Interpreting Predictive Models for Human-in-the-Loop Analytics

Hamsa Bastani
Wharton School, Operations Information and Decisions, hamsab@wharton.upenn.edu

Osbert Bastani
University of Pennsylvania, Computer Science Department, obastani@seas.upenn.edu

Carolyn Kim
Stanford University, Computer Science Department, ckim@cs.stanford.edu

Machine learning is increasingly used to inform consequential decisions. Yet, these predictive models have been found to exhibit unexpected defects when trained on real-world observational data, which are plagued with confounders and biases. Thus, it is critical to involve domain experts in an interactive process of developing predictive models; interpretability offers a promising way to facilitate this interaction. We propose a novel approach to interpreting complex, blackbox machine learning models by constructing simple decision trees that summarize their reasoning process. Our algorithm leverages active learning to extract richer and more accurate interpretations than several baselines. Furthermore, we prove that by generating a sufficient amount of data through our active learning strategy, the extracted decision tree converges to the exact decision tree, implying that we provably avoid overfitting. We evaluate our algorithm on a random forest to predict diabetes risk on a real electronic medical record dataset, and show that it produces significantly more accurate interpretations than several baselines. We also conduct a user study demonstrating that humans are able to better reason about our interpretations than state-of-the-art rule lists. We then perform a case study with domain experts (physicians) regarding our diabetes risk prediction model, and describe several insights they derived using our interpretation. Of particular note, the physicians discovered an unexpected causal issue by investigating a subtree in our interpretation; we were able to then verify that this endogeneity indeed existed in our data, underscoring the value of interpretability.

Key words: interpretability, risk prediction, observational data, predictive analytics, human-in-the-loop, active learning

History: This paper is under preparation.

1. Introduction

Machine learning has revolutionized our ability to use data to inform important decisions in a variety of domains such as healthcare, criminal justice, and retail. For instance, predictive analytics have been used to score patient risk (Ustun and Rudin 2016), identify personalized medical treatment regimens (Wang et al. 2016, Bertsimas et al. 2017), make bail decisions for defendants (Kleinberg et al. 2017, Jung et al. 2017), price products (Ferreira et al. 2015), and set retail staffing levels (Fisher et al. 2017). At the same time, machine learning models have been shown to exhibit unexpected defects when trained on real-world observational data, stemming from endogenous explanatory variables (LaLonde 1986, Angrist et al. 1996), unobserved censoring, or other systematic biases in reported outcomes data (Bastani et al. 2015, Mullainathan and Obermeyer 2017). In such cases, the predictive model may achieve good out-of-sample accuracy on the original observational dataset, but perform poorly when deployed in the real world. While these issues can sometimes be addressed through the use of carefully chosen instrumental variables (e.g., as is the case in Wang et al. 2016, Fisher et al. 2017), this process first requires understanding the source of confounding or bias, which in turn requires significant domain knowledge. Oftentimes, with the growing number of explanatory variables and the complexity of machine learning, even domain experts may need to examine and understand the learned model before the presence of confounding or bias becomes apparent. Thus, it is critical to involve domain experts in an iterative process of developing the predictive model to ensure that its predictions are unbiased; we refer to this process as human-in-the-loop analytics.

Yet, state-of-the-art machine learning models such as random forests and deep neural nets tend to be *blackbox* in nature (Rudin 2014); in other words, these models have a complex, opaque structure and tend to use many explanatory variables, making it difficult for humans to understand and verify the model's reasoning process. Thus, one proposed solution for facilitating human-in-the-loop analytics is the use of *interpretable* machine learning models (Doshi-Velez and Kim 2017). Examples of previously-proposed interpretable models include sparse linear models (Ustun and Rudin 2016), rule lists (Letham et al. 2015), and decision sets (Lakkaraju et al. 2016). These models are simple and transparent, allowing domain experts to easily understand how predictions are made; with this knowledge, they can identify potential sources of bias or errors in the model by checking if the model mimics their own reasoning process.

However, the constraint of using an interpretable model instead of a complex blackbox model comes at a significant cost in predictive accuracy, which in turn may result in poor decision-making (Ribeiro et al. 2016, Koh and Liang 2017). Thus, decision-makers are often faced with a tough decision: either (1) use an interpretable model which may produce worse decisions due to poor predictive performance, or (2) use a blackbox model which has strong predictive performance on observational data, but may exhibit arbitrary unexpected defects upon deployment in the real world. In this paper, motivated by discussions with industry experts, we propose a third alternative: to extract a simple interpretation that *approximates* a complex blackbox model. We express our interpretation in the form of a decision tree (Breiman et al. 1984), whose size can be chosen based on the desired strength of the approximation to the blackbox model. Then, the domain expert can restrict her focus on understanding and verifying the extracted decision tree rather than the original blackbox model. As long as the decision tree is a good approximation of the blackbox model, any significant confounding or bias in the blackbox model should translate to the tree. Thus, if the expert validates the tree's reasoning, then we may deploy the high-performing blackbox model with the confidence that it is likely free of significant bias or confounding as well.

1.1. Contributions

We extract simple, accurate decision trees from complex blackbox machine learning models. We make no assumptions on the structure of the blackbox model, and only require the ability to run it on a chosen set of inputs. We choose decision trees as our interpretations, since they are easy to understand, nonparametric, and can compactly represent complex functions.

Algorithm. We require an algorithm for constructing decision trees that accurately represents a given blackbox model. While decision trees are easy to interpret, a common problem is that they typically achieve poor predictive performance since they easily overfit to data. To overcome this difficulty, we leverage the ability to generate arbitrarily large amounts of training data by sampling new inputs and labeling them using the blackbox model. We propose a novel algorithm that uses *active learning* to generate inputs that flow down a given path in the decision tree, and then use these newly generated training points to avoid overfitting.

Theory. We prove that by actively sampling a sufficient number of points using our algorithm, our extracted decision tree converges to the *exact* decision tree. In other words, the estimation error of our decision tree interpretation goes to zero asymptotically, implying that our decision tree avoids overfitting the small initial training set. The key challenge to establishing this result is that the branches in a greedy decision tree are estimated by maximizing a non-convex objective function. As a result, even very small errors in the estimated objective function can dramatically change its maximizer. Under mild technical conditions, we establish that asymptotically, the estimated objective converges uniformly to the true objective with high probability, and consequently, the maximizer converges as well.

Evaluation. We first evaluate the accuracy and interpretability of our decision tree interpretations on a real dataset from a leading electronic medical record provider. We use a random forest as our blackbox model to predict the risk of a diabetes diagnosis for patients; as expected, the random forest is significantly more accurate than interpretable models such as decision trees (Breiman et al. 1984), rule lists (Yang et al. 2017), and sparse linear models (Tibshirani 1996). We then examine the faithfulness of our extracted decision tree to the blackbox model compared to several baselines; We find that the predictions of our extracted decision tree much more closely match the predictions of the random forest, implying that it is a more faithful interpretation. Next, we perform a user study on 46 graduate students (with at least some machine learning or data science background), and ask them to reason about both our decision tree interpretation and a state-of-the-art rule list interpretation. We find that users are more accurate on similar questions regarding decision trees than rule lists, suggesting that trees may be more interpretable to humans.

Finally, we conduct a case study with three domain experts (physicians) about our diabetes risk prediction model using our tree interpretation. We describe a number of insights they gained by examining our interpretations. Of particular note, the physicians discovered an unexpected causal issue by investigating a subtree in our interpretation; we were able to then verify that this endogeneity indeed existed in our data, underscoring the value of interpretability and human-in-the-loop analytics.

1.2. Related Work

The interaction between decision-makers (managers) and algorithms is an emerging topic of interest in the operations management literature. A number of papers have pointed out that humans experience "algorithm aversion" and erroneously distrust algorithmic predictions (Dietvorst et al. 2015). This is especially true when the task at hand is something that appears to require human intuition – for example, recommending jokes (Yeomans et al. 2016), or setting sale prices (Phillips et al. 2015, Caro and de Tejada Cuenca 2018). However, this aversion can be overcome by better explaining how the algorithm produces its prediction (Yeomans et al. 2016), or by giving the decision-maker power to even slightly modify the predictions (Dietvorst et al. 2016). Caro and de Tejada Cuenca (2018) further explore ways to improve managers' adherence to the algorithm's price recommendations by setting better reference points. On a similar note, from an implementation perspective, Ferreira et al. (2015) discuss that Rue La La managers were concerned about adopting a blackbox price-setting algorithm; the authors were able to ensure successful adoption by using interpretable models and explaining the algorithm's reasoning process.

Generally, the above literature has assumed that humans *should* trust algorithms (since they produce more accurate predictions or decisions), and study ways (such as using interpretable models) to create this trust. Van Donselaar et al. (2010) is a notable exception: they empirically demonstrate that an automated inventory replenishment algorithm for retail stores does not optimize for hidden costs (e.g., in-store handling costs). In this case, incorporating manager input can improve decisions. Our discussions with industry experts revealed many additional instances where the algorithm produced suboptimal predictions due to biases or confounders that were not accounted for during model development; their primary concerns about blackbox algorithms stemmed from the possibility that such an algorithm may be deployed in the real world without catching the error (see discussion in §6 for details). We propose using an approximate interpretation of the blackbox model to help decision-makers identify these problems before deploying the model. Once a problem is identified, one can use existing methods to correct for the bias or confounders (e.g., using instrumental variables in decision trees as proposed by Wang et al. 2017).

There are three general approaches to interpretability. We can directly deploy an interpretable model, derive *local* interpretations for individual predictions from a blackbox model, or extract a *global* interpretation for the entire blackbox model. We expand on this literature below.

Interpretable models. There has been a long history of learning simple, interpretable models such as decision trees, rule lists and sparse linear models. This approach is particularly relevant when accuracy is less important than fairness and transparency, e.g., in making bail decisions (Kleinberg et al. 2017, Jung et al. 2017) or predicting crime recidivism (Zeng et al. 2017).

Decision trees (Breiman et al. 1984) are considered highly interpretable, but unfortunately, they often produce poor predictive accuracy due to overfitting. This concern has been alleviated using non-greedy approaches, such as rule lists (Wang and Rudin 2015, Letham et al. 2015), decision sets (Lakkaraju et al. 2016), and optimal decision trees (Bertsimas and Dunn 2017). Interpretable model families based on sparse linear models (Tibshirani 1996, Ustun and Rudin 2016, Jung et al. 2017) have also been proposed; relatedly, Caruana et al. (2012) propose generalized additive models, which are linear combinations of nonparametric single-feature models. However, ultimately, such methods still come at the cost of predictive accuracy; consequently, the machine learning community has generally favored blackbox models such as random forests and deep neural nets for tasks where predictive accuracy is of foremost importance (Ribeiro et al. 2016, Koh and Liang 2017).

Local Interpretations. Another proposed approach is to use a complex, blackbox model, but to generate local interpretations for every prediction that can be verified by an expert. Specifically, given a new test point x, Ribeiro et al. (2016) generates an interpretation for the prediction $f(x)$ by fitting an interpretable model locally around x and using it as the explanation for the prediction. While this technique can help experts understand a specific prediction, they cannot help understand the model as a whole, making it less useful for diagnosing problems with the data or model itself (e.g., confounders or systematic bias). Furthermore, such an approach may not be suitable for largely automated tasks such as product pricing or nurse-led interventions, where a domain expert may not be able to verify each prediction individually (see discussion in §6).

Global interpretations. We propose using a complex, blackbox model and producing a simple interpretation for its overall reasoning process. Past techniques have focused on identifying influential features. The *relative influence* scores the contribution of each feature in tree-based models such as random forests (Friedman 2001). Similarly, Datta et al. (2016) use the Shapley value to quantify the influence of each feature. In our evaluation, we show that these approaches cannot help understand more complex reasoning performed by the model, as is needed to understand the dependence of a model on potentially endogenous features. In concurrent work, Lakkaraju et al. (2017) extract global explanations in the form of decision sets. However, this approach does not scale to datasets with many features (e.g., our diabetes dataset has hundreds of features), which we verified using the authors' original implementation. Since decision sets are quite similar to rule lists, we compare the optimized implementation provided by Yang et al. (2017) trained on blackbox model labels (as proposed by Lakkaraju et al. (2017)) against our interpretations. We demonstrate that our active learning strategy produces much more accurate interpretations, enabling experts to understand and validate a larger portion of the blackbox model's reasoning process (which manifests in our evaluation with physicians). Furthermore, the richness and accuracy of our interpretations can be gracefully (and provably) tuned by actively sampling more points and growing larger trees; in contrast, prior approaches (using rule lists or decision sets) do not use active learning and cannot produce richer interpretations without overfitting the training data.

2. Problem Formulation

We are given a complex, blackbox machine learning model $f : \mathcal{X} \to \mathcal{Y}$. Typically such a model is learned from an observational dataset: $X_{train} \in \mathbb{R}^{n \times d}$ is the feature matrix, whose rows X_i correspond to the d observed features (explanatory variables) of the i^{th} sample; $Y_{train} \in \mathbb{R}^n$ is a vector of outcomes for each of the n samples. In general, depending on the data, there is a feasible set of feature values (i.e., $X_{train} \subseteq \mathcal{X}$) and outcomes (i.e., $Y_{train} \subseteq \mathcal{Y}$). A data scientist then chooses f from a desired model family (e.g., random forests, neural nets) to best fit $Y_{train} \approx f(X_{train})$. To maintain full generality, we assume we do not know f and simply have blackbox access to it, i.e., given any feasible point $x \in \mathcal{X}$, we can obtain $f(x)$. Our goal is to approximate f using an axis-aligned decision tree T (Breiman et al. 1984).

We start by establishing some notation to describe decision trees. For brevity, we will denote the set $[k] = \{1, ...k\}$.

Definition 1. An *axis-aligned constraint* is a constraint on a chosen feature value, e.g., $C = (x_i \leq t)$, where $i \in [d]$, $t \in \mathbb{R}$, and d is the dimension of the input space. The *feasible set* of C is $\mathcal{F}(C) = \{x \in \mathcal{X} \mid x \text{ satisfies } C\}$.

Note that more general constraints can be built from existing constraints using negations $\neg C$, conjunctions $C_1 \wedge C_2$, and disjunctions $C_1 \vee C_2$.

Definition 2. A *decision tree* T is a binary tree. An *internal node* $N = (N_L, N_R, C)$ of T has a left child node N_L and a right child node N_R, and is labeled with an axis-aligned constraint $C = (x_i \leq t)$. A *leaf node* $N = (y)$ of T is associated with a label $y \in \mathcal{Y}$. We use N_T to denote the root node of T.

The decision tree is a function $T : \mathcal{X} \to \mathcal{Y}$ as well. More precisely, a leaf node $N = (y)$ is interpreted as a function $N(x) = y$. An internal node $N = (N_L, N_R, C)$ is interpreted as a function $N(x) = N_L(x)$ if $x \in \mathcal{F}(C)$, and $N(x) = N_R(x)$ otherwise. In other words, at an internal node, we take the left path if the point x satisfies the node's constraint, and we take the right path otherwise. Then,

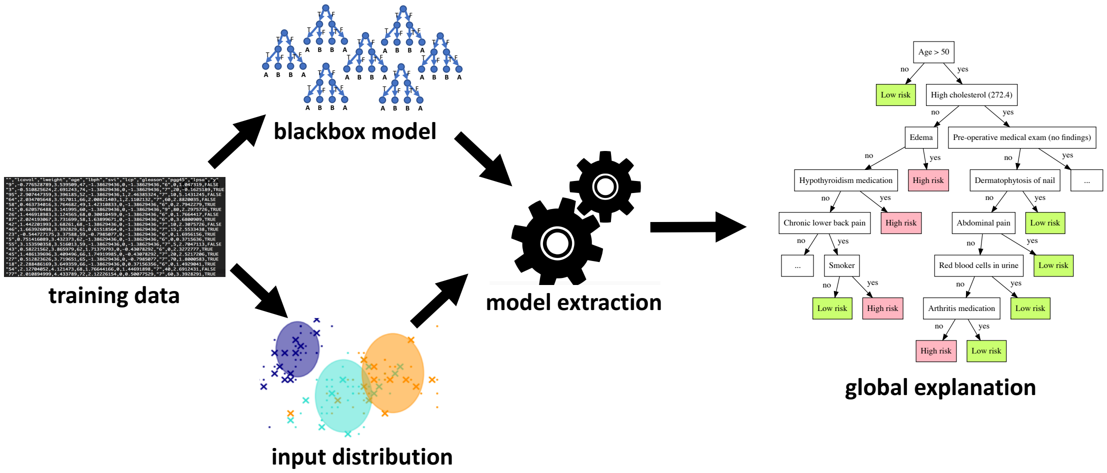

Figure 1 **Overview of our decision tree extraction algorithm.**

$T(x) = N_T(x)$, i.e., any point $x \in \mathcal{X}$ follows the appropriate path of the decision tree until a leaf node is reached.

For a node $N \in T$, we let C_N denote the conjunction of the constraints along the path from the root of T to N. Note that C_N is defined recursively: for the root N_T, we have $C_{N_T} = \text{True}$, and for an internal node $N = (N_L, N_R, C)$, we have $C_{N_L} = C_N \wedge C$ and $C_{N_R} = C_N \wedge \neg C$.

Definition 3. We say an input $x \in \mathcal{X}$ is *routed* to the leaf node $N \in T$ if $x \in \mathcal{F}(C_N)$.

In what follows, we focus on the case $\mathcal{X} = \mathbb{R}^d$ and $\mathcal{Y} = [m]$, i.e., our features are binary or continuous, and our task is classification into one of $m > 1$ groups. Our approach easily generalizes to the case where $\mathcal{X}$ contains categorical features, and to the case where $\mathcal{Y} = \mathbb{R}$ (i.e., regression task). We omit formalizing this case for brevity of notation.

We measure the performance of an interpretation using a metric called *fidelity*, which measures how well the interpretation's predictions match those of the original blackbox model on a test set.

Definition 4. The fidelity of an interpretation g with respect to a blackbox classifier f on a test set X_{test} is the performance of g on $\{x, f(x) \mid x \in X_{test}\}$. The performance metric is typically AUROC if f is a binary classifier, accuracy if f is a multi-class classifier, and mean-squared error if f is a regression.

Note that the purpose of an interpretation is not to perform well on the test set itself, but to mimic the performance of the blackbox model; this ensures that any errors or biases in the blackbox model are likely to translate to the interpretation as well.

3. Decision Tree Extraction

We give an overview of our decision tree extraction algorithm in Figure 1. At a high level, our algorithm proceeds as follows. We first fit a Gaussian mixture model to the training set X_{train} to estimate the joint feature distribution $\mathcal{P}$ over (future) inputs from $\mathcal{X}$. We then iteratively (i) use

$\mathcal{P}$ to generate new samples that are labeled using the blackbox model, and (ii) grow our decision tree interpretation by an additional node. In particular, for any given leaf node N in our current extracted tree, we can sample a new point $x \sim \mathcal{P} \mid C_N$ from the corresponding subpopulation (i.e., subpopulation of inputs that are routed to N), and compute its corresponding label $y = f(x)$ using the blackbox model f. Our algorithm uses this newly generated data to grow the extracted tree by determining the next best leaf node to split (i.e., convert to an internal node and two leaf nodes). The procedure is shown in Algorithm 1.

Algorithm 1 Decision Tree Extraction

```
Inputs: Blackbox model $f$; training set $X_{train}$; # of points $n$ to actively sample at each node;
maximum size $k$ of extracted tree
procedure ExtractTree
    Estimate $\mathcal{P}$ from $X_{train}$ using a Gaussian mixture model
    Let $y_0$ be the majority label on $X_{train}$ using $f$
    Initialize $T$ to be a decision tree with a single (root) node $N_T = (y_0)$
    Initialize $\mathsf{leaves} \leftarrow \{(N_T,\ \textsc{ProcessLeaf}(N_T))\}$
    for $t \in [k]$ do
        Identify $(N, (i^*, t^*, y_L, y_R, G)) \in \mathsf{leaves}$ with the highest gain $G$, and remove it
        In $T$, replace $N$ with $N' = (N_L, N_R, (x_{i^*} \leq t^*))$, where $N_L = (y_L)$ and $N_R = (y_R)$
        Add $(N_L,\ \textsc{ProcessLeaf}(N_L))$ and $(N_R,\ \textsc{ProcessLeaf}(N_R))$ to $\mathsf{leaves}$
    end for
    return $T$
end procedure
procedure ProcessLeaf(Leaf node $N$)
    Sample points $x^{(1)}, ..., x^{(n)} \sim \mathcal{P} \mid C_N$ and let $\hat{\mathcal{P}}_N = \text{Uniform}(\{x^{(1)}, ..., x^{(n)}\})$
    Using $\hat{\mathcal{P}}_N$, compute $(i^*, t^*)$ as in Eq. (1), estimate $y_L, y_R$ as in Eq. (3), and let $G = G(i^*, t^*; \mathcal{P}_N)$
    return $(i^*, t^*, y_L, y_R, G)$
end procedure
```

Remark 1. Algorithm 1 does not split a leaf node N if the classification accuracy is perfect in that node (i.e., $\Pr_{x \sim \mathcal{P}}[f(x) = y_N \mid C_N] = 1$). In these cases, Algorithm 1 may terminate early (i.e., before growing to size k) and produce a smaller tree.

Input distribution. We first use expectation maximization to fit a mixture $\mathcal{P}$ of axis-aligned Gaussian distributions over $\mathcal{X}$:

$$p_{\mathcal{P}}(x) = \sum_{i=1}^{K} \phi_i \mathcal{N}(\mu_i, \Sigma_i)\,,$$

where $p_{\mathcal{P}}$ is the probability density function associated with the distribution $\mathcal{P}$, the weights $\phi \in [0,1]^K$ satisfy $\sum_{i=1}^{K} \phi_i = 1$, and the ith Gaussian distribution in the mixture has mean $\mu_i \in \mathbb{R}^d$ and a diagonal covariance matrix $\Sigma_i \in \mathbb{R}^{d^2}$. Note that we have imposed that the features are independent within each Gaussian (which will enable efficient active sampling), but the resulting mixture $\mathcal{P}$ can still fit well-behaved joint distributions with a sufficient number of mixture components.

Growing the Tree. We initialize the tree with a single leaf node $N_T = (y_0)$, where y_0 is the majority label on X_{train} using the blackbox model f. We then proceed iteratively: at each iteration, we choose a leaf node $N = (y)$ and replace it with a new internal node and two child leaf nodes. Specifically, we replace N with the internal node $N' = (N_L, N_R, C)$, where $N_L = (y_L)$ and $N_R = (y_R)$ are the two new leaf nodes, and $C = (x_{i^*} \leq t^*)$ is the constraint for the new internal node. The choice of which leaf node N to split in each iteration, as well as the resulting parameters (N_L, N_R, C), is dictated by maximizing the *gain function* (Eq. 1), which is estimated for each leaf node by sampling new points from the subpopulation of points routed to that node (detailed below). This process is repeated $k-1$ times to grow a tree of size at most k.

Estimating the Gain. Given a distribution $\mathcal{Q}$ over $\mathcal{X}$ (we will later take $\mathcal{Q}$ to be the distribution of input points routed to a leaf node N in the decision tree), the gain function G measures the quality of a candidate split $x_i \leq t$ (where $i \in [d]$ and $t \in \mathbb{R}$):

$$\begin{aligned} G(i,t;\mathcal{Q}) &= -H(f,(x_i \leq t);\mathcal{Q}) - H(f,(x_i > t);\mathcal{Q}) + H(f,(\text{True});\mathcal{Q}), \qquad (1) \\ H(f,C;\mathcal{Q}) &= \left(1 - \sum_{y \in \mathcal{Y}} \Pr_{x \sim \mathcal{Q}}[f(x) = y \mid C]^2\right) \cdot \Pr_{x \sim \mathcal{Q}}[C], \end{aligned}$$

where H is the Gini impurity (Breiman et al. 1984), which is the standard metric used in the gain function for constructing decision tree classifiers. This metric can be replaced with other measures such as entropy (for classification) or mean-squared error (for regression).

For any leaf node N, we would ideally maximize G to obtain the optimal split for that node:

$$(i^*, t^*) = \underset{i \in [d], t \in \mathbb{R}}{\arg\max}\ G(i,t;\mathcal{P} \mid C_N),$$

where $\mathcal{P} \mid C_N$ is the distribution of points that are routed to N in the decision tree. However, it is impossible to optimize G using the exact distribution $\mathcal{P} \mid C_N$, since this would require integrating over f (recall that we only have blackbox access to f). Instead, we use a finite-sample estimate $\hat{\mathcal{P}}_N \approx \mathcal{P} \mid C_N$. More precisely, we sample points $x^{(1)}, ..., x^{(n)} \sim \mathcal{P} \mid C_N$ (the exact procedure is detailed later in this section), and let

$$\hat{\mathcal{P}}_N = \text{Uniform}(\{x^{(1)}, ..., x^{(n)}\}).$$

Then, we maximize G to obtain the optimal split:

$$(i^*, t^*) = \underset{i \in [d], t \in \mathbb{R}}{\arg\max}\ G(i,t;\hat{\mathcal{P}}_N). \qquad (2)$$

Given the optimal constraint $x_{i^*} \leq t^*$ from Eq. 2 for node N, the new leaf node labels (if we were to split on node N) would be

$$\begin{aligned} y_L &= \underset{y \in \mathcal{Y}}{\arg\max} \Pr_{x \sim \hat{\mathcal{P}}_N}[f(x) = y \mid C_N \wedge (x_i \leq t)], \qquad (3) \\ y_R &= \underset{y \in \mathcal{Y}}{\arg\max} \Pr_{x \sim \hat{\mathcal{P}}_N}[f(x) = y \mid C_N \wedge (x_i > t)]. \end{aligned}$$

The node chosen to be replaced is the one with the highest potential gain in the current tree. As described in Algorithm 1, we maintain a mapping leaves between the current leaf nodes of the tree and the potential improvement in the gain function if we were to split that node (as well as the constraint and child nodes associated with such a split). In each iteration, we select the leaf node with the highest gain from leaves to split. We then update leaves by removing the node that was converted to an internal node and adding the resulting two new leaf nodes (note that adding each new node requires sampling an additional n points to estimate the gain and optimal potential split in that subpopulation).

Sampling points. Finally, we describe how our algorithm samples $x \sim \mathcal{P} \mid C$, where C is a conjunction of axis-aligned constraints corresponding to some leaf node:

$$C = (x_{i_1} \leq t_1) \wedge ... \wedge (x_{i_k} \leq t_k) \wedge (x_{j_1} > s_1) \wedge ... \wedge (x_{j_h} > s_h).$$

In general, some inequalities in C may be redundant, so we first simplify the expression. First, for two constraints $x_i \leq t$ and $x_i \leq t'$ such that $t \leq t'$, the first constraint implies the second, so we can discard the latter. Similarly, for two constraints $x_i > s$ and $x_i > s'$ such that $s \geq s'$, we can discard the latter. Second, given two constraints $x_i \leq t$ and $x_i > s$, we can assume that $t \geq s$; otherwise C is unsatisfiable, so the gain (1) would have been zero and the algorithm would have terminated. In summary, we can assume C contains at most one inequality $(x_i \leq t)$ and at most one inequality $(x_i > s)$ per $i \in [d]$, and if both are present, then the two are not mutually exclusive. For simplicity, we assume C contains both inequalities for each $i \in [d]$:

$$C = (s_1 \leq x_1 \leq t_1) \wedge ... \wedge (s_d \leq x_d \leq t_d).$$

Now, recall that $\mathcal{P}$ is a mixture of axis-aligned Gaussians, so it has probability density function

$$p_{\mathcal{P}}(x) = \sum_{j=1}^{K} \phi_j \cdot p_{\mathcal{N}(\mu_j, \Sigma_j)}(x) \ = \sum_{j=1}^{K} \phi_j \prod_{i=1}^{d} p_{\mathcal{N}(\mu_{ji}, \sigma_{ji})}(x_i),$$

where $\sigma_{ji} = (\Sigma_j)_{ii}$. The conditional distribution is

$$\begin{aligned} p_{\mathcal{P}|C}(x) &\propto \sum_{j=1}^{K} \phi_j \prod_{i=1}^{d} p_{\mathcal{N}(\mu_{ji}, \sigma_{ji})|C}(x_i) \\ &= \sum_{j=1}^{K} \phi_j \prod_{i=1}^{d} p_{\mathcal{N}(\mu_{ji}, \sigma_{ji})|(s_i \leq x_i \leq t_i)}(x_i). \end{aligned}$$

Since the Gaussians are axis-aligned, the unnormalized probability of each component is

$$\begin{aligned} \tilde{\phi}_j' &= \int \phi_j \prod_{i=1}^{d} p_{\mathcal{N}(\mu_{ji}, \sigma_{ji})|(s_i \leq x_i \leq t_i)}(x_i) dx \\ &= \phi_j \prod_{i=1}^{d} \left(\Phi\left(\frac{t_i - \mu_{ji}}{\sigma_{ji}} \right) - \Phi\left(\frac{s_i - \mu_{ji}}{\sigma_{ji}} \right) \right), \end{aligned}$$

where Φ is the cumulative density function of the standard Gaussian distribution $\mathcal{N}(0,1)$. Then, the normalization constant is $Z = \sum_{j=1}^{K} \tilde{\phi}'_j$, and the component probabilities are $\tilde{\phi} = Z^{-1}\tilde{\phi}'$. Thus, to sample $x \sim \mathcal{P} \mid C$, we sample $j \sim \text{Categorical}(\tilde{\phi})$, and

$$x_i \sim \mathcal{N}(\mu_{ji}, \sigma_{ji}) \mid (s_i \leq x_i \leq t_i) \quad \text{for each } i \in [d].$$

We use standard algorithms for sampling truncated Gaussian distributions to sample each x_i.

Remark 2. Note that we construct decision trees greedily, following the widely-used approach of Breiman et al. (1984). Our active learning strategy can be adapted around a non-greedy decision learning algorithm (e.g., Bertsimas and Dunn 2017) as well. However, we prefer the greedy algorithm because (i) it is significantly more scalable and thus, more user-friendly, and (ii) we believe the greedy algorithm is more intuitive as an interpretation since it routes points based on the most explanatory feature (which we believe mimics human reasoning). The main advantages of non-greedy approaches are that they do not overfit to data as easily, and produce smaller trees. We prove in Section 4 that our algorithm does not overfit as long as we sample enough points, and our user study in Section 5.3 demonstrates that humans are able to reason more accurately about our interpretations compared to non-greedy rule lists despite our trees being significantly larger.

4. Theoretical Guarantees

We now show that our interpretation converges asymptotically as the number of sampled points n grows large. To provide intuition, we begin with the case where all features are binary (§4.1). In this setting, we show that an interpretation of depth $D = d+1$ (where d is the dimension of the observed features) is equivalent to the underlying (nonparametric) blackbox model with high probability as long as a sufficient number of samples are drawn in each round (Theorem 1). It is well known that a nonparametric function over $\{0,1\}^d$ requires $\mathcal{O}(2^d)$ degrees of freedom to express; equivalently, a tree of depth $d+1$ has $\mathcal{O}(2^d)$ leaves. Theorem 1 demonstrates that our interpretation recovers any nonparametric function exactly with high probability as the number of samples grows large. Note that this simple setting often holds since many features are binary in practice (e.g., indicator variables for diagnoses).

However, it is not always feasible to use an interpretation of depth $d+1$, particularly in "big data" settings when d is very large. In our user studies, we observed that users found it difficult to reason with trees with more than 32 leaves (corresponding to depth $D = 5$ for uniform trees). Thus, we introduce the notion of an "exact tree" of size k, which is the greedy decision tree with k leaves and zero estimation error (i.e., no overfitting to statistical noise). When the features are binary, we prove that the exact tree of depth $d+1$ is equivalent to an arbitrary blackbox model

(Lemma 2). In §4.2, we consider general features, and prove that our interpretation is arbitrarily close to the exact tree as long as sufficient samples are drawn in each round (Theorem 2).

Exact Tree. We begin by describing the construction of the exact decision tree. The construction mirrors that of the estimated tree in our algorithm, but we use the exact value of f integrated over the distribution $\mathcal{P}$ (which cannot be computed in practice since we only have blackbox access to f) instead of sampling points.

We initialize a tree with a single leaf node $N_{T^*} = (y_0^*)$, where y_0^* is the majority label of f on the data distribution $\mathcal{P}$ on $\mathcal{X}$. Then, in each iteration, we choose the leaf node $N = (y)$ with the highest gain in the current tree and replace it with an internal node $N' = (N_L, N_R, C)$. Here, the constraint $C = (x_{i^*} \leq t^*)$ is computed using the exact distribution $\mathcal{P} \mid C_N$ instead of $\hat{\mathcal{P}}_N$:

$$(i^*, t^*) = \operatorname*{arg\,max}_{i\in[d],t\in\mathbb{R}} G(i,t;\mathcal{P} \mid C_N)\,. \tag{4}$$

Again, in the gain function G (defined in Eq. 1), we take $\mathcal{Q}$ to be the exact distribution $\mathcal{P} \mid C_N$ instead of $\hat{\mathcal{P}}_N$. This results in the new exact leaf node labels:

$$\begin{aligned} y_L &= \operatorname*{arg\,max}_{y\in\mathcal{Y}} \Pr_{x\sim\mathcal{P}|C_N}[f(x) = y \mid C_N \wedge (x_{i^*} \leq t^*))] \\ y_R &= \operatorname*{arg\,max}_{y\in\mathcal{Y}} \Pr_{x\sim\mathcal{P}|C_N}[f(x) = y \mid C_N \wedge (x_{i^*} > t^*))]. \end{aligned} \tag{5}$$

As before, we iterate $k-1$ times, replacing the node with the highest gain $G(i^*, t^*; \mathcal{P} \mid C_N)$.

4.1. Binary Features

In this section, we consider classification trees with binary features and binary outcomes. In particular, let $f : \mathcal{X} \to \mathcal{Y}$, where $\mathcal{X} = \{0,1\}^d$ and $\mathcal{Y} = \{0,1\}$, and let T^* be the exact greedy decision tree of depth D. In this case, when constructing the exact greedy decision tree T^*, it suffices to restrict to $t = 0.5$ in the optimization problem

$$i^*, t^* = \operatorname*{arg\,max}_{i,t} G(i,t)$$

used to choose the branch condition $x_i \leq t$ labeling each node N in T^*. In particular, note that the gain $G(i,t)$ and the sets $\mathcal{F}(C_N \wedge x_i \leq t)$ and $\mathcal{F}(C_N \wedge x_i > t)$ are all constant for all $t \in [0,1)$. Thus, it suffices to consider a single choice; we choose $t = 0.5$. Furthermore, for all $t \in (-\infty, 0)$, the branch $x_i \leq 0$ is satisfied by none of the $x \in \mathcal{X}$, so $\Pr_{x\sim\mathcal{P}}[C_N \wedge x_i \leq t] = 0$; similarly, for all $t \in [1,\infty)$, the branch $x_i > t$ is satisfied by none of the $x \in X$. Therefore, our algorithm ignores these choices. Thus, it suffices to consider $t = 0.5$, as claimed. For simplicity, we write $G(i) = G(i, 0.5)$.

Definition 5. We say the exact greedy decision tree T^* is Δ-gapped if (i) for each internal node N, we have $G(i^*) \geq G(i') + \Delta$, where i^* is the maximizer of $G(i)$, and i' is the maximizer of

$G(i)$ subject to $i \neq i^*$, and (ii) for each leaf node N, we have $F(y^*) \geq F(y') + \Delta$. Here we define $F : \mathcal{Y} \to \mathbb{R}$ as

$$F(y) = \mathrm{Pr}_{x \sim \mathcal{P}}[f(x) = y \mid C_N],$$

where y^* is the maximizer of $F(y)$, and y' is the maximizer of $G(y)$ subject to $y \neq y^*$ (i.e., $y' = 1 - y$).

Definition 6. For functions $f, g : \mathcal{X} \to \mathcal{Y}$, we write $f \equiv g$ if and only if $f(x) = g(x)$ for all $x \in \mathcal{X}$.

Theorem 1 **(Discrete Features)**. *Let $f : \mathcal{X} \to \mathcal{Y}$ be any function, let T be the estimated greedy decision tree of depth $d+1$ trained on*

$$n_{tot} = O(\Delta^{-2} 2^{d+3\log d} \log \delta^{-1}),$$

samples, and assume that the exact greedy decision tree is Δ-gapped. Then, $Pr[T \equiv f] \geq 1 - \delta$.

Proof of Theorem 1 This result follows from Lemmas 2 and 3, given in Appendix A. □

4.2. General Features

We now consider general features, and show that for any fixed tree size k, our algorithm extracts a decision tree that is arbitrarily close to the exact decision tree of the same size for sufficiently large n. The key challenge to establishing this result is that the branches in a greedy decision tree are estimated by maximizing a non-convex objective function (the gain). As a result, even very small errors in the estimated objective function (i.e., errors in $\hat{\mathcal{P}}_N$) can dramatically change its maximizer (the chosen leaf node N to split, as well as the constraint given by i^*, t^*). Under mild technical conditions, we establish that asymptotically, the estimated objective converges uniformly to the true objective with high probability, and consequently, the maximizer converges as well.

Assumptions. We first make a mild assumption about the distribution $\mathcal{P}$:

Assumption 1. The probability density function $p(x)$ of the distribution $\mathcal{P}$ over $\mathcal{X}$ is continuous, bounded (i.e., $p(x) \leq p_{\max}$), and has bounded domain (i.e., $p(x) = 0$ for $\|x\|_\infty > x_{\max}$).

This is a standard assumption since features are almost always bounded in practice. To satisfy this, we can simply truncate the Gaussian mixture models in our algorithm to $\mathcal{X} = \{x \in \mathbb{R}^d \mid \|x\|_\infty \leq x_{\max}\}$, for some $x_{\max} \in \mathbb{R}$.

Our next assumption ensures that the exact tree is well-defined:

Assumption 2. The maximizers (i^*, t^*) in (4), and y_L and y_R in (3) are unique.

In other words, there are no nodes where the gain for two different choices of a branch are exactly tied; such a tie is very unlikely in practice since the gain is real-valued.

Convergence. We now define the notion in which the extracted tree converges to the exact tree. For simplicity of notation, we additionally assume that we are learning *complete* trees (i.e., the maximum gain is not zero during any iteration) with depth D (i.e., $k = 2^{D-1}$ leaf nodes).

Definition 7. Let T, T' be complete decision trees of depth D. For $\epsilon > 0$, we say T is an ϵ approximation of T' if

$$\Pr_{x \sim \mathcal{P}}[T(x) = T'(x)] \geq 1 - \epsilon\,.$$

Let T^* be the exact tree that is complete with depth D. For any $\epsilon, \delta > 0$, we say T is (ϵ, δ)-exact if

$$\Pr[T \text{ is an } \epsilon \text{ approximation of } T^*] \geq 1 - \delta,$$

where the randomness is taken over the training samples $x \sim \mathcal{P}$.

Main Result. We now state our main result:

Theorem 2 **(General Features)**. For any $\epsilon, \delta > 0$, there exists $n \in \mathbb{N}$ such that the decision tree extracted using n samples per node is (ϵ, δ)-exact.

4.3. Proof Overview of Theorem 2

At a high level, the idea behind our proof of Theorem 2 is to show that the internal structure of T converges to that of T^*. Intuitively, this holds because as we increase the number of samples used to estimate T, the parameters (i, t) of each internal node of T and the parameters (y) of each leaf node of T should converge to the parameters of T^*. As long as the internal node parameters converge, then an input $x \in \mathcal{X}$ should be routed to leaf nodes in T and T^* at the same position. Then, as long as the internal node parameters converge, x should furthermore be assigned the same label by T and T^*.

The main challenge is that the internal node parameter t is continuous, so T always has some error compared to T^*. Furthemore, errors that occur in early branches of the tree propagate to lower branches and may be potentially magnified due to the non-convexity of the gain. Thus, to prove Theorem 2, we have to quantify this error and show that it goes to zero as n goes to infinity. We quantify this error as the probability that an input is routed to the wrong leaf node in T.

Our main lemma formalizes this notion. We begin by establishing some notation. Consider a node N^* in the exact decision tree T^*. We define the function $\phi : T^* \to T$ to map N^* to the node $N = \phi(N^*)$ at the corresponding position in the estimated decision tree T estimated using n samples. Now, given an input $x \in \mathcal{X}$, we write $x \xrightarrow{T^*} N^*$ if x is routed to node N^* in T^*, and similarly $x \xrightarrow{T} N$ if x is routed to node N in T. Finally, we denote the leaves of T^* and T by $\mathsf{leaves}(T^*)$ and $\mathsf{leaves}(T)$, respectively.

Then, we have the following key result:

Lemma 1. *Let $p(x)$ be the probability density function for the distribution $\mathcal{P}$, let $N^* \in T^*$ and $N = \phi(N^*)$, and let*

$$p_{N^*}(x) = p(x) \cdot \mathbb{I}[x \xrightarrow{T^*} N^*]$$
$$p_N(x) = p(x) \cdot \mathbb{I}[x \xrightarrow{T} N].$$

Then, $\|p_N - p_{N^*}\|_1$ *converges in probability to* 0 *(where the randomness is taken over the* n *samples used to extract* T*), i.e., for any* $\epsilon, \delta > 0$*, there exists* $n > 0$ *such that*

$$\|p_N - p_{N^*}\|_1 \le \epsilon$$

with probability at least $1 - \delta$*.*

Intuitively, p_{N^*} captures the distribution of points that are routed to N^* in T^*, and p_N captures the distribution of points that are routed to N in T. Then, this lemma says that the distribution of points routed to N^* and N are similar. We prove this lemma in Section B.1.

4.4. Proof of Theorem 2

We now use Lemma 1 to prove Theorem 2. In particular, we must show that the quantity $P = \Pr_{x\sim\mathcal{P}}[T(x) \ne T^*(x)]$ is bounded by ϵ with probability at least $1 - \delta$. Throughout the proof, we use Lemma 1 with parameters $\left(\frac{\epsilon}{K}, \frac{\delta}{2K}\right)$, i.e., we have $\|p_N - p_{N^*}\|_1 \le \frac{\epsilon}{K}$ with probability at least $1 - \frac{\delta}{2K}$. By a union bound, this fact holds for every leaf node in T^* with probability at least $1 - \frac{\delta}{2}$.

Then, our proof proceeds in two steps:

1. We show that a leaf node $N \in T$ is correctly labeled as long as ϵ is sufficiently small. More precisely, let $N^* \in \mathsf{leaves}(T^*)$ such that $N^* = (y^*)$, and let $N = \phi(N^*) \in \mathsf{leaves}(T)$ such that $N = (y)$; then, we show that for any $\delta' > 0$, there exists $n \in \mathbb{N}$ such that $y = y^*$ with probability at least $1 - \delta'$ (where the randomness is taken over the n samples used to extract T).

2. Using the Lemma 1 together with the first step, we show that $P \le \epsilon$ with probability at least $1 - \delta$.

Proving $y = y^*$***.*** Let $p(x)$ be the probability density function for the distribution $\mathcal{P}$, and let $N^* \in \mathsf{leaves}(T^*)$ such that $N^* = (y^*)$ and $N = \phi(N^*) \in \mathsf{leaves}(T)$ such that $N = (y)$. First, we rewrite the objective (5) in terms of p_{N^*}. In particular, for each $y' \in \mathcal{Y}$, let

$$\begin{aligned} p^*_{y'} &= \Pr_{x\sim\mathcal{P}}[f(x) = y' \wedge (x \xrightarrow{T^*} N^*)] \\ &= \int \mathbb{I}[f(x) = y'] \cdot \mathbb{I}[x \xrightarrow{T^*} N^*] \cdot p(x)dx. \end{aligned}$$

Then, we have $y^* = \arg\max_{y'\in\mathcal{Y}} p^*_{y'}$, since the denominator $\Pr_{x\sim\mathcal{P}}[x \xrightarrow{T^*} N^*]$ in (5) is constant with respect to y'. Similarly, we rewrite the objective (3) in terms of p_{N^*}, letting

$$p_{y'} = \frac{1}{n}\sum_{j=1}^{n} \mathbb{I}[f(x^{(j)}) = y'] \cdot \mathbb{I}[x^{(j)} \xrightarrow{T} N]$$

for each $y' \in \mathcal{Y}$, in which case we have $y = \arg\max_{y'\in\mathcal{Y}} p_{y'}$.

By Assumption 2, we know that y^* is the unique maximizer of p^*_y, i.e.,

$$\Delta = p^*_{y^*} - \arg\max_{y'\ne y^*} p^*_{y'} > 0.$$

Therefore, to show that $y = y^*$, it suffices to show that for each $y' \in \mathcal{Y}$, we have

$$|p_{y'} - p^*_{y'}| \le \frac{\Delta}{3},$$

since then, for each $y' \in \mathcal{Y}$, we have

$$p_{y^*} - p_{y'} \ge \left(p^*_{y^*} - \frac{\Delta}{3}\right) - \left(p^*_{y'} + \frac{\Delta}{3}\right) \ge \frac{\Delta}{3} > 0,$$

which implies that $y = y^*$ since y^* is the maximizer of $p_{y'}$.

To show that $|p_{y'} - p^*_{y'}| \le \Delta/3$, we first define

$$\tilde{p}_{y'} = \int \mathbb{I}[f(x) = y'] \cdot \mathbb{I}[x \xrightarrow{T} N] \cdot p(x) dx.$$

Then, we have

$$|p_{y'} - p^*_{y'}| \le |p_{y'} - \tilde{p}_{y'}| + |\tilde{p}_{y'} - p^*_{y'}|.$$

To bound the first term, let

$$d_{y'} = \mathbb{I}[f(x) = y'] \cdot \mathbb{I}[x \xrightarrow{T} N]$$

be a Bernoulli random variable, so

$$d^{(j)}_{y'} = \mathbb{I}[f(x^{(j)}) = y'] \cdot \mathbb{I}[x^{(j)} \xrightarrow{T} N]$$

are samples of $d_{y'}$ for $j \in [n]$. Then, we have $\tilde{p}_{y'} = \mathbb{E}[d_{y'}]$ and $p_{y'} = n^{-1} \sum_{j=1}^{n} d^{(j)}_{y'}$, so we can apply Hoeffding's inequality to get

$$\Pr\left[|p_{y'} - \tilde{p}_{y'}| > \frac{\Delta}{6}\right] \le 2 \exp\left(-\frac{n\Delta^2}{18}\right).$$

To bound the second term, note that

$$\begin{aligned}
|\tilde{p}_{y'} - p^*_{y'}| &= \left| \int \mathbb{I}[f(x) = y'] \cdot (\mathbb{I}[x \xrightarrow{T} N] - \mathbb{I}[x \xrightarrow{T^*} N^*]) \cdot p(x) dx \right| \\
&\le \int |\mathbb{I}[x \xrightarrow{T} N] - \mathbb{I}[x \xrightarrow{T^*} N^*]| \cdot p(x) dx \\
&= \|p_N - p_{N^*}\|_1 \\
&\le \epsilon.
\end{aligned}$$

Finally, assume that $\epsilon < \Delta/6$; then, taking a union bound over $y' \in \mathcal{Y}$, we have that

$$|p_{y'} - p^*_{y'}| \le \frac{\Delta}{3}$$

for all $y' \in \mathcal{Y}$ with probability at least $1 - \delta'$, where

$$\delta' = 2 \cdot |\mathcal{Y}| \cdot \exp\left(-\frac{n\Delta^2}{18}\right).$$

In particular, it follows that $y = y^*$ with probability at least $1 - \delta'$.

Bounding P. First, we separate the contribution of each leaf node to P:

$$\begin{aligned}P &= \Pr_{x\sim\mathcal{P}}[T(x)\neq T^*(x)]\\ &= \sum_{N^*\in\mathsf{leaves}(T^*)} \Pr_{x\sim\mathcal{P}}[T(x)\neq T^*(x) \text{ and } x \xrightarrow{T^*} N^*].\end{aligned}$$

Next, we apply the result from the first step of this proof with parameter $\delta' = \frac{\delta}{2K}$ (where K is the number of nodes in each T^* and T); then, for any leaf node $N^*\in\mathsf{leaves}(T^*)$, the label assigned to N^* equals the label assigned to N with probability at least $1-\frac{\delta}{2K}$. Taking a union bound over the leaf nodes, this fact holds true for all the leaf nodes with probability at least $1-\frac{\delta}{2}$. For the remainder of the proof, we assume that this fact holds.

Consider an input x such that $x\xrightarrow{T^*}N^*$; as long as N^* and $\phi(N^*)$ have the same label, and additionally $x\xrightarrow{T}\phi(N^*)$, then $T(x)=T^*(x)$. Thus, we have

$$\Pr_{x\sim\mathcal{P}}[T(x)\neq T^*(x) \text{ and } x\xrightarrow{T^*}N^*] \leq \Pr_{x\sim\mathcal{P}}[\neg(x\xrightarrow{T}\phi(N^*)) \text{ and } x\xrightarrow{T^*}N^*].$$

As a consequence, we have

$$\begin{aligned}P &\leq \sum_{N^*\in\mathsf{leaves}(T^*)} \Pr_{x\sim\mathcal{P}}[\neg(x\xrightarrow{T}\phi(N^*)) \text{ and } x\xrightarrow{T^*}N^*]\\ &= \sum_{N^*\in\mathsf{leaves}(T^*)} \int (1-\mathbb{I}[x\xrightarrow{T}\phi(N^*)])\cdot\mathbb{I}[x\xrightarrow{T^*}N^*]\cdot p(x)dx.\end{aligned}$$

Now, we claim that

$$(1-\mathbb{I}[x\xrightarrow{T}\phi(N^*)])\cdot\mathbb{I}[x\xrightarrow{T^*}N^*] \leq |\mathbb{I}[x\xrightarrow{T^*}N^*]-\mathbb{I}[x\xrightarrow{T}\phi(N^*)]|.$$

To see this claim, note that both sides of inequality take values in $\{0,1\}$. Furthermore, the right-hand side equals 0 only if the two indicators are equal. In this case, the left-hand side also equals 0, so the claim follows. Thus, we have

$$\begin{aligned}P &\leq \sum_{N^*\in\mathsf{leaves}(T^*)} \int |\mathbb{I}[x\xrightarrow{T^*}N^*]-\mathbb{I}[x\xrightarrow{T}\phi(N^*)]|\cdot p(x)dx\\ &= \sum_{N^*\in\mathsf{leaves}(T^*)} \int |\mathbb{I}[x\xrightarrow{T^*}N^*]\cdot p(x)-\mathbb{I}[x\xrightarrow{T}\phi(N^*)]\cdot p(x)|dx\\ &= \sum_{N^*\in\mathsf{leaves}(T^*)} \|p_{\phi(N^*)}-p_{N^*}\|_1.\end{aligned}$$

By Lemma 1, we have

$$P \leq \sum_{N^*\in\mathsf{leaves}(T^*)} \frac{\epsilon}{K} \leq \epsilon.$$

Since Lemma 1 holds with probability at least $1-\frac{\delta}{2}$, and the first part of this proof holds with probability at least $1-\frac{\delta}{2}$, by a union bound, we have $P\leq\epsilon$ with probability at least $1-\delta$, which completes the proof. □

5. Evaluation

We illustrate a potential use case of our technique for predicting diabetes risk among patients on a real electronic medical record (EMR) dataset. In 2012, approximately 8.3% of the world's adult population had diabetes, which is a leading cause of cardiovascular disease, renal disease, blindness, and limb amputation (Läll et al. 2017). To make matters worse, an estimated 40% of diabetics in the US are undiagnosed, placing them at risk for major health complications (Cowie et al. 2009). At the same time, several clinical trials have demonstrated the potential to prevent type II diabetes among high-risk individuals through lifestyle interventions (Tuomilehto et al. 2011). Thus, there is significant interest in accurately predicting patients at risk for Type II diabetes in order to prescribe lifestyle interventions. We learn an effective random forest classifier for this task (out-of-sample AUROC = 0.84), and extract an interpretation from this model.

First, we find that our interpretation is much more accurate in mimicing the random forest (i.e., higher fidelity) compared to vanilla decision trees (Breiman et al. 1984), sparse logistic regression (Tibshirani 1996), and state-of-the-art rule lists (Yang et al. 2017). Second, we perform a user study to assess how well individuals can understand the resulting interpretations, e.g., by computing counterfactuals or identifying relevant patient subpopulations. We find that humans are more accurate when answering similar questions about our extracted trees than the rule list. However, the proof is in the pudding and to this end, we interview several domain experts (physicians) using our interpretation to discover useful insights about the data and model; most importantly, they find an unexpected causal issue that is important to control for in the chosen prediction task.

5.1. Preliminaries

We obtained a database of patient electronic medical records from a leading EMR company. Our dataset contains patients from multiple providers; however, there is significant variation in how diagnoses and procedures are coded among providers, and so we primarily restrict our analysis to the largest provider who has 578 unique patients. In our discussions with physicians, we also include a separate model and corresponding interpretation from the next largest provider (with 402 unique patients), helping us gain insight into coding variations across providers.

For each patient, we constructed 382 features based on their EMR data from the last three years. These included demographic features (age, gender) and whether the patient had had one of the 200 most frequent diagnoses, been prescribed one of the 150 most frequent medications, and results from the 30 most frequent lab tests. The outcome variable was whether a patient had a type II diabetes diagnosis in their most recent visit (not included in the data); the data was pre-processed by experts to ensure that any relevant lab results and medications (from prior visits) that indicated a diabetes diagnosis were removed from the data.

Model	**Test Set AUROC**
Decision Trees (Breiman et al. 1984)	0.70
Sparse Logistic Regression (Tibshirani 1996)	0.79
Rule Lists (Yang et al. 2017)	0.80
Random Forest (Breiman 2001)	0.84

Table 1 **The performance (AUROC) of different machine learning classifiers for predicting diabetes risk.**

We used 70% of our data for training, and the remaining 30% as a test set. As is the case in many classification tasks, our dataset is very imbalanced: only 11.8% of the patients in the data have positive labels, i.e., have been diagnosed with diabetes. Thus, to improve precision, we use the standard trick of up-sampling all instances with positive labels to balance the training set; however, we maintain the test set in its original form. We use AUROC (area under the ROC curve) on the test set as our performance metric, and report values for decision trees, sparse logistic regression, rule lists, and random forests in Table 1. As expected, the random forest outperforms its more interpretable counterparts. This improvement in predictive accuracy is desirable to better target patients for lifestyle interventions; in particular, we wish to target as many risky patients as possible while avoiding burdening patients who are not at risk.

Thus, the provider may wish to deploy the blackbox random forest model. However, it is important to understand its behavior and verify its reasoning process with domain experts to ensure good performance on new patients. To interpret the random forest, we use our algorithm to extract a decision tree. We first fit a Gaussian mixture model $\mathcal{P}$ using the same training data used to estimate the random forest. Then, we use our algorithm to extract a decision tree by sampling 1000 new training points per node. The resulting interpretation is shown in Fig 2.

For comparison, we train a rule list on the blackbox random forest labels $(X_{train}, f(X_{train}))$ as proposed in Lakkaraju et al. (2017); see Fig. 3. We can immediately see that the decision tree interpretation is significantly richer, providing more insight into the random forest's reasoning process. This difference manifests itself in our upcoming fidelity comparison: the decision tree interpretation has much higher fidelity than the rule list interpretation.

Remark 3. Lakkaraju et al. (2017) actually propose training decision sets (Lakkaraju et al. 2016) on the blackbox labels rather than rule lists (Yang et al. 2017). However, the decision set learning algorithm does not scale to our dataset. In both cases, we use the original implementation provided by the authors (open-source for rule lists, and through private communication for decision sets). We found that the decision set learning algorithm generally does not scale well to datasets with many features; the datasets in the evaluation by Lakkaraju et al. (2016) only have tens of features, whereas our diabetes risk dataset has hundreds of features. In contrast, the algorithm for learning rule lists (which are closely related to decision sets) is designed to scale to large datasets.

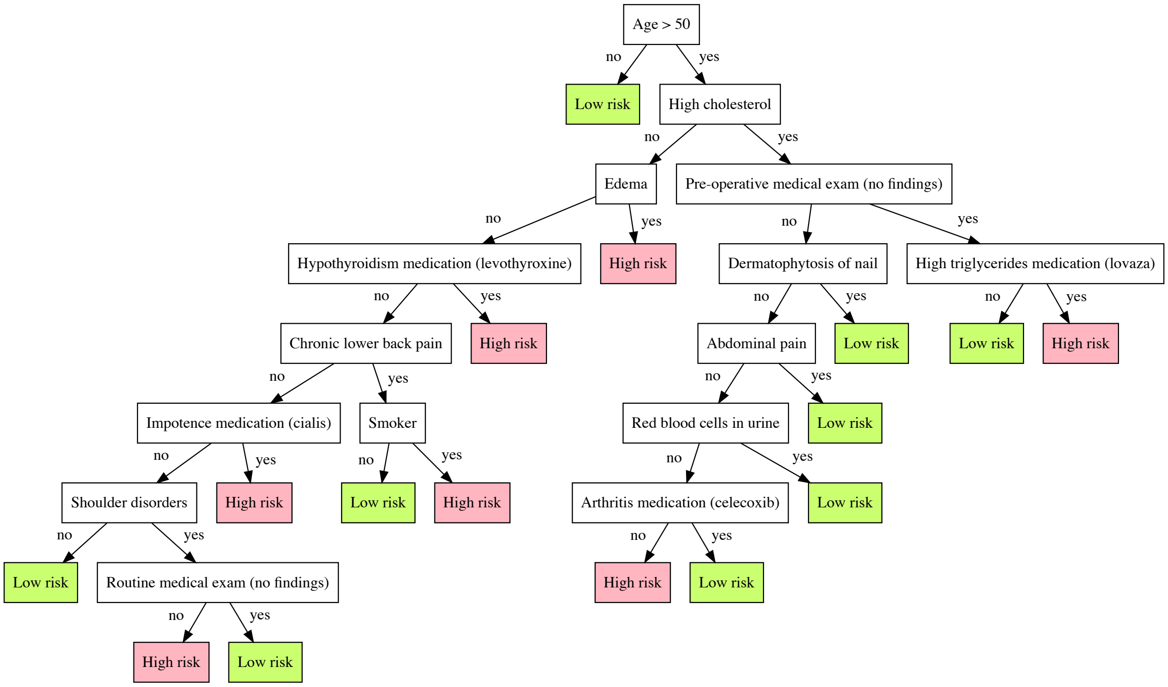

Figure 2 Our algorithm's extracted interpretation of the blackbox random forest diabetes risk classifier.

if Age $<$ 41 **then** Low risk
else if Moderate/severe pain medication (tramadol) **then** High risk
else if Arthritis medication (etodolac) **then** Low risk
else if High cholesterol **and** Smoker **then** High risk
else if High blood pressure **then** High risk
else if Age $<$ 53 **then** Low risk
else if Restless legs syndrome **then** Low risk
else if not High cholesterol **then** Low risk
else High risk

Figure 3 Rule list interpretation of the blackbox random forest diabetes classifier.

We note that in our experiments, we find that our algorithm is still roughly 3-4 times faster than the rule list learning algorithm.

5.2. Fidelity

First, we evaluate fidelity (Definition 4), which measures the AUROC of an interpretation relative to the predictions of the blackbox random forest. Achieving high fidelity is important, because it ensures that the insights obtained from the interpretation actually hold for the blackbox model that we hope to deploy in practice. Fig. 4(a) shows the fidelity of rule lists (Yang et al. 2017), vanilla decision trees (Breiman et al. 1984), and our extracted tree interpretation. We find that our extracted trees are significantly better at mimicing the blackbox model, thus providing a much richer understanding of the random forest to an expert.

Next, we consider how fidelity improves with the size of our interpretation (the number of leaves in our extracted tree). The size of the rule lists cannot be modified easily, so we compare ourselves to vanilla decision trees. As can be seen, our extracted tree outperforms the vanilla decision tree for every size. Moreover, we are able to produce better and better approximations of the blackbox model as the size of our interpretation is allowed to grow; thus, the expert can easily choose a tradeoff between fidelity (accuracy of insights) and the bulkiness of the interpretation.

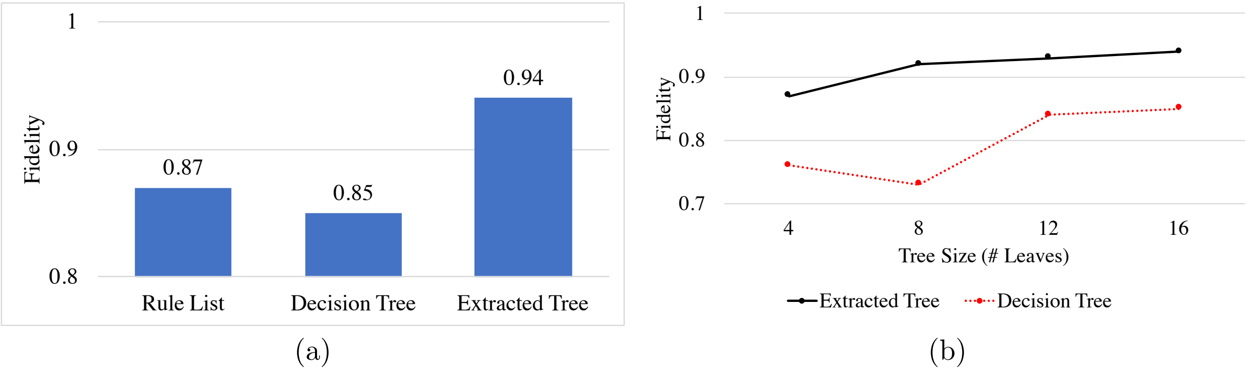

Figure 4 Fidelity to the random forest diabetes risk classifier. (a) compares the fidelity of rule lists, a decision tree, and an extracted tree using our approach. (b) shows fidelity as a function of the size of the tree.

5.3. Human Accuracy on Interpretations

Aside from fidelity, an equally important element for achieving effective human-in-the-loop analytics is the ability for humans to accurately reason about the interpretation. Given an interpretation, an expert may wish to compute counterfactuals or identify relevant subpopulations in order to better understand the consequences of deploying the blackbox machine learning model. Then, there are two potential sources of errors in their resulting insights: (i) error from the interpretation's approximation of the blackbox model, and (ii) error from the human's comprehension of the interpretation. We've established that our extracted trees produce smaller error in the former category (higher fidelity), so we now study the latter category (human comprehension).

We perform a user study to evaluate human ability to reason about our proposed interpretation versus state-of-the-art rule lists. The goal of our approach is to enable experts to better understand and validate blackbox models; thus, we recruited 46 graduate students with some background in machine learning or data science to participate in our study. Each participant answered questions intended to test their understanding of the rule list (shown in Fig. 3) and our extracted tree interpretation (shown in Fig. 2) of the diabetes risk classifier. We note that the purpose of this study is only to evaluate how well users can comprehend the given interpretations, and not to obtain insights or validate the model. Our study participants are not medical experts (we discuss insights from physicians in the next subsection), and we do not assume any prior medical knowledge to correctly answer our survey questions.

Consider patients over 50 years old who are otherwise healthy and are not taking any medications. According to the decision tree, are these patients at a high risk for diabetes?

- Yes
- No

Smoking is known to increase risk of diabetes, so the local hospital has started a program to help smokers quit smoking. According to the decision tree, which patient subpopulation should we target in this program if we want to reduce diabetes risk?

- Patients over 50 years old with high cholesterol
- Patients over 50 years old with chronic lower back pain
- Patients over 50 years old with high cholesterol, edema, chronic lower back pain, and who take medication for hypothyroidism

Consider patients over 53 years old who are otherwise healthy and are not taking any medications. According to the rule list, are these patients at a high risk for diabetes?

- Yes
- No

Smoking is known to increase risk of diabetes, so the local hospital has started a program to help smokers quit smoking. According to the rule list, which patient subpopulation should we target in this program if we want to reduce diabetes risk?

- Patients over 41 years old
- Patients over 41 years old with high cholesterol
- Patients over 41 years old with high cholesterol, and take medication for arthritis

Figure 5 Examples of questions asked in our user study on the diabetes risk classifier for our extracted decision tree (left) and for the rule list (right).

We designed five pairs of questions for the two types of interpretations; each pair was similar in construction and wording, but the exact question was adapted to the structure of the corresponding interpretation (ensuring that there is a single correct answer for each question based on the interpretation). Two examples of our questions are shown in Fig. 5; the variants on the left are for our extracted decision tree, and those on the right are for the rule list. (The remaining questions can be found in Appendix C.) The first question tests whether the user can determine how the model would classify a given patient. The second question tests whether the user is able to identify the subpopulation for which "Smoker" is a relevant feature; we believe that enabling end users to understand these subpopulation-level effects is a major benefit of global interpretations.

We randomized the order of the interpretations and the corresponding questions. Users were asked to skip a question if they were unable to determine the answer in 1-2 minutes. We averaged each user's accuracy over all 5 questions for each interpretation. The average human accuracy for rule lists versus our extracted decision tree are shown with standard errors in Fig. 6. Users responded more accurately when using our extracted tree, despite the fact that our tree was much larger than the rule list; this effect was statistically significant ($p = 0.02$ using a paired t-test clustered at the user level). Furthermore, a majority of users answered each question correctly, so we believe our questions were fair.

Upon examining the errors made by our users, we found that they had particular difficulty understanding the conditional structure of the rule list. For example, based on Fig. 3, if a patient is

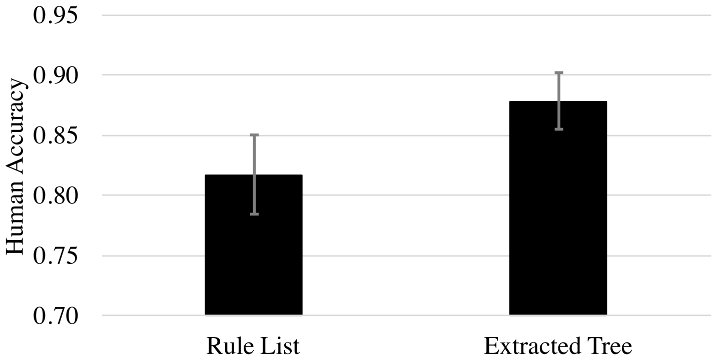

Figure 6 **Average human accuracy on similar questions designed around the rule list and our extracted tree interpretations; standard errors shown in grey bars. The difference is statistically significant (**$p = 0.02$**).**

taking arthritis medication, then only the first three rules are relevant (the else-if structure ensures that all remaining rules do not trigger). However, this proved challenging for users, and many continued to apply rules that were deeper down the rule list. On one such question, users answered correctly only 65% of the time using the rule list; in contrast, users were able to visualize which constraints triggered more easily in the decision tree, and correctly answered a similar question 91% of the time. This finding mirrors previous work showing that reasoning about long sequences of if-then-else rules can be difficult for humans (Lakkaraju et al. 2016). We note that our study is limited to a single dataset and a relatively small sample size (graduate students with some machine learning background are a highly specialized population); however, we believe it gives somewhat compelling evidence that our extracted trees are more or at least equally interpretable as rule lists.

5.4. Case Study on Physicians

Next, we interviewed three physicians about the proposed random forest diabetes risk classifier. We informed them of the goal (targeting interventions), our data curation and feature construction. They originally found this approach reasonable. We then presented them with the interpretation (see Fig. 2), upon which they made several observations. We also trained a separate random forest classifier and extracted a corresponding interpretation for the next largest provider (see Fig. 7); these results were useful in our discussions to understand variations across providers.

Endogeneity of diagnoses. One notable feature of our interpretation in Fig. 2 is the subtree rooted at the node labeled “Dermatophytosis of nail”. This subtree considers the patient subpopulation that is over 50, has high cholesterol, and has not had a pre-operative medical exam. Within this subpopulation, the default classification if the patient has no additional diagnoses is “high risk”. However, if the patient has dermatophytosis of nail, abdominal pain, red blood cells in urine, and/or is taking arthritis medication, then the decision tree classifies the patient as “low risk”. Our physicians found this effect surprising since these diagnoses have no known negative relationship to diabetes risk; if anything, dermatophytosis is more likely to occur in diabetic patients (Winston and Miller 2006), and so one might expect that patients with that diagnosis may have a somewhat higher risk of a diabetes diagnosis (rather than vice-versa). The physicians argued that, in general, additional diagnoses indicate poorer health and therefore higher risk of conditions such as diabetes; however, our extracted tree was predicting the opposite result within this patient subpopulation.

After this initial feedback, we checked that this effect is not simply an artifact of the extracted tree, but is indeed present (and quite strong) in our data. One may also be concerned that this effect may be specific to this provider or a chance finding on a particular patient subpopulation. In contrast, we found that this effect actually occurs among the other providers in our data as well. For example, it also occurs in the subtree rooted at "Chest pain" for the next largest provider (see Fig. 7). In this case, the patient subpopulation is over 48, has high blood pressure, and does not smoke (i.e., similarly high risk as the patient subpopulation considered earlier). Again, their default classification is "high risk" if they have no other diagnoses; however, if the patient has chest pain, muscle pain and inflammation, and/or takes anti-depressant medication, then they are classified as "low risk". Our experts found that this trend was highly similar to that we found with the original provider, and thus suggested that some systematic confounder was at play.

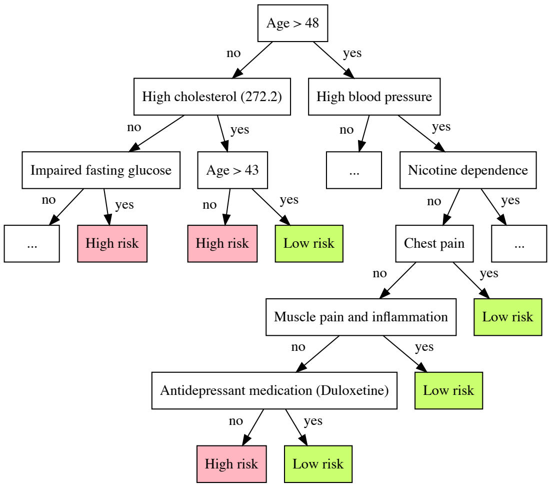

Figure 7 **Partial view of the extracted tree interpretation of the blackbox random forest diabetes classifier for the second largest provider. Ellipses represent subtrees that are not shown in the figure.**

After some thought, the physicians suggested a plausible explanation: patients with more diagnoses are also likely patients who frequently visit their healthcare provider. The patient subpopulation we are considering (i.e., over 50 years old with high cholesterol) already has some initial risk factors for diabetes; if they frequently visit their provider, their physician would have likely recommended pre-diabetic interventions that reduced the patient's risk for a diabetes diagnosis. On the other hand, patients who have not recently visited their provider may not have realized that they are at high risk for diabetes, and may not have been recommended lifestyle interventions. Thus, having additional diagnoses (even those that are unrelated to diabetes) may be correlated with lower diabetes risk since those patients receive greater attention and more interventions from their healthcare provider.

We investigated this hypothesis by looking at the fraction of patients with a positive label for diabetes as a function of the number of times they have visited their provider in the last year. We conditioned on the patient subpopulation that is over 50 years old and has high cholesterol to capture the fact that these patients already have some initial risk factors. Since we decided on this particular patient subpopulation based on results from the largest provider, we avoid overfitting to those results by only considering patients from the remaining providers in our data (9370 unique patients from 375 providers). The results are shown in Fig. 8.

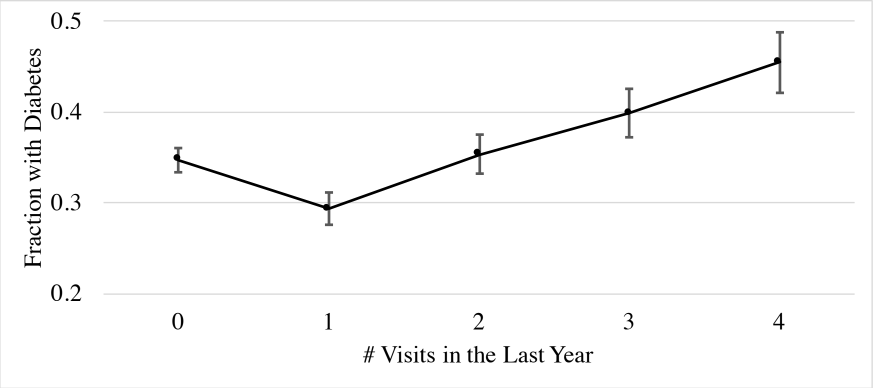

Figure 8 The fraction of patients diagnosed with diabetes (with standard errors) as a function of the number of visits to the provider in the last year, conditioned on being over 50 years old and having high cholesterol.

As the physicians suspected, we find a surprising V-shaped effect: while diabetes risk typically increases with the number of visits (since the patient is likely to be in poorer health), the risk is actually higher for patients with no provider visits in the last year compared to patients with a single visit. Thus, diagnoses appear to be endogenous explanatory variables that encode some (unobservable) physician effort at reducing a patient's diabetes risk. If we were to deploy the current predictive model to dictate which patients should receive interventions, a negative consequence may ensue. In particular, the model may recommend discontinuing interventions for some patients (e.g., those who are over 50, have high cholesterol, and are taking arthritis medication) because they have been classified as "low risk"; yet, these patients are only low risk in our data because they have historically received those interventions (but this was unobserved by our predictive model).

Our finding suggests that our interpretation was valuable for domain experts to form and test hypotheses about unexpected defects in the blackbox classifier. It is important to detect and understand such issues before deploying a model, especially in domains like healthcare. Controlling for this endogeneity to build a more reliable predictive model is beyond the scope of our paper.

We note that feature influence scores (Friedman 2001) are insufficient to tease out such an effect since they do not examine subpopulations, e.g., the effect we described only applies to the subpopulation of patients that are at least 50 and have high cholesterol. As an example, the correlation of "Abdominal pain" with diabetes in the overall population is 8.1×10^{-3}; however, within the subpopulation we consider, the correlation is -9.8×10^{-2}. In fact, none of the features in our subtree appear in the top 40 relative influence scores for the random forest.

Non-monotone dependence on age. Our physicians also found it interesting that age appears twice in the extracted decision tree for the second largest provider (Figure 7). They reasoned that younger patients are typically at lower risk for diabetes; however, conditioned on being less than 48 years old and having high cholesterol, the classifier predicts higher risk for younger patients. While we cannot be certain of the cause, they brainstormed a number of possible explanations. For example, a diagnosis of high cholesterol in younger patients is abnormal, and therefore it may suggest a much riskier disease trajectory for the patient. Alternatively, physicians are more likely to urge older patients with high cholesterol to take preventative measures to reduce diabetes risk (as prescribed by guidelines), and may not exert as much effort on younger patients.

We note that this structure demonstrates how our extracted decision tree can capture non-monotone dependencies on continuous features such as age. In contrast, non-monotone dependencies cannot be captured by feature influence scores. Rule lists can also capture such a dependence, but their restricted structure makes it more difficult to understand the effect ,e.g., to reason about the relationship between the first and sixth rules in the rule list (see Fig. 3), one has to reason about four intermediate rules, which proved to be a difficult task for users in our user study (§5.3).

Variations across providers. Our physicians also remarked on another interesting feature that differed between the two providers. "Impaired fasting glucose" (which is a type of pre-diabetes) is a very predictive feature for the second provider, but not for the first provider. After examining our data, we found that this was because 10% of patients were diagnosed with impaired fasting glucose for the second provider, but only 1% of patients had this diagnosis for the first provider. Thus, it may be the case that the second provider was more diligent about screening for pre-diabetes, or recording that information in the EMR. Understanding such differences among patient populations for different providers can aid data scientists in adapting existing models to new providers.

6. Discussion

Many academic papers (Doshi-Velez and Kim 2017) as well as popular media (O'Neil 2016) have claimed that interpretability is an important and necessary feature for predictive models deployed in practice. However, we wanted to gain an understanding of where interpretability fits (if at all) into the workflow of industry experts at large. Thus, we interviewed 17 industry experts from a variety of sectors (including healthcare, finance, retail, nonprofit, and consulting), who either currently work with or are planning on working with outputs from predictive algorithms. We now discuss their insights on the importance of interpretability, its role in their workflow, and the potential value of our proposed approach. We begin with situations where our approach is not applicable (interpretability is not important, or the application demands a local interpretation), and then discuss situations where our approach is applicable (global interpretation can be valuable).

No Interpretability. First, there was a consensus that interpretability is not important in "low stakes" settings such as online product recommendations (e.g., on Amazon or Netflix) or marketing campaigns. This is because poor decisions are not considered very costly in these applications. Furthermore, these firms liberally use A/B testing to ensure that their deployed models are indeed profitable to the company (thus avoiding causal issues), and A/B testing has a quick turnaround time enabling them to correct poor decisions almost immediately.

Second, interpretability was not considered useful in settings where decision-makers had limited expertise about the prediction task. We spoke to decision-makers at nonprofits who are interested in replacing costly survey-based methods with machine learning methods that produce reasonable estimates with cheap or public data. For example, one expert described using publicly-reported government indicators to predict regions with large numbers of out-of-school girls, while another described recent work by Jean et al. (2016) using satellite imagery to predict regions with the most poverty. This information is used to assign interventions to regions with the highest need. However, the decision-makers at these nonprofits had limited prior knowledge of how one might choose these regions, and as a result, did not feel that they could assess if the model was producing erroneous predictions by examining an interpretation. In other words, interpretations are more valuable when the decision-maker has regularly been making similar predictions as the algorithm, and thus, has some knowledge of what reasonable predictions should look like.

Local Interpretability. Several experts we interviewed thought that local per-prediction interpretations (e.g., Ribeiro et al. 2016) could be important for incorporating auxiliary information. One dermatologist referred to recent work by Esteva et al. (2017), which uses blackbox deep learning on images of skin lesions to predict whether a cancer is malignant or benign. Such methods are hard to combine with auxiliary information that is available only to the physician and not the algorithm. For instance, a dermatologist may look for similar skin patterns on other parts of the patient's skin, whose presence may signal a benign rather than malignant condition. In this case, an interpretation of why the algorithm thinks the cancer is benign/malignant may enable the dermatologist to better combine his/her auxiliary information with the algorithm's reasoning.

Another concern was that there are occasionally errors in the data inputs for the predictive model. For instance, two physicians noted that a predictive model may incorrectly produce warnings for patients based on vital signs from a sensor that had been accidentally knocked off. Similarly, experts from the financial industry noted examples where a model may recommend underwriting risk for an account that is already frozen (unbeknownst to the algorithm) or dramatically change its predictions upon observing that a store has lost 95% of its sales in one day (when any human would recognize that this was simply a typo). These experts believe that per-prediction interpretations can help the decision-maker easily identify such errors.

Global Interpretability. The majority of the experts we interviewed agreed that a global approximation of the blackbox model (our contribution) would be valuable in their workflow. First, aligned with the findings of behavioral studies by Yeomans et al. (2016), they said upper-level management is much more likely to trust and approve a predictive model if they can understand how it produces its predictions. Similarly, Ferreira et al. (2015) note that the use of an interpretable predictive pricing model helped ensure a higher likelihood of adoption by Rue La La management and merchants; in our interviews, experts believed that similar benefits of trust and adoption can be realized with an approximate global interpretation as well.

Experts also believe that global interpretations can aid in diagnosing errors during the development of the model. We spoke to two pricing experts who seek to predict customers' willingness-to-pay, which is never actually observed in the data; rather, they use a variety of proxies (e.g., demand for ancillary products or price shocks) to build their models. In these cases, they believe an interpretation can help them understand issues with their training data or how they constructed their proxy outcomes. Physicians are also concerned about biases in training data with respect to shifts in patient populations (e.g., the model is trained largely on Caucasians, but is intended to be deployed on Pacific Islanders), and believed interpretations may help them better understand whether the model may be appropriate or not.

Causal issues such as confounders (e.g., our case study) as well as "features from the future" were also raised. As an example, one data scientist at a food delivery startup told us that s/he had achieved excellent out-of-sample accuracy in predicting delays, but the deployed model did not perform well in the A/B test. After several iterations, the data scientist discovered that one of the features used in the model was actually measured in the *future* (after delivery) but the data was encoded in a feature that was intended to be measured *before* delivery. As a result, the model utilized information that did not exist yet, and performed well on observational data, but not in the real world. A physician mentioned a similar story, where an early warning system produced by machine learning experts had achieved promising performance on a test set, but failed upon deployment because it inadvertently relied on information from the future (since many timestamps in the electronic medical record are unreliably recorded). Both agreed that a global interpretation of the model would have helped diagnose these issues at a much earlier stage, since the futuristic features would have appeared much more predictive than the expert would have expected.

Transparency. When there is significant legal accountability, experts strongly prefer transparent predictive models over blackbox models, even at the cost of accuracy. These include financial settings (e.g., credit scoring) as well as healthcare settings (e.g., patient diagnosis), where firms/hospitals are answerable to legal authorities as to why each decision was made; consequently, they do not wish to rely on a blackbox model. In this case, they expressed interest in directly using our extracted decision tree (i.e., our interpretation) and discarding the blackbox model. Thus, our approach can also be viewed as a way to train more accurate decision trees.

7. Conclusion

We propose a novel approach for interpreting complex, blackbox machine learning models by extracting decision trees that effectively summarize their reasoning process. The key ingredient of our algorithm is the use of active learning, which we show can produce richer and more accurate interpretations than several baselines. We prove that our extracted decision tree converges to the exact decision tree, implying that we avoid overfitting. We evaluate our algorithm on a real electronic medical record dataset to demonstrate that it produces more accurate interpretations that are simultaneously easier for humans to comprehend. We then perform a case study on physicians regarding our diabetes risk prediction model, and describe a number of useful insights they derived using our interpretation, underscoring the value of interpretability. Finally, we interview a number of industry experts to better understand the role of interpretability in their workflow, and the potential value of our proposed approach in a variety of settings.

One important direction for future work is testing the operational value of interpretability in a field experiment. There has also been much discussion about the tension between offering experts too much information (which may result in cognitive overload) and too little (which may result in the expert being unable to factor in their domain knowledge effectively); this begs the question of how to optimize the design of interpretations to best facilitate overall improved decision-making.

Acknowledgments

This paper was partly written when H. Bastani was at IBM Thomas J. Watson Research and O. Bastani was at MIT Computer Science. This paper has benefitted from valuable feedback from Gad Allon, Anton Ovchinnikov, Ioannis Stamatopoulos, and various seminar participants.

References

Berk Ustun and Cynthia Rudin. Supersparse linear integer models for optimized medical scoring systems. *Machine Learning*, 102(3):349–391, 2016.

Guihua Wang, Jun Li, Wallace J Hopp, Franco L Fazzalari, and Steven Bolling. Using patient-centric quality information to unlock hidden health care capabilities. 2016.

Dimitris Bertsimas, Nathan Kallus, Alexander M Weinstein, and Ying Daisy Zhuo. Personalized diabetes management using electronic medical records. *Diabetes care*, 40(2):210–217, 2017.

Jon Kleinberg, Himabindu Lakkaraju, Jure Leskovec, Jens Ludwig, and Sendhil Mullainathan. Human decisions and machine predictions. Technical report, National Bureau of Economic Research, 2017.

Jongbin Jung, Connor Concannon, Ravi Shroff, Sharad Goel, and Daniel Goldstein. Simple rules for complex decisions. *arXiv:1702.04690*, 2017.

Kris Johnson Ferreira, Bin Hong Alex Lee, and David Simchi-Levi. Analytics for an online retailer: Demand forecasting and price optimization. *Manufacturing & Service Operations Management*, 18(1):69–88, 2015.

Marshall Fisher, Santiago Gallino, and Serguei Netessine. Setting retail staffing levels: A methodology validated with implementation. 2017.

Robert J LaLonde. Evaluating the econometric evaluations of training programs with experimental data. *The American economic review*, pages 604–620, 1986.

Joshua D Angrist, Guido W Imbens, and Donald B Rubin. Identification of causal effects using instrumental variables. *Journal of the American statistical Association*, 91(434):444–455, 1996.

Hamsa Bastani, Joel Goh, and Mohsen Bayati. Evidence of upcoding in pay-for-performance programs. 2015.

Sendhil Mullainathan and Ziad Obermeyer. Does machine learning automate moral hazard and error? *American Economic Review*, 107(5):476–80, 2017.

Cynthia Rudin. Algorithms for interpretable machine learning. In *KDD*, 2014.

Finale Doshi-Velez and Been Kim. A roadmap for a rigorous science of interpretability. *arXiv:1702.08608*, 2017.

Benjamin Letham, Cynthia Rudin, Tyler H McCormick, David Madigan, et al. Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model. *The Annals of Applied Statistics*, 9(3):1350–1371, 2015.

Himabindu Lakkaraju, Stephen H Bach, and Jure Leskovec. Interpretable decision sets: A joint framework for description and prediction. In *KDD*, 2016.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *KDD*, 2016.

P. W. Koh and P. Liang. Understanding black-box predictions via influence functions. *arXiv preprint arXiv:1703.04730*, 2017.

Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. *Classification and regression trees*. CRC press, 1984.

Hongyu Yang, Cynthia Rudin, and Margo Seltzer. Scalable bayesian rule lists. In *ICML*, 2017.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

Berkeley J Dietvorst, Joseph P Simmons, and Cade Massey. Algorithm aversion: People erroneously avoid algorithms after seeing them err. *Journal of Experimental Psychology: General*, 144(1):114, 2015.

Mike Yeomans, Anuj K Shah, Sendhil Mullainathan, and Jon Kleinberg. Making sense of recommendations. 2016.

Robert Phillips, A Serdar Şimşek, and Garrett Van Ryzin. The effectiveness of field price discretion: Empirical evidence from auto lending. *Management Science*, 61(8):1741–1759, 2015.

Felipe Caro and Anna S de Tejada Cuenca. Believing in analytics: Managers' adherence to price recommendations from a dss. *Working Paper*, 2018.

Berkeley J Dietvorst, Joseph P Simmons, and Cade Massey. Overcoming algorithm aversion: People will use imperfect algorithms if they can (even slightly) modify them. *Management Science*, 64(3):1155–1170, 2016.

Karel H Van Donselaar, Vishal Gaur, Tom Van Woensel, Rob ACM Broekmeulen, and Jan C Fransoo. Ordering behavior in retail stores and implications for automated replenishment. *Management Science*, 56(5):766–784, 2010.

Guihua Wang, Jun Li, and Wallace J Hopp. An instrumental variable tree approach for detecting heterogeneous treatment effects in observational studies. 2017.

Jiaming Zeng, Berk Ustun, and Cynthia Rudin. Interpretable classification models for recidivism prediction. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 180(3):689–722, 2017.

Fulton Wang and Cynthia Rudin. Falling rule lists. In *AISTATS*, 2015.

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

Rich Caruana, Yin Lou, and Johannes Gehrke. Intelligible models for classification and regression. In *KDD*, 2012.

Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

Anupam Datta, Shayak Sen, and Yair Zick. Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In *IEEE Symposium on Security and Privacy*, 2016.

Himabindu Lakkaraju, Ece Kamar, Rich Caruana, and Jure Leskovec. Interpretable & explorable approximations of black box models. In *FAT/ML*, 2017.

Kristi Läll, Reedik Mägi, Andrew Morris, Andres Metspalu, and Krista Fischer. Personalized risk prediction for type 2 diabetes: The potential of genetic risk scores. *Genetics in Medicine*, 19(3):322, 2017.

Catherine C Cowie, Keith F Rust, Earl S Ford, Mark S Eberhardt, Danita D Byrd-Holt, Chaoyang Li, Desmond E Williams, Edward W Gregg, Kathleen E Bainbridge, Sharon H Saydah, et al. Full accounting of diabetes and pre-diabetes in the us population in 1988–1994 and 2005–2006. *Diabetes care*, 32 (2):287–294, 2009.

Jaakko Tuomilehto, Peter Schwarz, and Jaana Lindström. Long-term benefits from lifestyle interventions for type 2 diabetes prevention: time to expand the efforts. *Diabetes Care*, 34(Supplement 2):S210–S214, 2011.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Jason A Winston and Jami L Miller. Treatment of onychomycosis in diabetic patients. *Clinical Diabetes*, 24 (4):160–166, 2006.

Cathy O'Neil. *Weapons of math destruction: How big data increases inequality and threatens democracy.* Broadway Books, 2016.

Neal Jean, Marshall Burke, Michael Xie, W Matthew Davis, David B Lobell, and Stefano Ermon. Combining satellite imagery and machine learning to predict poverty. *Science*, 353(6301):790–794, 2016.

Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639): 115, 2017.

Appendix

A. Proofs for Theorem 1

A.1. Main Lemmas

Lemma 2. *Let T^* be the exact greedy decision of depth d. Then, we have $T^* \equiv f$.*

Proof of Lemma 2 We claim that if node N in T^* branches on feature $i \in [d]$ (i.e., it is labeled with predict $x_i \leq t$), then none of the children of N branch on feature i. Without loss of generality, consider a child N' in the left branch of N. Then, since $C_{N'} = ... \wedge x_i \leq 0.5 \wedge ...$, so for all $x \in \mathcal{F}(C_{N'})$, we have $x_i = 0$. Thus, we have $\Pr_{x \sim \mathcal{P}}[C_N \wedge x_i > 0.5] = 0$, so our algorithm will not branch on feature i, as claimed.

Next, consider any path from the root of T^* to a leaf node N; we claim that $T^*(x) = f(x)$ for all $x \in \mathcal{F}(C_N)$. First, consider the case where N is at depth less than $d+1$. The only reason our algorithm would not branch at N before reaching the maximum depth is that classification accuracy is perfect at N. Thus, in this case, our current claim follows. Next, consider the case where N is at depth d. Let $N_0 = N_{T^*}, N_1, ..., N_d, N_{d+1} = N$ be the sequence of nodes along the path from the root N_{T^*} of T^* to leaf N. By our previous claim, each of the nodes $N_0, ..., N_d$ must branch on a different feature. Since there are exactly d features, each feature occurs exactly once in C_N. Without loss of generality, assume that each node N_{i+1} is the left branch of N_i; then, we have

$$C_N = x_1 \leq 0.5 \wedge ... \wedge x_d \leq 0.5.$$

Note that there is only a single point that satisfies C_N, i.e., $\mathcal{F}(C_N) = \{x_N\}$ where $x_N = \begin{bmatrix} 0 & ... & 0 \end{bmatrix}^T$. Thus, the label on N must be $f(x_N)$, and $T^*(x_N) = f(x_N)$. It follows that

$$\begin{aligned} \Pr[T^*(x) = f(x)] &= \sum_{N \in \mathsf{leaves}(T^*)} \Pr[T^*(x) = f(x) \mid C_N] \cdot \Pr[C_N] \\ &= \sum_{N \in \mathsf{leaves}(T^*)} \Pr[C_N] \\ &= 1, \end{aligned}$$

as claimed. □

Lemma 3. *Let $f : \mathcal{X} \to \mathcal{Y}$ be any function, let T^* be the exact greedy decision of depth $d+1$, and let $\hat{T}$ be the estimated greedy decision tree of depth $d+1$, and assume that T^* is Δ-gapped. Then, for*

$$n_{tot} = O(\Delta^{-2} 2^{d+3\log d} \log \delta^{-1}),$$

we have $Pr[T^* = \hat{T}] \geq 1 - \delta$ (where the randomness is taken over the samples $x \sim \mathcal{P}$ used to extract $\hat{T}$).

Proof of Lemma 3 First, note that by Lemma 4, for a given node N in the estimated greedy decision tree, we have

$$\Pr\left[\forall i \in [d]. |\hat{G}(i) - G(i)| > \frac{\Delta}{2}\right] \leq \delta$$

if we take n samples. This bound implies that for all $i \neq i^*$ (where $i^* = \arg\max_{i\in[d]} G(i)$ is the maximizer of the true gain), we have

$$\begin{aligned}\hat{G}(i^*) &\geq G(i^*) - \frac{\Delta}{2} \\ &\geq G(i) + \frac{\Delta}{2} \\ &\geq \hat{G}(i),\end{aligned}$$

where the second step follows from the definition of Δ. Thus, $\arg\max_{i\in[d]} \hat{G}(i) = i^*$, i.e., the index i chosen in the estimated greedy decision tree equals the one chosen in the exact greedy decision tree (assuming all the parents of N are chosen correctly). Taking a union bound over all 2^d internal nodes in the tree, the internal nodes of T^* and $\hat{T}$ are identical with probability at least $1 - \delta$.

Next, we compute the total number of samples needed. First, note that for each $i \in [d]$, there are four probabilities that must be estimated in $G(i)$, for a total of $4nd$ samples needed per node. Finally, since there are 2^d internal nodes, the total number of samples needed to estimate the internal nodes of T^* is $4nd2^d$, which is

$$n_{\text{tot}} = O(\Delta^{-2} 2^{d+3\log d} \log \delta^{-1}).$$

The number of samples required to choose the labels y_N assigned to each leaf node N is similar (and in fact simpler), since there are 2^d leaf nodes as well. Therefore, the claim follows. □

A.2. Properties of the Gain

Lemma 4. *Suppose that the gain $G(i)$ is estimated using*

$$n = \left(\frac{24}{\Delta}\right)^2 d \log \frac{24d}{\delta}$$

samples $x^{(1)}, ..., x^{(n)}$ for each probability. Then, we have

$$Pr\left[|\hat{G}(i) - G(i)| > \frac{\Delta}{2}\right] \leq \delta$$

for all $i \in [d]$, where $\hat{G}(i)$ is the estimated gain.

Proof of Lemma 4 Recall that

$$G(i) = -H(f, C_N \wedge x_i \leq 0.5) - H(f, C_N \wedge x_i > 0.5) + H(f, C_N),$$

where

$$H(f, C) = \left(1 - \sum_{y\in\mathcal{Y}} \Pr_{x\sim\mathcal{P}}[f(x) = y \mid C]^2\right) \cdot \Pr_{x\sim\mathcal{P}}[C].$$

Note that the last term of $G(i)$ is constant with respect to i, so it can be dropped without affecting the optimal value $i^* = \arg\max_{i\in[d]} G(i)$, as can the term

$$\Pr_{x\sim\mathcal{P}}[C_N \wedge x_i \leq 0.5] + \Pr_{x\sim\mathcal{P}}[C_N \wedge x_i > 0.5] = \Pr_{x\sim\mathcal{P}}[C_N]$$

that comes from the first terms of $H(f,C_N\wedge x_i\leq 0.5)$ and $H(f,C_N\wedge x_i>0.5)$. Then, we have

$$\begin{aligned}G(i)=&\sum_{y\in\mathcal{Y}}\mathrm{Pr}_{x\sim\mathcal{P}}[f(x)=y\mid C_N\wedge x_i\leq 0.5]^2\cdot\mathrm{Pr}_{x\sim\mathcal{P}}[C_N\wedge x_i\leq 0.5]\\&+\sum_{y\in\mathcal{Y}}\mathrm{Pr}_{x\sim\mathcal{P}}[f(x)=y\mid C_N\wedge x_i>0.5]^2\cdot\mathrm{Pr}_{x\sim\mathcal{P}}[C_N\wedge x_i>0.5].\end{aligned}$$

By Lemma 6, letting

$$E^*=\mathrm{Pr}_{x\sim\mathcal{P}}[f(x)=y\mid C_N\wedge x_i\leq 0.5]^2\cdot\mathrm{Pr}_{x\sim\mathcal{P}}[C_N\wedge x_i\leq 0.5]$$

and $\hat{E}$ be the estimate of E^*, and letting $\epsilon=\frac{\Delta}{24}$, we have

$$\mathrm{Pr}\left[|\hat{E}-E^*|>\frac{\Delta}{8}\right]\leq 6e^{-n(\Delta/24)^2}.$$

Similar bounds hold for the other three terms of $G(i)$; together, we have

$$\mathrm{Pr}\left[|\hat{G}(i)-G(i)|>\frac{\Delta}{2}\right]\leq 24e^{-n(\Delta/24)^2}.$$

Finally, taking a union bound over $i\in[d]$, we have

$$\mathrm{Pr}\left[\forall i\in[d].|\hat{G}(i)-G(i)|>\frac{\Delta}{2}\right]\leq 24de^{-n(\Delta/24)^2}=\frac{\delta}{2^d},$$

as claimed. □

A.3. Technical Lemmas

Lemma 5. *Let $C:\mathcal{X}\to\{0,1\}$ be any logical predicate, let*

$$E^*=Pr_{x\sim\mathcal{P}}[C]=\mathbb{E}_{x\sim\mathcal{P}}[\mathbb{I}[x\in\mathcal{F}(C)]]$$

be the probability that C holds, where $\mathbb{I}$ is the indicator function, and

$$\hat{E}_n=\frac{1}{n}\sum_{i=1}^{n}\mathbb{I}[x^{(i)}\in\mathcal{F}(C)]$$

be an estimate of E^* using n samples $x^{(1)},...,x^{(n)}\sim\mathcal{P}$. Then, we have

$$Pr[|\hat{E}-E|>\epsilon]\leq 2e^{-2n\epsilon^2},$$

where the probability is taken over the randomness in the samples $x^{(1)},...,x^{(n)}$.

Proof of Lemma 5 The lemma follows by applying Hoeffding's inequality to the random variable $z=\mathbb{I}[x\in\mathcal{F}(C)]$. □

Lemma 6. *Let $C_1,...,C_k:\mathcal{X}\to\{0,1\}$ be logical predicates, and $E_1^*,...,E_k^*$ and $\hat{E}_1,...,\hat{E}_k$ as in Lemma 5. Then, we have*

$$Pr[|\hat{E}_1\cdot...\cdot\hat{E}_k-E_1^*\cdot...\cdot E_k^*>k\epsilon]\leq 2ke^{-2n\epsilon^2}$$

Proof of Lemma 6 First, by applying Lemma 5 together with a union bound, we have

$$\Pr[|\hat{E}_1 - E_1^*| \leq \epsilon \wedge ... \wedge \hat{E}_k - E_k^* \leq \epsilon] \leq 2ke^{-2n\epsilon^2}.$$

Next, given that $|\hat{E}_i - E_i^*| \leq \epsilon$ for each $i \in [d]$, we prove that

$$|\hat{E}_1 ... \hat{E}_i - E_1^* ... E_i^* > i| \leq i\epsilon,$$

for each $i \in [d]$, by induction on i. The base case $i = 1$ follows by assumption. Then, we have

$$\begin{aligned}
&|\hat{E}_1 ... \hat{E}_i - E_1^* ... E_i^*| \\
&= |\hat{E}_1 ... \hat{E}_i - E_1^* ... E_{i-1}^* \hat{E}_i + E_1^* ... E_{i-1}^* \hat{E}_i - E_1^* ... E_i^*| \\
&= |(\hat{E}_1 ... \hat{E}_{i-1} - E_1^* ... E_{i-1}^*) \cdot \hat{E}_i + E_1^* ... E_{i-1}^* \cdot (\hat{E}_i - E_i^*)| \\
&\leq |\hat{E}_1 ... \hat{E}_{i-1} - E_1^* ... E_{i-1}^*| \cdot |\hat{E}_i| + |E_1^* ... E_{i-1}^*| \cdot |\hat{E}_i - E_i^*| \\
&\leq |\hat{E}_1 ... \hat{E}_{i-1} - E_1^* ... E_{i-1}^*| + |\hat{E}_i - E_i^*| \\
&\leq (i-1)\epsilon + \epsilon \\
&= i\epsilon,
\end{aligned}$$

where the second-to-last step follows since $E_i^* \in [0,1]$ and $\hat{E}_i \in [0,1]$ for each $i \in [d]$. Therefore, the inductive step holds, so the claim follows. □

B. Proofs for Theorem 2

Appendix B.1 gives the proof of Lemma 1. Appendices B.2 and B.3 provide proofs of the technical lemmas required for Theorem 2 and Lemma 1.

B.1. Proof of Main Lemma

The key idea behind proving Lemma 1 is to use induction on the structure of the tree. More precisely, it is clear that Lemma 1 holds for the root node N_{T^*} of T^*, since every input is routed to the root, i.e.,

$$\mathbb{I}[x \xrightarrow{T^*} N_{T^*}] = \mathbb{I}[x \xrightarrow{T} \phi(N_{T^*})] = 1$$

for all $x \in \mathcal{X}$. Then, it suffices to show that if Lemma 1 holds for the parent of a node $N^* \in T^*$, then it holds for N^* as well.

More precisely, let M^* be the parent of N^*, and let $M = \phi(M^*)$ be the parent of $N = \phi(N^*)$. Our goal is to prove that, assuming

$$\|p_M - p_{M^*}\|_1 \xrightarrow{p} 0,$$

then

$$\|p_N - p_{N^*}\|_1 \xrightarrow{p} 0$$

as well (note that we use $\xrightarrow{p}$ to denote convergence in probability).

For simplicity, we prove the one-dimensional case, i.e., $\mathcal{X} = \mathbb{R}$. Proving the general case is a straightforward extension of our proof, but requires extra bookkeeping that obscures the key ideas. In particular, let $N^* \in T^*$

have form $N^* = (i^*, t^*)$, and let $N = \phi(N^*) \in T$ have form $N = (i, t)$. When $d = 1$, we know that $i = i^* = 1$, so we only have to prove that t converges to t^*. Proving that i converges to i^* is straightforward since there are only finitely many choices for i. With this restriction, we can assume that internal nodes have only a single parameter, i.e., $N^* = (t^*)$ where $t^* \in \mathbb{R}$, and $N = \phi(N^*) = (t)$ where $t \in \mathbb{R}$.

We begin our proof by expressing p_N in terms of p_M. We assume without loss of generality that N is the left child of M. Then, note that

$$\mathbb{I}[x \xrightarrow{T} N] = \mathbb{I}[x \xrightarrow{T} M] \cdot \mathbb{I}[x \leq t],$$

where $M = (t)$, so we have

$$p_N(x) = p_M(x) \cdot \mathbb{I}[x \leq t].$$

Now, our proof proceeds in two steps:

1. First, we show assuming $\|p_M - p_{M^*}\|_1 \xrightarrow{p} 0$, then $t \xrightarrow{p} t^*$.
2. Second, we show that assuming $t \xrightarrow{p} t^*$, then $\|p_N - p_{N^*}\|_1 \xrightarrow{p} 0$.

Step 1: Proving $t \xrightarrow{p} t^*$***.*** First, we show that $\|p_M - p_{M^*}\|_1 \xrightarrow{p} 0$ implies

$$\|G - G^*\|_\infty \xrightarrow{p} 0,$$

where

$$G^*(s) = G(i, s; \mathcal{P} \mid C_{M^*})$$
$$G(s) = G(i, s; \mathcal{P}_M)$$

are the gain functions for T^* and T, respectively, where $G(i, s; \mathcal{Q})$ is defined in (1); as noted above, we have assumed $i = 1$ is a constant to simplify our exposition. Proving this step depends on the gain function being used to train the decision tree; we show that it holds for the gain function based on the Gini impurity in Lemma 7 (proof in Appendix B.2).

Next, we show that as long as $\|G - G^*\|_\infty$ is sufficiently small, then the difference between their corresponding maximizers

$$t^* = \arg\max_s G^*(s)$$
$$t = \arg\max_s G(s)$$

is small as well, i.e., $t \xrightarrow{p} t^*$.

By Assumption 2, we can prove the existence of a *gap*, which intuitively is an interval around t^* outside of which the $G^*(s)$ is "sufficiently smaller" than $G^*(t^*)$. More precisely:

DEFINITION 8. We say that a function $g : \mathbb{R} \to \mathbb{R}$ is (ϵ, δ)-*gapped* if it has a unique maximizer $s^* = \arg\max_{s \in \mathbb{R}} g(s)$, and for every $s \in \mathbb{R}$ such that $|s - s^*| > \epsilon$, we have $g(s^*) > g(s) + \delta$.

We show that as long as G^* is continuous and has bounded support, then for any $\epsilon' > 0$, there exists $\delta' > 0$ such that G^* is (ϵ', δ')-gapped; in Lemma 9 (proof in Appendix B.3), we show that the gain function G^*

based on the Gini impurity satisfies these technical assumptions. Then, let $s_{\max}$ bound the support of G^*, i.e., $G^*(s) = 0$ if $|s| > s_{\max}$. Let $\epsilon' > 0$ be arbitrary, and let

$$A_{\epsilon'} = \{s \in \mathbb{R} \mid |s| \leq s_{\max} \text{ and } |s - s^*| \geq \epsilon'\}.$$

Note that $A_{\epsilon'}$ is a compact set, so G^* achieves its maximum on $A_{\epsilon'}$, i.e.,

$$t^*_{\epsilon'} = \arg\max_{s \in A_{\epsilon'}} G^*(s).$$

Then, G^* is (ϵ', δ')-gapped, where

$$\delta' = \frac{G^*(t^*) - G^*(t^*_{\epsilon'})}{2} > 0.$$

Note that we divide by 2 since the inequality in Definition 8 is strict.

Now, we show that having a gap implies $t \xrightarrow{p} t^*$. In particular, suppose that $\|G^* - G\|_\infty \leq \frac{\delta'}{2}$. Then, we have

$$\begin{aligned} G^*(t^*) - G^*(t) &\leq \left(G(t^*) + \frac{\delta'}{2}\right) - \left(G(t) - \frac{\delta'}{2}\right) \\ &\leq G(t^*) - G(t) + \delta' \\ &\leq \delta', \end{aligned}$$

where the last step follows since t is the maximizer of G. In particular, we have shown that $|G^*(t^*) - G^*(t)| \leq \delta'$, so since G^* is (ϵ', δ')-gapped, it follows that $|t - t^*| \leq \epsilon'$. Since $\|G^* - G\|_\infty \xrightarrow{p} 0$, it follows that $t \xrightarrow{p} t^*$.

Step 2: Proving $\|p_N - p_{N^*}\|_1 \xrightarrow{p} 0$***.*** Note that

$$\begin{aligned} \|p_N - p_{N^*}\|_1 &= \int |p_N(x) - p_{N^*}(x)| dx \\ &= \int |p_M(x) \cdot \mathbb{I}[x \leq t] - p_{M^*}(x) \cdot \mathbb{I}[x \leq t^*]| dx \\ &= \int |p_M(x) \cdot \mathbb{I}[x \leq t] - (p_M(x) + p_{M^*}(x) - p_M(x)) \cdot \mathbb{I}[x \leq t^*]| dx \\ &\leq \int p_M(x) \cdot |\mathbb{I}[x \leq t] - \mathbb{I}[x \leq t^*]| dx + \int |p_M(x) - p_{M^*}(x)| \cdot \mathbb{I}[x \leq t^*] dx. \end{aligned}$$

Assume without loss of generality that $t \leq t^*$. Then, for the first integral, note that the integrand equals 0 for $x \notin [t, t^*]$ and equals 1 for $x \in [t, t^*]$. Thus,

$$\begin{aligned} \int p_M(x) \cdot |\mathbb{I}[x \leq t] - \mathbb{I}[x \leq t^*]| dx &= \int p_M(x) \cdot \mathbb{I}[t \leq x \leq t^*] dx \\ &= \int_t^{t^*} p_M(x) dx \\ &\leq |t - t^*| \cdot p_{\max}, \end{aligned}$$

where the last step follows by Assumption 1, which says that $p(x) \leq p_{\max}$ for all $x \in \mathbb{R}$.

Next, for the second integral, note that

$$\int |p_M(x) - p_{M^*}(x)| \cdot \mathbb{I}[x \leq t^*] dx \leq \|p_M - p_{M^*}\|_1.$$

Together, we have

$$\|p_N - p_{N^*}\|_1 \leq \|p_M - p_{M^*}\|_1 + |t - t^*| \cdot p_{\max}.$$

Since the left-hand side converges in probability to 0, so does the right-hand side, as claimed. □

B.2. Proof of Convergence of the Gain Function

In this section, we prove that the gain function G converges uniformly to G^* as $n \to \infty$. To simplify notation, we use slightly different notation for the Gini impurity H compared to the definition in (1).

Lemma 7. *Let*

$$G^*(t) = -H^*(f, C_{N^*} \wedge (x \le t)) - H^*(f, C_{N^*} \wedge (x > t)) + H^*(f, C_{N^*})$$
$$H^*(f, C) = \left(1 - \sum_{y \in \mathcal{Y}} \left(\frac{Pr_{x \sim \mathcal{P}}[f(x) = y \wedge C]}{Pr_{x \sim \mathcal{P}}[C]}\right)^2\right) \cdot Pr_{x \sim \mathcal{P}}[C]$$

be the gain function based on the Gini impurity for the exact decision tree, and let

$$G(t) = -H(f, C_N \wedge (x \le t)) - H(f, C_N \wedge (x > t)) + H(f, C_N)$$
$$H(f, C) = \left(1 - \sum_{y \in \mathcal{Y}} \left(\frac{\frac{1}{n}\sum_{j=1}^n \mathbb{I}[f(x^{(j)}) = y \wedge x^{(j)} \in \mathcal{F}(C)]}{\frac{1}{n}\sum_{j=1}^n \mathbb{I}[x^{(j)} \in \mathcal{F}(C)]}\right)^2\right) \cdot \frac{1}{n}\sum_{j=1}^n \mathbb{I}[x^{(j)} \in \mathcal{F}(C)]$$

be the corresponding gain function for the estimated decision tree.

If $\|p_N - p_{N^*}\|_1 \xrightarrow{p} 0$*, where*

$$p_{N^*}(x) = p(x) \cdot \mathbb{I}[x \xrightarrow{T^*} N^*]$$
$$p_N(x) = p(x) \cdot \mathbb{I}[x \xrightarrow{T} N],$$

then we have $\|G - G^*\|_\infty \xrightarrow{p} 0$.

Proof. First, note that

$$\begin{aligned}\|G - G^*\|_\infty \le & \sup_{t \in \mathbb{R}} |H^*(f, C_{N^*} \wedge (x \le t)) - H(f, C_N \wedge (x \le t))| \\ & + \sup_{t \in \mathbb{R}} |H^*(f, C_{N^*} \wedge (x > t)) - H(f, C_N \wedge (x > t))| \\ & + \sup_{t \in \mathbb{R}} |H^*(f, C_{N^*}) - H(f, C_N)|.\end{aligned}$$

We prove that the first term converges in probability to 0 as $n \to \infty$; the remaining two terms can be bounded using the same argument. In particular, let

$$H^*(t) = H^*(f, C_{N^*} \wedge (x \le t))$$
$$H(t) = H(f, C_N \wedge (x \le t)),$$

so our goal is to show that $\|H - H^*\|_\infty \xrightarrow{p} 0$. To simplify our expressions, define

$$\begin{aligned} g^*(t) &= \Pr_{x \sim \mathcal{P}}[C_{N^*} \wedge (x \le t)] \\ h_y^*(t) &= \Pr_{x \sim \mathcal{P}}[f(x) = y \wedge C_{N^*} \wedge (x \le t)] \\ g(t) &= \frac{1}{n}\sum_{j=1}^n \mathbb{I}[x^{(j)} \in \mathcal{F}(C_N \wedge (x \le t))] \\ h_y(t) &= \frac{1}{n}\sum_{j=1}^n \mathbb{I}[f(x^{(j)}) = y \wedge x^{(j)} \in \mathcal{F}(C_N) \wedge (x^{(j)} \le t)], \end{aligned}$$

A useful fact is that

$$0 \le h_y^*(t) \le g^*(t) \le 1$$

$$0 \le h_y(t) \le g(t) \le 1$$

for all $t \in \mathbb{R}$ and all $y \in \mathcal{Y}$ (but assuming the random samples $x^{(j)}$ are fixed). Now, we have

$$\begin{aligned} H^*(t) &= \left(1 - \sum_{y \in \mathcal{Y}} \left(\frac{h_y^*(t)}{g^*(t)}\right)^2\right) \cdot g^*(t) \\ &= g^*(t) - \sum_{y \in \mathcal{Y}} \frac{h_y^*(t)^2}{g^*(t)}, \end{aligned}$$

and similarly

$$H(t) = g(t) - \sum_{y \in \mathcal{Y}} \frac{h_y(t)^2}{g(t)}.$$

Then, we have

$$\|H - H^*\|_\infty \le \sup_{t \in \mathbb{R}} |g(t) - g^*(t)| + \sum_{y \in \mathcal{Y}} \sup_{t \in \mathbb{R}} \left| \frac{h_y^*(t)^2}{g^*(t)} - \frac{h_y(t)^2}{g(t)} \right|.$$

We show that for a fixed $y \in \mathcal{Y}$, we have

$$\sup_{t \in \mathbb{R}} \left| \frac{h_y^*(t)^2}{g^*(t)} - \frac{h_y(t)^2}{g(t)} \right| \xrightarrow{p} 0. \tag{6}$$

Bounding the first term of $\|H - H^*\|_\infty$ follows similarly; together, these limits imply that $\|H - H^*\|_\infty \xrightarrow{p} 0$ as well. We break the remainder of the proof into two steps:

1. First, we prove that $\|g - g^*\|_\infty \xrightarrow{p} 0$ and $\|h_y - h_y^*\|_\infty \xrightarrow{p} 0$.
2. Second, we use the first part to show that (6) holds.

Step 1. We prove that $\|h_y - h_y^*\|_\infty \xrightarrow{p}\rightarrow 0$; the claim $\|g - g^*\|_\infty \xrightarrow{p} 0$ follows similarly. First, note that

$$\begin{aligned} h_y^*(t) &= \int \mathbb{I}[f(x) = y] \cdot \mathbb{I}[x \xrightarrow{N^*} T^*] \cdot \mathbb{I}[x \le t] \cdot p(x) dx \\ &= \int \mathbb{I}[f(x) = y] \cdot \mathbb{I}[x \le t] \cdot p_{N^*}(x) dx, \end{aligned}$$

and define

$$\tilde{h}_y(t) = \int \mathbb{I}[f(x) = y] \cdot \mathbb{I}[x \le t] \cdot p_N(x) dx.$$

Then, note that

$$\|h_y - h_y^*\|_\infty \le \|h_y - \tilde{h}_y\|_\infty + \|\tilde{h}_y - h_y^*\|_\infty.$$

Bounding the first term, which represents the estimation error, is somewhat involved, so we relegate the proof to another lemma. In particular, taking $g = h_y$ and $g^* = \tilde{h}_y$ in Lemma 8, it follows that $\|h_y - \tilde{h}_y\|_\infty \xrightarrow{p} 0$.

To bound the second term, note that

$$\begin{aligned} \|\tilde{h}_y - h_y^*\|_\infty &= \sup_{t \in \mathbb{R}} \left| \int \mathbb{I}[f(x) = y] \cdot \mathbb{I}[x \le t] \cdot (p_N(x) - p_{N^*}(x)) dx \right| \\ &\le \sup_{t \in \mathbb{R}} \int |p_N(x) - p_{N^*}(x)| dx \\ &= \|p_N - p_{N^*}\|_1 \\ &\xrightarrow{p} 0, \end{aligned}$$

where the last step follows by our assumption.

Step 2. Let $\epsilon, \delta > 0$ be arbitrary. We need to show that

$$\left| \frac{h_y^*(t)^2}{g^*(t)} - \frac{h_y(t)^2}{g(t)} \right| \leq \epsilon$$

for every $t \in \mathbb{R}$ with probability at least $1 - \delta$. By the previous step, we can take

$$\|g - g^*\|_\infty \leq \frac{\epsilon}{8}$$
$$\|h_y - h_y^*\|_\infty \leq \frac{\epsilon^2}{16}$$

each with probability at least $1 - \frac{\delta}{2}$, so by a union bound, both these inequalities hold with probability at least $1 - \delta$.

We consider two cases. First, suppose that $g^*(t) \leq \frac{\epsilon}{4}$, in which case

$$g(t) \leq g^*(t) + \frac{\epsilon}{8} \leq \frac{\epsilon}{2}.$$

Then, since $h_y^*(t) \leq g^*(t)$ and $h_y(t) \leq g(t)$, we have

$$\begin{aligned} \left| \frac{h_y^*(t)^2}{g^*(t)} - \frac{h_y(t)^2}{g(t)} \right| &\leq \left| \frac{h_y^*(t)^2}{g^*(t)} \right| + \left| \frac{h_y(t)^2}{g(t)} \right| \\ &\leq |g^*(t)| + |g(t)| \\ &\leq \epsilon. \end{aligned}$$

One detail is that when $g^*(t) = 0$, then $H^*(t)$ is not well-defined. Defining $H^*(t) = 0$ in this case is standard practice, since $h_y^*(t) \leq g^*(t)$, so

$$H^*(t) = \frac{h_y^*(t)^2}{g^*(t)} \leq \frac{g^*(t)^2}{g^*(t)} \leq g^*(t) = 0.$$

Similarly, we define $H(t) = 0$ if $g(t) = 0$. In either case, the above argument still applies.

Second, suppose that $g^*(t) \geq \frac{\epsilon}{4}$, in which case

$$g(t) \geq g^*(t) - \frac{\epsilon}{8} \geq \frac{\epsilon}{8}.$$

Then, we have

$$\begin{aligned} \left| \frac{h_y^*(t)^2}{g^*(t)} - \frac{h_y(t)^2}{g(t)} \right| &\leq \frac{8}{\epsilon} \cdot |h_y^*(t)^2 - h_y(t)^2| \\ &= \frac{8}{\epsilon} \cdot |h_y^*(t) - h_y(t)| \cdot |h_y^*(t) + h_y(t)| \\ &\leq \frac{8}{\epsilon} \cdot \frac{\epsilon^2}{16} \cdot 2 \\ &\leq \epsilon. \end{aligned}$$

In either case, the claim follows, completing the proof. □

Next, we prove that the estimation error in Lemma 7 goes to zero.

LEMMA 8. Let $\mathcal{P}$ be a probability distribution over $\mathbb{R}$, let $p(x)$ be the probability density function for $\mathcal{P}$, let $F(x)$ be the cumulative distribution function for $\mathcal{P}$, let $\alpha : \mathbb{R} \to [0,1]$ be an arbitrary function, let

$$g^*(t) = \int \alpha(x) \cdot \mathbb{I}[x \leq t] \cdot p(x) dx,$$

and let $x^{(1)},...,x^{(n)}$ be i.i.d. random samples from $\mathcal{P}$, and let

$$g(t) = \frac{1}{n}\sum_{j=1}^{n} \alpha(x^{(j)}) \cdot \mathbb{I}[x^{(j)} \leq t]$$

be the empirical estimate of g^* on these samples. Then, we have

$$\text{Pr}_{x^{(1)},...,x^{(n)}\sim\mathcal{P}}\left[\|g - g^*\|_\infty \geq \frac{4\log n}{\sqrt{n}}\right] \leq \frac{2}{n^{3/2}},$$

for sufficiently large n.

Proof. First, we define points $t_0, t_1, ..., t_{\sqrt{n}} \in \mathbb{R}$ that divide $\mathbb{R}$ into $\sqrt{n}$ intervals according to the cumulative distribution function $F(x)$ (for convenience, we assume n is a perfect square). In particular, we choose t_i to satisfy

$$t_i \in F^{-1}\left(\frac{i}{\sqrt{n}}\right).$$

For convenience, we choose $t_0 = -\infty$ and $t_{\sqrt{n}} = \infty$, which satisfy the condition. Now, for each $i \in [\sqrt{n}]$, let $I_i = (t_{i-1}, t_i]$. Note that these intervals cover $\mathbb{R}$, i.e., $\mathbb{R} = I_1 \cup ... \cup I_{\sqrt{n}}$.

Then, we can decompose the quantity $\|g - g^*\|_\infty$ into three parts:

$$\begin{aligned}
\|g - g^*\|_\infty &= \sup_{t\in\mathbb{R}} |g(t) - g^*(t)| \\
&= \sup_{i\in[\sqrt{n}]} \sup_{t\in I_i} |g(t) - g^*(t)| \\
&\leq \sup_{i\in[\sqrt{n}]} \sup_{t\in I_i} \{|g(t) - g(t_i)| + |g(t_i) - g^*(t_i)| + |g^*(t_i) - g^*(t)|\} \\
&\leq \sup_{i\in[\sqrt{n}]} \sup_{t\in I_i} |g(t) - g(t_i)| + \sup_{i\in[\sqrt{n}]} |g(t_i) - g^*(t_i)| + \sup_{i\in[\sqrt{n}]} \sup_{t\in I_i} |g^*(t_i) - g^*(t)|.
\end{aligned}$$

We show that each of these three parts can be made arbitrarily small with high probability by taking n sufficiently large.

First term. We first show that for every $i \in [\sqrt{n}]$, the interval I_i contains at most $n^{1/2}\log n$ of the points $x^{(1)},...,x^{(n)}$ with high probability. By the definition of the points t_i, the probability that a single randomly selected point $x^{(j)}$ falls in I_i is $n^{-1/2}$ (since the points t_i were constructed according to the cumulative distribution function F):

$$M = \mathbb{E}_{x\sim\mathcal{P}}\left[\mathbb{I}[x \in I_i]\right] = \text{Pr}_{x\sim\mathcal{P}}[x \in I_i] = \frac{1}{\sqrt{n}}.$$

Then, the fraction of the n points $x^{(j)}$ that fall in the interval I_i is

$$\hat{M} = \frac{1}{n}\sum_{j=1}^{n} \mathbb{I}[x^{(j)} \in I_i].$$

Note that each $\mathbb{I}[x^{(j)} \in I_i]$ is an random variable in $[0, 1]$, so by Hoeffding's inequality, we have

$$\text{Pr}_{x^{(1)},...,x^{(n)}\sim\mathcal{P}}\left[|\hat{M} - M| \geq \frac{\log n}{\sqrt{n}}\right] \leq 2e^{-2(\log n)^2} \leq \frac{1}{n^2}$$

for sufficiently large n. Now, note that each point $x^{(j)}$ in I_i can increase the value of $|g(t) - g(t_i)|$ by at most n^{-1}. Since there are $n \cdot \hat{M}$ points $x^{(j)}$ in I_i, the total increase is bounded by $\hat{M}$, i.e.,

$$\text{Pr}_{x^{(1)},...,x^{(n)}\sim\mathcal{P}}\left[\sup_{t\in I_i} |g(t) - g(t_i)| \geq \frac{2\log n}{\sqrt{n}}\right] \leq \frac{1}{n^2}.$$

By a union bound, this inequality holds for every $i \in [\sqrt{n}]$ with probability $n^{-3/2}$.

Second term. Note that each $\alpha(x^{(j)}) \cdot \mathbb{I}[x^{(j)} \geq t]$ is a random variable in $[0,1]$. Therefore, by the Hoeffding inequality, we have

$$\mathrm{Pr}_{x^{(1)},\ldots,x^{(n)} \sim \mathcal{P}} \left[|g(t_i) - g^*(t_i)| \geq \frac{\log n}{\sqrt{n}} \right] \leq 2e^{-2(\log n)^2} \leq \frac{1}{n^2}$$

for sufficiently large n. By a union bound, this inequality holds for every $i \in [\sqrt{n}]$ with probability $n^{-3/2}$.

Third term. Since $t \leq t_i$, we have $\mathbb{I}[x \leq t] \leq \mathbb{I}[x \leq t_i]$. Thus, for all $t \in I_i$, we have

$$\begin{aligned}
|g^*(t_i) - g^*(t)| &= \left| \int \alpha(x) \cdot (\mathbb{I}[x \leq t_i] - \mathbb{I}[x \leq t]) \cdot p(x) dx \right| \\
&\leq \int (\mathbb{I}[x \leq t_i] - \mathbb{I}[x \leq t]) \cdot p(x) dx \\
&= F(t_i) - F(t) \\
&\leq n^{-1/2},
\end{aligned}$$

where the last inequality follows from the definition of t_i and the fact that $t \in I_i$.

Combined bound. Putting the three results together, we can conclude that for sufficiently large n, we have

$$\mathrm{Pr}_{x^{(1)},\ldots,x^{(n)} \sim \mathcal{P}} \left[\|g - g^*\|_\infty \geq \frac{4 \log n}{\sqrt{n}} \right] \leq \frac{2}{n^{3/2}},$$

as claimed. □

B.3. Proof of Regularity of the Gain Function

In this section, we prove that the gain function G^* satisfies certain regularity conditions.

LEMMA 9. *The function $G^* : \mathbb{R} \to \mathbb{R}$ is continuous and has bounded support.*

Proof. It is clear that G^* is continuous. To see that G^* has bounded support, recall that $p(x)$ has bounded support, i.e., $p(x) = 0$ for $|x| > x_{\max}$. Then, note that if $s > x_{\max}$, we have

$$\begin{aligned}
\mathrm{Pr}_{x \sim \mathcal{P}}[C_{N^*} \wedge (x \leq s)] &= \mathrm{Pr}_{x \sim \mathcal{P}}[C_{N^*}] \\
\mathrm{Pr}_{x \sim \mathcal{P}}[C_{N^*} \wedge (x > s)] &= 0 \\
\mathrm{Pr}_{x \sim \mathcal{P}}[f(x) = y \mid C_{N^*} \wedge (x \leq s)] &= \mathrm{Pr}_{x \sim \mathcal{P}}[f(x) = y \mid C_{N^*}] \\
\mathrm{Pr}_{x \sim \mathcal{P}}[f(x) = y \mid C_{N^*} \wedge (x > s)] &= 0
\end{aligned}$$

Therefore, we have

$$G^*(s) = -H^*(f, C_{N^*} \wedge (x \leq s)) - H(f, C_{N^*} \wedge (x > s)) + H(f, C_{N^*}) = 0.$$

By a similar argument, $G^*(s) = 0$ for $s < -x_{\max}$, so the claim follows. □

C. User Study

We provide the remaining questions in our user study here. Fig. 9 lists the remaining three pairs of questions for our extracted decision trees (left) and the rule lists (right) in our user study. The first two questions are in the main body of the paper (Fig. 5 in §5.3).

According to the decision tree, does being over 50 years old put patients at a relatively higher risk of diabetes?

- Yes
- No

According to the decision tree, for which patient subpopulation might a diagnosis of chronic lower back pain newly introduce a high risk of diabetes?

- Patients who are over 50 years old
- Patients who are over 50 years old and have high cholesterol
- Patients who are over 50 years old and who smoke

Consider patients over 50 years old who have high cholesterol, and have had a pre-operative medical exam with no findings. What additional information does the decision tree need to give an assessment of their diabetes risk?

- Whether they have edema
- Whether they have dermatophytosis of nail
- Whether they are taking high triglycerides medication
- No additional information is needed

According to the rule list, does being over 41 years old put patients at a relatively higher risk of diabetes?

- Yes
- No

According to the rule list, for which patient subpopulation might a diagnosis of high cholesterol newly introduce a high risk of diabetes?

- Patients who are over 41 years old
- Patients who are over 41 years old and take arthritis medication
- Patients who are over 41 years old who smoke

Consider patients who are 41–52 years old who are taking arthritis medication. What additional information does the rule list need to give an assessment of their diabetes risk?

- Whether they are taking pain medication
- Whether they are taking pain medication, have high cholesterol, have restless legs syndrome, and/or smoke
- Whether they have high blood pressure and/or smoke
- No additional information is needed

Figure 9 Remaining questions asked in our user study on the diabetes risk classifier for our extracted decision tree (left) and for the rule list (right).