Neural-Informed Decision Trees: A Fair, Interpretable and Expressive Tree Model

Georgia Perakis
Massachusetts Institute of Technology
Cambridge, MA, USA
georgiap@mit.edu

Asterios Tsiourvas
Massachusetts Institute of Technology
Cambridge, MA, USA
atsiour@mit.edu

ABSTRACT

We study the problem of creating a highly expressive, interpretable, and simultaneously fair machine learning model. We propose neural-informed decision trees (NIDTs), a fair model that combines the predictive power of neural networks with the inherent interpretability of decision trees. NIDTs perform axis-aligned splits on the features of the dataset to create an interpretable decision path, and at each leaf, use a linear predictor that uses both the features as well as the embeddings coming from a task-specific neural network to capture non-linearities in the data. To generate NIDTs we propose a decomposition training scheme. The proposed training method enables the direct integration of fairness constraints by solving a constrained convex optimization problem at each leaf, resulting in a certified fair model. We evaluate NIDTs on 15 publicly available datasets, where we show that NIDTs outperform multiple interpretable tree-based models, as well as the neural network that informs them. We also show the interpretable aspects of the method by extracting a drug-dosage prescription policy using a real-world dataset. Finally, we demonstrate the fairness of NIDTs on a real-world dataset by directly incorporating fairness constraints into the model, resulting in a certified fair model that eliminates gender bias in prediction.

CCS CONCEPTS

• **Computing methodologies → Machine learning**; **Machine learning approaches**; **Classification and regression trees**.

KEYWORDS

Neural Networks, Decision Trees, Fair ML, Interpretability

ACM Reference Format:
Georgia Perakis and Asterios Tsiourvas. 2023. Neural-Informed Decision Trees: A Fair, Interpretable and Expressive Tree Model. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '23).* ACM, New York, NY, USA, 8 pages.

1 INTRODUCTION

In recent years, practitioners and academics have been increasingly incorporating machine learning models to inform critical decision-making processes in various fields such as healthcare [7], justice [17], and education [9] among others. Although machine learning has greatly transformed critical decision-making processes, the deployment of machine learning models in real-world settings has exposed unexpected defects. Examples include fairness issues [22] where models propagate biases in existing data, privacy issues [20] where models use sensitive personal data and causality issues [24] where models cannot distinguish causal effects from correlations.

Interpretability [12] can mitigate the aforementioned challenges as it provides a clear understanding of the internal workings of the model and helps identify potential biases and errors. One way of achieving interpretability is to train inherently interpretable models such as decision trees [5] and sets [18] that allow for transparent inspection. However, such approaches typically sacrifice performance. On the other hand, one can deploy a highly predictive complex model, and then obtain a local explanation [25] for its predictions. Unfortunately, the explanation may not provide an understanding of the model's overall behavior as local explanations provide insight into specific predictions. In contrast, our approach aims to combine the best of both worlds by introducing an interpretable tree-based model that is informed by a highly predictive neural network.

Although interpretability provides a thorough understanding of the model's decision-making process, it does not suggest concrete actions for promoting fairness. To tackle substantial concerns regarding fairness in predictive models, several studies have concentrated on integrating fairness constraints [11]. However, given the complexity of most models, these constraints are usually implemented through proxies and therefore, fairness in prediction is not guaranteed. In contrast, our method proposes an alternative solution that allows the direct integration of fairness constraints on the training process, eliminating the need for proxies [28].

1.1 Contributions

In this work, we introduce NIDTs, a fair model that combines the predictive power of neural networks with the interpretability of decision trees. Our contributions encompass several key aspects.

(1) We propose a novel predictive model that has the structure of an axis-aligned decision tree with linear predictors at each leaf. NIDTs showcase enhanced predictive performance as they incorporate the embeddings obtained from a task-specific neural network into their linear predictors.
(2) We introduce an efficient decomposition training scheme that builds upon the CART training algorithm. We show analytically the existence of a NIDT that performs at least as well as the informing network, while we also provide an approximation bound for the performance of NIDTs.
(3) We demonstrate how fairness constraints can be directly incorporated, in the form of domain knowledge constraints, into NIDTs by solving a constrained convex optimization problem at each leaf, resulting in a certified fair model.
(4) Finally, to validate the previous claims, we conduct an experimental evaluation of NIDTs using multiple real-world datasets, where we verify the predictive power, the interpretability and the fairness of NIDTs.

KDD '23, August 6-10, 2023, Long Beach, CA, USA

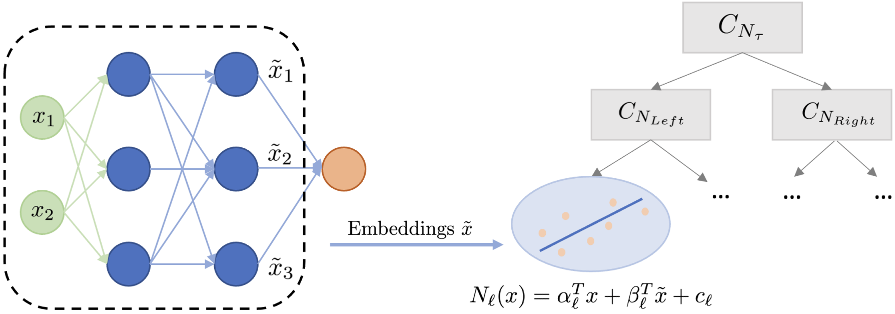

Figure 1: (Left) A neural network f_θ, trained on the problem at hand, generates new expressive features (embeddings $\tilde{x}$) that are *linearly* combined at the output layer to produce the network's final prediction. (Right) The embeddings $\tilde{x}$, generated by the network's last hidden layer, are used along with the initial features x by the linear model at each leaf of the NIDT τ.

2 RELATED LITERATURE

There has been much interest in learning interpretable models, especially decision trees [5], and in combining them with expressive machine learning models for enhanced predictive performance. [14] popularized the concept of knowledge distillation where first a neural network is trained and then a soft decision tree is trained on a dataset whose labels are produced by the network. Since then multiple works have focused on deriving decision trees from particular model families, like random forests [10]. Significant research has also been on model-agnostic approaches where decision trees are constructed to resemble black-box models. [6] proposed born-again trees, while [2] proposed an active sampling-based method to extract global explanations. The aforementioned works focus on creating decision trees that resemble complex predictive models. On the other hand, our model does not seek to resemble a neural network but leverages the network's embeddings to *inform* its predictions. Other methodologies for creating expressive decision trees include MIO [4] and using non-trivial predictors [21]. While constructing such models can be computationally demanding, our approach is tractable as the proposed training scheme builds upon the computationally efficient CART algorithm.

In the field of fair machine learning, bias mitigation algorithms are recognized as one of the most prominent approaches to address ethical issues in high-stakes automated decision making. These algorithms are categorized into pre-processing, in-processing, and post-processing approaches [26]. In the first case [16], the bias mitigation occurs during training. This is usually achieved by minimizing the empirical risk regularized by a fairness metric surrogate. However, using regularized fairness surrogates requires careful hyperparameter selection and lacks explicit guarantees regarding constraint satisfaction [3]. In contrast, we propose a certified fair model that integrates, without the use of surrogates, fairness constraints directly into training. In this work, we focus on notions of group fairness and specifically, demographic parity.

3 PROBLEM FORMULATION

NIDTs, which we denote as τ, are built upon axis-aligned decision trees [5] with linear predictors at each leaf [29]. We denote the input domain as $\mathcal{X} \subseteq \mathbb{R}^d$ and the output space as $\mathcal{Y} \subseteq \mathbb{R}$. An axis-aligned split is defined by the constraint $C = (x_i \leq c)$, where x_i is the i-th coordinate of $x \in \mathcal{X}$ with $i \in [d] := \{1, \dots, d\}$ and $c \in \mathbb{R}$. The feasible set of an axis-aligned constraint C is defined as $\mathcal{F}(C) = \{x \in \mathcal{X} | \ x \text{ satisfies } C\}$. A NIDT consists of three node categories: root, internal, and leaf. Following the notation of [2], we define N_τ to be the root node of τ and $N = (N_{Left}, N_{Right}, C)$ to be an internal node, with N_{Left} the left child node of N, N_{Right} the right child node of N and C the corresponding axis-aligned constraint of N. Finally, we define N_ℓ to be the ℓ-th leaf node of τ.

In a NIDT, predictions within each leaf node come from a linear model that uses both the features of the dataset as well as features generated by the hidden layers of a task-specific neural network. This approach is motivated by the following observation. Neural networks utilize their hidden layers to generate new expressive features, known as *embeddings* (symbolically $\tilde{x}$), that are combined *linearly* at the output layer to generate the network's prediction. The strong performance typically exhibited by neural networks can be attributed to their ability to capture complex relationships within the data through embeddings. By adding the embeddings into the linear predictor of each leaf, we expect the NIDT to exhibit enhanced predictive performance, similar to the performance achieved by the embedding-generating network. Furthermore, by allowing NIDT to perform axis-aligned splits only on the initial features of the dataset, the path to each leaf remains interpretable, like a typical decision tree. The proposed framework is depicted in Figure 1.

Formally, we denote a NIDT as the function $\tau : \mathcal{X} \times \tilde{\mathcal{X}} \to \mathcal{Y}$, where $\tilde{\mathcal{X}} \subseteq \mathbb{R}^{\tilde{d}}$ is the domain of the embeddings. Specifically, the ℓ-th leaf node N_ℓ can be interpreted as the function $N_\ell(x, \tilde{x}) = \alpha_\ell^T x + \beta_\ell^T \tilde{x} + c_\ell$, where $\alpha_\ell \in \mathbb{R}^d$, $\beta_\ell \in \mathbb{R}^{\tilde{d}}$ and $c_\ell \in \mathbb{R}$. An internal node $N = (N_{Left}, N_{Right}, C)$ can be interpreted as $N(x, \tilde{x}) = N_{Left}(x, \tilde{x})$, if $x \in \mathcal{F}(C)$ and $N(x, \tilde{x}) = N_{Right}(x, \tilde{x})$ otherwise. Based on the previous definition, $\tau(x, \tilde{x}) = N_\tau(x, \tilde{x})$. For a given node N, we denote the conjunction of constraints from the root to N as C_N. For an internal node $N = (N_{Left}, N_{Right}, C)$ we have $C_{N_{Left}} = C_N \wedge C$ and $C_{N_{Right}} = C_N \wedge (\neg C)$, while for the root we have $C_{N_\tau} = True$.

We also denote a neural network as the function $f_\theta : \mathcal{X} \to \mathcal{Y}$, where θ is the set of the trainable parameters. We consider the densely connected architecture, where each neuron receives as inputs the outputs of the previous layer. Formally, $f_\theta(x) = W_K x_K + b_K, \ x_k = \sigma(W_{k-1} x_{k-1} + b_{k-1}), \ k = 1, \dots, K$. In this definition, K is the number of the hidden layers, $\sigma : \mathbb{R} \to \mathbb{R}$ is the activation function, $W_k \in \mathbb{R}^{n_k \times n_{k-1}}$ is the weight matrix, $b_k \in \mathbb{R}^{n_k}$ the bias vector and n_k the number of neurons at layer k. For ease of notation, we define $x_0 = x$ and $n_0 = d$. We also define the output of the k-th layer as $f_\theta^k(x) \in \mathbb{R}^{n_k}$. In this notation, in leaf ℓ for $x \in \mathcal{F}(C_\ell)$, we have $N_\ell(x, \tilde{x}) = \alpha_\ell^T x + \beta_\ell^T \tilde{x} + c_\ell = \alpha_\ell^T x + \beta_\ell^T f_\theta^K(x) + c_\ell$ with $\tilde{d} = n_K$ and $f_\theta^K(x) = \tilde{x}$. In the rest of the paper, we focus on the regression task, where $\mathcal{X} = \mathbb{R}^d$ and $\mathcal{Y} = \mathbb{R}$.

3.1 NIDT Generation Algorithm

By denoting as $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_{\geq 0}$ the convex loss function used for regression, the nominal optimization problem for training NIDTs is

$$\min_{\theta,\tau} \sum_{(x,y)\in\mathcal{D}} \mathcal{L}(f_\theta(x), y) + \mathcal{L}(\tau(x, f_\theta^K(x)), y), \tag{1}$$

where the sum is taken over all pairs of features and labels (x, y) in the dataset $\mathcal{D}$. To solve problem (1) we propose the following decomposition approach. First, a neural network f_θ is trained by minimizing the first part of (1), i.e. $\min_\theta \sum_{(x,y)\in\mathcal{D}} \mathcal{L}(f_\theta(x), y)$, using SGD. Then, a NIDT that uses the embeddings generated by the trained f_θ is constructed to minimize $\min_\tau \sum_{(x,y)\in\mathcal{D}} \mathcal{L}(\tau(x, f_\theta^K(x)), y)$. Given that the network is already trained, we can augment the initial dataset with the generated embeddings by adding $\tilde{x} = f_\theta^K(x)$ to $\mathcal{D}$. Finally, the NIDT can be obtained efficiently by training a decision tree with linear predictors on the augmented dataset using the CART algorithm. To obtain an interpretable model we impose that the axis-aligned splits should be conducted solely upon the original features. The procedure is described in Algorithm 1.

Algorithm 1 Decomposition Training Scheme for (1)

```
1: Input: Training set $\mathcal{D}$, convex loss function $\mathcal{L}$, convex regularizer function $\mathcal{R}$.
2: Initialize neural network $f_\theta$, where $\theta$ is the set of trainable parameters.
3: Calculate $\theta^* = \arg\min_\theta \sum_{(x,y)\in\mathcal{D}} \mathcal{L}(f_\theta(x), y)$ using SGD.
4: Construct the augmented training dataset $\tilde{\mathcal{D}} = \{(x, \tilde{x}, y) | \forall (x, y) \in \mathcal{D}, \tilde{x} = f_{\theta^*}^K(x)\}$.
5: Construct the NIDT $\tau$ using the CART algorithm on $\tilde{\mathcal{D}}$, restricting the axis-aligned splits upon the initial features. The impurity for the CART algorithm is calculated using $\mathcal{L}$. Each linear model is trained on all features by solving a regularized least squares problem with regularizer $\mathcal{R}$.
6: Return $f_{\theta^*}, \tau$
```

By decomposing problem (1), we can train NIDTs efficiently. We first train the neural network f_θ and then, once f_θ is trained, a decision tree with linear predictions can be efficiently generated using the CART algorithm on the augmented dataset.

3.2 Analytical Guarantees

For the proposed training scheme, we begin by presenting the following existence theorem.

Theorem 3.1. *Given a trained neural network $f_\theta : \mathcal{X} \to \mathbb{R}$, there exists a NIDT $\tau^* : \mathcal{X} \times \tilde{\mathcal{X}} \to \mathbb{R}$ that utilizes the embeddings of f_θ and performs at least as well as f_θ.*

Additionally, we present, under two mild assumptions, an approximation bound for the constructed NIDT. For ease of exposition, we also assume $x \in [0, 1]^d, \tilde{x} \in [0, 1]^{\tilde{d}}$. We define $f : \mathcal{X} \to \mathbb{R}$ to be the underlying truth function and f_θ be its approximation by a neural network that is used to inform the NIDT τ^*.

Assumption 3.2. The estimation error is uniformly bounded, i.e. $|f(x) - f_\theta(x)| \leq \mathcal{K}(n), \forall x \in [0, 1]^d$.

This assumption has been used in several works in the literature regarding prescriptive trees and knowledge distillation [27].

Assumption 3.3. Each weight of f_θ and τ^* is bounded by $m > 0$.

Theorem 3.4. *Under the above assumptions, the difference between τ^* and f is bounded by $|f(x) - \tau^*(x, f_\theta^K(x))| \leq \mathcal{K}(n) + m(d + 2\tilde{d} + 2)$.*

Next, we describe how domain knowledge can be included in NIDTs by solving a convex optimization problem at each leaf.

3.3 Incorporating Domain Knowledge for Fairness

A crucial requirement for predictive tasks in practice is to incorporate domain knowledge. Adding domain knowledge, such as fairness constraints, can improve model fairness and safety. Explicit incorporation of knowledge through constraints is hard for many tree-based algorithms as they rely on recursive partitioning of the data, making it challenging to impose constraints across branches. We consider knowledge constraints of the form,

$$\textit{if } h_i(x) \leq 0, \forall i \in [m], \textit{ then } \tau(x, f_\theta^K(x)) \textit{ should belong in } \mathcal{I}, \tag{2}$$

where $h_i : \mathcal{X} \to \mathbb{R}, i \in [m]$, are convex functions and $\mathcal{I} \subseteq \mathcal{Y}$ is a closed or a half-line interval in which the NIDT's prediction should belong. An example for drug prescription would be *if BMI ≥ 30, then the prescribed dosage by $\tau(x, f_\theta^K(x))$ should be $\geq 20mg$.*

Typically, the CART algorithm fits a linear model at each leaf ℓ by solving the convex optimization problem

$$\min_{\alpha_\ell,\beta_\ell,c_\ell} \sum_{(x,\tilde{x},y)\in\tilde{\mathcal{D}}_\ell} \mathcal{L}(\alpha_\ell^T x + \beta_\ell^T \tilde{x} + c_\ell, y) + \lambda_\ell \mathcal{R}(\alpha_\ell, \beta_\ell, c_\ell), \tag{3}$$

where $\tilde{\mathcal{D}}_\ell$ is the enhanced dataset for leaf ℓ, i.e. $\tilde{\mathcal{D}}_\ell = \{(x, \tilde{x}, y) | \forall (x, y) \in \mathcal{D}, \tilde{x} = f_\theta^K(x), x \in \mathcal{F}(C_\ell)\}$, $\mathcal{R} : \mathbb{R}^d \times \mathbb{R}^{\tilde{d}} \times \mathbb{R} \to \mathbb{R}_{\geq 0}$ is the convex regularizing function and $\lambda_\ell \in \mathbb{R}_{\geq 0}$ is the leaf's regularizer multiplier. To explicitly incorporate domain knowledge constraints, we modify the CART algorithm to solve the following constrained convex optimization problem at leaf ℓ

$$\begin{aligned} \min_{\alpha_\ell,\beta_\ell,c_\ell} \quad & \sum_{(x,\tilde{x},y)\in\tilde{\mathcal{D}}_\ell} \mathcal{L}(\alpha_\ell^T x + \beta_\ell^T \tilde{x} + c_\ell, y) + \lambda_\ell \mathcal{R}(\alpha_\ell, \beta_\ell, c_\ell) \\ \text{s.t.} \quad & (\alpha_\ell^T x + \beta_\ell^T \tilde{x} + c_\ell) \in \mathcal{I}, \ \forall (x, \tilde{x}) \in \tilde{\mathcal{D}}_{\ell,h}. \end{aligned} \tag{4}$$

where $\tilde{\mathcal{D}}_{\ell,h} := \{(x, \tilde{x}) \in \tilde{\mathcal{D}}_\ell : h_i(x) \leq 0, \forall i \in [m]\}$. Given that in practice, the loss is the MSE and the regularizer the ℓ_2 norm (Ridge), (4) is a convex quadratically constrained quadratic problem that can be solved efficiently using commercial solvers.

3.4 Comparison with Benchmarks

We compare the R^2 (coefficient of determination) achieved by NIDTs against that of multiple tree models on 15 publicly available UCI

Table 1: Test R^2 for 15 UCI datasets across all models. The best R^2 per dataset is highlighted.

Models	abalone	ailerons	airfoil	cccp	cpu-act
CART	0.46 ± 0.02	0.67 ± 0.02	0.53 ± 0.04	0.92 ± 0.01	0.85 ± 0.12
CART-L	0.56 ± 0.04	0.80 ± 0.01	0.86 ± 0.03	0.94 ± 0.00	0.93 ± 0.15
BA	0.44 ± 0.02	0.66 ± 0.02	0.49 ± 0.05	0.91 ± 0.00	0.88 ± 0.04
BA-L	0.57 ± 0.02	0.80 ± 0.03	0.85 ± 0.02	0.94 ± 0.00	**0.98 ± 0.00**
ST	0.46 ± 0.02	0.62 ± 0.02	0.53 ± 0.07	0.85 ± 0.08	0.83 ± 0.10
ST-L	0.57 ± 0.03	0.66 ± 0.10	0.65 ± 0.09	0.91 ± 0.08	0.96 ± 0.01
ME	0.43 ± 0.04	0.65 ± 0.08	0.46 ± 0.07	0.91 ± 0.01	0.68 ± 0.14
ME-L	0.54 ± 0.05	0.63 ± 0.11	0.72 ± 0.05	0.93 ± 0.01	0.73 ± 0.03
ORT	0.47 ± 0.03	0.69 ± 0.01	0.61 ± 0.04	0.92 ± 0.00	0.89 ± 0.04
ORT-L	0.56 ± 0.03	0.80 ± 0.01	0.82 ± 0.02	0.93 ± 0.00	**0.98 ± 0.00**
NIDT (ours)	**0.59 ± 0.02**	**0.82 ± 0.02**	**0.88 ± 0.02**	**0.95 ± 0.00**	**0.98 ± 0.00**
NIDT relative	98.12%	105.67%	99.63%	100.72%	100.26%

Models	cpu-small	elevators	housing	kin8nm	life-exp
CART	0.84 ± 0.12	0.46 ± 0.03	0.72 ± 0.08	0.36 ± 0.01	0.85 ± 0.01
CART-L	0.97 ± 0.00	0.81 ± 0.01	0.77 ± 0.07	0.65 ± 0.02	0.74 ± 0.06
BA	0.88 ± 0.04	0.46 ± 0.02	0.71 ± 0.05	0.35 ± 0.02	0.84 ± 0.02
BA-L	0.97 ± 0.01	0.82 ± 0.01	0.78 ± 0.07	0.65 ± 0.02	0.89 ± 0.04
ST	0.82 ± 0.11	0.44 ± 0.05	0.7 ± 0.07	0.34 ± 0.02	0.73 ± 0.07
ST-L	0.95 ± 0.02	0.76 ± 0.10	0.76 ± 0.10	0.63 ± 0.01	0.78 ± 0.08
ME	0.72 ± 0.18	0.47 ± 0.02	0.69 ± 0.09	0.34 ± 0.04	0.64 ± 0.08
ME-L	0.91 ± 0.13	0.78 ± 0.02	0.74 ± 0.11	0.62 ± 0.05	0.80 ± 0.09
ORT	0.88 ± 0.04	0.49 ± 0.02	0.74 ± 0.08	0.43 ± 0.02	0.88 ± 0.00
ORT-L	0.96 ± 0.00	0.83 ± 0.02	0.78 ± 0.14	0.68 ± 0.02	0.94 ± 0.03
NIDT (ours)	**0.98 ± 0.00**	**0.87 ± 0.04**	**0.85 ± 0.04**	**0.72 ± 0.01**	**0.95 ± 0.01**
NIDT relative	100.12%	98.26%	100.72%	99.81%	100.49%

Models	parkinsons-m	parkinsons-t	steel	superconduct	wine
CART	0.12 ± 0.02	0.12 ± 0.02	0.97 ± 0.01	0.74 ± 0.01	0.27 ± 0.02
CART-L	0.23 ± 0.02	0.23 ± 0.02	**1.0 ± 0.00**	0.66 ± 0.25	0.35 ± 0.02
BA	0.12 ± 0.03	0.12 ± 0.03	0.97 ± 0.01	0.74 ± 0.01	0.26 ± 0.02
BA-L	0.25 ± 0.02	0.26 ± 0.02	**1.0 ± 0.00**	0.84 ± 0.02	0.33 ± 0.05
ST	0.13 ± 0.02	0.13 ± 0.03	0.94 ± 0.04	0.74 ± 0.01	0.24 ± 0.03
ST-L	0.25 ± 0.02	0.27 ± 0.02	**1.0 ± 0.00**	0.85 ± 0.01	0.31 ± 0.03
ME	0.13 ± 0.04	0.13 ± 0.03	0.89 ± 0.01	0.46 ± 0.09	0.21 ± 0.03
ME-L	0.26 ± 0.07	0.27 ± 0.10	0.97 ± 0.01	0.86 ± 0.11	0.32 ± 0.04
ORT	0.14 ± 0.02	0.15 ± 0.03	0.98 ± 0.00	0.77 ± 0.01	0.28 ± 0.03
ORT-L	0.27 ± 0.02	0.32 ± 0.04	0.99 ± 0.00	0.87 ± 0.01	0.34 ± 0.03
NIDT (ours)	**0.38 ± 0.03**	**0.43 ± 0.03**	**1.0 ± 0.00**	**0.95 ± 0.00**	**0.38 ± 0.02**
NIDT relative	94.42%	98.17%	100.01%	105.82%	99.46%

datasets [13]. Specifically, we compare against the CART trees [5], born-again trees (BA) [6], student-teacher (ST) trees [14], model extraction (ME) trees [2] and optimal regression trees (ORT) [4]. All methods are tested with both constant and linear predictors (-L). For the NIDT, we fit a Ridge regression model at each leaf. We also report the ratio between the R^2 of the NIDT and the R^2 of the informing network. The higher the ratio, the better the NIDT performs compared to the network. The hyperparameters of all models were optimized using 3-fold cross-validation, with the specific values detailed in the Appendix. For each dataset, we run 10 independent simulations and report the mean and standard deviation of the R^2 in Table 1.

We observe that the NIDTs outperform all benchmarks across all datasets, confirming their superior performance. Furthermore, we also observe that NIDTs perform at least 94.42% as well as the informing network, while in 8 out of the 15 datasets, they outperform it. Thus, it can be seen that in the majority of the cases, our proposed training scheme can indeed retrieve a NIDT that outperforms the informing network. Next, we describe how NIDTs can be used to extract an interpretable warfarin prescription policy.

3.5 An Interpretable Drug Dosage Policy

According to the International Warfarin Pharmacogenetics Consortium, warfarin is the most widely used oral anticoagulant agent worldwide. Determining the suitable dosage is challenging as it may differ by a factor of ten among patients, and inaccurate dosages can result in severe adverse effects [8]. For this study, we use the publicly available dataset collected by [8] and apply NIDTs to estimate the correct weekly stable dosage. The dataset contains the actual stable dose for $5,701$ patients. Upon data preprocessing, by eliminating missing values and outliers, we acquire a dataset of $3,109$ patients. The patient covariates consist of demographic information (gender, race, etc), diagnostic information (reason for treatment, e.g. cardiomyopathy), and genetic information (presence of CYP2C9, VKORC1 genotype polymorphisms).

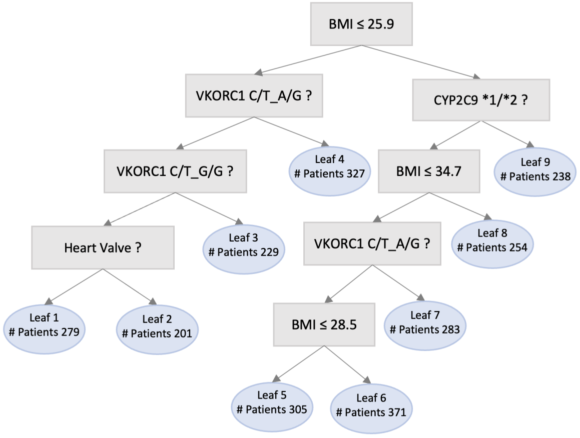

Figure 2: The NIDT for the warfarin prescription. The left arrow corresponds to "Yes" and the other to “No”.

We employ a NIDT with hyperparameters selected as in Section 3.4. Approximately 60.8% of NIDT's predictions fall within 7 mg from the true weekly stable dosage, which is an average daily deviation of at most 1 mg. This is the best performance compared to all benchmarks. Figure 2 displays the resulting NIDT. We observe that the most prevalent features in the NIDT are whether VKORC1 and CYP2C9 genotypes are present, the BMI of the patient, and whether the patient has a mechanical heart valve. These features are known to be strongly associated with warfarin dosage requirements [15, 19, 23]. The deployed NIDT learns this relationship and provides a trasparent dosing guideline where the effect of the relevant features is clear. Indeed, each path to a leaf creates an interpretable and meaningful partitioning of patients into groups. For each group, a distinct linear model that uses both the initial features and the embeddings of the informing network predicts the stable weekly dosage of each patient within the group.

3.6 Reducing Gender Bias by Integrating Fairness Constraints

We now describe how the methodology of Section 3.3 can be used for creating certified fair NIDTs. We consider a setting where for each instance i we have access to a sensitive attribute $s_i \in \{0,1\}$, i.e. $\mathcal{D} = \{(x_n, s_n, y_n)\}_{i=1}^{n}$, $|\mathcal{D}| = n$, that represents the protected group membership of the i-th instance. For example, $s_i = 1$ may represent that instance i belongs to a specific demographic group. To quantify disparities across different groups, we focus on demographic parity (DP) [1]. DP requires the regressor's predictions to be statistically independent of the sensitive attribute, i.e., $h(x|s=0) = h(x|s=1)$, where $h : \mathcal{X} \to \mathbb{R}$ is the regressor under examination. An approach frequently used to develop fair regressors is to enforce the absolute difference in DP, i.e. $\Delta_h(x) = |h(x|s=0) - h(x|s=1)|$, to be close to zero through proxies [28].

To tackle this problem, we take a different approach where instead of using proxies, we incorporate the constraint $\Delta_\tau(x) \leq \epsilon$, with ϵ being a small positive value that controls for the maximum allowed DP in each instance, *directly* into the training of the NIDT using the methodology described in Section 3.3. More specifically, for a given NIDT τ and $x \in \mathcal{F}(C_\ell)$, where ℓ is a leaf of τ, we have that $\Delta_\tau(x) = |\alpha_{\ell,s} + \beta_\ell^T (f^K(x|s=1) - f^K(x|s=0))| \leq \epsilon$, where $\alpha_{\ell,s}$ is the coefficient of the sensitive attribute s for the linear model of leaf ℓ. By linearizing the constraint, we obtain that during the training of the NIDT, at each leaf we need to solve the following convex optimization problem to satisfy explicitly fairness requirements.

$$\begin{aligned} \min_{\alpha_\ell, \beta_\ell, c_\ell} \quad & \sum_{(x,\tilde{x},y) \in \tilde{\mathcal{D}}_\ell} \mathcal{L}(\alpha_\ell^T x + \beta_\ell^T \tilde{x} + c_\ell, y) + \lambda_\ell \mathcal{R}(\alpha_\ell, \beta_\ell, c_\ell) \\ \text{s.t.} \quad & \alpha_{\ell,s} + \beta_\ell^T (f^K(x|s=1) - f^K(x|s=0)) \leq \epsilon, \ \forall x \in \tilde{\mathcal{D}}_\ell, \\ & -\alpha_{\ell,s} + \beta_\ell^T (f^K(x|s=0) - f^K(x|s=1)) \leq \epsilon, \ \forall x \in \tilde{\mathcal{D}}_\ell. \end{aligned} \tag{5}$$

We test our approach on the Student Performance Data Set [9]. The data describe student performance in secondary education of two Portuguese schools. The attributes include student grades, demographics, social, and other school-related features. The task is to predict the final grade (ranging from 0 to 20) of each student. Our goal is to apply our approach to mitigate gender bias in the prediction of the final grade. In the experiment, we consider sex as the sensitive attribute. We compare the error variance, i.e. $1 - R^2$, as well as the actual average DP achieved by the NIDT against that of the informing neural network. The hyperparameters were selected as in Section 3.4, while for the NIDT we test values of ϵ ranging from 10^{-3} to 1. We run 10 independent simulations for each model and we report the results in Figure 3.

We observe that the constrained (fair) NIDT outperforms in terms of predictive performance, i.e. R^2, the unconstrained informing neural network for $\epsilon > 0.005$. This implies that even with a minimal allowed influence from the sensitive attribute, our method can retrieve a fair NIDT with enhanced predictive capabilities. We also observe that the average DP achieved by the fair NIDT is significantly lower than the corresponding maximum allowed DP, i.e. ϵ, and also is always lower than the one achieved by the network. Overall, the fair NIDT consistently demonstrates superior fairness and higher accuracy than the informing network.

4 CONCLUSIONS

Our work adds to an emerging body of research, i.e. generating a simultaneously expressive, interpretable, and fair predictive model. We propose NIDT, a model that combines the predictive power of neural networks with the interpretability of decision trees. To generate a NIDT we introduce a training scheme that first trains

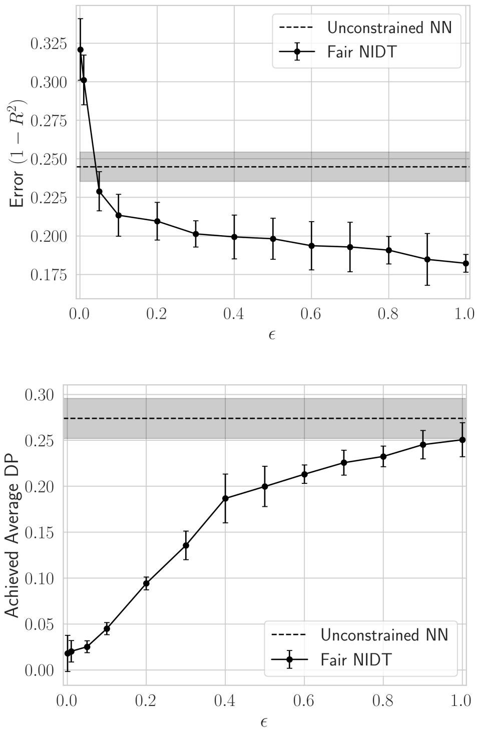

Figure 3: (Top) Error variance ($1 - R^2$) achieved by the unconstrained neural network and the constrained (fair) NIDT. (Bottom) Actual average DP achieved by the fair NIDT and the unconstrained neural network that informs the NIDT.

a task-specific neural network and then utilizes its embeddings to inform predictions. The proposed scheme facilitates fairness by providing the flexibility to incorporate fairness constraints directly into the training process, compared to other approaches that use surrogates, resulting in a certified fair model. Experiments on multiple real-world datasets verify the predictive power, the interpretability and the fairness of NIDTs. Future research could investigate an extension to policy learning scenarios and to decision-making.

REFERENCES

[1] Solon Barocas, Moritz Hardt, and Arvind Narayanan. 2017. Fairness in machine learning. *Nips tutorial* 1 (2017).

[2] Osbert Bastani, Carolyn Kim, and Hamsa Bastani. 2017. Interpreting blackbox models via model extraction. *arXiv preprint arXiv:1705.08504* (2017).

[3] Henry C Bendekgey and Erik Sudderth. 2021. Scalable and stable surrogates for flexible classifiers with fairness constraints. *Advances in Neural Information Processing Systems* 34 (2021), 30023–30036.

[4] Dimitris Bertsimas and Jack Dunn. 2017. Optimal classification trees. *Machine Learning* 106 (2017).

[5] Leo Breiman, Jerome Friedman, Charles J. Stone, and R.A. Olshen. 1984. *Classification and Regression Trees.* Chapman and Hall/CRC.

[6] Leo Breiman and Nong Shang. 1996. Born again trees. *University of California, Berkeley, Berkeley, CA, Technical Report* 1, 2 (1996).

[7] Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. 2015. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining.*

[8] International Warfarin Pharmacogenetics Consortium. 2009. Estimation of the warfarin dose with clinical and pharmacogenetic data. *New England Journal of Medicine* 360, 8 (2009).

[9] Paulo Cortez and Alice Maria Gonçalves Silva. 2008. Using data mining to predict secondary school student performance. (2008).

[10] Houtao Deng. 2019. Interpreting tree ensembles with intrees. *International Journal of Data Science and Analytics* 7, 4 (2019).

[11] Michele Donini, Luca Oneto, Shai Ben-David, John S Shawe-Taylor, and Massimiliano Pontil. 2018. Empirical risk minimization under fairness constraints. *Advances in neural information processing systems* 31 (2018).

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

[13] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository.

[14] Nicholas Frosst and Geoffrey E. Hinton. 2017. Distilling a Neural Network Into a Soft Decision Tree. *CoRR* abs/1711.09784 (2017). arXiv:1711.09784

[15] Amanda Hu, Chi-Ming Chow, Diem Dao, Lee Errett, and Mary Keith. 2006. Factors influencing patient knowledge of warfarin therapy after mechanical heart valve replacement. *Journal of cardiovascular Nursing* 21, 3 (2006).

[16] Toshihiro Kamishima, Shotaro Akaho, and Jun Sakuma. 2011. Fairness-aware learning through regularization approach. In *2011 IEEE 11th International Conference on Data Mining Workshops.* IEEE, 643–650.

[17] Jon Kleinberg, Himabindu Lakkaraju, Jure Leskovec, Jens Ludwig, and Sendhil Mullainathan. 2018. Human decisions and machine predictions. *The quarterly journal of economics* 133, 1 (2018).

[18] Himabindu Lakkaraju, Stephen H Bach, and Jure Leskovec. 2016. 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.*

[19] Tao Li, Leslie A Lange, Xiangli Li, Lisa Susswein, Betsy Bryant, Robb Malone, Ethan M Lange, Teng-Yi Huang, Darrel W Stafford, and James P Evans. 2006. Polymorphisms in the VKORC1 gene are strongly associated with warfarin dosage requirements in patients receiving anticoagulation. *Journal of medical genetics* 43, 9 (2006).

[20] Bo Liu, Ming Ding, Sina Shaham, Wenny Rahayu, Farhad Farokhi, and Zihuai Lin. 2021. When machine learning meets privacy: A survey and outlook. *ACM Computing Surveys (CSUR)* 54, 2 (2021).

[21] Wei-Yin Loh. 2011. Classification and regression trees. *Wiley interdisciplinary reviews: data mining and knowledge discovery* 1, 1 (2011).

[22] 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).

[23] Julia A Mueller, Tulsi Patel, Ahmad Halawa, Adrian Dumitrascu, and Nancy L Dawson. 2014. Warfarin dosing and body mass index. *Annals of Pharmacotherapy* 48, 5 (2014).

[24] Judea Pearl. 2009. *Causality.* Cambridge university press.

[25] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. " 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.*

[26] Kush R Varshney. 2019. Trustworthy machine learning and artificial intelligence. *XRDS: Crossroads, The ACM Magazine for Students* 25, 3 (2019), 26–29.

[27] Stefan Wager and Susan Athey. 2018. Estimation and inference of heterogeneous treatment effects using random forests. *J. Amer. Statist. Assoc.* 113, 523 (2018).

[28] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. 2017. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th international conference on world wide web.*

[29] Haozhe Zhang, Dan Nettleton, and Zhengyuan Zhu. 2019. Regression-enhanced random forests. *arXiv preprint arXiv:1904.10416* (2019).

A PROOFS OF THEOREMS

Theorem 3.1 *Given a trained neural network $f_\theta : \mathcal{X} \longrightarrow \mathbb{R}$, there exists a NIDT $\tau^* : \mathcal{X} \times \tilde{\mathcal{X}} \longrightarrow \mathbb{R}$ that utilizes the embeddings of f_θ and performs at least as well as f_θ.*

Proof. Let $f_\theta : \mathcal{X} \longrightarrow \mathbb{R}$ be a trained neural network at the task at hand, $\mathcal{L}(\cdot,\cdot) : \mathcal{Y} \times \mathcal{Y} \longrightarrow \mathbb{R}_{\geq 0}$ the loss function, and $\mathcal{D}$ the dataset under consideration. We consider the degenerate NIDT $\hat{\tau}$ of depth 0, i.e. $N_{\hat{\tau}}(x, \tilde{x}) = \alpha^T x + \beta^T \tilde{x} + c$, with $\alpha = 0$, $\beta = w_K$ and $c = b_K$, where w_K and b_K are the weights of the output layer of f_θ. We have that $\mathcal{L}(f_\theta(x), y) = \mathcal{L}(\hat{\tau}(x, f_\theta^K(x)), y), \forall (x, y) \in \mathcal{D}$, since $\hat{\tau}$ replicates exactly f_θ by construction.

Therefore, if we denote as τ^* the solution to $\min_\tau \sum_{(x,y)\in\mathcal{D}} \mathcal{L}(\tau(x, f_\theta^K(x)), y)$, i.e. $\tau^* = \arg\min_\tau \sum_{(x,y)\in\mathcal{D}} \mathcal{L}(\tau(x, f_\theta^K(x)), y)$, we obtain

$$\sum_{(x,y)\in\mathcal{D}} \mathcal{L}(\tau^*(x, f_\theta^K(x)), y) \leq \sum_{(x,y)\in\mathcal{D}} \mathcal{L}(\hat{\tau}(x, f_\theta^K(x)), y) = \sum_{(x,y)\in\mathcal{D}} \mathcal{L}(f_\theta(x), y). \tag{6}$$

□

Theorem 3.4 *Under the above assumptions, the difference between τ^* and f is bounded by $|f(x) - \tau^*(x, f_\theta^K(x))| \leq \mathcal{K}(n) + m(d + 2\tilde{d} + 2)$.*

Proof. Let $x \in [0, 1]^d$. We have that:

$$\begin{aligned} |f(x) - \tau^*(x, f_\theta^K(x))| &= |f(x) - f_\theta(x) + f_\theta(x) - \tau^*(x, f_\theta^K(x))| \leq \\ &\leq |f(x) - f_\theta(x)| + |f_\theta(x) - \tau^*(x, f_\theta^K(x))|. \end{aligned} \tag{7}$$

Assuming that $x \in \mathcal{F}(C_\ell)$, where ℓ is a leaf of τ^*, we have that

$$|f_\theta(x) - \tau^*(x, f_\theta^K(x))| = |\alpha_\ell^T x + (\beta_\ell - w_K)^T \tilde{x} + c_\ell - b_K|, \tag{8}$$

where w_K is the weight vector and b_K is the bias of the output layer of the network. By plugging equation (8) back into equation (7) and using the assumption on the uniformly bounded estimation error we obtain that

$$\begin{aligned} |f(x) - \tau^*(x, f_\theta^K(x))| &\leq \mathcal{K}(n) + |\alpha_\ell^T x + (\beta_\ell - w_K)^T \tilde{x} + c_\ell - b_K| \leq \\ &\leq \mathcal{K}(n) + \max_{\ell,\theta,x\in\mathcal{X},\tilde{x}\in\tilde{\mathcal{X}}} |\alpha_\ell^T x + (\beta_\ell - w_K)^T \tilde{x} + c_\ell - b_K| \leq \\ &\leq \mathcal{K}(n) + \max_{\ell,x\in\mathcal{X}} |\alpha_\ell^T x| + \max_{\ell,\theta,\tilde{x}\in\tilde{\mathcal{X}}} |(\beta_\ell - w_K)^T \tilde{x}| + \max_{\ell,\theta} |c_\ell - b_K| = \\ &= \mathcal{K}(n) + md + 2m\tilde{d} + 2m = \mathcal{K}(n) + m(d + 2\tilde{d} + 2). \end{aligned} \tag{9}$$

□

B EXPERIMENTAL SETUP

All computational experiments were performed using Python 3.9, Gurobi 9.5, Linear-Tree 0.3.5, Keras 2.10.0, and Scikit-learn 1.0. All experiments were executed on an internal cluster with a 2.20GHz Intel(R) Xeon(R) Gold 5120 CPU and 256 GB memory.

The hyperparameters used for tuning all tree-based methods are described in Table 2. For the neural networks, we used the dense, feed-forward architecture, with 10 neurons in the last hidden layer. The hyperparameters used for tuning the neural networks are described in Table 3.

Table 2: Hyperparameters used for tuning all tree-based methods.

Max Depth	Min Samples Split	Min Samples Leaf
$\{3, 5, 7, 10, 100\}$	$\{0.05, 0.1\}$	$\{0.04, 0.06, 0.08\}$

Table 3: Hyperparameters used for tuning the neural networks.

Hidden Layers	Neurons per Layer	Activation	Dropout	Early Stopping Patience
$\{2, 3\}$	$\{20, 50, 100\}$	{ReLU, ELU, tanh}	$\{0, 0.1, 0.2\}$	$\{20\}$

C DESCRIPTION OF DATASETS

A description of the datasets used in the experiments of Section 3.4 is provided in Table 4.

Table 4: Description of the 15 UCI datasets used. The number of features is calculated after converting categorical variables into dummy/indicator variables.

Dataset	Dataset Size	Features
abalone	4,177	9
ailerons	13,750	40
airfoil	1,503	5
cccp	9,568	4
cpu-act	8,192	21
cpu-small	8,192	12
elevators	16,599	18
housing	1,059	68
kin8nm	8,192	8
life-exp	1,649	152
parkinsons-m	5,875	16
parkinsons-t	5,875	16
steel	35,040	15
superconduct	21,263	81
wine	4,898	11

D WARFARIN DOSING - BENCHMARKS

To evaluate the performance of the NIDTs on Warfarin prescription, we compare our approach with the benchmarks described in Section 3.4. The hyperparameters of all benchmarks are tuned using 3-fold cross-validation, while the hyperparameters tested are described in Section B of the Appendix. We run 10 independent simulations for each benchmark and we report the mean and the standard deviation of the percentage of the predictions that fall within a range of 7 mg from the actual weekly stable dosage, which is an average deviation of at most 1 mg per day, in Table 5.

Table 5: Average percentage of the predictions that fall within a range of 7 mg from the actual weekly stable dosage.

Models	Warfarin
CART	55.32 ± 1.47
CART-L	56.19 ± 1.60
BA	55.71 ± 2.10
BA-L	57.91 ± 1.88
ST	55.08 ± 1.90
ST-L	56.98 ± 1.52
ME	55.32 ± 2.16
ME-L	58.01 ± 2.51
ORT	54.29 ± 2.06
ORT-L	56.95 ± 1.93
NN	59.08 ± 2.33