Learning Optimal Fair Decision Trees: Trade-offs Between Interpretability, Fairness, and Accuracy

Nathanael Jo
nathanael.jo@gmail.com
USC Center for AI in Society
Los Angeles, CA, USA

Sina Aghaei
saghaei@usc.edu
USC Center for AI in Society
Los Angeles, CA, USA

Jack Benson
jackhbenson17@gmail.com
USC Center for AI in Society
Los Angeles, CA, USA

Andrés Gómez
andgomez@usc.edu
University of Southern California
Los Angeles, CA, USA

Phebe Vayanos
phebe.vayanos@usc.edu
USC Center for AI in Society
Los Angeles, CA, USA

ABSTRACT

The increasing use of machine learning in high-stakes domains – where people's livelihoods are impacted – creates an urgent need for interpretable, fair, and highly accurate algorithms. With these needs in mind, we propose a mixed integer optimization (MIO) framework for learning optimal classification trees – one of the most interpretable models – that can be augmented with arbitrary fairness constraints. In order to better quantify the "price of interpretability", we also propose a new measure of model interpretability called *decision complexity* that allows for comparisons across different classes of machine learning models. We benchmark our method against state-of-the-art approaches for fair classification on popular datasets; in doing so, we conduct one of the first comprehensive analyses of the trade-offs between interpretability, fairness, and predictive accuracy. Given a fixed disparity threshold, our method has a price of interpretability of about 4.2 percentage points in terms of out-of-sample accuracy compared to the best performing, complex models. However, our method consistently finds decisions with almost full parity, while other methods rarely do.

KEYWORDS

fair machine learning, interpretability, decision trees, mixed-integer optimization

1 INTRODUCTION

There is growing interest in using machine learning (ML) to make decisions in high-stakes domains. For instance, ML algorithms are now commonly used to determine a criminal's risk of recidivism in the United States [7]. There is also a growing literature in designing algorithms to determine the best course of action for homeless individuals [9], to diagnose and treat of various illnesses [26], and many more. In these contexts, it is necessary for such models to be both *accurate* (in order to minimize erroneous predictions that negatively affect stakeholders) and *interpetable* (so that decisions are transparent and hence accountable). We therefore focus our attention to the problem of learning optimal classification trees. Classification trees are among the most interpretable of models [48], and optimal trees – rather than ones that are built using heuristics – maximize predictive accuracy. They belong to a broader set of models known as decision trees (the other type of decision tree being regression trees, which apply to datasets with real valued labels).

There exist numerous qualitative ways of characterizing interpretability [42]. These notions often involve querying humans (especially practitioners, community stakeholders, etc.), so that interpretability desiderata can be tailored to the application or population at hand [21]. Unfortunately, quantitative notions of interpretability are lacking in the machine learning literature, making it hard to compare models systematically without humans in the loop. One such measure – sparsity – is one of the only quantitative proxies for interpretability, but does not allow for equivalent comparisons between different model classes [49] (see Section 1.3 for more details). Therefore, we seek to address this gap by proposing a new notion of interpretability in order to more formally quantify the price of interpretability between our classification trees and more complex models.

Apart from interpretability, another crucial consideration in machine learning for high-impact situations is *fairness*. After all, an algorithm that affects people's well-being should be aware of the particular historical and/or social contexts that surround the learning problem. However, what constitutes as fair may vary widely depending on one's goals, domains of interest, etc. This discussion surrounding "algorithmic fairness" consists of a rich literature that bridges philosophy, economics, and computer science [23, 46, 47]. Consider, for example, a naïve approach where a classifier learns from data where the sensitive attributes are omitted (such as race, sex, and others). Also known as "fairness through unawareness",

this approach often leads to discrimination because some other non-sensitive attribute(s) may be correlated with the withheld features (e.g., race is inextricably linked to ZIP code and income) [43].

Another approach – and one that this paper adopts – is the notion of "group" statistical fairness. A classifier satisfying group fairness is now aware of the sensitive features in the data but will only produce decisions that enforce parity between segments of the population. This is in contrast to "individual" fairness, which requires that individuals with similar characteristics be classified similarly [23]. Both perspectives have their advantages and drawbacks, but in this work we will focus on group fairness primarily because of it is simpler to define and easier for stakeholders to understand; as such, many practitioners in practice value assessing and enforcing group fairness (see, e.g., [8]). In the following sections, we will discuss in more depth the use of many notions of group fairness in the machine learning literature.

1.1 Problem Statement

We now formalize the problem we study. Let $\mathcal{D} := \{(x^i, y^i)\}_{i \in \mathcal{I}}$ be training data indexed in the set $\mathcal{I} \in \{1, \ldots, I\}$. Each datapoint $i \in \mathcal{I}$ consists of a vector x^i with F features (i.e., $x^i \in \mathbb{R}^F$), and a class label $y^i \in \mathcal{K}$, where $\mathcal{K}$ is the finite set of possible classes. Throughout this paper, we will consider the case of binary classes, i.e., $\mathcal{K} := \{0, 1\}$ where $y = 1$ will be referred to as the positive class. Note that the literature on fairness for multi-class learning is limited, see e.g., Denis et al. [20]. The goal is to learn, over all possible trees of maximum depth d, the tree that maps x^i to y^i and maximizes out-of-sample accuracy, using in-sample performance as a proxy. Further, suppose the population can be divided into different sensitive groups (whereby discrimination exists or is a concern within the classification problem, e.g., race, gender). Let $\mathcal{P}$ denote levels of a sensitive attribute, and let each datapoint i have a value $p^i \in \mathcal{P}$. The features in $\mathcal{P}$ may or may not be included in the vector x^i since there may be legal or ethical considerations barring the use of protected features in the predictive task [17].

1.2 Common Notions of Group Fairness

In this section, we define five common notions of group fairness in the machine learning literature that we will use. Let $\hat{Y}$ be the classifier's prediction, and let $X, Y, \hat{Y}$, and P be random variables for features, classes, classifier's predictions, and protected features, respectively; their joint distribution is unknown and denoted by $\mathbb{P}$.
Statistical Parity. A classifier satisfies statistical parity if the probability of receiving a positive class is equal across all protected groups [23]. Formally, this means

$$\mathbb{P}[\hat{Y} = 1 | P = p] = \mathbb{P}[\hat{Y} = 1 | P = p'] \quad \forall p, p' \in \mathcal{P}.$$

Conditional Statistical Parity. A classifier satisfies *conditional* statistical parity if the probability of receiving a positive class is equal across all protected groups, conditional on some legitimate feature(s) indicative of risk [19]. This may be considered as a fairer notion than statistical parity because it takes into account the distribution of risk factors within each sensitive group. Letting L (which is a subvector of X) represent the random variable taken from a set

$\mathcal{L}$ of legitimate features, conditional statistical parity is satisfied if

$$\mathbb{P}[\hat{Y} = 1 | P = p, L = \ell] = \mathbb{P}[\hat{Y} = 1 | P = p', L = \ell]$$
$$\forall p, p' \in \mathcal{P}, \ell \in \mathcal{L}.$$

Predictive Equality. A classifier satisfies predictive equality if all protected groups have the same false positive rates (FPR) [18], i.e.,

$$\mathbb{P}[\hat{Y} = 1 | P = p, Y = 0] = \mathbb{P}[\hat{Y} = 1 | P = p', Y = 0] \quad \forall p, p' \in \mathcal{P}.$$

Equal Opportunity. A classifier satisfies equal opportunity if all protected groups have the same true positive rate (TPR) [30]. The formal definition of equal opportunity is

$$\mathbb{P}[\hat{Y} = 1 | P = p, Y = 1] = \mathbb{P}[\hat{Y} = 1 | P = p', Y = 1] \quad \forall p, p' \in \mathcal{P}.$$

Equalized Odds. Equalized odds combines predictive equality and equal opportunity such that both FPR and TPR must be similar across all protected groups [30]. Equalized odds is, of course, a stronger condition than predictive equality and equal opportunity individually. Formally, equalized odds is satisfied if

$$\mathbb{P}[\hat{Y} = 1 | P = p, Y = y] = \mathbb{P}[\hat{Y} = 1 | P = p', Y = y]$$
$$\forall p, p' \in \mathcal{P}, y \in \{0, 1\}.$$

1.3 Related Works

Our work relates to five streams of literature in machine learning, which we review in turn. In this section we briefly review related works in the machine learning literature.

Discrimination Prevention in Machine Learning. The literature on fairness in machine learning is extensive – we point the interested reader to [38] and [59] for surveys of this topic. In general, there are three approaches that existing works employ in order to promote fairness. The first is *pre-processing*, which entails eliminating discrimination within the training data, see [27, 29, 35, 56], among others. The second is *post-processing*, which takes the model's output and alters its decisions in ways that promote some fairness metric, see [30, 34, 44]. Finally, one may also employ *in-processing* by modifying existing models so that fairness is integrated within its learning goal; for instance, approaches have utilized regularization [3, 11, 36] or different heuristics [16, 34]. Our work falls under the category of in-processing discrimination prevention techniques.

Fair Decision Trees. Within the stream of literature on discrimination prevention, our work most closely relates to approaches for learning fair decision trees. For instance, Ranzato et al. [45] adapt a genetic algorithm for learning robust decision trees in order to prioritize individual fairness. In the online setting, Zhang et al. [58] propose a fairness-aware Hoeffding tree that builds decision trees over streams of data. Grari et al. [28] take an adversarial approach to training fair gradient-boosted trees that promote statistical parity. Kanamori and Arimura [37] propose a post-processing step that uses mixed-integer optimization (MIO) to edit the branching thresholds of the tree's internal nodes to satisfy some fairness constraint; in the paper they use statistical parity and equalized odds. In a similar manner, Zhang et al. [57] flip the outcomes of different paths of a learned tree in order to improve fairness. Closely related to our work is that of Aghaei et al. [3], who propose a mixed-integer optimization framework to build optimal and fair decision trees that prevent disparate impact and/or disparate treatment. In contrast to

our work, Aghaei et al. [3] enforces fairness via regularization while our method uses constraints in the optimization. Finally, Kamiran et al. [34] propose a two-pronged approach: the first is to incorporate sensitivity gain (IGS) with information gain (IGC) as a heuristic to build a fairer decision tree, while the second is to relabel the predictions in the tree's leaf nodes to further promote statistical parity.

Optimal Decision Trees. Mixed-integer optimization (MIO) has recently gained traction as a framework to solve various machine learning problems. Particularly related to this paper are approaches that use MIO to learn optimal decision trees in order to improve on the traditional classification/regression tree (CART) algorithms [15], which rely on a heuristic. Many works introduce novel formulations for learning optimal classification trees [3, 4, 12, 53]. Elmachtoub et al. [25] use MIO to learn decision trees that minimize a loss function derived from the "predict-then-optimize" framework. There exist several extensions to learning decision trees using MIO as well. For instance, Mišić [40] uses MIO to solve tree ensemble models to optimality; Jo et al. [31] to learn optimal prescriptive trees from observational data; and Justin et al. [33] and Bertsimas et al. [13] to learn optimal robust decision trees. In this paper, we build on the MIO method introduced by Aghaei et al. [4] by conducting extensive experiments when we add various fairness constraints to the model.

Notions of Interpretability. There exist numerous proxies or desiderata for interpretability in the machine learning literature, including:

- **Sparsity**: The simplicity of a model. There are many ways to define sparsity, which also differ across model classes. For instance, within the context of decision trees, Rudin et al. Rudin et al. [49] define sparsity using the number of leaves, where trees with fewer leaves are sparser and thus preferable. In a regression model, sparsity is widely associated with the number of nonzero regression coefficients [5, 39, 52]. In general, a numeric value for sparsity is only useful when comparing models within the same class. This is because some model classes have sparsity that does not grow uncontrollably (e.g., regression is constrained by the number of features), while others are a function of design and data (e.g., trees become increasingly complex as depth increases) [49].
- **Simulatability**: The extent to which a human can internally simulate and reason about part of the entire decision-making schema [42]. Shallow decision trees are some of the most simulatable models since we can easily visualize and understand if-else rules. This is in contrast to a neural network, where the numerous connections between nodes result in complicated calculations that a human cannot keep track of.
- **Scope** (Global vs. Local interpretability): Global interpretability means that a human can wholly understand the decision schema (as is the case with, say, regression, where everything one needs to know about the model is encoded in its coefficients). On the other hand, local interpretability means that one could reason about how and why a particular datapoint gets classified a certain way [41]. For instance, while the entire behavior of a k-nearest neighbor (kNN) algorithm is incomprehensible, especially when a dataset is large, humans can reason about local behavior: a datapoint is classified a certain way because most of its k-nearest neighbors are classified the same way.

There are many other proxies (e.g., uncertainty, algorithmic transparency, monotonicity, etc.), as well as other notions of interpretabilty that apply to various stages of the predictive pipeline (e.g., considerations in feature engineering). We refer the interested reader to [41, 42, 49] for an extensive overview of interpretable machine learning. As we have mentioned previously, interpretability is broad, loosely-defined, and often context dependent. Nonetheless, existing quantitative measures (such as sparsity) are only useful for comparisons within the same model class; our work attempts to address this shortcoming by proposing a new measure of interpretability that allows for comparisons *across* model classes.

Interpretability, Accuracy, and Fairness. Several works have explored the possible trade-offs between predictive accuracy and interpretability (without considering fairness), often in application-specific contexts [10, 32] or within a certain model class [5, 6, 50, 55] where interpretability desiderata differ widely. In a more general setting, Dziugaite et al. [24] propose a learning framework via empirical risk minimization that imposes interpretability constraints, and characterizes when trade-offs between accuracy and interpretability may exist – this is in line with other works that find the optimal model within a specified model class subject to interpretability constraints (e.g., Azizi et al. [9], Rudin et al. [49]). On the other hand, Semenova et al. [51] allow for comparisons *across* various model classes by using Rashomon sets to gauge the likelihood of simpler models with competing accuracy performance existing within a hypothesis space. However, none of these works consider fairness as a dimension in model selection. There is a dearth of literature touching the trade-offs between all three: interpretability, accuracy, *and* fairness. Agarwal [2] finds theoretical results that in general, there exist more complex models that perform strictly better with respect to fairness and accuracy. Our work extends this finding by experimentally characterizing these trade-offs. Wang et al. [54] compare models with varying levels of interpretability in predicting criminal recidivism, considering notions of fairness where the prediction task is continuous (e.g., calibration). In contrast, our work considers binary fairness notions and we run experiments more generally on various benchmark datasets – which include predicting recidivism. We additionally compare several fairness-promoting algorithms in the literature, while Wang et al. [54] only consider off-the-shelf ML models.

1.4 Contributions

In this work, we build upon an MIO formulation to learn optimal decision trees initially proposed in Aghaei et al. [4] by showcasing its flexible modeling power in considering various notions of fairness. We benchmark these experiments against two state-of-the-art methods that similarly learn fair decision trees, as well as three fair classification algorithms that are compatible with a variety of ML models. In order to more formally juxtapose the interpretability of ML models, we propose a new measure called *decision complexity*. In doing so, we conduct one of the first experiments to characterize the trade-offs between performance and interpretability within the

algorithmic fairness literature, and discuss the practical considerations among these dimensions. In particular, we observe that the best performing, complex models have on average 4.2 percentage points higher out-of-sample accuracy than our interpretable approach, indicating the price of interpretability. We also observe that our method is particularly well-suited in finding decisions with full parity, while other methods do not boast the same guarantees.

The remainder of this paper is structured as follows. In Section 2, we introduce a new notion of interpretability – decision complexity – that allows for comparisons across model classes. We then outline the MIO formulation to learn optimal and fair decision trees in Section 3. Finally, we conduct and analyze our computational experiments in Section 4.

2 DECISION COMPLEXITY

As discussed in Section 1.3, sparsity/simplicity is one of the only metrics in the literature to quantify interpretability, and is mainly relevant when comparing the models within the same class. In our work, however, we are concerned with comparing a measure of interpretability *across* model classes (e.g., comparing decision trees with kNN). To the best of our knowledge, there is no such universal definition or framework for interpretability. We therefore propose a new notion that can quantify the interpretability of predictive models belonging either to the same or to different model classes.

Definition 2.1 (Decision Complexity). Given a trained classifier, decision complexity captures the minimum number of parameters needed for the classifier to make a prediction on a new datapoint.

Example 2.1 (Binary Classification Trees). The decision complexity of binary classification trees is measured by the number of nodes in the tree (branching plus leaves), which corresponds to the number of times a datapoint is routed through the tree and how it is classified.

Example 2.2 (Random Forest). Building from binary classification trees, a random forest's decision complexity is equal to the sum of nodes (branching and leaves) over all trees in the forest.

Example 2.3 (Linear/Logistic Regression). Assuming full linearity (i.e., no interaction or quadratic terms), the decision complexity of simple regression models is always equal to the number of features in the data (plus a possible bias term).

Example 2.4 (k-Nearest Neighbors). In order to classify a new datapoint, we must find the distance between said datapoint with all training points, and therefore a kNN's decision complexity is equal to the size of the training data. While there exist more efficient algorithms in practice that do not require finding all pairwise comparisons, we are concerned primarily with how a human can walk through the algorithm's decisions rather than the computational complexity to train the classifier.

Example 2.5 (Support Vector Machines). Decision complexity in SVMs highly depends on the choice of kernel. Linear kernels have a complexity equal to the number of features in the data, which correspond to the coefficients in the hyperplane that separates classes. However, gaussian RBF kernels have a decision complexity equal to the number of support vectors in the training set. This is

because, upon training, each of the support vectors are associated with weights that determine how a new datapoint gets classified.

Example 2.6 (Neural Network). A neural network's decision complexity is equal to the number of "connections" between all nodes, since a new datapoint is classified through matrix calculations using solved weights from each connection.

Decision complexity can be viewed as an extension to sparsity (in that sparsity is often used analogously to simplicity) but has the advantage of being general enough for all model classes. It can also be interpreted as attempting to quantify a part of simulatability – the more decisions a model takes, the harder it tends to be for humans to simulate the decision making process.

3 FORMULATION

In this section, we present the MIO formulation to learn optimal decision trees proposed in Aghaei et al. [4]. The paper introduced the formulation without an emphasis on fairness, but in this work we use the formulation as a building block to which we add various fairness constraints. From hereon, we will refer to the combination of the MIO formulation and the fairness constraints as *FairOCT*.

3.1 From Decision Tree to Flow Graph

We first introduce the modeling framework we use for decision trees. The key idea is to convert a decision tree into a directed, acyclic graph where all arcs "flow" from the tree's root to its leaves. We start with a perfect binary tree of depth d, whose nodes are labeled 1 through $(2^{d+1} - 1)$ in order of a breadth-first search. Let $\mathcal{B} := \{1, \ldots, 2^d - 1\}$ denote the set of branching nodes and $\mathcal{T} := \{2^d, \ldots, 2^{d+1} - 1\}$ the set of terminal nodes. We then convert all arcs in the tree to point from the parent node to its child node (see Figure 1, left). From the binary tree, we connect a source s to the root and all nodes $n \in \mathcal{B} \cup \mathcal{T}$ to sinks t_k, one for every class $k \in \mathcal{K}$ (see Figure 1, right). While we assume that we have binary classes, this formulation generalizes to arbitrary finite classes. All arcs have a capacity of 1, so each datapoint is weighted equally and flows from the source s to one sink t_k, where k is the class that the decision tree assigns to that datapoint. From hereon, we refer to this structure as a "flow graph".
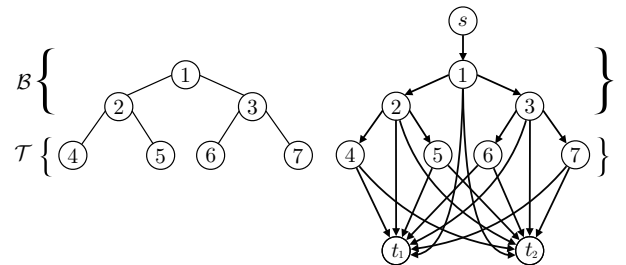
Figure 1: A decision tree of depth 2 (left) and its associated flow graph (right) with two classes.

3.2 MIO Formulation

With the flow graph at hand, we present the MIO formulation for learning optimal classification trees. Without loss of generality, we assume that features are binary, i.e., $x^i \in \{0, 1\}^F$ – this assumption can easily be relaxed to cater for integer or categorical features, see Remark 1 in [4]. We encode the branching structure of the tree with variables $b_{nf} \in \{0, 1\}$, for all $n \in \mathcal{B}, f \in \mathcal{F}$, which indicate if feature f is selected for branching at node n. We also encode the prediction scheme of the tree using variables $w_{nk} \in \{0, 1\}$, for all $n \in \mathcal{B} \cup \mathcal{T}, k \in \mathcal{K}$, which equal 1 if and only if (iff) node n assigns class k to all datapoints that land on that node. We let $p_n \in \{0, 1\}$ indicate if node n is a prediction node, in which case it must assign a class to all datapoints that land on that node and no further branching is allowed. We also define "flow variables" z to capture the flow of datapoints, where $z^i_{a(n),n} \in \{0, 1\}$ equals 1 iff datapoint i flows from the direct ancestor of node n, $a(n)$, to n.

The formulation is as follows:

$$\text{maximize} \quad \sum_{i \in \mathcal{I}} \sum_{n \in \mathcal{B} \cup \mathcal{T}} z^i_{n,t_{y^i}} \qquad \text{subject to} \quad (1a)$$

$$\sum_{f \in \mathcal{F}} b_{nf} + p_n + \sum_{m \in \mathcal{A}(n)} p_m = 1 \qquad \forall n \in \mathcal{B} \quad (1b)$$

$$p_n + \sum_{m \in \mathcal{A}(n)} p_m = 1 \qquad \forall n \in \mathcal{T} \quad (1c)$$

$$z^i_{a(n),n} = z^i_{n,\ell(n)} + z^i_{n,r(n)} + \sum_{k \in \mathcal{K}} z^i_{n,t_k} \qquad \forall n \in \mathcal{B}, i \in \mathcal{I} \quad (1d)$$

$$z^i_{a(n),n} = \sum_{k \in \mathcal{K}} z^i_{n,t_k} \qquad \forall i \in \mathcal{I}, n \in \mathcal{T} \quad (1e)$$

$$z^i_{s,1} \le 1 \qquad \forall i \in \mathcal{I} \quad (1f)$$

$$z^i_{n,\ell(n)} \le \sum_{f \in \mathcal{F}: x^i_f = 0} b_{nf} \qquad \forall n \in \mathcal{B}, i \in \mathcal{I} \quad (1g)$$

$$z^i_{n,r(n)} \le \sum_{f \in \mathcal{F}: x^i_f = 1} b_{nf} \qquad \forall n \in \mathcal{B}, i \in \mathcal{I} \quad (1h)$$

$$z^i_{n,t_k} \le w^n_k \qquad \forall i \in \mathcal{I}, n \in \mathcal{B} \cup \mathcal{T}, k \in \mathcal{K} \quad (1i)$$

$$\sum_{k \in \mathcal{K}} w^n_k = p_n \qquad \forall n \in \mathcal{B} \cup \mathcal{T} \quad (1j)$$

$$w^n_k \in \{0, 1\} \qquad \forall n \in \mathcal{B} \cup \mathcal{T}, k \in \mathcal{K} \quad (1k)$$

$$b_{nf} \in \{0, 1\} \qquad \forall n \in \mathcal{B}, f \in \mathcal{F} \quad (1l)$$

$$p_n \in \{0, 1\} \qquad \forall n \in \mathcal{B} \cup \mathcal{T} \quad (1m)$$

$$z^i_{a(n),n}, z^i_{n,t_k} \in \{0, 1\} \qquad \forall n \in \mathcal{B} \cup \mathcal{T}, i \in \mathcal{I}, k \in \mathcal{K}, \quad (1n)$$

where $\ell(n)$ (resp. $r(n)$) is the left (resp. right) descendant of n and $\mathcal{A}(n)$ is the set of all ancestors of node $n \in \mathcal{B} \cup \mathcal{T}$. The objective (1a) maximizes the number of correctly classified datapoints. Note that we can add a regularization term in the objective to control for overfitting, see Aghaei et al. [4]. Constraints (1b) ensure that each node is either a branching node, a prediction node, or neither of the two because one of its ancestors is already a prediction node; the last option means that the node is pruned out. Constraints (1c) similarly ensure that each terminal node either makes a prediction or has an ancestor that is a prediction node. Constraints (1d) and (1e) enable flow conservation, whereby every datapoint that flows into node n must exit to its left (or right) descendant, or flow directly to a sink t_k that corresponds to a class k. Constraints (1f) ensure that the flow value of each datapoint entering source s is at most 1. Constraints (1g) and (1h) enforce datapoints to flow to a node's left (resp. right) child if their branching feature is 0 (resp. 1). Constraints (1i) require that a datapoint must be directed to the

sink corresponding to the class that the prediction node assigns. Finally, constraints (1j) in conjunction with (1k) ensure that if we make a prediction at node n, then exactly one class is associated with its prediction. In sum, each datapoint flows into the graph and lands on exactly one of the sinks via a path from source to sink depending on its feature vector.

3.3 Fairness Constraints

One main advantage of formulation (1) is its flexible modeling power. In the following, we showcase the variety of constraints that can be added to the formulation to learn trees that satisfy the definitions of fairness we introduced in Section 1.2. We previously defined the various notions of fairness using strict equalities; in practice, however, we may relax this condition by introducing a bias δ, where δ is the maximum disparity allowed between groups.

Statistical Parity. Statistical parity is satisfied up to a bias of δ when we add the following constraint to (1):

$$\left| \frac{\sum_{n \in \mathcal{B} \cup \mathcal{T}} \sum_{i \in \mathcal{I}: p^i = p} z^i_{n,t_1}}{|\{i \in \mathcal{I}: p^i = p\}|} - \frac{\sum_{n \in \mathcal{B} \cup \mathcal{T}} \sum_{i \in \mathcal{I}: p^i = p'} z^i_{n,t_1}}{|\{i \in \mathcal{I}: p^i = p'\}|} \right| \le \delta \quad (2)$$

$$\forall p, p' \in \mathcal{P}: p \ne p',$$

where the left-hand side of the inequality is the absolute difference between the proportion of positive classes assigned by the classification tree to groups p and p', respectively.

Conditional Statistical Parity. In order to describe the CSP constraint, we will let ℓ^i denote the value of datapoint i's legitimate factor(s). To ensure that the learned tree satisfies conditional statistical parity up to a bias δ and for all $\ell \in \mathcal{L}$, we may augment (1) with the constraint

$$\left| \frac{\sum_{n \in \mathcal{B} \cup \mathcal{T}} \sum_{i \in \mathcal{I}} \mathbb{I}(p^i = p \wedge \ell^i = \ell) z^i_{n,t_1}}{|\{i \in \mathcal{I}: p^i = p \wedge \ell^i = \ell\}|} - \frac{\sum_{n \in \mathcal{B} \cup \mathcal{T}} \sum_{i \in \mathcal{I}} \mathbb{I}(p^i = p' \wedge \ell^i = \ell) z^i_{n,t_1}}{|\{i \in \mathcal{I}: p^i = p' \wedge \ell^i = \ell\}|} \right| \le \delta \quad (3)$$

$$\forall p, p' \in \mathcal{P}: p \ne p', \ell \in \mathcal{L},$$

where the left-hand side of the inequality is the absolute difference between the proportions of positive classes assigned by the classification tree to groups p and p' respectively, in the training data, and whose legitimate feature(s) equal(s) ℓ.

Equalized Odds. We may augment (1) to satisfy equalized odds up to a bias δ using the constraint

$$\left| \frac{\sum_{n \in \mathcal{B} \cup \mathcal{T}} \sum_{i \in \mathcal{I}} \mathbb{I}(p^i = p \wedge y^i = k) z^i_{n,t_1}}{|\{i \in \mathcal{I}: p^i = p \wedge y^i = k\}|} - \frac{\sum_{n \in \mathcal{B} \cup \mathcal{T}} \sum_{i \in \mathcal{I}} \mathbb{I}(p^i = p' \wedge y^i = k) z^i_{n,t_1}}{|\{i \in \mathcal{I}: p^i = p' \wedge y^i = k\}|} \right| \le \delta \quad (4)$$

$$\forall k \in \mathcal{K}, p, p' \in \mathcal{P}: p \ne p',$$

where the left-hand side of the inequality is the absolute difference between the proportions of false positive and true positive assignments made by the classification tree in groups p and p'. Note that predictive equality and equal opportunity are equivalent to setting $k = 0$ and $k = 1$ in (4), respectively.

Note that any of the above constraints can be easily linearized by decomposing them into two. In the general case where we have

constraint $|f(x)| \leq \delta$, we can reformulate it into $f(x) \leq \delta$ and $-f(x) \leq \delta$. If f is affine (as is the case here), then these constraints are linear in the decision variables of the problem and the resulting problem can be solved with powerful off-the-shelf mixed-integer *linear* optimization solvers such as Gurobi[1]. We also emphasize that the above constraints are illustrative examples of the approach's flexibility, and that they may be amended or combined with other fairness considerations. For example, instead of restricting the absolute value of the difference between groups, we may impose that the minority group should be better off by a margin δ, see Section 4.4.

4 EXPERIMENTS

We now evaluate the empirical performance of the model outlined in Section 3. We first compare the interpretability of several popular machine learning models based on three desiderata – decision complexity, simulatability, and scope – illustrating that our approach (*FairOCT*) yields one of the most interpretable models. Then, we compare our approach to a suite of methods for learning fair classifiers (both ones that are model agnostic and those that specifically learn decision trees).

4.1 Datasets

COMPAS. The Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) is a popular dataset used originally to predict a criminal's risk of recidivism after 2 years. Angwin et al. [7] published a seminal article analyzing that the algorithm deployed then was biased in favor of White criminals. We therefore let race be the sensitive attribute. The original race attribute has 6 levels, but Asian and Native American criminals are quite rare in the data, so we group them under the "Other" category in order to make it possible to obtain better estimates of fairness metrics. The dataset consists of 6,172 datapoints.

Adult. The UCI Adult dataset is taken from a 1994 Census database [22]. The goal is to predict whether or not someone's income exceeds \$50,000 per year, and we treat sex as the sensitive attribute in accordance with [1, 28, 36] with females being the marginalized group. The full dataset contains 30,162 datapoints.

German. The German dataset classifies people as having good or bad credit. We use age as the sensitive attribute, following the works of [27, 34] to split the population into people 25 and younger and older than 25, with the former as the marginalized group since younger people are often assigned worse credit under the basis of age. The dataset contains 1,000 datapoints.

All three datasets are popular benchmark datasets in the algorithmic fairness literature. We focus on these three because they each have different and societally important prediction tasks.

4.2 Benchmark Methods

Fair Decision Tree-Based Methods

Optimal and Fair Decision Trees via Regularization (RegOCT). We compare our approach to the method for learning optimal fair trees proposed by Aghaei et al. [3], which considers both disparate impact and disparate treatment. In our setting, we are only interested in disparate impact, which reduces to statistical parity in the case of binary outcomes. Our method differs from *RegOCT* in two ways:

1) *FairOCT* is formulated much more efficiently, resulting in faster solve times, and 2) *RegOCT* promotes fairness via a regularization term in the objective rather than as a constraint (like ours). Its objective function minimizes misclassification rate plus a regularization term controlled by a "fairness parameter" ρ.

Discrimination Aware Decision Trees (DADT). The second tree-based method we compare to is *DADT* proposed by Kamiran et al. [34], which consists of a two-pronged approach. The first is an in-processing step that uses sensitivity gain (IGS) and information gain (IGC) as heuristics to build a fair decision tree. If the tree still has a discrimination level greater than ϵ, a post-processing step can be used that relabels the predictions of the tree's leaf nodes until the discrimination reaches ϵ. Note that the original paper proposes to use a combination of these steps where appropriate, such as IGC+IGS, IGC-IGS, or only IGC (which is equivalent to the CART algorithm) – all of which can be combined with the relabeling step post-training ("Relab"). We compare our method with respect to IGC+IGS_Relab because Kamiran et al. [34] cited that it in general performs best, although we display results for all three heuristics in the Appendix Section A. We grow trees of maximum depth 2 and 3 for interpretability. *DADT* has two modeling limitations: it only considers statistical parity as its fairness metric, and it assumes only two sensitive groups exist. In further contrast to *FairOCT*, *DADT* subtracts the probability of the marginalized group from the dominant group. A "fair" result under this definition may include a scenario in which the marginalized group receives better outcomes at a much higher rate than the dominant group, which constraint (2) does not allow. However, since our approach is very flexible, we run experiments on *FairOCT* with this notion of disparity – refer to Appendix Section A.

Note that methods other than *DADT* and *RegOCT* in the literature either: consider individual fairness as opposed to group fairness (e.g., Ranzato et al. [45]); learn trees in an online setting (e.g., Zhang and Ntoutsi [58]); or are not interpretable (e.g., Grari et al. [28]).

Model Agnostic Fairness Methods

Apart from strictly tree-based methods, we also compare our approach to three methods – one from each umbrella of approaches: pre-processing, in-processing, and post-processing. Critically, these three methods can learn most ML model classes (such as logistic regression, random forests, kNN, etc.). This feature is important because our goal is to analyze the accuracy and discrimination trade-offs when opting to use our method (one of the most interpretable) in lieu of more complex models.

[Pre-Processing] Correlation Remover (CR). We benchmark our method against *CR*, which reduces the correlation between the data and the sensitive feature by regressing the "centered" sensitive feature with the non-sensitive features. A linear transformation using the learned coefficients is then applied, resulting in new data X_{corr}. The final transformation X_{tfm} is controlled via a fairness parameter $\alpha \in [0, 1]$:

$$X_{\mathrm{tfm}} = \alpha X_{\mathrm{corr}} + (1 - \alpha)X,$$

where $\alpha = 0$ corresponds to the original feature vector X and $\alpha = 1$ means all correlation is removed. Further detail can be found in [14] and the API documentation for FairLearn. Note that we chose

---

[1]See https://www.gurobi.com/products/gurobi-optimizer/

Model	Decision Complexity (COMPAS)		Decision Complexity (Adult)		Decision Complexity (German)		Simulatability	Scope
Full Tree (d=1)	3		3		3		High	Global
Full Tree (d=2)	7		7		7		High	Global
Full Tree (d=3)	15		15		15		High	Global
Logistic Regression	6		12		20		Medium-High	Global
Decision Tree	779		5,101		329		Medium	Global
k-Nearest Neighbors	4,629		22,621		750		Medium-Low	Local
SVM (RBF Kernel)	3,255		10,542		454		Low	Local
Multilayer Perceptron	700		1,300		2,100		Low	None
Random Forest	59,126		273,861		17,956		Low	None

Table 1: Interpretability of various machine learning models with respect to three desiderata: decision complexity (in the context of the *COMPAS*, *Adult*, and *German* datasets), simulaltability, and scope.

this method out of all other pre-processing approaches because it is lightweight and yields better performance upon testing.

[In-Processing] Exponentiated-Gradient Reduction (ExpG). Algorithm 1 in Agarwal et al. [1] finds a classifier and λ with the highest accuracy subject to a fairness constraint, where λ is a vector consisting of k Lagrangian multipliers, each corresponding to a fairness constraint in the algorithm (i.e., $\lambda \in \mathbb{R}^k_+$). Since we are interested in obtaining a breadth of accuracy-discrimination datapoints, we opted to implement their "grid search" method, which searches through a grid of λ and yields the best estimator from a given model class. In our case, we use *all* the results from the possible values of λ. This grid search method is what we will refer to as *ExpG* in our experiments. We chose to compare our method with *ExpG* because it is one of the only in-processing approaches that take in nearly any model class and also optimize for many fairness notions.

[Post-Processing] Randomized Threshold Optimizer (RTO). Lastly, we compare *FairOCT* with a method proposed by Hardt et al. [30] that applies a randomized thresholding transformation to the classifier's prediction to enforce a fairness notion. We refer to Section 3.2 of Hardt et al. [30] for a full treatment of the algorithm. Similar to *ExpG*, we chose *RTO* because it is one of the only post-processing methods that is model agnostic and can optimize for all the fairness notions we consider in this work.

4.3 Interpretability of Machine Learning Models

Table 1 provides a comparison of several popular machine learning models and our assessment of their interpretability with respect to several desiderata: model complexity (defined in Section 2), simulatability, and scope. In the following, we elucidate our judgments on the simulatability and scope of these models. We will classify models as having "High" to "Low" simulatability, and "Global", "Local", or "None" in terms of scope. Unless otherwise noted, all models except for optimal trees are trained using the standard parameters of the scikit-learn package:

- **Full Binary Trees**: For simplicity, we consider learning full binary trees up to a fixed depth d for our optimal tree methods, although in practice we may easily prune the tree via regularization. We argue that trees have a global scope and

are among the most simulatable because humans can visualize the entire decision rule and trace predictions with relative ease (especially when the trees are shallow and simple).

- **Decision Trees – CART (DT)**: Trees that are grown via a heuristic typically have a stopping point to avoid overfitting, and in our case we impose the minimum number of samples in each leaf to be 3. Since there is stochasticity in tree depth and pruning, we report decision complexity as the average over all trees grown for a particular dataset. While DT has a global scope similar to full binary trees, it is much less simulatable given how deep the trees tend to grow.

- **Logistic Regression (LR)**: Assuming full linearity, logistic regression has global scope because one can wholly observe its decision rule via its coefficients. However, we argue that logistic regression has medium simulatability because the coefficients need to be calculated and contextualized for full interpretability.

- **k-Nearest Neighbors (kNN)**: While kNN's simulatability can be high – especially in lower dimensions where the decision space can be visualized ($F \leq 3$) – kNN's are often not simulatable in practice. Without visualizing the entire feature space, we can only observe the local behavior of the model, i.e., which k points are nearest to our new datapoint.

- **Support Vector Machines (SVM), RBF Kernel**: Due to stochasticity, we report the average number of support vectors in our experiments for SVM's decision complexity. We argue that SVM's have low simulatability, especially when $F > 2$ and the decision space is hard to visualize. We can, however, classify SVM's as having local scope because the intuition is similar to kNN – support vectors closer to the new datapoint will have much higher weights.

- **Random Forest (RF)**: We grow 100 trees via the CART algorithm to build the random forest model, and argue that simulatability is low because a human cannot keep track of the complicated decisions that are aggregated over numerous trees. The intricacy of the random forest also means that it is not interpretable neither globally or locally.

- **Multilayer Perceptron (MLP)**: A multilayer perceptron is a neural network. We train MLPs with 1 hidden layer consisting of 100 nodes, while the input layer has the number

Algorithm	Fairness Definitions	# Sensitive Levels	Disparity	Model	Fairness Parameter	Time Limit
FairOCT	SP, CSP, PE, EOpp, EOdds	Any	absmax	Full Tree ($d \in \{1, 2, 3\}$)	$\delta \in [0.01, 0.55]$ $\Delta = 0.01$	3 hrs
RegOCT	SP	Any	absmax	Full Tree ($d \in \{1, 2, 3\}$)	$\rho \in [0, 1]$ $\Delta = 0.02$	3 hrs
DADT	SP	2	dom-marg	DT	$\epsilon \in [0.01, 0.55]$ $\Delta = 0.01$	N/A
CR	SP, PE, EOpp, EOdds	Any	absmax	DT, LR, kNN, SVM, RF, MLP	$\alpha \in [0, 1]$ $\Delta = 0.1$	N/A
ExpG	SP, PE, EOpp, EOdds	Any	absmax	DT, LR, kNN, SVM, RF, MLP	# of $\lambda : 100$	N/A
RTO	SP, PE, EOpp, EOdds	Any	absmax	DT, LR, kNN, SVM, RF, MLP	N/A	N/A

Table 2: List of algorithms run on *COMPAS*, *Adult*, and *German* datasets. Each algorithm can accommodate different fairness definitions: statistical parity (SP), conditional statistical parity (CSP), predictive equality (PE), equal opportunity (EOpp), and equalized odds (EOdds). They also take varying number of sensitive levels and disparity measures, where "absmax" corresponds to the absolute value of the maximum pairwise disparity between sensitive groups and "dom-marg" is the difference in positive classification rate between the dominant group and the marginalized group. Different machine learning models are also considered: full binary trees with branching depths $d \in \{1, 2, 3\}$, decision trees grown via CART (DT), logistic regression (LR), k-nearest neighbors (kNN), support vector machine with RBF kernel (SVM), random forest (RF), and multilayer perceptron (MLP). Our experiments also varied the fairness parameters for all methods, whose role is outlined in detail in Section 4.2. Δ denotes the increments taken in each of these fairness parameters.

of nodes equal to the number of features in the dataset. Its decision complexity, therefore, is equal to $100 \times F + 100$. Since the prediction relies on potentially thousands of matrix calculations, MLPs have very low simulatability and have neither global nor local interpretability.

4.4 Methodology

We compare the performance of *FairOCT* with the benchmark methods outlined in Section 4.2 on all three datasets: *COMPAS*, *German* and *Adult*. Where relevant and possible, we incorporate various fairness definitions in all our experiments. All datasets are tested on statistical parity, predictive equality, equal opportunity, and equalized odds, and *COMPAS* additionally considers conditional statistical parity (conditioned on the number of prior crimes). However, note that not all methods can accommodate all these notions. For instance, only our method can incorporate conditional statistical parity. On the other hand, *RegOCT* and *DADT* can only consider statistical parity. *DADT* additionally only takes in two sensitive levels while all the other methods are robust to more than two sensitive levels. Therefore, in our experiments on *COMPAS*, *DADT* only considers binary race – "Black" vs. "non-Black" – which yields results that are not fully comparable to other methods.

A full summary of our experiments for each method, including capabilities, ML models trained, and fairness parameters tested is provided in Table 2. We consider fairness parameters in increments of Δ and display results for all these values in order to fully observe the accuracy-discrimination trade-off. We use 75% of the data for training and 25% for testing on all three datasets outlined in Section 4.1 – split five times with different seeds to account for random variability – and evaluate model performance on the test set (i.e., out-of-sample or OOS set). The *Adult* dataset is both large and high-dimensional, so we train *FlowOCT* and *RegOCT* on 2,700 datapoints due to computational limitations (while evaluating on the same test set as other methods to allow for an equivalent comparison); we will further discuss this choice in Section 4.5. All experiments have a time limit of 3 hours, and utilized four Intel Xeon E5-2640 v4, 2.40GHz CPUs, each having 4GB of memory.

4.5 Results

Comparison of Fairness Methods. Figure 2 plots the accuracy and discrimination trade-off for all experiments outlined in Table 2. In general, *FairOCT* trained on all depths ($d = \{1, 2, 3\}$) obtain similar performances, though this is mainly attributed to the experiments on $d = 3$ not solving to optimality within the time limit. Trends vary across datasets; for instance, *FairOCT* found decisions with accuracies between $[0.55, 0.68]$ with disparities between $[0, 0.25]$ for *COMPAS*. In contrast, its results for *Adult* varied only by at most 5 percentage points (p.p.) in accuracy and 10 p.p. in disparity. Nonetheless, *FairOCT* learns a range of heterogeneous performances, which allows us to choose between various accuracy-discrimination trade-offs.

Overall, for any given disparity threshold, *FairOCT* consistently outperforms *DADT* within the notion of statistical parity (recall that *DADT* is only trained on this notion). This is expected since *FairOCT* finds an optimal solution whereas *DADT* relies on a heuristic. A
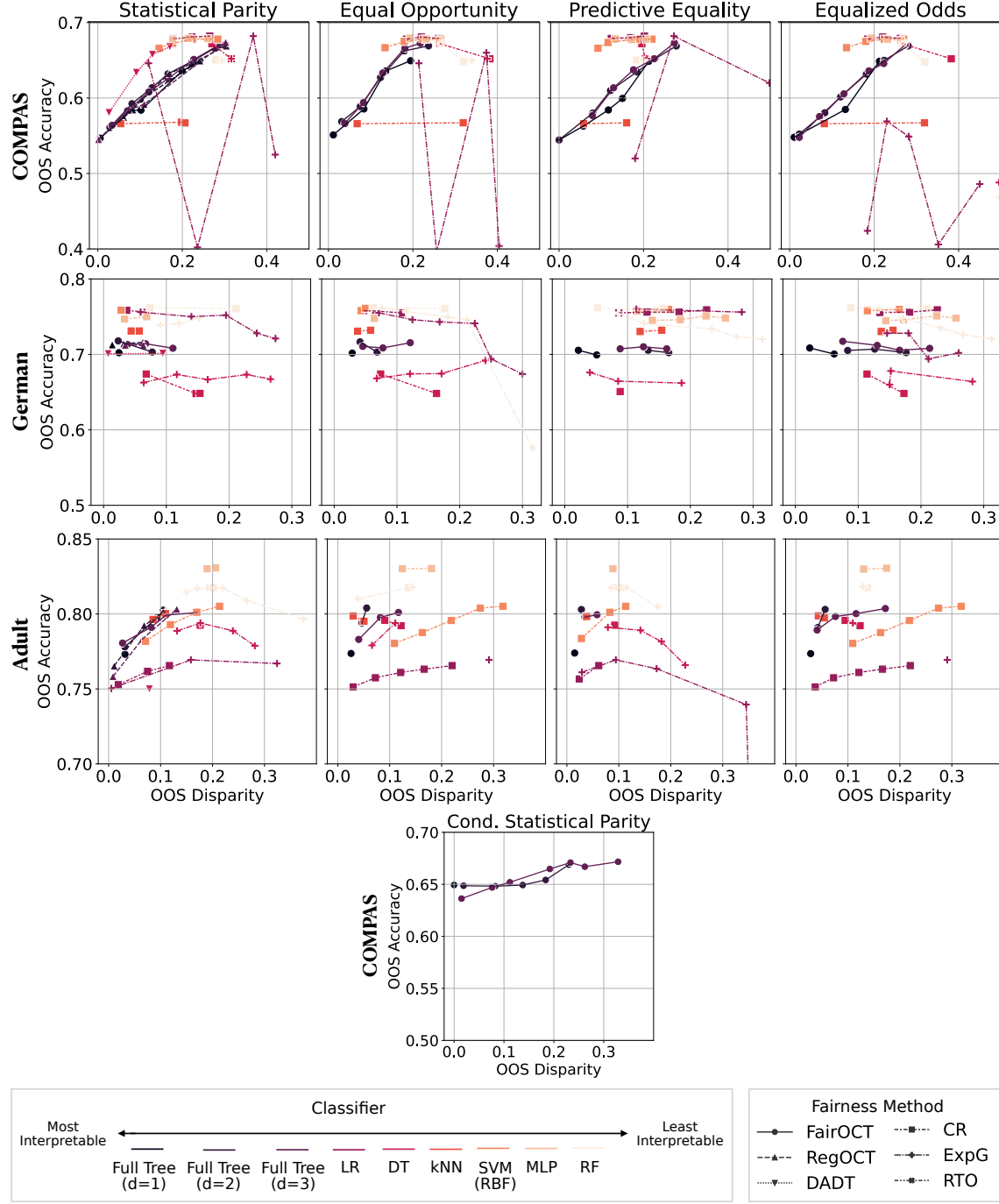
Figure 2: Accuracy and discrimination of *FairOCT* on the *COMPAS* (top), *Adult* (middle), and *German* (bottom) datasets with varying fairness metrics (from left to right – statistical parity equal opportunity, predictive equality, and equalized odds, and at the bottom, conditional statistical parity), averaged for each fairness parameter over 5 random samples. Each datapoint in the graph corresponds to an average over 0.05 increments of disparity (i.e., for every method and model, each datapoint is averaged from [0, 0.05], [0.05, 0.1), and so on). Classifiers are ordered from darkest (most interpretable) to lightest (least interpretable) according to our interpretability desiderata outlined in Table 1. Different fairness methods are outlined by different marker and line styles. For ease of visualization, we average the results for *DADT* with trees of maximum depth 2 and 3.

notable aberration is *DADT*'s superior performance to *FairOCT* on *COMPAS*, but this is because *DADT* only considers two levels of race while all other experiments have four levels. Even when only considering two levels of race, *DADT* does not result in significant improvements in performance. Refer to Appendix Section A for a comparison of *FairOCT* and *DADT* within the same learning setting, which shows that *FairOCT* consistently outperforms *DADT*.

In contrast, the other tree-based benchmark method – *RegOCT* – performs at around the same level as *FairOCT* (again only on statistical parity because *RegOCT* only considers this notion). This result is unsurprising given that *RegOCT* similarly learns optimal trees, so its performance will be as good as *FairOCT*, which already finds the best-performing partition for a fixed maximum depth. However, we emphasize that *RegOCT* is much less flexible in considering other notions of fairness. Moreover, *FairOCT* benefits from a stronger formulation and is faster by at least an order-of-magnitude. Refer to Appendix Table 3 for a full comparison of computational times.

With regard to non-tree-based methods, *CR* in general yields smooth and heterogeneous results throughout the disparity axis. We observe the same trend for *ExpG*, with the exception of the *COMPAS* dataset, whose performance jumps widely. [1] mentioned that as the number of sensitive levels (and therefore the number of constraints) increase, *ExpG* is not recommended because the search space grows exponentially. This limitation may be circumvented by testing an arbitrarily large number of fairness parameters and ignoring several outliers to smooth the curve. Note also that most of the parameters tested led to a disparity level of 1.0 or near 1.0, but we decided to truncate the results to a more reasonable level for all datasets. Finally, *RTO* is mainly missing from Figure 2 because we cannot tune any fairness parameters – resulting in a single datapoint, none of which is displayed because the performance hovers at around 0.6-0.9 disparity.

In comparison, *FairOCT* consistently finds results in the lower end of the disparity axis (which is the space we care most given the task of promoting full parity). For instance, in the *COMPAS* dataset on all the fairness notions, *FairOCT* resulted in disparities ranging from 0 to 0.3 – no other method boasts this range. Even in the *Adult* and *German* datasets where that range is attenuated, no other fairness-promoting method consistently finds datapoints with low disparity (i.e., < 0.05). This result likely arises from our method taking in a hyperparemeter δ that upper bounds resulting disparity. The optimization problem is then forced to find the highest-accuracy decision rule given a (potentially low) tolerance on disparity. This is in contrast to all other methods; RegOCT uses a regularization term, CR tunes the extent of correlation removed, and ExpG passes numerous values of λ, all of which do not guarantee certain disparity results.

Finally, we note that the *FairOCT* results trained on the 2,700 datapoints of *Adult* have comparable performances with other baseline methods, indicating that while optimization-based methods like *FairOCT* may run into computational problems, training on a smaller, representative dataset yields predictions that generalize very well to the entire population. In the bottom graph of Figure 2, we also conduct experiments where *FairOCT* considers conditional statistical parity on *COMPAS*; no other method we compare to can equivalently compare this fairness notion, highlighting our approach's flexible modeling power.

The Price of Interpretability. As expected, more interpretable models like full trees perform (marginally) worse than more complex models like random forests and MLP's. This difference denotes the price of interpretability. For instance, in the *German* dataset, a random forest trained on *CR* has on average 6 percentage points higher accuracy than a full tree of depth 2 trained on *FairOCT*, across all discrimination thresholds. However, as mentioned previously, many fairness-promoting methods fail to reach full or near perfect parity in contrast to FairOCT, rendering this aggregated comparison less meaningful. Notably, in the *COMPAS* dataset, a neural net trained on CR might perform on average 3 percentage points better than a *FairOCT* ($d = 2$), but only within the disparity ranges for which both methods have results. We must further consider the other datapoints *FairOCT* ($d = 2$) yields for disparities less than 0.15, of which have no equivalent comparison with the neural net. Nonetheless, given a fixed disparity threshold, the best performing complex model performs on average 4.2 p.p. better in terms of OOS accuracy than *FairOCT* over the range of disparities for which both models have results.

5 CONCLUSION

In this work, we presented an MIO formulation for learning optimal classification trees that can be modeled to consider a variety of algorithmic fairness notions. We also propose a new measure of interpretability named *decision complexity* in order to compare our interpretable method with other classes of models. In doing so, we conduct one of the first experiments that analyze in-depth the trade-offs between interpretability, fairness, and predictive accuracy.

Our experiments show that while we observe a (often small) price of interpretability with trees of shallow depth, one must ultimately consider not only the trade-offs between interpretability, accuracy, and discrimination, but also the various fairness-promoting methods that may yield vastly different results (e.g., with regard to the range of disparities found, how different methods might be better for certain constraints, etc.). In reality, decision makers face the incredibly hard task of balancing these trade-offs given the application at hand. Our work attempts to enrich these considerations in the hopes of guiding practitioners when they make these difficult decisions.

REFERENCES

[1] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. 2018. A reductions approach to fair classification. In *Proceedings of the 35th International Conference on Machine Learning (Proceedings of Machine Learning Research)*. PMLR, Stockholm, Sweden, 60–69. https://proceedings.mlr.press/v80/agarwal18a.html

[2] Sushant Agarwal. 2021. Trade-offs between fairness and interpretability in machine learning. In *IJCAI 2021 Workshop on AI for Social Good*. IJCAI, Virtual, 1–6.

[3] Sina Aghaei, Mohammad Javad Azizi, and Phebe Vayanos. 2019. Learning optimal and fair decision trees for non-discriminative decision-making. In *33rd AAAI Conference on Artificial Intelligence*, Vol. 33. AAAI, Honolulu, USA, 1418–1426.

[4] Sina Aghaei, Andrés Gómez, and Phebe Vayanos. 2021. Strong Optimal Classification Trees.

[5] Hirotugu Akaike. 1974. A new look at the statistical model identification. *IEEE transactions on automatic control* 19, 6 (1974), 716–723.

[6] Rafael Alcalá, Jesús Alcalá-Fdez, Jorge Casillas, Oscar Cordón, and Francisco Herrera. 2006. Hybrid learning models to get the interpretability–accuracy trade-off in fuzzy modeling. *Soft Computing* 10 (2006), 717–734.

[7] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. 2016. Machine bias. https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

[8] Los Angeles Homeless Services Authority. 2018. Report and Recommendations of the Ad Hoc Committee on Black People Experiencing Homelessness. https://www.lahsa.org/documents?id=2823-report-and-recommendations-of-the-ad-hoc-committee-on-black-people-experiencing-homelessness. Accessed: 2023-03-24.

[9] Mohammad Javad Azizi, Phebe Vayanos, Bryan Wilder, Eric Rice, and Milind Tambe. 2018. Designing fair, efficient, and interpretable policies for prioritizing homeless youth for housing resources. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 10848 LNCS (2018), 35–51. https://doi.org/10.1007/978-3-319-93031-2_3

[10] George Baryannis, Samir Dani, and Grigoris Antoniou. 2019. Predicting supply chain risks using machine learning: The trade-off between performance and interpretability. *Future Generation Computer Systems* 101 (2019), 993–1004.

[11] Richard Berk, Hoda Heidari, Shahin Jabbari, Matthew Joseph, Michael Kearns, Jamie Morgenstern, Seth Neel, and Aaron Roth. 2017. A convex framework for fair regression. *arXiv preprint arXiv:1706.02409*.

[12] Dimitris Bertsimas and Jack Dunn. 2017. Optimal classification trees. *Machine Learning* 106, 7 (2017), 1039–1082.

[13] Dimitris Bertsimas, Jack Dunn, Colin Pawlowski, and Ying Daisy Zhuo. 2019. Robust classification. *INFORMS Journal on Optimization* 1, 1 (2019), 2–34.

[14] Sarah Bird, Miro Dudík, Richard Edgar, Brandon Horn, Roman Lutz, Vanessa Milan, Mehrnoosh Sameki, Hanna Wallach, and Kathleen Walker. 2020. *Fairlearn: A toolkit for assessing and improving fairness in AI*. Technical Report MSR-TR-2020-32. Microsoft. https://www.microsoft.com/en-us/research/publication/fairlearn-a-toolkit-for-assessing-and-improving-fairness-in-ai/

[15] Leo Breiman, Jerome H Friedman, Richard A Olshen, and Charles J Stone. 1984. *Classification and regression trees*. Routledge, Boca Raton, USA.

[16] Toon Calders and Sicco Verwer. 2010. Three naive bayes approaches for discrimination-free classification. *Data mining and knowledge discovery* 21, 2 (2010), 277–292.

[17] Jiahao Chen, Nathan Kallus, Xiaojie Mao, Geoffry Svacha, and Madeleine Udell. 2019. Fairness under unawareness: Assessing disparity when protected class is unobserved. In *Proceedings of the 2019 Conference on Fairness, Accountability, and Transparency*. Association for Computing Machinery, Atlanta, USA, 339–348.

[18] Alexandra Chouldechova. 2017. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data* 5, 2 (2017), 153–163.

[19] Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. 2017. Algorithmic decision making and the cost of fairness. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Vol. Part F129685. ACM, Halifax, Canada, 797–806. https://doi.org/10.1145/3097983.3098095

[20] Christophe Denis, Romuald Elie, Mohamed Hebiri, and François Hu. 2021. Fairness guarantee in multi-class classification.

[21] Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

[22] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml

[23] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness Through Awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference* (Cambridge, Massachusetts) *(ITCS '12)*. ACM, New York, NY, USA, 214–226. https://doi.org/10.1145/2090236.2090255

[24] Gintare Karolina Dziugaite, Shai Ben-David, and Daniel M Roy. 2020. Enforcing interpretability and its statistical impacts: Trade-offs between accuracy and interpretability. *arXiv preprint arXiv:2010.13764*.

[25] Adam Elmachtoub, Jason Cheuk Nam Liang, and Ryan McNellis. 2020. Decision trees for decision-making under the predict-then-optimize framework. In *37th International Conference on Machine Learning*. PMLR, ICML, Vienna, Austria, 2858–2867.

[26] Meherwar Fatima, Maruf Pasha, et al. 2017. Survey of machine learning algorithms for disease diagnostic. *Journal of Intelligent Learning Systems and Applications* 9, 01 (2017), 1.

[27] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. 2015. Certifying and removing disparate impact. In *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, Sydney, Australia, 259–268.

[28] Vincent Grari, Boris Ruf, Sylvain Lamprier, and Marcin Detyniecki. 2019. Fair adversarial gradient tree boosting. In *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, IEEE, Beijing, China, 1060–1065.

[29] Sara Hajian and Josep Domingo-Ferrer. 2012. A methodology for direct and indirect discrimination prevention in data mining. *IEEE transactions on knowledge and data engineering* 25, 7 (2012), 1445–1459.

[30] Moritz Hardt, Eric Price, and Nathan Srebro. 2016. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems*. NeurIPS, Barcelona, Spain, 3315–3323.

[31] Nathanael Jo, Sina Aghaei, Andrés Gómez, and Phebe Vayanos. 2021. Learning optimal prescriptive trees from observational data.

[32] Ulf Johansson, Cecilia Sönströd, Ulf Norinder, and Henrik Boström. 2011. Trade-off between accuracy and interpretability for predictive in silico modeling. *Future medicinal chemistry* 3, 6 (2011), 647–663.

[33] Nathan Justin, Sina Aghaei, Andres Gomez, and Phebe Vayanos. 2021. Optimal Robust Classification Trees. In *The AAAI-22 Workshop on Adversarial Machine Learning and Beyond*. AAAI Press, Vancouver, Canada.

[34] Faisal Kamiran, Toon Calders, and Mykola Pechenizkiy. 2010. Discrimination aware decision tree learning. In *Proceedings - IEEE International Conference on Data Mining, ICDM*. IEEE, Sydney, Australia, 869–874. https://doi.org/10.1109/ICDM.2010.50

[35] Faisal Kamiran, Indrė Žliobaitė, and Toon Calders. 2013. Quantifying explainable discrimination and removing illegal discrimination in automated decision making. *Knowledge and information systems* 35, 3 (2013), 613–644.

[36] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. 2012. Fairness-aware classifier with prejudice remover regularizer. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, ECML PKDD, Prague, Czech Republic, 35–50.

[37] Kentaro Kanamori and Hiroki Arimura. 2019. Fairness-aware Edit of Thresholds in a Learned Decision Tree Using a Mixed Integer Programming Formulation. In *33rd Annual Conference of the Japanese Society for Artificial Intelligence, (2019)*. Japanese Society for Artificial Intelligence, Japanese Society for Artificial Intelligence, Niigata City, Japan, 3Rin211–3Rin211.

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

[39] Alan Miller. 2002. *Subset selection in regression*. Chapman and Hall/CRC, United Kingdom.

[40] Velibor V Mišić. 2020. Optimization of tree ensembles. *Operations Research* 68, 5 (2020), 1605–1624.

[41] Christoph Molnar. 2022. *Interpretable Machine Learning* (2 ed.). Github, Online. https://christophm.github.io/interpretable-ml-book

[42] W James Murdoch, Chandan Singh, Karl Kumbier, Reza Abbasi-Asl, and Bin Yu. 2019. Definitions, methods, and applications in interpretable machine learning. *Proceedings of the National Academy of Sciences* 116, 44 (2019), 22071–22080.

[43] Dino Pedreshi, Salvatore Ruggieri, and Franco Turini. 2008. Discrimination-aware data mining. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, Las vegas, USA, 560–568.

[44] Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q. Weinberger. 2017. On fairness and calibration. In *31st International Conference on Neural Information Processing Systems*, Vol. 2017-December. NeurIPS, Long Beach, USA, 5684—5693.

[45] Francesco Ranzato, Caterina Urban, and Marco Zanella. 2021. Fair Training of Decision Tree Classifiers.

[46] John Rawls. 1971. *A theory of justice*. Harvard University Press, Cambridge, Massachusetts.

[47] John E Roemer and Alain Trannoy. 2000. *Equality of opportunity*. Harvard University Press, Cambridge, Massachusetts.

[48] Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence* 1, 5 (2019), 206–215.

[49] Cynthia Rudin, Chaofan Chen, Zhi Chen, Haiyang Huang, Lesia Semenova, and Chudi Zhong. 2022. Interpretable machine learning: Fundamental principles and 10 grand challenges. *Statistics Surveys* 16 (2022), 1–85.

[50] Gideon Schwarz. 1978. Estimating the dimension of a model. *The annals of statistics* 6, 2 (1978), 461–464.

[51] Lesia Semenova, Cynthia Rudin, and Ronald Parr. 2022. On the existence of simpler machine learning models. In *2022 ACM Conference on Fairness, Accountability, and Transparency*. ACM, Seoul, South Korea, 1827–1858.

[52] Robert Tibshirani. 1996. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)* 58, 1 (1996), 267–288.

[53] Sicco Verwer and Yingqian Zhang. 2019. Learning optimal classification trees using a binary linear program formulation. In *33rd AAAI Conference on Artificial*

Intelligence, Vol. 33. Association for the Advancement of Artificial Intelligence, Honolulu, USA, 1625–1632.

[54] Caroline Wang, Bin Han, Bhrij Patel, and Cynthia Rudin. 2022. In pursuit of interpretable, fair and accurate machine learning for criminal recidivism prediction. *Journal of Quantitative Criminology* 39, 2 (2022), 1–63.

[55] Yaqian You, Jianbin Sun, Yu Guo, Yuejin Tan, and Jiang Jiang. 2022. Interpretability and accuracy trade-off in the modeling of belief rule-based systems. *Knowledge-Based Systems* 236 (2022), 107491.

[56] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. 2013. Learning fair representations. In *30th International Conference on Machine Learning*.

PMLR, ICML, Atlanta, USA, 325–333.

[57] Jiang Zhang, Ivan Beschastnikh, Sergey Mechtaev, and Abhik Roychoudhury. 2020. Fairness-guided SMT-based Rectification of Decision Trees and Random Forests.

[58] Wenbin Zhang and Eirini Ntoutsi. 2019. FaHT: An adaptive fairness-aware decision tree classifier. In *IJCAI International Joint Conference on Artificial Intelligence*, Vol. 2019-August. IJCAI, Macao, China, 1480–1486. https://doi.org/10.24963/ijcai.2019/205

[59] Indrė Žliobaitė. 2017. Measuring discrimination in algorithmic decision making. *Data Mining and Knowledge Discovery* 31, 4 (2017), 1060–1089.