \varepsilon)$ is the one minimizing the total number of examples to be removed (C.2) to build the closest (in the Jaccard sense) subset of $\mathcal{D}$.*

$$\mathcal{IPSR}(h, \mathcal{D}, \varepsilon): \tag{C.1}$$

$$\min_{f_p, f_u, g_p, g_u} \quad f_p + f_u + g_p + g_u \tag{C.2}$$

$$s.t. \quad \left|\frac{F_p^{\mathcal{D}} - f_p}{G_p^{\mathcal{D}} - f_p - g_p} - \frac{F_u^{\mathcal{D}} - f_u}{G_u^{\mathcal{D}} - f_u - g_u}\right| > \varepsilon \tag{C.3}$$

$$0 \leq f_p \leq F_p^{\mathcal{D}} \tag{C.4}$$

$$0 \leq f_u \leq F_u^{\mathcal{D}} \tag{C.5}$$

$$0 \leq g_p \leq G_p^{\mathcal{D}} - F_p^{\mathcal{D}} \tag{C.6}$$

$$0 \leq g_u \leq G_u^{\mathcal{D}} - F_u^{\mathcal{D}} \tag{C.7}$$

$$f_p + g_p < G_p^{\mathcal{D}} \tag{C.8}$$

$$f_u + g_u < G_u^{\mathcal{D}}. \tag{C.9}$$

*Constraint (C.3) encodes the fact that the fairness constraint must be violated on the resulting subset. Constraints (C.4) to (C.7) capture the variables' domains. Finally, constraints (C.8) and (C.9) enforce that at least one example of each group is kept (otherwise unfairness is undefined).*

***Illustration of $\mathcal{IPSR}(h, \mathcal{D}, \varepsilon)$ for an example metric:*** For the Equal Opportunity metric, recall that $F_s^{\mathcal{D}}$ is the number of positively labeled examples belonging to group $s$ that are positively predicted by $h$ (true positives). For this metric, $G_s^{\mathcal{D}}$ is the total number of positively labeled examples belonging to group $s$. Then, $f_s$ represents the number of examples removed from $\mathcal{D}$ that belong to group $s$ and are positively labeled and positively predicted by $h$. Removing $f_s$ such examples decrements both $F_s^{\mathcal{D}}$ and $G_s^{\mathcal{D}}$. On the other side, $g_s$ is the number of examples removed from $\mathcal{D}$ that belong to group $s$ and are positively labeled and negatively predicted by $h$. Removing $g_s$ such examples decrements only $G_s^{\mathcal{D}}$.

In the next proposition, we show that $\mathcal{IPSR}(h, \mathcal{D}, \varepsilon)$ can be used to exactly compute a classifier's fairness sample-robustness $\mathcal{SR}(h, \mathcal{D}, \varepsilon)$.

**Proposition 5.** *(**Quantifying Sample-Robustness for fairness using** $\mathcal{IPSR}(h, \mathcal{D}, \varepsilon)$) Let $(f_p^*, f_u^*, g_p^*, g_u^*)$ be the optimal solution of $\mathcal{IPSR}(h, \mathcal{D}, \varepsilon)$. Then:*

$$\mathcal{SR}(h, \mathcal{D}, \varepsilon) = \frac{f_p^* + f_u^* + g_p^* + g_u^*}{|\mathcal{D}|}.$$

*Proof.* To prove this equality, we will need to prove the two conditions of Definition 4.

Let $z^* = f_p^* + f_u^* + g_p^* + g_u^*$ be the value of the objective function of the optimal solution of $\mathcal{IPSR}(h, \mathcal{D}, \varepsilon)$. We define $\tau^* = \frac{z^*}{|\mathcal{D}|} = \frac{f_p^* + f_u^* + g_p^* + g_u^*}{|\mathcal{D}|}$. Then:

1. Consider $\mathcal{D}^*$, the subset of $\mathcal{D}$ formed by removing $f_p^*$ (respectively $f_u^*$) examples of group $p$ (respectively $u$) satisfying the statistical criterion, and $g_p^*$ (respectively $g_u^*$) examples of group $p$ (respectively $u$) not satisfying the statistical criterion. The bounds of the decision variables of Problem (C.1) enforce that $\mathcal{D}^*$ exists. We have: $J(\mathcal{D}, \mathcal{D}^*) = \frac{f_p^* + f_u^* + g_p^* + g_u^*}{|\mathcal{D}|} = \tau^*$. Additionally, we know that $\mathsf{unf}(h, \mathcal{D}^*) > \varepsilon$, because $(f_p^*, f_u^*, g_p^*, g_u^*)$ is a solution of $\mathcal{IPSR}(h, \mathcal{D}, \varepsilon)$ and then necessarily satisfies Constraint (C.3). Hence, $\forall \tau \geq \tau^*, \exists \mathcal{D}' = \mathcal{D}^* \in \mathcal{B}(\mathcal{D}, \tau)$ such that $\mathsf{unf}(h, \mathcal{D}') > \varepsilon$.

2. Assume that $\exists \mathcal{D}'' \in \mathcal{B}(\mathcal{D}, \tau)$ with $\tau < \tau^*$ such that $\mathsf{unf}(h, \mathcal{D}'') > \varepsilon$. Then, $\mathcal{D}''$ is formed by removing $z'' < z^*$ examples from $\mathcal{D}$. In addition, $\mathcal{D}''$ is a solution to Problem (C.1) as $\mathsf{unf}(h, \mathcal{D}'') > \varepsilon$. This contradicts the fact that $z^*$ is the optimal objective value of Problem (C.1). Hence, $\forall \tau < \tau^*, \forall \mathcal{D}' \in \mathcal{B}(\mathcal{D}, \tau), \mathsf{unf}(h, \mathcal{D}') \leq \varepsilon$.

Finally, by (1) and (2), $\tau^* = \mathcal{SR}(h, \mathcal{D}, \varepsilon)$. □

Finally, one can compute $\mathcal{SR}(h, \mathcal{D}, \varepsilon)$, the fairness sample-robustness of any fixed classifier $h$ on dataset $\mathcal{D}$ for a chosen unfairness metric and tolerance value $\varepsilon$, solving $\mathcal{IPSR}(h, \mathcal{D}, \varepsilon)$. In all our experiments, this simple integer program is solved within fractions of seconds using the `OR-Tools`[Perron & Furnon 2019] CP-SAT solver[1]. As mentioned in Section 2.5.3, we also proposed in [Ferry *et al.* 2023b] a simple, linear-time greedy algorithm upper-bounding $\mathcal{SR}(h, \mathcal{D}, \varepsilon)$. In a nutshell, the intuition behind this algorithm is to start from the entire training set $\mathcal{D}$, and iteratively remove the example maximizing the unfairness violation increase until the chosen fairness violation level is reached (or there is no more example to remove). Importantly, at each step, the algorithm needs only consider the four possible moves corresponding to the four decision variables of $\mathcal{IPSR}(h, \mathcal{D}, \varepsilon)$.

[1] https://github.com/google/or-tools

Appendix D

# Reconstruction Correction Models for Multi-Valued Sensitive Attributes

In this appendix section, we discuss the most general setting in which the sensitive attribute is multi-valued and takes one of $|\mathcal{S}|$ values (hence effectively defining $|\mathcal{S}|$ protected groups). One may also observe that this general setting covers the *intersectional fairness* notions [Kearns *et al.* 2018] (also called *subgroup fairness*) in which protected groups are defined with respect to combinations of values of several sensitive attributes. Indeed, the intersectional fairness case can be cast to the scenario in which we have a single, multi-valued sensitive attribute, by creating one sensitive attribute value per combination of the attributes considered for intersectional fairness.

Hereafter, we explain how both models can be extended to handle multi-valued sensitive attributes reconstruction and discuss the complexity cost induced by this extension. We begin with the general reconstruction correction model, which is suitable to encode any constraint on the protected attributes. We then treat the efficient model, which can be used to encode any rate constraints on the protected attributes (such as, but not restricted to, statistical fairness constraints).

## D.1 General Reconstruction Correction Model

The general reconstruction correction model $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$ uses exactly one decision variable to encode each training example's sensitive attribute. Extension to the general multi-valued sensitive attributes case hence requires modifying the domains of such variables to match that of the sensitive attributes (with $|\mathcal{S}|$ different possible values). The $N$ decision variables now have domain of cardinality $|\mathcal{S}|$. The objective function sums the (weighted) changes in the adversary's sensitive attributes guess, as was done in the binary case in (3.1). $|\mathcal{S}|$ constraints ensure that there is at least one example from each protected group (as was done with (3.2) for the binary sensitive attribute setting). Finally, one fairness constraint is declared for each protected group (sensitive attribute value), ensuring that its positive prediction rate is no further than $\varepsilon$ from that of the entire dataset (as was done with (3.3) and (3.4) for the binary sensitive attribute setting).

Overall, the size of the search space of $\mathcal{RC}(\hat{S}, P, \hat{Y}, \varepsilon)$ is $O(|\mathcal{S}|^N)$, which generalizes the binary sensitive attribute case for which it was $O(2^N)$.

## D.2 Efficient Model for Statistical Fairness

The efficient reconstruction correction model $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$ uses one decision variable to count the number of changes from one sensitive attribute value to another, for each pair of sensitive attributes values. Extension to the general multi-valued sensitive attributes case hence requires declaring $O(|\mathcal{S}|^2)$ variables. To ensure that each example is counted only once, $O(|\mathcal{S}|)$ constraints must be declared. Furthermore, to quantify the total cost of the performed changes, $O(|\mathcal{S}|^2)$ `element` constraints have to be summed in the objective function, as was performed in (3.5) in the binary sensitive attributes case. $|\mathcal{S}|$ constraints ensure that there is at least one example from each protected group (as was done with (3.6) and (3.7) for the binary sensitive attribute setting). Finally, one fairness constraint is declared for each protected group (sensitive attribute value), ensuring that its positive prediction rate is no further than $\varepsilon$ from that of the entire dataset (as was done with (3.8) and (3.9) for the binary sensitive attribute setting).

Overall, the size of the search space of $\mathcal{RC}_{\mathcal{E}}(\hat{S}, P, \hat{Y}, \varepsilon)$ is $O(N^{|\mathcal{S}|^2})$, which generalizes the binary sensitive attribute case for which it was $O(N^4)$.

Appendix E

# Sensitive Attributes Reconstruction Correction: Additional Experiments

In this appendix section, we provide results for additional experiments using a pre-processing method for fairness: the CorrelationRemover method, implemented in the `Fairlearn` library [Bird *et al.* 2020]. In a nutshell, the CorrelationRemover transforms the training set unsensitive attributes in order to remove their correlations with the sensitive ones. A traditional machine learning algorithm is then used on the sanitized data (pre-processed unsensitive attributes) to produce a fair model. When using this model for inference on unseen data, the transformation learnt by the CorrelationRemover has to be performed (on the unsensitive attributes) first.

The CorrelationRemover does not optimize statistical fairness metrics explicitly. Indeed, bias against sensitive attributes is removed before training the model, in the data pre-processing step. Hence, in order to perform sensitive attributes reconstruction correction, one has to infer some fairness information. To do so, we use the strategy described in Section 3.4.2: the attacker measures the target model's unfairness on its own attack set, and chooses the metric with the smallest value. The experimental setup is similar to that of Section 3.3.3. However, because the CorrelationRemover method does not optimize a particular fairness metric nor a particular tolerance value, we only perform one experiment for each dataset (repeated 100 times with different random seeds).

The results presented in Table E.1 show that even in this context, the reconstruction correction step still provides significant reconstruction accuracy improvements. In all situations, the attacker was able to infer a valid fairness constraint and to leverage it to improve the initial sensitive attributes reconstruction. Finally, these additional experiments confirm that the type of fairness intervention does not influence the performances of our proposed reconstruction correction step. The key factor for allowing reconstruction correction is that the predictions of the target model should be more fair than the original data. In this situation, the original attacker's reconstruction will likely be more biased than the (fair) target model's predictions, which will allow some reconstruction correction.

Table E.1: Summary of the results of our sensitive attributes reconstruction correction experiments using a pre-processing method for fairness, with the attacker inferring the fairness information. We report the accuracy performances of the trained (target) model, the results of the fairness constraint estimation process (inferred metrics and average inferred tolerance), and the reconstruction performances.

| **Target model $h$ (under attack)** | | **Estimated Constraint** | | **Reconstruction Accuracy** | | | |
|---|---|---|---|---|---|---|---|
| | | | | **Baseline** | | **Corrected** | |
| ***Train Acc.*** | ***Test Acc.*** | ***Estimated Metric*** | ***Estimated Tolerance*** | $\mathcal{A}$ | $\mathcal{A}'$ | $\mathcal{A}$ | $\mathcal{A}'$ |
| **UCI Adult Income dataset** | | | | | | | |
| 0.860 ±0.003 | 0.848 ±0.003 | PE (68%), EO (32%) | 0.023 ±0.013 | 0.808 ±0.005 | 0.806 ±0.005 | 0.828 ±0.013 | **0.827 ±0.014** |
| **ACSPublicCoverage dataset** | | | | | | | |
| 0.862 ±0.001 | 0.852 ±0.002 | PE (92%), SP (8%) | 0.006 ±0.004 | 0.861 ±0.005 | 0.860 ±0.006 | 0.863 ±0.005 | **0.872 ±0.010** |
| **ACSIncome dataset** | | | | | | | |
| 0.798 ±0.002 | 0.785 ±0.003 | PE (100%) | 0.056 ±0.016 | 0.690 ±0.007 | 0.685 ±0.008 | 0.704 ±0.014 | **0.763 ±0.009** |

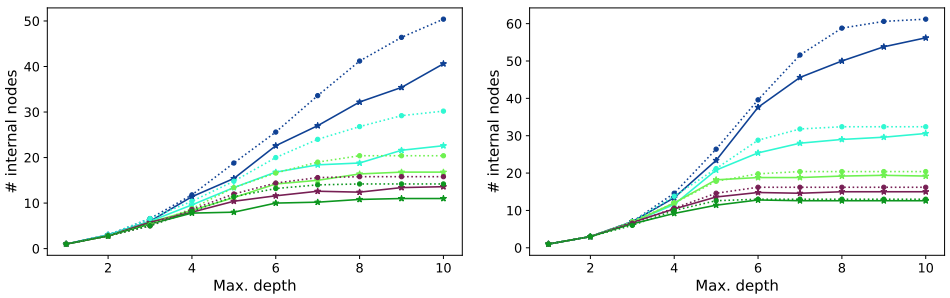
Appendix F

# Dataset Reconstruction from Interpretable Models: Additional Results

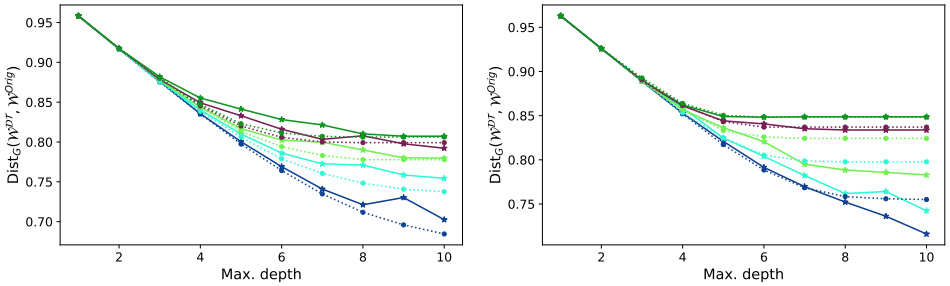
We observed in Section 4.5.2 that optimal models (either decision trees or rule lists) usually contain more information than greedily-built ones of the same size. However, when related to the models' utility (accuracy on the training data), this trend is reversed, and optimal models leak less information regarding their training data than greedily-built ones for the same performances level. This was explained by the fact that greedy learning algorithms iteratively make local choices that are sub-optimal, overall adding unnecessary information to the resulting model (*e.g.,* performing more splits than necessary within decision trees).

In this appendix section, we relate the amount of information an interpretable model carries to the size constraints that were enforced to build it. More precisely, we report in Figures F.1a and F.2a the resulting model size as a function of the maximum depth constraint, for the different minimum support constraints. Again, the model size is quantified as the number of internal nodes for a decision tree, or as the number of rules for width-1 rule lists. We also report in Figures F.1b and F.2b the overall entropy reduction ratio, as a function of the maximum depth constraint.

One can observe in Figure F.1a that, as expected, the number of internal nodes within the built decision trees grows with the maximum depth value. Enforcing large values of the (relative) minimum leaf support quickly prevents the trees from expanding, as no split can be performed while satisfying the minimum support constraint. Hence, as expected, lowering the minimum support value leads to the computation of larger decision trees. Comparing greedily-built and optimal decision trees, one can note that the models learnt by `sklearn_DT` contain more nodes than the optimal ones built using `DL8.5`, for the same provided parameters (*i.e.,* minimum leaf support and maximum depth values). This can be explained by the fact that `sklearn_DT` often adds non-necessary splits as it iteratively performs local, sub-optimal choices. Meanwhile, many branches do not reach the enforced maximum depth within the optimal decision trees thanks to the performed global optimization which considers a split only if it is necessary. As a consequence, we observe in Figure F.1b (left) that, for fixed parameters, the decision trees produced by `sklearn_DT` on the Adult Income dataset contain more information than those learnt by `DL8.5`. For the COMPAS dataset (Figure F.1b (right)), we observe the opposite trend. This can be explained by two observations. First, the size difference




(a) Experiments relating the actual models' sizes to the size constraints enforced during learning. We report tree size (number of splits/internal nodes) as a function of the maximum depth constraint.

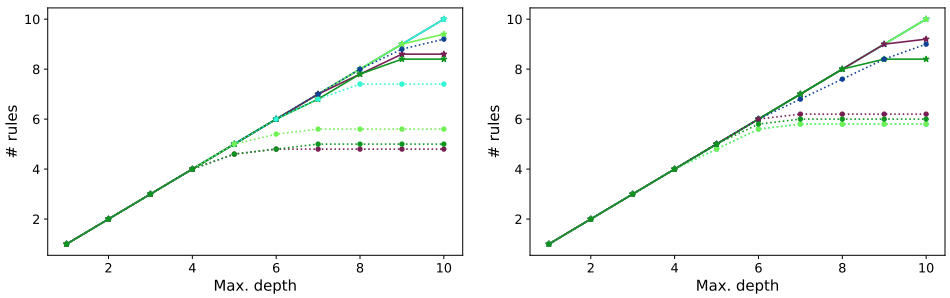



(b) Experiments relating the entropy reduction ratio to the size constraints enforced during learning. We report the entropy reduction as a function of the maximum depth constraint.

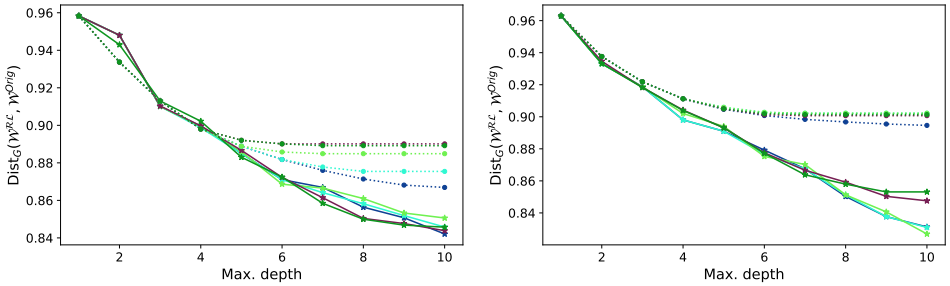

Figure F.1: Results of our experiments comparing optimal and greedily-built decision trees (learnt respectively with `DL8.5` and `sklearn_DT`), for different (relative) minimum leaf support values. Left: Adult Income dataset, Right: COMPAS dataset.

between optimal and greedily-built decision trees is smaller on the COMPAS dataset (Figure F.1a (right)) than on the Adult dataset (Figure F.1a (left)). Then, in average, we saw within Section 4.5.2 (Figure 4.2a) that, for equivalent sizes, the optimal decision trees carry more information than the greedily-built ones.

Figure F.2a shows that, as expected, the number of rules within the built rule lists grows with the enforced maximum depth value. As for the decision trees, largest values of the enforced minimum rule support prevent expansion of the rule lists, when no rule satisfying the minimum support constraint can be found. This is particularly true for the greedy learning algorithm. Indeed, at each iteration, the algorithm selects a rule maximizing a given criterion (*i.e.,* minimizing Gini Impurity). Then, the examples not captured by the rules fall into the rest of the rule list, and are used for the next iterations. If the algorithm selects rules with large supports during the first iterations, there may be too few remaining examples to be




(a) Experiments relating the actual models' sizes to the size constraints enforced during learning. We report rule list size (number of rules) as a function of the maximum depth constraint.




(b) Experiments relating the entropy reduction ratio to the size constraints enforced during learning. We report the entropy reduction as a function of the maximum depth constraint.


Figure F.2: Results of our experiments comparing optimal and greedily-built rule lists (learnt respectively with the `CORELS` and `GreedyRL` algorithms), for different (relative) minimum rule support values. Left: Adult Income dataset, Right: COMPAS dataset.

able to add new rules. This drawback is not observed with `CORELS`, as it performs global optimization. As a direct consequence, one can see in Figure F.2b that, for fixed parameters (*i.e.,* minimum rule support and maximum depth values), the rule lists built using `CORELS` contain more information than those produced by `GreedyRL`. This trend is related to the observed size difference, but is also exacerbated by the fact that, as observed in Section 4.5.2 (Figure 4.3a), optimal rule lists usually encode more information that greedily-built ones of equivalent size.

# Bibliography


[Aalmoes *et al.* 2022] Jan Aalmoes, Vasisht Duddu and Antoine Boutet. *Dikaios: Privacy Auditing of Algorithmic Fairness via Attribute Inference Attacks.* arXiv preprint arXiv:2202.02242, 2022. (Cited on pages 42, 43, 87, 98, 105 and 108.)

[Abadi *et al.* 2016] Martín Abadi, Andy Chu, Ian J. Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar and Li Zhang. *Deep Learning with Differential Privacy.* In Edgar R. Weippl, Stefan Katzenbeisser, Christopher Kruegel, Andrew C. Myers and Shai Halevi, editors, Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016, pages 308–318. ACM, 2016. (Cited on pages 45, 47, 90 and 120.)

[Abowd 2018] John M. Abowd. *The U.S. Census Bureau Adopts Differential Privacy.* In Yike Guo and Faisal Farooq, editors, Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018, page 2867. ACM, 2018. (Cited on page 40.)

[Agarwal *et al.* 2018] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford and Hanna M. Wallach. *A Reductions Approach to Fair Classification.* In Jennifer G. Dy and Andreas Krause, editors, Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018, volume 80 of *Proceedings of Machine Learning Research*, pages 60–69. PMLR, 2018. (Cited on pages 21, 26, 27, 94, 95, 107, 109 and 115.)

[Agarwal 2021a] Sushant Agarwal. *Trade-Offs between Fairness and Interpretability in Machine Learning.* In IJCAI 2021 Workshop on AI for Social Good, 2021. (Cited on pages 33 and 52.)

[Agarwal 2021b] Sushant Agarwal. *Trade-Offs between Fairness and Privacy in Machine Learning.* In IJCAI 2021 Workshop on AI for Social Good, 2021. (Cited on pages 87, 88 and 113.)

[Aghaei *et al.* 2019] Sina Aghaei, Mohammad Javad Azizi and Phebe Vayanos. *Learning Optimal and Fair Decision Trees for Non-Discriminative Decision-Making.* In The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019, pages 1418–1426. AAAI Press, 2019. (Cited on pages 12, 16, 30 and 53.)

[Aglin *et al.* 2020a] Gaël Aglin, Siegfried Nijssen and Pierre Schaus. *Learning Optimal Decision Trees Using Caching Branch-and-Bound Search.* In The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020, pages 3146–3153. AAAI Press, 2020. (Cited on page 139.)

[Aglin *et al.* 2020b] Gaël Aglin, Siegfried Nijssen and Pierre Schaus. *PyDL8.5: a Library for Learning Optimal Decision Trees.* In Christian Bessiere, editor, Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020, pages 5222–5224. ijcai.org, 2020. (Cited on page 139.)

[Aïvodji *et al.* 2019a] Ulrich Aïvodji, Hiromi Arai, Olivier Fortineau, Sébastien Gambs, Satoshi Hara and Alain Tapp. *Fairwashing: the risk of rationalization.* In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA, volume 97 of *Proceedings of Machine Learning Research*, pages 161–170. PMLR, 2019. (Cited on pages 54 and 55.)

[Aïvodji *et al.* 2019b] Ulrich Aïvodji, Julien Ferry, Sébastien Gambs, Marie-José Huguet and Mohamed Siala. *Learning Fair Rule Lists.* CoRR, vol. abs/1909.03977, 2019. (Cited on pages 3, 50 and 71.)

[Aïvodji *et al.* 2020] Ulrich Aïvodji, Alexandre Bolot and Sébastien Gambs. *Model extraction from counterfactual explanations.* CoRR, vol. abs/2009.01884, 2020. (Cited on pages 123 and 126.)

[Aïvodji *et al.* 2021a] Ulrich Aïvodji, Hiromi Arai, Sébastien Gambs and Satoshi Hara. *Characterizing the risk of fairwashing.* In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang and Jennifer Wortman Vaughan, editors, Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual, pages 14822–14834, 2021. (Cited on pages 34 and 55.)

[Aïvodji *et al.* 2021b] Ulrich Aïvodji, François Bidet, Sébastien Gambs, Rosin Claude Ngueveu and Alain Tapp. *Local Data Debiasing for Fairness Based on Generative Adversarial Training.* Algorithms, vol. 14, no. 3, page 87, 2021. (Cited on page 21.)

[Aïvodji *et al.* 2021c] Ulrich Aïvodji, Julien Ferry, Sébastien Gambs, Marie-José Huguet and Mohamed Siala. *FairCORELS, an Open-Source Library for Learning Fair Rule Lists.* In Gianluca Demartini, Guido Zuccon, J. Shane

Culpepper, Zi Huang and Hanghang Tong, editors, CIKM '21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021, pages 4665–4669. ACM, 2021. (Cited on pages 3 and 50.)

[Aïvodji *et al.* 2022] Ulrich Aïvodji, Julien Ferry, Sébastien Gambs, Marie-José Huguet and Mohamed Siala. *Leveraging Integer Linear Programming to Learn Optimal Fair Rule Lists.* In Pierre Schaus, editor, Integration of Constraint Programming, Artificial Intelligence, and Operations Research - 19th International Conference, CPAIOR 2022, Los Angeles, CA, USA, June 20-23, 2022, Proceedings, volume 13292 of *Lecture Notes in Computer Science*, pages 103–119. Springer, 2022. (Cited on page 3.)

[Aktay *et al.* 2020] Ahmet Aktay, Shailesh Bavadekar, Gwen Cossoul, John Davis, Damien Desfontaines, Alex Fabrikant, Evgeniy Gabrilovich, Krishna Gadepalli, Bryant Gipson, Miguel Guevara, Chaitanya Kamath, Mansi Kansal, Ali Lange, Chinmoy Mandayam, Andrew Oplinger, Christopher Pluntke, Thomas Roessler, Arran Schlosberg, Tomer Shekel, Swapnil Vispute, Mia Vu, Gregory Wellenius, Brian Williams and Royce J. Wilson. *Google COVID-19 Community Mobility Reports: Anonymization Process Description (version 1.0).* CoRR, vol. abs/2004.04145, 2020. (Cited on page 40.)

[Alber *et al.* 2019] Maximilian Alber, Sebastian Lapuschkin, Philipp Seegerer, Miriam Hägele, Kristof T. Schütt, Grégoire Montavon, Wojciech Samek, Klaus-Robert Müller, Sven Dähne and Pieter-Jan Kindermans. *iNNvestigate Neural Networks!* Journal of Machine Learning Research, vol. 20, no. 93, pages 1–8, 2019. (Cited on page 11.)

[Alghamdi *et al.* 2022] Wael Alghamdi, Hsiang Hsu, Haewon Jeong, Hao Wang, P. Winston Michalak, Shahab Asoodeh and Flávio P. Calmon. *Beyond Adult and COMPAS: Fairness in Multi-Class Prediction.* CoRR, vol. abs/2206.07801, 2022. (Cited on page 25.)

[Angelino *et al.* 2017] Elaine Angelino, Nicholas Larus-Stone, Daniel Alabi, Margo Seltzer and Cynthia Rudin. *Learning Certifiably Optimal Rule Lists.* In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '17, page 35–44. Association for Computing Machinery, 2017. (Cited on pages 50, 57, 61, 130, 139 and 149.)

[Angelino *et al.* 2018] Elaine Angelino, Nicholas Larus-Stone, Daniel Alabi, Margo Seltzer and Cynthia Rudin. *Learning Certifiably Optimal Rule Lists for Categorical Data.* Journal of Machine Learning Research, vol. 18, no. 234, pages 1–78, 2018. (Cited on pages 50, 57, 61, 66, 130 and 139.)

[Angwin *et al.* 2016] Julia Angwin, Jeff Larson, Surya Mattu and Lauren Kirchner. *Machine bias: There's software used across the country to predict future*

*criminals. And it's biased against blacks. ProPublica (2016).* ProPublica, May, vol. 23, 2016. (Cited on pages 10, 67 and 139.)

[Ateniese *et al.* 2015] Giuseppe Ateniese, Luigi V. Mancini, Angelo Spognardi, Antonio Villani, Domenico Vitali and Giovanni Felici. *Hacking Smart Machines with Smarter Ones: How to Extract Meaningful Data from Machine Learning Classifiers.* Int. J. Secur. Netw., vol. 10, no. 3, page 137–150, sep 2015. (Cited on page 41.)

[Aziz *et al.* 2021] Haris Aziz, Ágnes Cseh, John P. Dickerson and Duncan C. McElfresh. *Optimal Kidney Exchange with Immunosuppressants.* In Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021, pages 21–29. AAAI Press, 2021. (Cited on page 1.)

[Bach *et al.* 2015] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller and Wojciech Samek. *On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation.* PloS one, vol. 10, no. 7, page e0130140, 2015. (Cited on page 125.)

[Bagdasaryan *et al.* 2019] Eugene Bagdasaryan, Omid Poursaeed and Vitaly Shmatikov. *Differential Privacy Has Disparate Impact on Model Accuracy.* In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox and Roman Garnett, editors, Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, pages 15453–15462, 2019. (Cited on pages 87 and 90.)

[Balagopalan *et al.* 2022] Aparna Balagopalan, Haoran Zhang, Kimia Hamidieh, Thomas Hartvigsen, Frank Rudzicz and Marzyeh Ghassemi. *The Road to Explainability is Paved with Bias: Measuring the Fairness of Explanations.* In FAccT '22: 2022 ACM Conference on Fairness, Accountability, and Transparency, Seoul, Republic of Korea, June 21 - 24, 2022, pages 1194–1206. ACM, 2022. (Cited on page 55.)

[Banisar 2011] David Banisar. *The right to information and privacy: balancing rights and managing conflicts.* World Bank Institute Governance Working Paper, 2011. (Cited on page 122.)

[Barocas & Selbst 2016] Solon Barocas and Andrew D. Selbst. *Big Data's Disparate Impact.* California Law Review, vol. 104, no. 3, pages 671–732, 2016. (Cited on pages 9, 16, 21, 43, 86 and 97.)

[Barocas *et al.* 2019] Solon Barocas, Moritz Hardt and Arvind Narayanan. Fairness and machine learning. fairmlbook.org, 2019. http://www.fairmlbook.org. (Cited on pages 1, 9, 22, 25, 26 and 86.)

[Begley *et al.* 2020] Tom Begley, Tobias Schwedes, Christopher Frye and Ilya Feige. *Explainability for fair machine learning.* arXiv preprint arXiv:2010.07389, 2020. (Cited on page 52.)

[Bell *et al.* 2022] Andrew Bell, Ian Solano-Kamaiko, Oded Nov and Julia Stoyanovich. *It's Just Not That Simple: An Empirical Study of the Accuracy-Explainability Trade-off in Machine Learning for Public Policy.* In FAccT '22: 2022 ACM Conference on Fairness, Accountability, and Transparency, Seoul, Republic of Korea, June 21 - 24, 2022, pages 248–266. ACM, 2022. (Cited on page 35.)

[Bellamy *et al.* 2019] Rachel K. E. Bellamy, Kuntal Dey, Michael Hind, Samuel C. Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilovic, Seema Nagar, Karthikeyan Natesan Ramamurthy, John T. Richards, Diptikalyan Saha, Prasanna Sattigeri, Moninder Singh, Kush R. Varshney and Yunfeng Zhang. *AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias.* IBM J. Res. Dev., vol. 63, no. 4/5, pages 4:1–4:15, 2019. (Cited on pages 11 and 20.)

[Ben-Tal *et al.* 2013] Aharon Ben-Tal, Dick Den Hertog, Anja De Waegenaere, Bertrand Melenberg and Gijs Rennen. *Robust solutions of optimization problems affected by uncertain probabilities.* Management Science, vol. 59, no. 2, pages 341–357, 2013. (Cited on page 73.)

[Berthet *et al.* 2020] Quentin Berthet, Mathieu Blondel, Olivier Teboul, Marco Cuturi, Jean-Philippe Vert and Francis Bach. *Learning with differentiable pertubed optimizers.* Advances in neural information processing systems, vol. 33, pages 9508–9519, 2020. (Cited on pages 116 and 150.)

[Beutel *et al.* 2017] Alex Beutel, Jilin Chen, Zhe Zhao and Ed H. Chi. *Data Decisions and Theoretical Implications when Adversarially Learning Fair Representations.* CoRR, vol. abs/1707.00075, 2017. (Cited on page 22.)

[Binns 2020] Reuben Binns. *On the apparent conflict between individual and group fairness.* In FAT* 2020 - Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, pages 514–524. Association for Computing Machinery, Inc, jan 2020. (Cited on pages 23 and 24.)

[Bird *et al.* 2020] Sarah Bird, Miro Dudík, Richard Edgar, Brandon Horn, Roman Lutz, Vanessa Milan, Mehrnoosh Sameki, Hanna Wallach and Kathleen Walker. *Fairlearn: A toolkit for assessing and improving fairness in AI.* Technical report MSR-TR-2020-32, Microsoft, May 2020. (Cited on pages 11, 107 and 169.)

[Breiman *et al.* 1984] Leo Breiman, J. H. Friedman, R. A. Olshen and C. J. Stone. Classification and regression trees. Wadsworth, 1984. (Cited on pages 12 and 139.)

[Calmon *et al.* 2017] Flávio P. Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy and Kush R. Varshney. *Optimized Pre-Processing for Discrimination Prevention.* In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan and Roman Garnett, editors, Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, pages 3992–4001, 2017. (Cited on page 21.)

[Carlini *et al.* 2019] Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos and Dawn Song. *The Secret Sharer: Evaluating and Testing Unintended Memorization in Neural Networks.* In Nadia Heninger and Patrick Traynor, editors, 28th USENIX Security Symposium, USENIX Security 2019, Santa Clara, CA, USA, August 14-16, 2019, pages 267–284. USENIX Association, 2019. (Cited on page 42.)

[Carlini *et al.* 2022] Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis and Florian Tramèr. *Membership Inference Attacks From First Principles.* In 43rd IEEE Symposium on Security and Privacy, SP 2022, San Francisco, CA, USA, May 22-26, 2022, pages 1897–1914. IEEE, 2022. (Cited on page 40.)

[Caton & Haas 2023] Simon Caton and Christian Haas. *Fairness in Machine Learning: A Survey.* ACM Comput. Surv., aug 2023. (Cited on pages 20, 25, 26 and 86.)

[Chang & Shokri 2021] Hongyan Chang and Reza Shokri. *On the Privacy Risks of Algorithmic Fairness.* In IEEE European Symposium on Security and Privacy, EuroS&P 2021, Vienna, Austria, September 6-10, 2021, pages 292–303. IEEE, 2021. (Cited on pages 86, 87 and 89.)

[Chen *et al.* 2019] Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin and Jonathan Su. *This Looks Like That: Deep Learning for Interpretable Image Recognition.* In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox and Roman Garnett, editors, Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, pages 8928–8939, 2019. (Cited on page 35.)

[Chhabra *et al.* 2021] Anshuman Chhabra, Karina Masalkovaite and Prasant Mohapatra. *An Overview of Fairness in Clustering.* IEEE Access, vol. 9, pages 130698–130720, 2021. (Cited on page 17.)

[Chiappa & Isaac 2018] Silvia Chiappa and William S Isaac. *A causal bayesian networks viewpoint on fairness.* In IFIP International Summer School on

Privacy and Identity Management, pages 3–20. Springer, 2018. (Cited on page 20.)

[Chierichetti *et al.* 2017] Flavio Chierichetti, Ravi Kumar, Silvio Lattanzi and Sergei Vassilvitskii. *Fair Clustering Through Fairlets*. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan and Roman Garnett, editors, Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, pages 5029–5037, 2017. (Cited on page 17.)

[Chikalov *et al.* 2013] Igor Chikalov, Vadim Lozin, Irina Lozina, Mikhail Moshkov, Hung Son Nguyen, Andrzej Skowron and Beata Zielosko. Logical analysis of data: Theory, methodology and applications, pages 147–192. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. (Cited on pages 33 and 58.)

[Chouldechova 2017] Alexandra Chouldechova. *Fair prediction with disparate impact: A study of bias in recidivism prediction instruments*. Big data, vol. 5, no. 2, pages 153–163, 2017. (Cited on page 17.)

[Chuang & Mroueh 2021] Ching-Yao Chuang and Youssef Mroueh. *Fair Mixup: Fairness via Interpolation*. In 9th International Conference on Learning Representations, ICLR, 2021. (Cited on pages 26, 28 and 51.)

[Clifton & Tassa 2013] Chris Clifton and Tamir Tassa. *On Syntactic Anonymity and Differential Privacy*. Trans. Data Priv., vol. 6, no. 2, pages 161–183, 2013. (Cited on pages 39, 46 and 47.)

[Cohen & Nissim 2020] Aloni Cohen and Kobbi Nissim. *Linear Program Reconstruction in Practice*. J. Priv. Confidentiality, vol. 10, no. 1, 2020. (Cited on page 42.)

[Cormode 2010] Graham Cormode. *Individual Privacy vs Population Privacy: Learning to Attack Anonymization*. CoRR, vol. abs/1011.2511, 2010. (Cited on page 113.)

[Coston *et al.* 2021] Amanda Coston, Ashesh Rambachan and Alexandra Chouldechova. *Characterizing Fairness Over the Set of Good Models Under Selective Labels*. In Marina Meila and Tong Zhang, editors, Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event, volume 139 of *Proceedings of Machine Learning Research*, pages 2144–2155. PMLR, 2021. (Cited on page 150.)

[Cotter *et al.* 2018] Andrew Cotter, Maya Gupta, Heinrich Jiang, Nathan Srebro, Karthik Sridharan, Serena Wang, Blake Woodworth and Seungil You. *Training Fairness-Constrained Classifiers to Generalize*. FATML, 2018. (Cited on pages 26, 27, 51, 74 and 81.)

[Cotter *et al.* 2019] Andrew Cotter, Maya Gupta, Heinrich Jiang, Nathan Srebro, Karthik Sridharan, Serena Wang, Blake Woodworth and Seungil You. *Training well-generalizing classifiers for fairness metrics and other data-dependent constraints.* In Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA, volume 97 of *Proceedings of Machine Learning Research*, pages 1397–1405. PMLR, 2019. (Cited on pages 15, 26, 27, 51, 74 and 81.)

[Cristofaro 2020] Emiliano De Cristofaro. *An Overview of Privacy in Machine Learning.* CoRR, vol. abs/2005.08679, 2020. (Cited on pages 1, 38, 40, 41, 86, 113 and 122.)

[Cummings *et al.* 2019] Rachel Cummings, Varun Gupta, Dhamma Kimpara and Jamie Morgenstern. *On the Compatibility of Privacy and Fairness.* In Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization, UMAP'19 Adjunct, page 309–315, New York, NY, USA, 2019. Association for Computing Machinery. (Cited on pages 87, 88, 92 and 113.)

[Dai *et al.* 2021] Jessica Dai, Sohini Upadhyay, Stephen H. Bach and Himabindu Lakkaraju. *What will it take to generate fairness-preserving explanations?* CoRR, vol. abs/2106.13346, 2021. (Cited on page 54.)

[Dai *et al.* 2022] Jessica Dai, Sohini Upadhyay, Ulrich Aïvodji, Stephen H. Bach and Himabindu Lakkaraju. *Fairness via Explanation Quality: Evaluating Disparities in the Quality of Post hoc Explanations.* In Vincent Conitzer, John Tasioulas, Matthias Scheutz, Ryan Calo, Martina Mara and Annette Zimmermann, editors, AIES '22: AAAI/ACM Conference on AI, Ethics, and Society, Oxford, United Kingdom, May 19 - 21, 2021, pages 203–214. ACM, 2022. (Cited on page 54.)

[Dash *et al.* 2018] Sanjeeb Dash, Oktay Günlük and Dennis Wei. *Boolean Decision Rules via Column Generation.* In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi and Roman Garnett, editors, Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada, pages 4660–4670, 2018. (Cited on page 58.)

[Datta *et al.* 2016] Anupam Datta, Shayak Sen and Yair Zick. *Algorithmic Transparency via Quantitative Input Influence: Theory and Experiments with Learning Systems.* In IEEE Symposium on Security and Privacy, SP 2016, San Jose, CA, USA, May 22-26, 2016, pages 598–617. IEEE Computer Society, 2016. (Cited on pages 91 and 120.)

[Datta *et al.* 2023] Teresa Datta, Daniel Nissani, Max Cembalest, Akash Khanna, Haley Massa and John P Dickerson. *Position: Tensions Between the Proxies*

*of Human Values in AI.* In First IEEE Conference on Secure and Trustworthy Machine Learning, 2023. (Cited on pages 2, 11, 24, 47 and 122.)

[Defrance & Bie 2023] MaryBeth Defrance and Tijl De Bie. *Maximal fairness.* In Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency, FAccT 2023, Chicago, IL, USA, June 12-15, 2023, pages 851–880. ACM, 2023. (Cited on page 24.)

[Delaunay *et al.* 2022] Julien Delaunay, Luis Galárraga and Christine Largouët. *When Should We Use Linear Explanations?* In Mohammad Al Hasan and Li Xiong, editors, Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022, pages 355–364. ACM, 2022. (Cited on page 34.)

[Dimanov *et al.* 2020] Botty Dimanov, Umang Bhatt, Mateja Jamnik and Adrian Weller. *You Shouldn't Trust Me: Learning Models Which Conceal Unfairness from Multiple Explanation Methods.* In Giuseppe De Giacomo, Alejandro Catalá, Bistra Dilkina, Michela Milano, Senén Barro, Alberto Bugarín and Jérôme Lang, editors, ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, August 29 - September 8, 2020 - Including 10th Conference on Prestigious Applications of Artificial Intelligence (PAIS 2020), volume 325 of *Frontiers in Artificial Intelligence and Applications*, pages 2473–2480. IOS Press, 2020. (Cited on page 56.)

[Ding *et al.* 2020] Jiahao Ding, Xinyue Zhang, Xiaohuan Li, Junyi Wang, Rong Yu and Miao Pan. *Differentially Private and Fair Classification via Calibrated Functional Mechanism.* In The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020, pages 622–629. AAAI Press, 2020. (Cited on page 93.)

[Ding *et al.* 2021] Frances Ding, Moritz Hardt, John Miller and Ludwig Schmidt. *Retiring Adult: New Datasets for Fair Machine Learning.* In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang and Jennifer Wortman Vaughan, editors, Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual, pages 6478–6490, 2021. (Cited on pages 9, 37, 86, 106 and 107.)

[Dinur & Nissim 2003] Irit Dinur and Kobbi Nissim. *Revealing information while preserving privacy.* In Frank Neven, Catriel Beeri and Tova Milo, editors, Proceedings of the Twenty-Second ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, June 9-12, 2003, San Diego, CA, USA, pages 202–210. ACM, 2003. (Cited on pages 38, 40, 41 and 105.)

[Dodge *et al.* 2019] Jonathan Dodge, Q Vera Liao, Yunfeng Zhang, Rachel KE Bellamy and Casey Dugan. *Explaining models: an empirical study of how explanations impact fairness judgment.* In Proceedings of the 24th international conference on intelligent user interfaces, pages 275–285, 2019. (Cited on page 54.)

[Domingo-Ferrer *et al.* 2021] Josep Domingo-Ferrer, David Sánchez and Alberto Blanco-Justicia. *The limits of differential privacy (and its misuse in data release and machine learning).* Commun. ACM, vol. 64, no. 7, pages 33–35, 2021. (Cited on page 46.)

[Doshi-Velez & Kim 2017] Finale Doshi-Velez and Been Kim. *Towards a rigorous science of interpretable machine learning.* arXiv preprint arXiv:1702.08608, 2017. (Cited on pages 29, 30, 31, 52 and 119.)

[Du & Wu 2021] Wei Du and Xintao Wu. *Fair and Robust Classification Under Sample Selection Bias.* In Proceedings of the 30th ACM International Conference on Information & Knowledge Management, CIKM '21, page 2999–3003, New York, NY, USA, 2021. Association for Computing Machinery. (Cited on page 28.)

[Dua & Graff 2017] Dheeru Dua and Casey Graff. *UCI Machine Learning Repository*, 2017. (Cited on pages 67, 71, 106 and 139.)

[Duchi *et al.* 2013] John C. Duchi, Michael I. Jordan and Martin J. Wainwright. *Local privacy and statistical minimax rates.* In 51st Annual Allerton Conference on Communication, Control, and Computing, Allerton 2013, Allerton Park & Retreat Center, Monticello, IL, USA, October 2-4, 2013, page 1592. IEEE, 2013. (Cited on page 46.)

[Duchi *et al.* 2020] John C. Duchi, Tatsunori Hashimoto and Hongseok Namkoong. *Distributionally Robust Losses for Latent Covariate Mixtures.* arXiv preprint arXiv:2007.13982, 2020. (Cited on page 27.)

[Duchi *et al.* 2021] John C Duchi, Peter W Glynn and Hongseok Namkoong. *Statistics of robust optimization: A generalized empirical likelihood approach.* Mathematics of Operations Research, vol. 46, no. 3, pages 946–969, 2021. (Cited on page 73.)

[Duddu & Boutet 2022] Vasisht Duddu and Antoine Boutet. *Inferring Sensitive Attributes from Model Explanations.* In Mohammad Al Hasan and Li Xiong, editors, Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022, pages 416–425. ACM, 2022. (Cited on pages 43 and 125.)

[Dwork & Roth 2014] Cynthia Dwork and Aaron Roth. *The Algorithmic Foundations of Differential Privacy.* Found. Trends Theor. Comput. Sci., vol. 9, no. 3–4, page 211–407, aug 2014. (Cited on pages 43, 44, 112 and 145.)

[Dwork *et al.* 2006] Cynthia Dwork, Frank McSherry, Kobbi Nissim and Adam D. Smith. *Calibrating Noise to Sensitivity in Private Data Analysis.* In Shai Halevi and Tal Rabin, editors, Theory of Cryptography, Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006, Proceedings, volume 3876 of *Lecture Notes in Computer Science*, pages 265–284. Springer, 2006. (Cited on pages 40, 43, 44, 112 and 145.)

[Dwork *et al.* 2007] Cynthia Dwork, Frank McSherry and Kunal Talwar. *The Price of Privacy and the Limits of LP Decoding.* In Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing, STOC '07, page 85–94, New York, NY, USA, 2007. Association for Computing Machinery. (Cited on pages 41 and 105.)

[Dwork *et al.* 2012] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold and Richard Zemel. *Fairness through Awareness.* In Proceedings of the 3rd Innovations in Theoretical Computer Science Conference, ITCS '12, page 214–226, New York, NY, USA, 2012. Association for Computing Machinery. (Cited on pages 17, 19, 23, 24 and 95.)

[Dwork *et al.* 2017] Cynthia Dwork, Adam Smith, Thomas Steinke and Jonathan Ullman. *Exposed! A Survey of Attacks on Private Data.* Annual Review of Statistics and Its Application, vol. 4, no. 1, pages 61–84, 2017. (Cited on pages 40, 41, 86, 105 and 122.)

[Dziugaite *et al.* 2020] Gintare Karolina Dziugaite, Shai Ben-David and Daniel M Roy. *Enforcing Interpretability and its Statistical Impacts: Trade-offs between Accuracy and Interpretability.* arXiv preprint arXiv:2010.13764, 2020. (Cited on pages 30, 35 and 53.)

[EEOC. 1979] The U.S. EEOC. *Uniform guidelines on employee selection procedures.* March 2, 1979. (Cited on pages 87 and 113.)

[Ekstrand *et al.* 2018] Michael D Ekstrand, Rezvan Joshaghani and Hoda Mehrpouyan. *Privacy for all: Ensuring fair and equitable privacy protections.* In Conference on Fairness, Accountability and Transparency, pages 35–47. PMLR, 2018. (Cited on pages 16, 38, 86 and 89.)

[Elmachtoub & Grigas 2022] Adam N Elmachtoub and Paul Grigas. *Smart "predict, then optimize".* Management Science, vol. 68, no. 1, pages 9–26, 2022. (Cited on pages 116 and 150.)

[Farrand *et al.* 2020] Tom Farrand, Fatemehsadat Mireshghallah, Sahib Singh and Andrew Trask. *Neither Private Nor Fair: Impact of Data Imbalance on Utility and Fairness in Differential Privacy.* In Benyu Zhang, Raluca Ada Popa, Matei Zaharia, Guofei Gu and Shouling Ji, editors, PPMLP'20: Proceedings of the 2020 Workshop on Privacy-Preserving Machine Learning in

Practice, Virtual Event, USA, November, 2020, pages 15–19. ACM, 2020. (Cited on page 90.)

[Feldman *et al.* 2015] Michael Feldman, Sorelle A. Friedler, John Moeller, Carlos Scheidegger and Suresh Venkatasubramanian. *Certifying and Removing Disparate Impact.* In Longbing Cao, Chengqi Zhang, Thorsten Joachims, Geoffrey I. Webb, Dragos D. Margineantu and Graham Williams, editors, Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13, 2015, pages 259–268. ACM, 2015. (Cited on pages 20, 87 and 113.)

[Ferry *et al.* 2023a] Julien Ferry, Ulrich Aïvodji, Sébastien Gambs, Marie-José Huguet and Mohamed Siala. *Exploiting Fairness to Enhance Sensitive Attributes Reconstruction.* In First IEEE Conference on Secure and Trustworthy Machine Learning, 2023. (Cited on pages 3, 108 and 115.)

[Ferry *et al.* 2023b] Julien Ferry, Ulrich Aivodji, Sébastien Gambs, Marie-José Huguet and Mohamed Siala. *Improving fairness generalization through a sample-robust optimization method.* Machine Learning, vol. 112, no. 6, pages 2131–2192, 2023. (Cited on pages 3 and 165.)

[Ferry *et al.* 2023c] Julien Ferry, Ulrich Aïvodji, Sébastien Gambs, Marie-José Huguet and Mohamed Siala. *Probabilistic Dataset Reconstruction from Interpretable Models.* CoRR, vol. abs/2308.15099, 2023. (Cited on page 4.)

[Ferry *et al.* 2023d] Julien Ferry, Gabriel Laberge and Ulrich Aïvodji. *Learning Hybrid Interpretable Models: Theory, Taxonomy, and Methods.* arXiv preprint arXiv:2303.04437, 2023. (Cited on pages 3, 36 and 37.)

[Fioretto *et al.* 2022] Ferdinando Fioretto, Cuong Tran, Pascal Van Hentenryck and Keyu Zhu. *Differential Privacy and Fairness in Decisions and Learning Tasks: A Survey.* In Luc De Raedt, editor, Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022, pages 5470–5477. ijcai.org, 2022. (Cited on pages 87, 88, 90 and 91.)

[Fiscella & Fremont 2006] Kevin Fiscella and Allen M Fremont. *Use of geocoding and surname analysis to estimate race and ethnicity.* Health services research, vol. 41, no. 4p1, pages 1482–1500, 2006. (Cited on page 24.)

[Fletcher & Islam 2019] Sam Fletcher and Md Zahidul Islam. *Decision Tree Classification with Differential Privacy: A Survey.* ACM Comput. Surv., vol. 52, no. 4, pages 83:1–83:33, 2019. (Cited on page 120.)

[Fredrikson *et al.* 2014] Matthew Fredrikson, Eric Lantz, Somesh Jha, Simon M. Lin, David Page and Thomas Ristenpart. *Privacy in Pharmacogenetics: An End-to-End Case Study of Personalized Warfarin Dosing.* In Kevin Fu

and Jaeyeon Jung, editors, Proceedings of the 23rd USENIX Security Symposium, San Diego, CA, USA, August 20-22, 2014, pages 17–32. USENIX Association, 2014. (Cited on page 42.)

[Fredrikson *et al.* 2015] Matt Fredrikson, Somesh Jha and Thomas Ristenpart. *Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures.* In Indrajit Ray, Ninghui Li and Christopher Kruegel, editors, Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, CO, USA, October 12-16, 2015, pages 1322–1333. ACM, 2015. (Cited on pages 41 and 42.)

[Freitas 2014] Alex A. Freitas. *Comprehensible Classification Models: A Position Paper.* SIGKDD Explor. Newsl., vol. 15, no. 1, page 1–10, March 2014. (Cited on pages 29, 30 and 33.)

[Friedler *et al.* 2016] Sorelle A Friedler, Carlos Scheidegger and Suresh Venkatasubramanian. *On the (im) possibility of fairness.* arXiv preprint arXiv:1609.07236, 2016. (Cited on page 22.)

[Friedler *et al.* 2019] Sorelle A. Friedler, Carlos Scheidegger, Suresh Venkatasubramanian, Sonam Choudhary, Evan P. Hamilton and Derek Roth. *A comparative study of fairness-enhancing interventions in machine learning.* In danah boyd and Jamie H. Morgenstern, editors, Proceedings of the Conference on Fairness, Accountability, and Transparency, FAT* 2019, Atlanta, GA, USA, January 29-31, 2019, pages 329–338. ACM, 2019. (Cited on pages 20 and 22.)

[Friedman & Schuster 2010] Arik Friedman and Assaf Schuster. *Data mining with differential privacy.* In Bharat Rao, Balaji Krishnapuram, Andrew Tomkins and Qiang Yang, editors, Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, July 25-28, 2010, pages 493–502. ACM, 2010. (Cited on pages 45, 119 and 145.)

[gabriel laberge *et al.* 2023] gabriel laberge, Ulrich Aïvodji, Satoshi Hara, Mario Marchand and Foutse Khomh. *Fooling SHAP with Stealthily Biased Sampling.* In The Eleventh International Conference on Learning Representations, 2023. (Cited on page 56.)

[Gadotti *et al.* 2019] Andrea Gadotti, Florimond Houssiau, Luc Rocher, Benjamin Livshits and Yves-Alexandre de Montjoye. *When the Signal is in the Noise: Exploiting Diffix's Sticky Noise.* In Nadia Heninger and Patrick Traynor, editors, 28th USENIX Security Symposium, USENIX Security 2019, Santa Clara, CA, USA, August 14-16, 2019, pages 1081–1098. USENIX Association, 2019. (Cited on pages 42 and 105.)

[Gambs *et al.* 2012] Sébastien Gambs, Ahmed Gmati and Michel Hurfin. *Reconstruction Attack through Classifier Analysis.* In Nora Cuppens-Boulahia,

Frédéric Cuppens and Joaquín García-Alfaro, editors, Data and Applications Security and Privacy XXVI - 26th Annual IFIP WG 11.3 Conference, DBSec 2012, Paris, France, July 11-13,2012. Proceedings, volume 7371 of *Lecture Notes in Computer Science*, pages 274–281. Springer, 2012. (Cited on pages 43, 118, 126, 127, 128, 129, 135 and 145.)

[Garcia *et al.* 2018] Washington Garcia, Joseph I. Choi, Suman Kalyan Adari, Somesh Jha and Kevin R. B. Butler. *Explainable Black-Box Attacks Against Model-based Authentication.* CoRR, vol. abs/1810.00024, 2018. (Cited on page 122.)

[Garfinkel *et al.* 2018] Simson Garfinkel, John M. Abowd and Christian Martindale. *Understanding Database Reconstruction Attacks on Public Data: These Attacks on Statistical Databases Are No Longer a Theoretical Danger.* Queue, vol. 16, no. 5, page 28–53, oct 2018. (Cited on page 42.)

[Ghorbani *et al.* 2019] Amirata Ghorbani, Abubakar Abid and James Y. Zou. *Interpretation of Neural Networks Is Fragile.* In The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019, pages 3681–3688. AAAI Press, 2019. (Cited on page 34.)

[Gong *et al.* 2020] Maoguo Gong, Yu Xie, Ke Pan, Kaiyuan Feng and Alex Kai Qin. *A Survey on Differentially Private Machine Learning [Review Article].* IEEE Comput. Intell. Mag., vol. 15, no. 2, pages 49–64, 2020. (Cited on pages 45 and 119.)

[Guidotti *et al.* 2018] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti and Dino Pedreschi. *A survey of methods for explaining black box models.* ACM computing surveys (CSUR), vol. 51, no. 5, pages 1–42, 2018. (Cited on pages 2, 29, 30, 31, 33, 35 and 50.)

[Gupta *et al.* 2019] Vivek Gupta, Pegah Nokhiz, Chitradeep Dutta Roy and Suresh Venkatasubramanian. *Equalizing Recourse across Groups.* CoRR, vol. abs/1909.03166, 2019. (Cited on page 55.)

[Haimes 1971] Yacov Haimes. *On a bicriterion formulation of the problems of integrated system identification and system optimization.* IEEE transactions on systems, man, and cybernetics, no. 3, pages 296–297, 1971. (Cited on pages 15 and 59.)

[Hajian & Domingo-Ferrer 2013] Sara Hajian and Josep Domingo-Ferrer. *A Methodology for Direct and Indirect Discrimination Prevention in Data Mining.* IEEE Trans. Knowl. Data Eng., vol. 25, no. 7, pages 1445–1459, 2013. (Cited on pages 16 and 17.)

[Hajian *et al.* 2015] Sara Hajian, Josep Domingo-Ferrer, Anna Monreale, Dino Pedreschi and Fosca Giannotti. *Discrimination- and privacy-aware patterns.* Data Min. Knowl. Discov., vol. 29, no. 6, pages 1733–1782, 2015. (Cited on page 96.)

[Hamman *et al.* 2022] Faisal Hamman, Jiahao Chen and Sanghamitra Dutta. *Can Querying for Bias Leak Protected Attributes? Achieving Privacy With Smooth Sensitivity.* In NeurIPS 2022 Workshop on Algorithmic Fairness through the Lens of Causality and Privacy, 2022. (Cited on pages 42 and 43.)

[Harder *et al.* 2020] Frederik Harder, Matthias Bauer and Mijung Park. *Interpretable and Differentially Private Predictions.* In The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020, pages 4083–4090. AAAI Press, 2020. (Cited on page 120.)

[Hardt *et al.* 2016] Moritz Hardt, Eric Price, Eric Price and Nati Srebro. *Equality of Opportunity in Supervised Learning.* In Advances in Neural Information Processing Systems, volume 29. Curran Associates, Inc., 2016. (Cited on pages 17, 21, 94, 95, 107, 111 and 113.)

[Hebrard & Siala 2017a] Emmanuel Hebrard and Mohamed Siala. *Explanation-Based Weighted Degree.* In Domenico Salvagnin and Michele Lombardi, editors, Integration of AI and OR Techniques in Constraint Programming - 14th International Conference, CPAIOR 2017, Padua, Italy, June 5-8, 2017, Proceedings, volume 10335 of *Lecture Notes in Computer Science*, pages 167–175. Springer, 2017. (Cited on page 69.)

[Hebrard & Siala 2017b] Emmanuel Hebrard and Mohamed Siala. *Mistral Solver Engine*, 2017. (Cited on page 67.)

[Hebrard 2008] Emmanuel Hebrard. *Mistral, a constraint satisfaction library.* Proceedings of the Third International CSP Solver Competition, vol. 3, no. 3, pages 31–39, 2008. (Cited on page 67.)

[Heo *et al.* 2019] Juyeon Heo, Sunghwan Joo and Taesup Moon. *Fooling Neural Network Interpretations via Adversarial Model Manipulation.* In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox and Roman Garnett, editors, Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, pages 2921–2932, 2019. (Cited on page 56.)

[Herdagdelen *et al.* 2020] Amaç Herdagdelen, Alex Dow, Bogdan State, Payman Mohassel and Alex Pompe. *Protecting privacy in facebook mobility data during the covid- 19 response.* online, 2020. (Cited on page 40.)

[Hu & Lan 2020] Hui Hu and Chao Lan. *Inference attack and defense on the distributed private fair learning framework*. In The AAAI Workshop on Privacy-Preserving Artificial Intelligence, 2020. (Cited on pages 42, 43, 87 and 89.)

[Hu *et al.* 2020] Hao Hu, Mohamed Siala, Emmanuel Hebrard and Marie-José Huguet. *Learning Optimal Decision Trees with MaxSAT and its Integration in AdaBoost*. In Christian Bessiere, editor, Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020, pages 1170–1176. ijcai.org, 2020. (Cited on page 14.)

[Hu *et al.* 2022a] Hao Hu, Marie-José Huguet and Mohamed Siala. *Optimizing Binary Decision Diagrams with MaxSAT for Classification*. In Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022, pages 3767–3775. AAAI Press, 2022. (Cited on page 14.)

[Hu *et al.* 2022b] Hongsheng Hu, Zoran Salcic, Lichao Sun, Gillian Dobbie, Philip S. Yu and Xuyun Zhang. *Membership Inference Attacks on Machine Learning: A Survey*. ACM Comput. Surv., vol. 54, no. 11s, pages 235:1–235:37, 2022. (Cited on page 40.)

[Huang & Vishnoi 2019] Lingxiao Huang and Nisheeth K. Vishnoi. *Stable and Fair Classification*. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA, volume 97 of *Proceedings of Machine Learning Research*, pages 2879–2890. PMLR, 2019. (Cited on pages 26, 28, 51 and 74.)

[Ignatiev *et al.* 2020] Alexey Ignatiev, Martin C Cooper, Mohamed Siala, Emmanuel Hebrard and Joao Marques-Silva. *Towards Formal Fairness in Machine Learning*. In International Conference on Principles and Practice of Constraint Programming, pages 846–867. Springer, 2020. (Cited on pages 24 and 95.)

[Ilvento 2020] Christina Ilvento. *Metric Learning for Individual Fairness*. In Aaron Roth, editor, 1st Symposium on Foundations of Responsible Computing, FORC 2020, June 1-3, 2020, Harvard University, Cambridge, MA, USA (virtual conference), volume 156 of *LIPIcs*, pages 2:1–2:11. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. (Cited on page 19.)

[Iofinova *et al.* 2021] Eugenia Iofinova, Nikola Konstantinov and Christoph H Lampert. *FLEA: Provably Fair Multisource Learning from Unreliable Training Data*. arXiv preprint arXiv:2106.11732, 2021. (Cited on page 28.)

[Jabbari *et al.* 2020] Shahin Jabbari, Han-Ching Ou, Himabindu Lakkaraju and Milind Tambe. *An empirical study of the trade-offs between interpretability and fairness.* In ICML 2020 Workshop on Human Interpretability in Machine Learning, 2020. (Cited on page 53.)

[Jagielski *et al.* 2019] Matthew Jagielski, Michael J. Kearns, Jieming Mao, Alina Oprea, Aaron Roth, Saeed Sharifi-Malvajerdi and Jonathan R. Ullman. *Differentially Private Fair Learning.* In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA, volume 97 of *Proceedings of Machine Learning Research*, pages 3000–3008. PMLR, 2019. (Cited on pages 94 and 95.)

[Ji *et al.* 2014] Zhanglong Ji, Zachary Chase Lipton and Charles Elkan. *Differential Privacy and Machine Learning: a Survey and Review.* CoRR, vol. abs/1412.7584, 2014. (Cited on page 45.)

[Jiang & Nachum 2020] Heinrich Jiang and Ofir Nachum. *Identifying and correcting label bias in machine learning.* In International Conference on Artificial Intelligence and Statistics, pages 702–712. PMLR, 2020. (Cited on pages 20 and 22.)

[Jo *et al.* 2023] Nathanael Jo, Sina Aghaei, Jack Benson, Andres Gomez and Phebe Vayanos. *Learning Optimal Fair Decision Trees: Trade-offs Between Interpretability, Fairness, and Accuracy.* In Proceedings of the 2023 AAAI/ACM Conference on AI, Ethics, and Society, pages 181–192, 2023. (Cited on page 33.)

[Joseph *et al.* 2016] Matthew Joseph, Michael J. Kearns, Jamie Morgenstern and Aaron Roth. *Fairness in Learning: Classic and Contextual Bandits.* In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon and Roman Garnett, editors, Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain, pages 325–333, 2016. (Cited on page 19.)

[Jung *et al.* 2021] Christopher Jung, Michael Kearns, Seth Neel, Aaron Roth, Logan Stapleton and Zhiwei Steven Wu. *An Algorithmic Framework for Fairness Elicitation.* In Katrina Ligett and Swati Gupta, editors, 2nd Symposium on Foundations of Responsible Computing, FORC 2021, June 9-11, 2021, Virtual Conference, volume 192 of *LIPIcs*, pages 2:1–2:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021. (Cited on page 19.)

[Kairouz *et al.* 2015] Peter Kairouz, Sewoong Oh and Pramod Viswanath. *Secure Multi-party Differential Privacy.* In Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama and Roman Garnett, editors, Advances

in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada, pages 2008–2016, 2015. (Cited on page 39.)

[Kamiran & Calders 2012] Faisal Kamiran and Toon Calders. *Data preprocessing techniques for classification without discrimination.* Knowledge and Information Systems, vol. 33, no. 1, pages 1–33, 2012. (Cited on pages 20 and 54.)

[Kamiran *et al.* 2010] Faisal Kamiran, Toon Calders and Mykola Pechenizkiy. *Discrimination Aware Decision Tree Learning.* In Geoffrey I. Webb, Bing Liu, Chengqi Zhang, Dimitrios Gunopulos and Xindong Wu, editors, ICDM 2010, The 10th IEEE International Conference on Data Mining, Sydney, Australia, 14-17 December 2010, pages 869–874. IEEE Computer Society, 2010. (Cited on page 21.)

[Kamiran *et al.* 2012] Faisal Kamiran, Asim Karim and Xiangliang Zhang. *Decision Theory for Discrimination-Aware Classification.* In Mohammed Javeed Zaki, Arno Siebes, Jeffrey Xu Yu, Bart Goethals, Geoffrey I. Webb and Xindong Wu, editors, 12th IEEE International Conference on Data Mining, ICDM 2012, Brussels, Belgium, December 10-13, 2012, pages 924–929. IEEE Computer Society, 2012. (Cited on page 21.)

[Kamishima *et al.* 2012] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh and Jun Sakuma. *Fairness-Aware Classifier with Prejudice Remover Regularizer.* In Peter A. Flach, Tijl De Bie and Nello Cristianini, editors, Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2012, Bristol, UK, September 24-28, 2012. Proceedings, Part II, volume 7524 of *Lecture Notes in Computer Science*, pages 35–50. Springer, 2012. (Cited on page 21.)

[Kang 2017] Yang Kang. *Distributionally Robust Optimization and its Applications in Machine Learning.* PhD thesis, Columbia University, 2017. (Cited on page 74.)

[Karimi *et al.* 2023] Amir-Hossein Karimi, Gilles Barthe, Bernhard Schölkopf and Isabel Valera. *A Survey of Algorithmic Recourse: Contrastive Explanations and Consequential Recommendations.* ACM Comput. Surv., vol. 55, no. 5, pages 95:1–95:29, 2023. (Cited on page 55.)

[Kearns *et al.* 2018] Michael J. Kearns, Seth Neel, Aaron Roth and Zhiwei Steven Wu. *Preventing Fairness Gerrymandering: Auditing and Learning for Subgroup Fairness.* In Jennifer G. Dy and Andreas Krause, editors, Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018, volume 80 of *Proceedings of Machine Learning Research*, pages 2569–2577. PMLR, 2018. (Cited on page 167.)

[Khalili *et al.* 2021] Mohammad Mahdi Khalili, Xueru Zhang, Mahed Abroshan and Somayeh Sojoudi. *Improving Fairness and Privacy in Selection Problems.* In Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021, pages 8092–8100. AAAI Press, 2021. (Cited on page 96.)

[Khoshgoftaar *et al.* 2013] Taghi M Khoshgoftaar, Alireza Fazelpour, Huanjing Wang and Randall Wald. *A survey of stability analysis of feature subset selection techniques.* In 2013 IEEE 14th International Conference on Information Reuse & Integration (IRI), pages 424–431, 2013. (Cited on page 75.)

[Kilbertus *et al.* 2017] Niki Kilbertus, Mateo Rojas-Carulla, Giambattista Parascandolo, Moritz Hardt, Dominik Janzing and Bernhard Schölkopf. *Avoiding Discrimination through Causal Reasoning.* In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan and Roman Garnett, editors, Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, pages 656–666, 2017. (Cited on page 19.)

[Kilbertus *et al.* 2018] Niki Kilbertus, Adrià Gascón, Matt J. Kusner, Michael Veale, Krishna P. Gummadi and Adrian Weller. *Blind Justice: Fairness with Encrypted Sensitive Attributes.* In Jennifer G. Dy and Andreas Krause, editors, Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018, volume 80 of *Proceedings of Machine Learning Research*, pages 2635–2644. PMLR, 2018. (Cited on pages 16, 52 and 88.)

[Kleinberg & Mullainathan 2019] Jon Kleinberg and Sendhil Mullainathan. *Simplicity creates inequity: implications for fairness, stereotypes, and interpretability.* In Proceedings of the 2019 ACM Conference on Economics and Computation, pages 807–808, 2019. (Cited on pages 52 and 53.)

[Koch & Soll 2023] Korbinian Koch and Marcus Soll. *No Matter How You Slice It: Machine Unlearning with SISA Comes at the Expense of Minority Classes.* In First IEEE Conference on Secure and Trustworthy Machine Learning, 2023. (Cited on page 91.)

[Koh & Liang 2017] Pang Wei Koh and Percy Liang. *Understanding black-box predictions via influence functions.* In International conference on machine learning, pages 1885–1894. PMLR, 2017. (Cited on pages 32 and 124.)

[Komer *et al.* 2014] Brent Komer, James Bergstra and Chris Eliasmith. *Hyperopt-sklearn: automatic hyperparameter configuration for scikit-learn.* In ICML

workshop on AutoML, volume 9, page 50. Citeseer, 2014. (Cited on page 108.)

[Kosub 2019] Sven Kosub. *A note on the triangle inequality for the Jaccard distance.* Pattern Recognition Letters, vol. 120, pages 36–38, 2019. (Cited on page 75.)

[Krco *et al.* 2023] Natasa Krco, Thibault Laugel, Jean-Michel Loubes and Marcin Detyniecki. *When Mitigating Bias is Unfair: A Comprehensive Study on the Impact of Bias Mitigation Algorithms.* CoRR, vol. abs/2302.07185, 2023. (Cited on page 24.)

[Kuhn 1955] Harold W Kuhn. *The Hungarian method for the assignment problem.* Naval research logistics quarterly, vol. 2, no. 1-2, pages 83–97, 1955. (Cited on page 129.)

[Kulynych *et al.* 2022] Bogdan Kulynych, Mohammad Yaghini, Giovanni Cherubin, Michael Veale and Carmela Troncoso. *Disparate Vulnerability to Membership Inference Attacks.* Proc. Priv. Enhancing Technol., vol. 2022, no. 1, pages 460–480, 2022. (Cited on pages 40, 41 and 89.)

[Kumar *et al.* 2020] I Elizabeth Kumar, Suresh Venkatasubramanian, Carlos Scheidegger and Sorelle Friedler. *Problems with Shapley-value-based explanations as feature importance measures.* In International Conference on Machine Learning, pages 5491–5500. PMLR, 2020. (Cited on page 124.)

[Kuppa & Le-Khac 2021] Aditya Kuppa and Nhien-An Le-Khac. *Adversarial XAI Methods in Cybersecurity.* IEEE Trans. Inf. Forensics Secur., vol. 16, pages 4924–4938, 2021. (Cited on pages 122, 123 and 124.)

[Lahoti *et al.* 2019] Preethi Lahoti, Krishna P Gummadi and Gerhard Weikum. *ifair: Learning individually fair data representations for algorithmic decision making.* In 2019 IEEE 35th International Conference on Data Engineering (ICDE), pages 1334–1345. IEEE, 2019. (Cited on pages 19 and 23.)

[Lakkaraju & Bastani 2020] Himabindu Lakkaraju and Osbert Bastani. *"How Do I Fool You?": Manipulating User Trust via Misleading Black Box Explanations.* In Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society, AIES '20, page 79–85, New York, NY, USA, 2020. Association for Computing Machinery. (Cited on pages 34 and 55.)

[Lakkaraju *et al.* 2016] Himabindu Lakkaraju, Stephen H Bach and Jure Leskovec. *Interpretable decision sets: A joint framework for description and prediction.* In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pages 1675–1684, 2016. (Cited on page 33.)

[Lakkaraju *et al.* 2019] Himabindu Lakkaraju, Ece Kamar, Rich Caruana and Jure Leskovec. *Faithful and Customizable Explanations of Black Box Models.* In

Vincent Conitzer, Gillian K. Hadfield and Shannon Vallor, editors, Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, AIES 2019, Honolulu, HI, USA, January 27-28, 2019, pages 131–138. ACM, 2019. (Cited on page 55.)

[Lakkaraju *et al.* 2020] Himabindu Lakkaraju, Nino Arsov and Osbert Bastani. *Robust and Stable Black Box Explanations.* In Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event, volume 119 of *Proceedings of Machine Learning Research*, pages 5628–5638. PMLR, 2020. (Cited on page 34.)

[Li *et al.* 2007] Ninghui Li, Tiancheng Li and Suresh Venkatasubramanian. *t-Closeness: Privacy Beyond k-Anonymity and l-Diversity.* In Rada Chirkova, Asuman Dogac, M. Tamer Özsu and Timos K. Sellis, editors, Proceedings of the 23rd International Conference on Data Engineering, ICDE 2007, The Marmara Hotel, Istanbul, Turkey, April 15-20, 2007, pages 106–115. IEEE Computer Society, 2007. (Cited on page 39.)

[Li *et al.* 2023] Zhe Li, Honglong Chen, Zhichen Ni and Huajie Shao. *Balancing Privacy Protection and Interpretability in Federated Learning.* CoRR, vol. abs/2302.08044, 2023. (Cited on page 121.)

[Lin *et al.* 2020] Jimmy Lin, Chudi Zhong, Diane Hu, Cynthia Rudin and Margo I. Seltzer. *Generalized and Scalable Optimal Sparse Decision Trees.* In Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event, volume 119 of *Proceedings of Machine Learning Research*, pages 6150–6160. PMLR, 2020. (Cited on pages 12 and 149.)

[Lipton 2018] Zachary C Lipton. *The mythos of model interpretability.* Queue, vol. 16, no. 3, pages 31–57, 2018. (Cited on pages 31, 33 and 50.)

[Liu *et al.* 2019] Lydia T. Liu, Sarah Dean, Esther Rolf, Max Simchowitz and Moritz Hardt. *Delayed Impact of Fair Machine Learning.* In Sarit Kraus, editor, Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019, pages 6196–6200. ijcai.org, 2019. (Cited on page 24.)

[Liu *et al.* 2021a] Evan Zheran Liu, Behzad Haghgoo, Annie S. Chen, Aditi Raghunathan, Pang Wei Koh, Shiori Sagawa, Percy Liang and Chelsea Finn. *Just Train Twice: Improving Group Robustness without Training Group Information.* In Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event, volume 139 of *Proceedings of Machine Learning Research*, pages 6781–6792. PMLR, 2021. (Cited on page 27.)

[Liu *et al.* 2021b] Hongbin Liu, Jinyuan Jia and Neil Zhenqiang Gong. *On the Intrinsic Differential Privacy of Bagging.* In Zhi-Hua Zhou, editor, Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021, pages 2730–2736. ijcai.org, 2021. (Cited on page 46.)

[Locatello *et al.* 2019] Francesco Locatello, Gabriele Abbati, Thomas Rainforth, Stefan Bauer, Bernhard Schölkopf and Olivier Bachem. *On the fairness of disentangled representations.* Advances in neural information processing systems, vol. 32, 2019. (Cited on page 150.)

[Lohia *et al.* 2019] Pranay K Lohia, Karthikeyan Natesan Ramamurthy, Manish Bhide, Diptikalyan Saha, Kush R Varshney and Ruchir Puri. *Bias mitigation post-processing for individual and group fairness.* In Icassp 2019-2019 ieee international conference on acoustics, speech and signal processing (icassp), pages 2847–2851. IEEE, 2019. (Cited on pages 21 and 23.)

[Long & Albert 2021] Kevin D Long and Steven M Albert. *Use of Zip Code Based Aggregate Indicators to Assess Race Disparities in COVID-19.* Ethnicity & Disease, vol. 31, no. 3, page 399, 2021. (Cited on page 24.)

[Luo *et al.* 2022] Xinjian Luo, Yangfan Jiang and Xiaokui Xiao. *Feature Inference Attack on Shapley Values.* In Heng Yin, Angelos Stavrou, Cas Cremers and Elaine Shi, editors, Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security, CCS 2022, Los Angeles, CA, USA, November 7-11, 2022, pages 2233–2247. ACM, 2022. (Cited on page 125.)

[Machanavajjhala *et al.* 2007] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke and Muthuramakrishnan Venkitasubramaniam. L-*diversity: Privacy beyond* k-*anonymity.* ACM Trans. Knowl. Discov. Data, vol. 1, no. 1, page 3, 2007. (Cited on page 39.)

[Makhlouf *et al.* 2021] Karima Makhlouf, Sami Zhioua and Catuscia Palamidessi. *On the Applicability of Machine Learning Fairness Notions.* SIGKDD Explor., vol. 23, no. 1, pages 14–23, 2021. (Cited on page 24.)

[Malgieri 2020] Gianclaudio Malgieri. *The concept of fairness in the GDPR: a linguistic and contextual interpretation.* In Mireille Hildebrandt, Carlos Castillo, L. Elisa Celis, Salvatore Ruggieri, Linnet Taylor and Gabriela Zanfir-Fortuna, editors, FAT* '20: Conference on Fairness, Accountability, and Transparency, Barcelona, Spain, January 27-30, 2020, pages 154–166. ACM, 2020. (Cited on page 16.)

[Mandal *et al.* 2020] Debmalya Mandal, Samuel Deng, Suman Jana, Jeannette Wing and Daniel J Hsu. *Ensuring Fairness Beyond the Training Data.* In Advances in Neural Information Processing Systems, volume 33, pages

18445–18456. Curran Associates, Inc., 2020. (Cited on pages 26, 27, 51, 74 and 75.)

[Manerba & Guidotti 2022] Marta Marchiori Manerba and Riccardo Guidotti. *Investigating Debiasing Effects on Classification and Explainability.* In Vincent Conitzer, John Tasioulas, Matthias Scheutz, Ryan Calo, Martina Mara and Annette Zimmermann, editors, AIES '22: AAAI/ACM Conference on AI, Ethics, and Society, Oxford, United Kingdom, May 19 - 21, 2021, pages 468–478. ACM, 2022. (Cited on page 54.)

[Mangold *et al.* 2023] Paul Mangold, Michaël Perrot, Aurélien Bellet and Marc Tommasi. *Differential Privacy has Bounded Impact on Fairness in Classification.* In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato and Jonathan Scarlett, editors, International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA, volume 202 of *Proceedings of Machine Learning Research*, pages 23681–23705. PMLR, 2023. (Cited on page 95.)

[Mata *et al.* 2022] Kota Mata, Kentaro Kanamori and Hiroki Arimura. *Computing the Collection of Good Models for Rule Lists.* CoRR, vol. abs/2204.11285, 2022. (Cited on page 150.)

[McSherry & Talwar 2007] Frank McSherry and Kunal Talwar. *Mechanism Design via Differential Privacy.* In 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2007), October 20-23, 2007, Providence, RI, USA, Proceedings, pages 94–103. IEEE Computer Society, 2007. (Cited on pages 44 and 95.)

[Mehrabi *et al.* 2022] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman and Aram Galstyan. *A Survey on Bias and Fairness in Machine Learning.* ACM Comput. Surv., vol. 54, no. 6, pages 115:1–115:35, 2022. (Cited on pages 16 and 86.)

[Merrer & Tredan 2019] Erwan Le Merrer and Gilles Tredan. *The Bouncer Problem: Challenges to Remote Explainability.* arXiv preprint arXiv:1910.01432, 2019. (Cited on page 56.)

[Milli *et al.* 2019] Smitha Milli, Ludwig Schmidt, Anca D Dragan and Moritz Hardt. *Model reconstruction from model explanations.* In Proceedings of the Conference on Fairness, Accountability, and Transparency, pages 1–9, 2019. (Cited on pages 123 and 126.)

[Miura *et al.* 2021] Takayuki Miura, Satoshi Hasegawa and Toshiki Shibahara. *MEGEX: Data-Free Model Extraction Attack against Gradient-Based Explainable AI.* CoRR, vol. abs/2107.08909, 2021. (Cited on page 123.)

[Mochaourab *et al.* 2021] Rami Mochaourab, Sugandh Sinha, Stanley Greenstein and Panagiotis Papapetrou. *Robust Explanations for Private Support Vector Machines*. CoRR, vol. abs/2102.03785, 2021. (Cited on page 121.)

[Molnar 2020] Christoph Molnar. Interpretable machine learning. Lulu. com, 2020. (Cited on page 31.)

[Mougan *et al.* 2023] Carlos Mougan, Laura State, Antonio Ferrara, Salvatore Ruggieri and Steffen Staab. *Demographic Parity Inspector: Fairness Audits via the Explanation Space*. CoRR, vol. abs/2303.08040, 2023. (Cited on page 52.)

[Mozannar *et al.* 2020] Hussein Mozannar, Mesrob Ohannessian and Nathan Srebro. *Fair learning with private demographic data*. In International Conference on Machine Learning, pages 7066–7075. PMLR, 2020. (Cited on page 95.)

[Munkres 1957] James Munkres. *Algorithms for the assignment and transportation problems*. Journal of the society for industrial and applied mathematics, vol. 5, no. 1, pages 32–38, 1957. (Cited on page 129.)

[Nabi & Shpitser 2018] Razieh Nabi and Ilya Shpitser. *Fair Inference on Outcomes*. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018, pages 1931–1940. AAAI Press, 2018. (Cited on page 19.)

[Naidu *et al.* 2021] Rakshit Naidu, Aman Priyanshu, Aadith Kumar, Sasikanth Kotti, Haofan Wang and Fatemehsadat Mireshghallah. *When Differential Privacy Meets Interpretability: A Case Study*. CoRR, vol. abs/2106.13203, 2021. (Cited on page 121.)

[Nam *et al.* 2020] Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee and Jinwoo Shin. *Learning from Failure: De-biasing Classifier from Biased Classifier*. In Advances in Neural Information Processing Systems, volume 33, pages 20673–20684. Curran Associates, Inc., 2020. (Cited on page 27.)

[Nguyen *et al.* 2023] Truc D. T. Nguyen, Phung Lai, Hai Phan and My T. Thai. *XRand: Differentially Private Defense against Explanation-Guided Attacks*. In Brian Williams, Yiling Chen and Jennifer Neville, editors, Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023, pages 11873–11881. AAAI Press, 2023. (Cited on page 121.)

[Øverby & Audestad 2021] Harald Øverby and Jan Arild Audestad. Big data economics, pages 305–322. Springer International Publishing, Cham, 2021. (Cited on page 1.)

[Pan *et al.* 2020] Danqing Pan, Tong Wang and Satoshi Hara. *Interpretable companions for black-box models.* In International conference on artificial intelligence and statistics, pages 2444–2454. PMLR, 2020. (Cited on pages 35 and 36.)

[Papernot *et al.* 2017] Nicolas Papernot, Martín Abadi, Úlfar Erlingsson, Ian J. Goodfellow and Kunal Talwar. *Semi-supervised Knowledge Transfer for Deep Learning from Private Training Data.* In 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings. OpenReview.net, 2017. (Cited on pages 45 and 90.)

[Papernot *et al.* 2018] Nicolas Papernot, Shuang Song, Ilya Mironov, Ananth Raghunathan, Kunal Talwar and Úlfar Erlingsson. *Scalable Private Learning with PATE.* In 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings. OpenReview.net, 2018. (Cited on pages 45 and 90.)

[Parmentier & Vidal 2021] Axel Parmentier and Thibaut Vidal. *Optimal Counterfactual Explanations in Tree Ensembles.* In Marina Meila and Tong Zhang, editors, Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event, volume 139 of *Proceedings of Machine Learning Research*, pages 8422–8431. PMLR, 2021. (Cited on page 32.)

[Patel *et al.* 2022] Neel Patel, Reza Shokri and Yair Zick. *Model Explanations with Differential Privacy.* In FAccT '22: 2022 ACM Conference on Fairness, Accountability, and Transparency, Seoul, Republic of Korea, June 21 - 24, 2022, pages 1895–1904. ACM, 2022. (Cited on pages 91 and 120.)

[Pedregosa *et al.* 2011] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and E. Duchesnay. *Scikit-learn: Machine Learning in Python.* Journal of Machine Learning Research, vol. 12, pages 2825–2830, 2011. (Cited on pages 107, 108, 128 and 139.)

[Pedreschi *et al.* 2008] Dino Pedreschi, Salvatore Ruggieri and Franco Turini. *Discrimination-aware data mining.* In Ying Li, Bing Liu and Sunita Sarawagi, editors, Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27, 2008, pages 560–568. ACM, 2008. (Cited on page 17.)

[Perron & Furnon 2019] Laurent Perron and Vincent Furnon. *OR-Tools*, 2019. (Cited on pages 14 and 165.)

[Phong *et al.* 2017] Le Trieu Phong, Yoshinori Aono, Takuya Hayashi, Lihua Wang and Shiho Moriai. *Privacy-Preserving Deep Learning: Revisited and Enhanced*. In Lynn Batten, Dong Seong Kim, Xuyun Zhang and Gang Li, editors, Applications and Techniques in Information Security - 8th International Conference, ATIS 2017, Auckland, New Zealand, July 6-7, 2017, Proceedings, volume 719 of *Communications in Computer and Information Science*, pages 100–110. Springer, 2017. (Cited on page 42.)

[Pleiss *et al.* 2017] Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg and Kilian Q Weinberger. *On fairness and calibration*. Advances in neural information processing systems, vol. 30, 2017. (Cited on pages 21 and 54.)

[Pruthi *et al.* 2020] Danish Pruthi, Mansi Gupta, Bhuwan Dhingra, Graham Neubig and Zachary C. Lipton. *Learning to Deceive with Attention-Based Explanations*. In Dan Jurafsky, Joyce Chai, Natalie Schluter and Joel R. Tetreault, editors, Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020, pages 4782–4793. Association for Computational Linguistics, 2020. (Cited on page 56.)

[Pujol *et al.* 2020] David Pujol, Ryan McKenna, Satya Kuppam, Michael Hay, Ashwin Machanavajjhala and Gerome Miklau. *Fair decision making using privacy-protected data*. In FAT* 2020 - Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, pages 189–199. Association for Computing Machinery, Inc, jan 2020. (Cited on pages 47 and 91.)

[Putzel & Lee 2022] Preston Putzel and Scott Lee. *Blackbox Post-Processing for Multiclass Fairness*. In Gabriel Pedroza, José Hernández-Orallo, Xin Cynthia Chen, Xiaowei Huang, Huáscar Espinoza, Mauricio Castillo-Effen, John A. McDermid, Richard Mallah and Seán Ó hÉigeartaigh, editors, Proceedings of the Workshop on Artificial Intelligence Safety 2022 (SafeAI 2022) co-located with the Thirty-Sixth AAAI Conference on Artificial Intelligence (AAAI2022), Virtual, February, 2022, volume 3087 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2022. (Cited on page 25.)

[Raff *et al.* 2018] Edward Raff, Jared Sylvester and Steven Mills. *Fair Forests: Regularized Tree Induction to Minimize Model Bias*. In Jason Furman, Gary E. Marchant, Huw Price and Francesca Rossi, editors, Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society, AIES 2018, New Orleans, LA, USA, February 02-03, 2018, pages 243–250. ACM, 2018. (Cited on page 21.)

[Rahimian & Mehrotra 2019] Hamed Rahimian and Sanjay Mehrotra. *Distributionally robust optimization: A review*. arXiv preprint arXiv:1908.05659, 2019. (Cited on pages 73 and 74.)

[Ramaswamy *et al.* 2020] Harish Guruprasad Ramaswamy*et al. Ablation-cam: Visual explanations for deep convolutional network via gradient-free localization.* In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pages 983–991, 2020. (Cited on page 125.)

[Rezaei *et al.* 2020] Ashkan Rezaei, Rizal Fathony, Omid Memarrast and Brian D. Ziebart. *Fairness for Robust Log Loss Classification.* In The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020, pages 5511–5518. AAAI Press, 2020. (Cited on page 27.)

[Ribeiro *et al.* 2016] Marco Tulio Ribeiro, Sameer Singh and Carlos Guestrin. *" Why should i trust you?" Explaining the predictions of any classifier.* In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pages 1135–1144, 2016. (Cited on pages 32 and 124.)

[Rigaki & Garcia 2023] Maria Rigaki and Sebastian Garcia. *A Survey of Privacy Attacks in Machine Learning.* ACM Comput. Surv., sep 2023. (Cited on pages 40, 41 and 86.)

[Rivest 1987] Ronald L. Rivest. *Learning Decision Lists.* Mach. Learn., vol. 2, no. 3, pages 229–246, 1987. (Cited on pages 57, 137 and 138.)

[Rouzot *et al.* 2022] Julien Rouzot, Julien Ferry and Marie-José Huguet. *Learning Optimal Fair Scoring Systems for Multi-Class Classification.* In ICTAI 2022-The 34th IEEE International Conference on Tools with Artificial Intelligence, 2022. (Cited on pages 3, 25 and 26.)

[Rudin & Ustun 2018] Cynthia Rudin and Berk Ustun. *Optimized scoring systems: Toward trust in machine learning for healthcare and criminal justice.* Interfaces, vol. 48, no. 5, pages 449–466, 2018. (Cited on page 25.)

[Rudin *et al.* 2022] Cynthia Rudin, Chaofan Chen, Zhi Chen, Haiyang Huang, Lesia Semenova and Chudi Zhong. *Interpretable machine learning: Fundamental principles and 10 grand challenges.* Statistic Surveys, vol. 16, pages 1–85, 2022. (Cited on pages 35, 50, 53 and 118.)

[Rudin 2019] Cynthia Rudin. *Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead.* Nature Machine Intelligence, vol. 1, no. 5, pages 206–215, 2019. (Cited on pages 30, 31, 34, 35, 50, 52 and 119.)

[Ruggieri 2013] Salvatore Ruggieri. *Data anonymity meets non-discrimination.* In 2013 IEEE 13th International Conference on Data Mining Workshops, pages 875–882. IEEE, 2013. (Cited on page 96.)

[Russell & Norvig 2020] Stuart Russell and Peter Norvig. Artificial intelligence: A modern approach (4th edition). Pearson, 2020. (Cited on page 9.)

[Saeys *et al.* 2008] Yvan Saeys, Thomas Abeel and Yves Van de Peer. *Robust Feature Selection Using Ensemble Feature Selection Techniques.* In Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pages 313–325. Springer, 2008. (Cited on page 75.)

[Sagawa *et al.* 2020] Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto and Percy Liang. *Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization.* In 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020. OpenReview.net, 2020. (Cited on pages 27, 74 and 75.)

[Salem *et al.* 2020] Ahmed Salem, Apratim Bhattacharya, Michael Backes, Mario Fritz and Yang Zhang. *Updates-Leak: Data Set Inference and Reconstruction Attacks in Online Learning.* In Srdjan Capkun and Franziska Roesner, editors, 29th USENIX Security Symposium, USENIX Security 2020, August 12-14, 2020, pages 1291–1308. USENIX Association, 2020. (Cited on page 42.)

[Samarati 2001] Pierangela Samarati. *Protecting Respondents' Identities in Microdata Release.* IEEE Trans. Knowl. Data Eng., vol. 13, no. 6, pages 1010–1027, 2001. (Cited on page 39.)

[Samuel 1959] Arthur L. Samuel. *Some Studies in Machine Learning Using the Game of Checkers.* IBM J. Res. Dev., vol. 3, no. 3, pages 210–229, 1959. (Cited on page 1.)

[Sarathy & Muralidhar 2011] Rathindra Sarathy and Krishnamurty Muralidhar. *Evaluating Laplace Noise Addition to Satisfy Differential Privacy for Numeric Data.* Trans. Data Priv., vol. 4, no. 1, pages 1–17, 2011. (Cited on page 47.)

[Schneider & Handali 2019] Johannes Schneider and Joshua Peter Handali. *Personalized Explanation for Machine Learning: a Conceptualization.* In Jan vom Brocke, Shirley Gregor and Oliver Müller, editors, 27th European Conference on Information Systems - Information Systems for a Sharing Society, ECIS 2019, Stockholm and Uppsala, Sweden, June 8-14, 2019, 2019. (Cited on page 150.)

[Selbst *et al.* 2019] Andrew D Selbst, Danah Boyd, Sorelle A Friedler, Suresh Venkatasubramanian and Janet Vertesi. *Fairness and abstraction in sociotechnical systems.* In Proceedings of the Conference on Fairness, Accountability, and Transparency, pages 59–68, 2019. (Cited on page 24.)

[Selvaraju *et al.* 2017] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh and Dhruv Batra. *Grad-cam: Visual explanations from deep networks via gradient-based localization.* In Proceedings of the IEEE international conference on computer vision, pages 618–626, 2017. (Cited on pages 32, 121 and 125.)

[Severi *et al.* 2021] Giorgio Severi, Jim Meyer, Scott E. Coull and Alina Oprea. *Explanation-Guided Backdoor Poisoning Attacks Against Malware Classifiers.* In Michael Bailey and Rachel Greenstadt, editors, 30th USENIX Security Symposium, USENIX Security 2021, August 11-13, 2021, pages 1487–1504. USENIX Association, 2021. (Cited on pages 121 and 122.)

[Sharma *et al.* 2020] Shubham Sharma, Jette Henderson and Joydeep Ghosh. *CERTIFAI: A Common Framework to Provide Explanations and Analyse the Fairness and Robustness of Black-box Models.* In Annette N. Markham, Julia Powles, Toby Walsh and Anne L. Washington, editors, AIES '20: AAAI/ACM Conference on AI, Ethics, and Society, New York, NY, USA, February 7-8, 2020, pages 166–172. ACM, 2020. (Cited on page 55.)

[Shokri *et al.* 2017] Reza Shokri, Marco Stronati, Congzheng Song and Vitaly Shmatikov. *Membership Inference Attacks Against Machine Learning Models.* In 2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017, pages 3–18. IEEE Computer Society, 2017. (Cited on page 40.)

[Shokri *et al.* 2020] Reza Shokri, Martin Strobel and Yair Zick. *Exploiting transparency measures for membership inference: a cautionary tale.* In The AAAI Workshop on Privacy-Preserving Artificial Intelligence (PPAI). AAAI, volume 13, 2020. (Cited on page 54.)

[Shokri *et al.* 2021] Reza Shokri, Martin Strobel and Yair Zick. *On the privacy risks of model explanations.* In Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society, pages 231–241, 2021. (Cited on pages 54, 123, 124 and 126.)

[Simonyan *et al.* 2014] Karen Simonyan, Andrea Vedaldi and Andrew Zisserman. *Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps.* In Yoshua Bengio and Yann LeCun, editors, 2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Workshop Track Proceedings, 2014. (Cited on page 125.)

[Singh *et al.* 2021] Chandan Singh, Keyan Nasseri, Yan Shuo Tan, Tiffany Tang and Bin Yu. *imodels: a python package for fitting interpretable models*, 2021. (Cited on page 139.)

[Slack *et al.* 2020a] Dylan Slack, Sorelle A. Friedler and Emile Givental. *Fairness warnings and fair-MAML: learning fairly with minimal data.* In Mireille Hildebrandt, Carlos Castillo, L. Elisa Celis, Salvatore Ruggieri, Linnet Taylor and Gabriela Zanfir-Fortuna, editors, FAT* '20: Conference on Fairness, Accountability, and Transparency, Barcelona, Spain, January 27-30, 2020, pages 200–209. ACM, 2020. (Cited on page 28.)

[Slack *et al.* 2020b] Dylan Slack, Sophie Hilgard, Emily Jia, Sameer Singh and Himabindu Lakkaraju. *Fooling LIME and SHAP: Adversarial Attacks on Post Hoc Explanation Methods.* In Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society, AIES '20, page 180–186, New York, NY, USA, 2020. Association for Computing Machinery. (Cited on page 56.)

[Slack *et al.* 2021a] Dylan Slack, Anna Hilgard, Himabindu Lakkaraju and Sameer Singh. *Counterfactual Explanations Can Be Manipulated.* In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang and Jennifer Wortman Vaughan, editors, Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual, pages 62–75, 2021. (Cited on page 56.)

[Slack *et al.* 2021b] Dylan Slack, Sophie Hilgard, Sameer Singh and Himabindu Lakkaraju. *Feature Attributions and Counterfactual Explanations Can Be Manipulated.* CoRR, vol. abs/2106.12563, 2021. (Cited on page 56.)

[Slowik & Bottou 2021] Agnieszka Slowik and Léon Bottou. *Algorithmic Bias and Data Bias: Understanding the Relation between Distributionally Robust Optimization and Data Curation.* CoRR, vol. abs/2106.09467, 2021. (Cited on page 27.)

[Smilkov *et al.* 2017] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas and Martin Wattenberg. *Smoothgrad: removing noise by adding noise.* arXiv preprint arXiv:1706.03825, 2017. (Cited on page 124.)

[Song *et al.* 2017] Congzheng Song, Thomas Ristenpart and Vitaly Shmatikov. *Machine Learning Models that Remember Too Much.* In Bhavani M. Thuraisingham, David Evans, Tal Malkin and Dongyan Xu, editors, Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017, pages 587–601. ACM, 2017. (Cited on page 42.)

[Speith 2022] Timo Speith. *A review of taxonomies of explainable artificial intelligence (XAI) methods.* In 2022 ACM Conference on Fairness, Accountability, and Transparency, pages 2239–2250, 2022. (Cited on page 31.)

[Srivastava *et al.* 2022] Gautam Srivastava, Rutvij H. Jhaveri, Sweta Bhattacharya, Sharnil Pandya, Rajeswari, Praveen Kumar Reddy Maddikunta, Gokul Yenduri, Jon G. Hall, Mamoun Alazab and Thippa Reddy Gadekallu. *XAI for*

*Cybersecurity: State of the Art, Challenges, Open Issues and Future Directions.* CoRR, vol. abs/2206.03585, 2022. (Cited on page 119.)

[Suciu *et al.* 2011] Dan Suciu, Dan Olteanu, Christopher Ré and Christoph Koch. Probabilistic databases. Synthesis Lectures on Data Management. Morgan & Claypool Publishers, 2011. (Cited on pages 131 and 132.)

[Suriyakumar *et al.* 2021] Vinith M. Suriyakumar, Nicolas Papernot, Anna Goldenberg and Marzyeh Ghassemi. *Chasing Your Long Tails: Differentially Private Prediction in Health Care Settings.* In Madeleine Clare Elish, William Isaac and Richard S. Zemel, editors, FAccT '21: 2021 ACM Conference on Fairness, Accountability, and Transparency, Virtual Event / Toronto, Canada, March 3-10, 2021, pages 723–734. ACM, 2021. (Cited on page 90.)

[Sweeney 2002] Latanya Sweeney. *k-Anonymity: A Model for Protecting Privacy.* Int. J. Uncertain. Fuzziness Knowl. Based Syst., vol. 10, no. 5, pages 557–570, 2002. (Cited on page 39.)

[Taskesen *et al.* 2020] Bahar Taskesen, Viet Anh Nguyen, Daniel Kuhn and Jose Blanchet. *A distributionally robust approach to fair classification.* arXiv preprint arXiv:2007.09530, 2020. (Cited on pages 27, 51, 74 and 75.)

[Team 2017] Apple Differential Privacy Team. *Learning with Privacy at Scale.* 2017. (Cited on page 40.)

[Tommasi *et al.* 2017] Tatiana Tommasi, Novi Patricia, Barbara Caputo and Tinne Tuytelaars. *A deeper look at dataset bias.* In Domain adaptation in computer vision applications, pages 37–55. Springer, 2017. (Cited on page 16.)

[Torralba & Efros 2011] Antonio Torralba and Alexei A Efros. *Unbiased look at dataset bias.* In CVPR 2011, pages 1521–1528. IEEE, 2011. (Cited on page 16.)

[Tramèr *et al.* 2016] Florian Tramèr, Fan Zhang, Ari Juels, Michael K. Reiter and Thomas Ristenpart. *Stealing Machine Learning Models via Prediction APIs.* In Thorsten Holz and Stefan Savage, editors, 25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, August 10-12, 2016, pages 601–618. USENIX Association, 2016. (Cited on page 41.)

[Tran *et al.* 2021a] Cuong Tran, My H. Dinh, Kyle Beiter and Ferdinando Fioretto. *A Fairness Analysis on Private Aggregation of Teacher Ensembles.* CoRR, vol. abs/2109.08630, 2021. (Cited on pages 90 and 91.)

[Tran *et al.* 2021b] Cuong Tran, My H. Dinh and Ferdinando Fioretto. *Differentially Private Empirical Risk Minimization under the Fairness Lens.* In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang and Jennifer Wortman Vaughan, editors, Advances in Neural Information

Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual, pages 27555–27565, 2021. (Cited on page 90.)

[Tran *et al.* 2021c] Cuong Tran, Ferdinando Fioretto and Pascal Van Hentenryck. *Differentially Private and Fair Deep Learning: A Lagrangian Dual Approach.* In Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021, pages 9932–9939. AAAI Press, 2021. (Cited on page 93.)

[Truta *et al.* 2007] Traian Marius Truta, Alina Campan and Paul Meyer. *Generating Microdata with* P *-Sensitive* K *-Anonymity Property.* In Willem Jonker and Milan Petkovic, editors, Secure Data Management, 4th VLDB Workshop, SDM 2007, Vienna, Austria, September 23-24, 2007, Proceedings, volume 4721 of *Lecture Notes in Computer Science*, pages 124–141. Springer, 2007. (Cited on page 39.)

[Uniyal *et al.* 2021] Archit Uniyal, Rakshit Naidu, Sasikanth Kotti, Sahib Singh, Patrik Joslin Kenfack, Fatemehsadat Mireshghallah and Andrew Trask. *DP-SGD vs PATE: Which Has Less Disparate Impact on Model Accuracy?* CoRR, vol. abs/2106.12576, 2021. (Cited on page 90.)

[Ustun & Rudin 2016] Berk Ustun and Cynthia Rudin. *Supersparse linear integer models for optimized medical scoring systems.* Machine Learning, vol. 102, pages 349–391, 2016. (Cited on page 25.)

[Ustun *et al.* 2019] Berk Ustun, Alexander Spangher and Yang Liu. *Actionable Recourse in Linear Classification.* In danah boyd and Jamie H. Morgenstern, editors, Proceedings of the Conference on Fairness, Accountability, and Transparency, FAT* 2019, Atlanta, GA, USA, January 29-31, 2019, pages 10–19. ACM, 2019. (Cited on page 55.)

[Verhaeghe *et al.* 2020] Hélène Verhaeghe, Siegfried Nijssen, Gilles Pesant, Claude-Guy Quimper and Pierre Schaus. *Learning optimal decision trees using constraint programming.* Constraints An Int. J., vol. 25, no. 3-4, pages 226–250, 2020. (Cited on page 12.)

[Verma & Rubin 2018] Sahil Verma and Julia Rubin. *Fairness definitions explained.* In Yuriy Brun, Brittany Johnson and Alexandra Meliou, editors, Proceedings of the International Workshop on Software Fairness, FairWare@ICSE 2018, Gothenburg, Sweden, May 29, 2018, pages 1–7. ACM, 2018. (Cited on page 17.)

[Vidal & Schiffer 2020] Thibaut Vidal and Maximilian Schiffer. *Born-again tree ensembles.* In International conference on machine learning, pages 9743–9753. PMLR, 2020. (Cited on page 31.)

[Voigt & Von dem Bussche 2017] Paul Voigt and Axel Von dem Bussche. *The eu general data protection regulation (gdpr).* A Practical Guide, 1st Ed., Cham: Springer International Publishing, vol. 10, no. 3152676, pages 10–5555, 2017. (Cited on pages 1, 16, 29 and 38.)

[Wachter *et al.* 2020] Sandra Wachter, Brent Mittelstadt and Chris Russell. *Bias preservation in machine learning: the legality of fairness metrics under EU non-discrimination law.* W. Va. L. Rev., vol. 123, page 735, 2020. (Cited on page 17.)

[Wang & Lin 2021] Tong Wang and Qihang Lin. *Hybrid Predictive Models: When an Interpretable Model Collaborates with a Black-box Model.* J. Mach. Learn. Res., vol. 22, pages 137–1, 2021. (Cited on page 36.)

[Wang *et al.* 2021] Yijie Wang, Viet Anh Nguyen and Grani A Hanasusanto. *Wasserstein Robust Support Vector Machines with Fairness Constraints.* arXiv preprint arXiv:2103.06828, 2021. (Cited on pages 27, 51 and 74.)

[Wang *et al.* 2022a] Caroline Wang, Bin Han, Bhrij Patel and Cynthia Rudin. *In pursuit of interpretable, fair and accurate machine learning for criminal recidivism prediction.* Journal of Quantitative Criminology, pages 1–63, 2022. (Cited on page 54.)

[Wang *et al.* 2022b] Yongjie Wang, Hangwei Qian and Chunyan Miao. *DualCF: Efficient Model Extraction Attack from Counterfactual Explanations.* In FAccT '22: 2022 ACM Conference on Fairness, Accountability, and Transparency, Seoul, Republic of Korea, June 21 - 24, 2022, pages 1318–1329. ACM, 2022. (Cited on page 123.)

[Wang 2019] Tong Wang. *Gaining free or low-cost interpretability with interpretable partial substitute.* In International Conference on Machine Learning, pages 6505–6514. PMLR, 2019. (Cited on page 36.)

[Warner 1965] Stanley L Warner. *Randomized response: A survey technique for eliminating evasive answer bias.* Journal of the American Statistical Association, vol. 60, no. 309, pages 63–69, 1965. (Cited on page 44.)

[Watkins *et al.* 2022] Elizabeth Anne Watkins, Michael McKenna and Jiahao Chen. *The four-fifths rule is not disparate impact: a woeful tale of epistemic trespassing in algorithmic fairness.* CoRR, vol. abs/2202.09519, 2022. (Cited on page 24.)

[Weller 2019] Adrian Weller. *Transparency: Motivations and Challenges.* In Wojciech Samek, Grégoire Montavon, Andrea Vedaldi, Lars Kai Hansen and Klaus-Robert Müller, editors, Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, volume 11700 of *Lecture Notes in Computer Science*, pages 23–40. Springer, 2019. (Cited on page 35.)

[Woodworth *et al.* 2017] Blake E. Woodworth, Suriya Gunasekar, Mesrob I. Ohannessian and Nathan Srebro. *Learning Non-Discriminatory Predictors.* In Satyen Kale and Ohad Shamir, editors, Proceedings of the 30th Conference on Learning Theory, COLT 2017, Amsterdam, The Netherlands, 7-10 July 2017, volume 65 of *Proceedings of Machine Learning Research*, pages 1920–1953. PMLR, 2017. (Cited on page 21.)

[Xiang & Raji 2019] Alice Xiang and Inioluwa Deborah Raji. *On the Legal Compatibility of Fairness Definitions.* CoRR, vol. abs/1912.00761, 2019. (Cited on page 16.)

[Xin *et al.* 2022] Rui Xin, Chudi Zhong, Zhi Chen, Takuya Takagi, Margo I. Seltzer and Cynthia Rudin. *Exploring the Whole Rashomon Set of Sparse Decision Trees.* In NeurIPS, 2022. (Cited on page 150.)

[Xu *et al.* 2019] Depeng Xu, Shuhan Yuan and Xintao Wu. *Achieving Differential Privacy and Fairness in Logistic Regression.* In Sihem Amer-Yahia, Mohammad Mahdian, Ashish Goel, Geert-Jan Houben, Kristina Lerman, Julian J. McAuley, Ricardo Baeza-Yates and Leila Zia, editors, Companion of The 2019 World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019, pages 594–599. ACM, 2019. (Cited on page 93.)

[Xu *et al.* 2021] Depeng Xu, Wei Du and Xintao Wu. *Removing Disparate Impact on Model Accuracy in Differentially Private Stochastic Gradient Descent.* In Feida Zhu, Beng Chin Ooi and Chunyan Miao, editors, KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021, pages 1924–1932. ACM, 2021. (Cited on page 90.)

[Yang *et al.* 2022] Fan Yang, Qizhang Feng, Kaixiong Zhou, Jiahao Chen and Xia Hu. *Differentially Private Counterfactuals via Functional Mechanism.* CoRR, vol. abs/2208.02878, 2022. (Cited on page 121.)

[Yu *et al.* 2020] Jinqiang Yu, Alexey Ignatiev, Pierre Le Bodic and Peter J. Stuckey. *Optimal Decision Lists using SAT.* CoRR, vol. abs/2010.09919, 2020. (Cited on page 58.)

[Yurochkin *et al.* 2020] Mikhail Yurochkin, Amanda Bower and Yuekai Sun. *Training individually fair ML models with sensitive subspace robustness.* In 8th International Conference on Learning Representations, ICLR Addis Ababa, Ethiopia, April 26-30, 2020. OpenReview.net, 2020. (Cited on page 29.)

[Zafar *et al.* 2017] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez and Krishna P. Gummadi. *Fairness Beyond Disparate Treatment & Disparate Impact: Learning Classification without Disparate Mistreatment.* In Proceedings of the 26th International Conference on World Wide Web, WWW '17, page 1171–1180, Republic and Canton of Geneva, CHE, 2017.

International World Wide Web Conferences Steering Committee. (Cited on pages 16, 21, 22, 43 and 67.)

[Zemel *et al.* 2013] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi and Cynthia Dwork. *Learning fair representations.* In International Conference on Machine Learning, pages 325–333, 2013. (Cited on pages 19, 20, 23, 54 and 95.)

[Zhang *et al.* 2012] Jun Zhang, Zhenjie Zhang, Xiaokui Xiao, Yin Yang and Marianne Winslett. *Functional Mechanism: Regression Analysis under Differential Privacy.* Proc. VLDB Endow., vol. 5, no. 11, pages 1364–1375, 2012. (Cited on pages 44, 45 and 93.)

[Zhang *et al.* 2018a] Brian Hu Zhang, Blake Lemoine and Margaret Mitchell. *Mitigating Unwanted Biases with Adversarial Learning.* In Jason Furman, Gary E. Marchant, Huw Price and Francesca Rossi, editors, Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society, AIES 2018, New Orleans, LA, USA, February 02-03, 2018, pages 335–340. ACM, 2018. (Cited on page 22.)

[Zhang *et al.* 2018b] Quanshi Zhang, Wenguan Wang and Song-Chun Zhu. *Examining CNN Representations With Respect to Dataset Bias.* In Sheila A. McIlraith and Kilian Q. Weinberger, editors, Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018, pages 4464–4473. AAAI Press, 2018. (Cited on page 20.)

[Zhang *et al.* 2020] Xinyang Zhang, Ningfei Wang, Hua Shen, Shouling Ji, Xiapu Luo and Ting Wang. *Interpretable Deep Learning under Fire.* In Srdjan Capkun and Franziska Roesner, editors, 29th USENIX Security Symposium, USENIX Security 2020, August 12-14, 2020, pages 1659–1676. USENIX Association, 2020. (Cited on page 56.)

[Zhang *et al.* 2021] Tao Zhang, Tianqing Zhu, Kun Gao, Wanlei Zhou and Philip S. Yu. *Balancing Learning Model Privacy, Fairness, and Accuracy With Early Stopping Criteria.* IEEE Transactions on Neural Networks and Learning Systems, pages 1–13, 2021. (Cited on page 90.)

[Zhao *et al.* 2021] Xuejun Zhao, Wencan Zhang, Xiaokui Xiao and Brian Lim. *Exploiting explanations for model inversion attacks.* In Proceedings of the IEEE/CVF international conference on computer vision, pages 682–692, 2021. (Cited on page 125.)

[Zhou 2012] Zhi-Hua Zhou. Ensemble methods: Foundations and algorithms. Chapman & Hall/CRC, 1st édition, 2012. (Cited on page 78.)

[Zliobaite & Custers 2016] Indre Zliobaite and Bart Custers. *Using sensitive personal data may be necessary for avoiding discrimination in data-driven decision models.* Artif. Intell. Law, vol. 24, no. 2, pages 183–201, 2016. (Cited on page 86.)

[Zou *et al.* 2016] Quan Zou, Jiancang Zeng, Liujuan Cao and Rongrong Ji. *A novel features ranking metric with application to scalable visual and bioinformatics data classification.* Neurocomputing, vol. 173, pages 346–354, 2016. (Cited on page 75.)

[Zytek *et al.* 2022] Alexandra Zytek, Ignacio Arnaldo, Dongyu Liu, Laure Berti-Equille and Kalyan Veeramachaneni. *The need for interpretable features: Motivation and taxonomy.* ACM SIGKDD Explorations Newsletter, vol. 24, no. 1, pages 1–13, 2022. (Cited on page 35.)