The New England Journal of Statistics in Data Science Volume 0, 1–24 (2025)
DOI: https://doi.org/10.51387/25-NEJSDS84

Knowledge Distillation Decision Tree for Unravelling Black-Box Machine Learning Models

Xuetao LU and J. Jack LEE*

Abstract

Machine learning models, particularly the black-box models, are widely favored for their outstanding predictive capabilities. However, they often face scrutiny and criticism due to the lack of interpretability. Paradoxically, their strong predictive capabilities may indicate a deep understanding of the underlying data, implying significant potential for interpretation. Leveraging the emerging concept of knowledge distillation, we introduce the method of knowledge distillation decision tree (KDDT). This method enables the distillation of knowledge about the data from a black-box model into a decision tree, thereby facilitating the interpretation of the black-box model. Essential attributes for a good interpretable model include simplicity, stability, and predictivity. The primary challenge of constructing an interpretable tree lies in ensuring structural stability under the randomness of the training data. KDDT is developed with the theoretical foundations demonstrating that structure stability can be achieved under mild assumptions. Furthermore, we propose the hybrid KDDT to achieve both simplicity and predictivity. An efficient algorithm is provided for constructing the hybrid KDDT. Simulation studies and a real-data analysis validate the hybrid KDDT's capability to deliver accurate and reliable interpretations. KDDT is an excellent interpretable model with great potential for practical applications.

KEYWORDS AND PHRASES: Knowledge distillation, Decision tree, Machine learning, Model interpretability, Prediction accuracy, Structural stability.

1. INTRODUCTION

In recent decades, machine learning (ML) has gained significant popularity in various fields. However, the widespread adoption of black-box ML models, such as neural networks and ensemble models, has led to growing concerns about their interpretability. This lack of interpretability has triggered skepticism and criticism, particularly in decision-based applications. For instance, in fields like medical diagnostics or treatment choice, without straightforward and concise interpretability the model could lead to erroneous diagnoses and potentially harmful treatment decisions. Knowledge distillation [2, 8, 23, 20, 22, 1] provides a way to interpret the black-box ML model through a transparent model, following a teacher-student architecture [9]. Knowledge about the data is distilled from the teacher model (black-box ML model) to train the student model (transparent model). As a result, the student model (transparent model) inherits the teacher model's knowledge about the complex structure of the data and the underlying mechanisms of the domain question, enabling it to achieve both high interpretability and strong predictive performance. [17] employed several simple models, including linear models and decision trees, as transparent models. Similarly, [15] utilized kernel methods and local linear approaches to construct the transparent model. In this study, we focus on the decision tree [11, 6, 4, 14, 21, 5] which emerges as an ideal transparent model for two reasons. First, it is inherently interpretable. Second, it possesses the capacity to capture complex data structures. Several studies have employed decision tree as transparent model alongside knowledge distillation. [6] explored the distillation of a neural network into a soft decision tree. [4] discussed the decision tree model in interpreting deep reinforcement model. [19] used decision tree for explaining data in the field of e-commerce. However, none of these studies considered the stability of decision trees constructed through knowledge distillation. The interpretability of decision tree relies heavily on the stability of its structure, which may be sensitive to the specific datasets used for training. Interpretations may become questionable if minor changes in the training data significantly affect the tree's structure. Given that the training data is generated randomly through the knowledge distillation process, ensuring the stability of the tree's structure becomes a key challenge to address. [27] explored tree structure stability in knowledge distillation, while their study focused on a single splitting criterion and did not provide conclusive conditions under which the tree structure (or split) converge.

We refer to the decision tree generated from the knowledge distillation process as the "knowledge distillation decision tree" (KDDT). In this paper, we conduct a comprehensive theoretical study for the split stability of KDDT, demonstrating that split will converge in probability with

*Corresponding author.

a specific convergence rate, subject to mild assumptions. Our theoretical findings encompass the most commonly used splitting criteria and are applicable to both classification and regression applications. Additionally, we propose and implement algorithms for KDDT induction. Note that KDDT provides a global approximation to the black-box model, meaning it approximates the entire black-box model at once using a single interpretable model. This approach may be less efficient for interpreting very large and complex black-box models, such as deep neural networks, compared to the local approximation models discussed in [26], which approximate the black-box model piecewisely. For local approximation models, each segment of the black-box model can be represented by a different local approximation, allowing for more tailored interpretations. However, large-scale black-box models have a large number (e.g., millions) of parameters and require large training datasets, making them not suitable for small or medium datasets, such as those with fewer than or equal to $O(10^3)$ samples. For these datasets, the global approximation provided by KDDT can offer more accurate interpretations for the global effects of covariates than simple linear models. Through a simulation study, we validate KDDT's ability to provide precise interpretations while maintaining a stable structure. We also include real data analysis to demonstrate its practical applicability.

The remainder of the paper is organized as follows. In Section 2, we introduce the concept and stability theory of KDDT. The algorithms for constructing KDDT are proposed in Section 3. Section 4 presents the simulation study. In Section 5, we apply KDDT on real datasets. Finally, we conclude and engage in a discussion in Section 6. Theorems and proofs are in Appendix A. Supplementary materials can be found in Appendix B. Additionally, an open-source R implementation of KDDT is accessible on GitHub at https://github.com/lxtpvt/kddt.git.

2. KNOWLEDGE DISTILLATION DECISION TREE

A knowledge distillation decision tree is essentially a decision tree. Instead of being constructed from real observations, it is generated from the knowledge distillation process.

2.1 Knowledge Distillation Process

A typical knowledge distillation process with the teacher-student architecture is illustrated in Figure 1. The specific components of this process can be adapted based on application requirements. For example, the teacher model can be a Convolutional Neural Network (CNN) [7, 12] or a Large Language Model (LLM) [25], while the student model can be a decision tree [11, 6, 4] or a lightweight neural network [7, 12]. In this paper, we specify the components as follows:

- **Data.** $D = \{Y, X\}$, where Y is the set of observations of response variable y, X is the set of observations of covariates $\mathbf{x} = (x_1, \ldots, x_p)$. Both response and covariates can be categorical or continuous variables.
- **Teacher model.** $y = f(\mathbf{x})$, we specify it as a small scale black-box ML model for the size of data $O(10^3)$.
- **Knowledge distillation.** Includes two steps: 1) random sampling of covariate values on their support, denoted as X', and 2) generating the corresponding response values $Y' = f(X')$ through the fitted teacher model f.
- **Knowledge.** $D' = \{Y', X'\}$, we call it pseudo-data.
- **Student model.** A decision tree, we refer to it as a knowledge distillation decision tree (KDDT). The KDDT is constructed from the pseudo-data D' and keeps a stable structure under the randomness of D'.

The knowledge distillation process can also be viewed as a model approximation process, as illustrated in Figure B.12 in Appendix B. The student model, KDDT, is used to approximate the teacher model through the pseudo-data $D' = \{Y', X'\}$. Additionally, Figure B.12 also highlights the differences between the model approximation and generalization processes.

2.2 Tree Structure Stability

In this paper, we focus on the second half of the knowledge distillation process, specifically from the knowledge distillation to the student model. The teacher model and original data are fixed. Our task is to handle the randomness in the pseudo-data D' and construct a stable KDDT. It is based on the hypothesis that we can generate arbitrarily large D' to achieve the stability of KDDT. In this section, we will prove this hypothesis.

2.2.1 Prerequisites

It is essential to introduce the key concepts and notations of decision tree that will be used in the theoretical study.

(1) Splitting criteria

Typically, different criteria are used for regression and classification trees. In regression, the primary criteria include minimizing the sum of squared errors (SSE) or the mean squared error (MSE) after splitting:

$$\begin{aligned} &min\left\{\sum_{i=1}^{n_l}(y_{li}-\bar{y}_l)^2+\sum_{j=1}^{n_r}(y_{rj}-\bar{y}_r)^2\right\},\\ &min\left\{\frac{1}{n_l}\sum_{i=1}^{n_l}(y_{li}-\bar{y}_l)^2+\frac{1}{n_r}\sum_{j=1}^{n_r}(y_{rj}-\bar{y}_r)^2\right\},\end{aligned} \tag{2.1}$$

where the subscripts l, r represent the left and right nodes of a stump, $n_l + n_r = n$, $\bar{y}_l = \frac{1}{n_l}\sum_{i=1}^{n_l} y_{li}$ and $\bar{y}_r = \frac{1}{n_r}\sum_{j=1}^{n_r} y_{rj}$.

In classification, the criterion for selecting the best split is to maximize the reduction of impurity after splitting:

$$max\{E-(E_l+E_r)\},$$

where E is the total impurity before splitting, and E_l and E_r are the left and right child impurities, respectively, af-

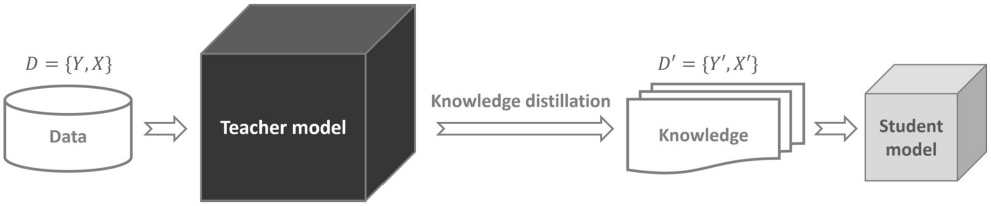


Figure 1: Teacher-student architecture for knowledge distillation.

ter splitting. Since the split does not impact E, the above criterion can be simplified as follows:

$$min\{E_l + E_r\}, \tag{2.2}$$

The well-known impurity measures include Shannon entropy, gain ratio, and Gini index [16, 3]. [24] proposed the Tsallis entropy in (2.3) to unify these measures in a single framework.

$$E = S_q(Y) = \frac{1}{1-q}\left(\sum_{i=1}^{C} p(y_i)^q - 1\right), \qquad q \in \mathbb{R}, \tag{2.3}$$

where Y is a random variable that takes value in $\{y_1, \ldots, y_C\}$, $p(y_i)$ is the corresponding probability of y_i, $i = 1, \ldots, C$, and q is an adjustable parameter.
(2) Split search algorithm

The most commonly used split search algorithm is the greedy search algorithm, which makes a locally optimal choice at each stage in a heuristic manner, to find the global optimum. The algorithm involves the steps: (a) for each split, searching through all covariates and their observed values; (b) for each candidate pair (covariate, value), calculating the loss (gain) defined by splitting criterion; and (c) identifying the best split by minimizing the loss (maximizing the gain). Although the greedy search algorithm may not guarantee the global optimum which is theoretically an NP problem [10], we still choose it for our study due to its simplicity and popularity in practice.

2.2.2 Split Convergence

As discussed in the introduction, studying the stability of the entire tree is challenging. A practical approach is to focus on individual splits. If all splits are stable, the entire tree is stable naturally. We refer to a split as achieving stability when it converges to a unique optimal split. The concepts of optimal split is defined as follows.

Definition 1 (Optimal split). Let Ω be the support of univariate x, and $z_i^l(x)$ and $z_i^r(x)$ be functions $\Omega \to \mathbb{R}$, where $i = 1, \ldots, C$ and C is a constant in $\mathbb{N}^+$. Let $g(z_1^t(x), \ldots, z_C^t(x))$ be a continuous function $\mathbb{R}^{\mathbb{C}} \to \mathbb{R}$, where $t = l$ or r. Then, the optimal split $x_s \in \Omega$ is defined as follows.

$$x_s = \underset{x\in\Omega}{\operatorname{argmin}}\big[g\big(z_1^l(x), \ldots, z_C^l(x)\big) + g\big(z_1^r(x), \ldots, z_C^r(x)\big)\big]. \tag{2.4}$$

Definition 1 is somewhat abstract. To clarify, we provide two examples to illustrate this definition in both regression and classification contexts.

- Regression: we assume that both y and x are continuous variables, and that the split criterion is the MSE as defined in (2.1). The components in Definition 1 are outlined as follows.

$$g\big(z_1^l(x)\big) = z_1^l(x), \quad g\big(z_1^r(x)\big) = z_1^r(x),$$

$$z_1^l(x) = \int_a^x \big(f(t) - \mu_l(x)\big)^2\, dt,$$

$$z_1^r(x) = \int_x^b \big(f(t) - \mu_r(x)\big)^2\, dt,$$

where,

$$\mu_l(x) = \frac{1}{x-a}\int_a^x f(u)\, du, \quad \mu_r(x) = \frac{1}{b-x}\int_x^b f(u)\, du.$$

- Classification: we assume that $y \in \{y_1, \ldots, y_C\}$ is a categorical variable with C categories, x is a continuous variable, and the split criterion is defined by (2.2) using the Tsallis entropy as in (2.3). The components in Definition 1 are specified as follows.

$$g\big(z_1^l(x), \ldots, z_C^l(x)\big) = \frac{1}{1-q}\left(\sum_{i=1}^{C} z_i^l(x)^q - 1\right),$$

$$g\big(z_1^r(x), \ldots, z_C^r(x)\big) = \frac{1}{1-q}\left(\sum_{i=1}^{C} z_i^r(x)^q - 1\right),$$

$$z_i^l(x) = \int_a^x \frac{1}{x-a} * I_{y_i}\big(f(t)\big)\, dt,$$

$$z_i^r(x) = \int_x^b \frac{1}{b-x} * I_{y_i}\big(f(t)\big)\, dt,$$

where $I_{y_i}(f(x))$ is an indicator function that is equal to 1 at $f(x) = y_i$ and 0 elsewhere.

The concept of split convergence can be defined based on the definition of an optimal split as follows.

Definition 2 (Split convergence). A split x_s^n is estimated via greedy search algorithm on the sampled data $D' = \{Y', X'\}$ with size n. Let x_s be the unique optimal split on Ω. If x_s^n converges to x_s in probability as $n \to \infty$, we refer to this case as split convergence and x_s^n as a convergent split.

Our theoretical study demonstrates that split convergence can be guaranteed under three assumptions: (1) the existence of unique optimal split; (2) the uniform random sampling of pseudo-data $D' = \{Y', X'\}$; and (3) the greedy search algorithm. Since continuous and categorical response variables have different split criteria for regression and classification, and different types of covariates require distinct treatments in the proof, we divide the theory into four theorems, each corresponding to one of the combinations of variable types listed in Table 1. The details of all theorems, lemma, and their proofs can be found in Appendix A.

Table 1. Theorems classified based on the combinations of variable types.

	y	
x	Continuous	Categorical
Continuous	Theorem 1	Theorem 2
Categorical	Theorem 3	Theorem 4

Fair assumptions help establish a theory with a solid foundation and broad applicability. Regarding the greedy search assumption, as discussed in Section 2.2.1, it has the advantages of simplicity and popularity in practice. The uniform random sampling assumption ensures the sampling space covers the teacher model and simplifies theoretical proofs. However, it may lead to efficiency issues as the dimension of covariates increases. For problems with modest dimensions (i.e., fewer than 20 continuous variables), uniform random sampling works well (see real data analysis in Section 5). For high-dimensional problems, non-uniform random sampling strategies may be more appealing and worth investigating. As for the unique optimal split assumption, let us consider its opposite first: assume there are multiple optimal splits. We can define the concept of split oscillation in Definition 3. Although split oscillation may occur in theory, it rarely happens in practice. Let us consider a scenario where two optimal splits exist. When applying the greedy search algorithm with real data, the likelihood of two splits yielding identical numerical results (e.g., impurity reduction) will be extremely low. Even if such a rare situation arises, it is not a significant concern. It simply indicates that the two splits are equivalent, and selecting either of them is reasonable.

Definition 3 (Split oscillation). A split x_s^n is estimated via greedy search algorithm on the sampled data $D' = \{Y', X'\}$ with size n. If x_s^n has multiple limits as $n \to \infty$, we refer to this case as split oscillation and the split as an oscillating split.

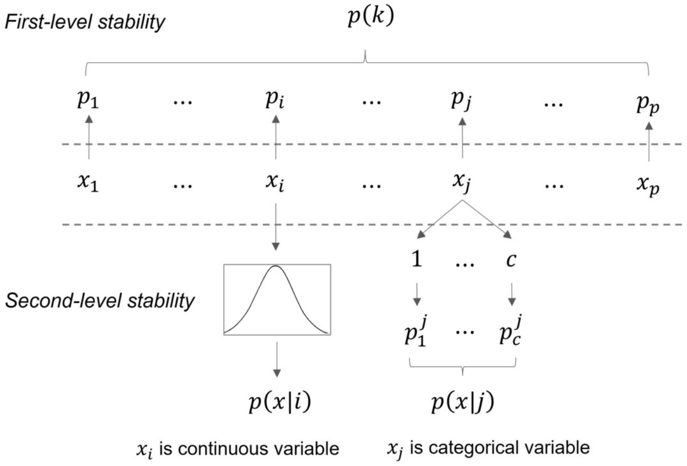

First-level stability
$p(k)$
p_1 ... p_i ... p_j ... p_p
x_1 ... x_i ... x_j ... x_p
Second-level stability
1 ... c
p_1^j ... p_c^j
$p(x|i)$
$p(x|j)$
x_i is continuous variable
x_j is categorical variable

Figure 2: Two-level split stability. First-level stability is denoted as the pmf of choosing a split variable. Second-level stability is denoted as either a pdf or a pmf conditioning on the selected split variable.

2.2.3 Measure of Split Stability

In practice, the pseudo-data must be finite. Therefore, we need a way to measure the split stability with finite data. Motivated by the greedy search algorithm, we propose the two-level split stability (see Figure 2) as follows.

- **First-level stability.** It is defined as a discrete distribution with the probability mass function $p(k)$, which quantifies the stability of selecting the k-th covariate x_k, $(k = 1, \ldots, p)$ as the splitting variable.
- **Second-level stability.** It is defined as the conditional distribution of the splitting value given the covariate to split on. The second-level stability can be either a probability density function (e.g., $p(x|i)$) or a probability mass function (e.g., $p(x|j)$), depending on the type of selected splitting variable.

Since the two-level stability is difficult to calculate analytically, we use Monte Carlo simulation for its estimation.

3. ALGORITHMS FOR CONSTRUCTING KDDT

There are two fundamental distinctions in the construction of KDDT compared to ordinary decision tree (ODT). Firstly, ODT is built directly from a limited dataset, whereas KDDT is constructed using unlimited (in theory) pseudo-data. Secondly, ODT's goal is to best fit the dataset, whereas KDDT's objective is to best approximate the teacher model. These distinctions result in a different construction algorithm of KDDT compared to ODT.

3.1 KDDT Induction Algorithm

We first introduce the concept of the sampling region, which will be utilized in the KDDT induction algorithm.

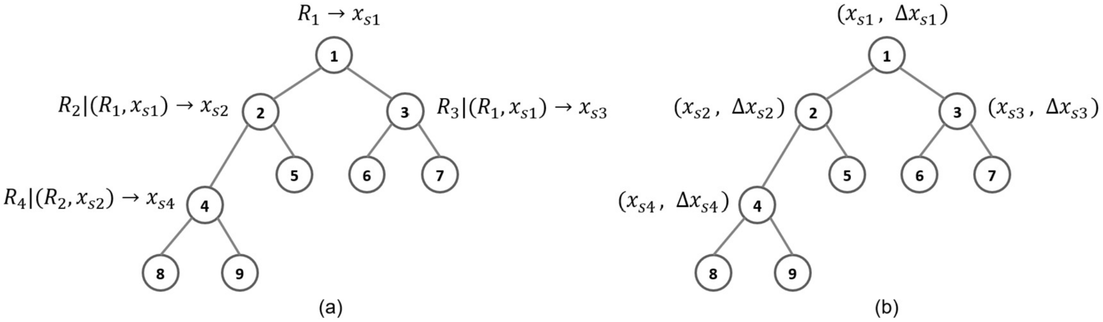

$R_1 \to x_{s1}$
$R_2|(R_1, x_{s1}) \to x_{s2}$
$R_3|(R_1, x_{s1}) \to x_{s3}$
$R_4|(R_2, x_{s2}) \to x_{s4}$
1
2
3
4
5
6
7
8
9
(a)
$(x_{s1}, \Delta x_{s1})$
$(x_{s2}, \Delta x_{s2})$
$(x_{s3}, \Delta x_{s3})$
$(x_{s4}, \Delta x_{s4})$
(b)

Figure 3: Examples of the dependency chain and variance propagation. (a) Two dependency chains in a tree. (b) The corresponding variance propagation. Note: R_i denotes sampling region i. $R_2|(R_1, x_{s1}) \to x_{s2}$ indicates that x_{s2} is determined by R_2, given R_1 and x_{s1}. The variance of x_{si} is denoted by Δx_{si}.

Definition 4 (Sampling region). Let $\mathbb{S}$ be the bounded space defined by the observed data. For node i in KDDT, its ancestors define a subspace on $\mathbb{S}$. We denote it as R_i and refer to it as the sampling region of node i. The sampling region of any inner node is exactly the union of the sampling regions of its two child nodes. (Note: Since the boundary of observed data is limited, $\mathbb{S}$ is bounded.)

As an extension of the sampling region, the concept of the sampling path will also be used later in this paper.

Definition 5 (Sampling path). A sampling path is a series of nested sampling regions defined by the nodes in a KDDT path. The sampling path $P_{i,j}$ starts from sampling region R_i and ends at sampling region R_j, i.e., $P_{i,j} = \{(R_i, \ldots, R_j)|R_i \supset \cdots \supset R_j\}$. Two sampling paths intersect if there exists a sampling region in one sampling path that includes any sampling region in the other sampling path.

The most commonly used induction algorithm for constructing an ODT is a top-down recursive approach [18], referred to as the ODT induction algorithm in this paper. It starts with the entire input dataset in the root node, where a locally optimal split is identified using the greedy search algorithm, and conditional branches based on the split are created. This process is repeated in the generated nodes until the stopping criterion is met. A naive approach to constructing a KDDT is to directly apply the ODT induction algorithm on a large pseudo-dataset. However, this method may not perform well in practice. The pseudo-data introduces variation (uncertainty) due to the random sampling process. This variation propagates in the constructed tree along dependency chains created by the top-down induction strategy. For instance, as illustrated in panel (a) of Figure 3, the split x_{s4} depends on (R_2, x_{s2}), which, in turn, depends on (R_1, x_{s1}). The propagation of split variance follows the inverse direction of these dependencies. We denote the variance of x_{si} as Δx_{si}. In panel (b) of Figure 3, the variance Δx_{s1} will affect $(x_{s2}, \Delta x_{s2})$ and $(x_{s3}, \Delta x_{s3})$, and subsequently, Δx_{s2} will impact $(x_{s4}, \Delta x_{s4})$. This results in rapid inflation of variance as it propagates to deeper levels. For example, a small Δx_{s1} may lead to a substantial Δx_{s4} or even a change in the split variable.

Algorithm 1: Steps of KDDT induction algorithm.

```
Data: pseudo-data
Result: A knowledge distillation decision tree
Starting from the root node, set $i = 1$, and create an
  empty set $X_s$ to store splits.
while the stopping criterion is not met, do
    1. For node $i$, repeat the following processes $N_i$ times.
        (1) Generate pseudo-data, which includes $n_i$
            samples from the sampling region $R_i$
            corresponding to node $i$.
        (2) Fit a stump on the pseudo-data and store the
            split of the stump into $X_s$.
    2. Compute two-level stability with $X_s$ to identify the
       best split $x_s^*$ (the mode of the second-level stability).
    3. Apply $x_s^*$ to create child nodes and set their id as
       $2i$, $2i + 1$, respectively.
    4. Move to the next node that needs to be split.
end
```

Incorporating the two-level stability, we proposed a KDDT induction algorithm to avoid the variation inflation issue in the ODT algorithm. As shown in the steps of KDDT induction algorithm, for a given node i, we measure its split N_i times. Utilizing these repeated measurements represented as X_s, we can calculate the two-level stability and choose a split value with the lowest variance. The first-level stability aids in reducing the variance when selecting the split variable, while the second-level stability assists in reducing the variance when identifying the split value. For example, if the split variable is continuous, considering the mean of all fitted split values $\bar{x}_s$, by central limit theorem, the variance of $\bar{x}_s$ will reduce at a rate of N_i^{-1}. Instead of using the mean of all fitted split values, the two-level stability approach choose the mode, which not only reduces variance but also mitigates the influence of outliers. Additionally, choosing the mode aligns with the likelihood principle, as it corresponds to selecting the value with the maximal likelihood, given that two-level stability is defined by probability mass/density function. By repeating this process at each split, we can construct a KDDT with a stable structure. In practice, it is common to choose a reasonably large value for N_i ($N_i = 100$ works well for our study). The sample size of the pseudo data n_i can be estimated by using (proportional to) the potential explanation index (see Definition 6). In practice, if the number of nodes is small (see interpretable nodes in Section 3.2), we can simply set all n_i to be the same at the same tree level and assign their values equal to 90% of the corresponding value in the preceding upper level. We select 90% to maintain a large number of pseudo-data, ensuring a stable estimation of the split value. We determine the value of x_s^* by selecting the mode of the second-level stability (pmf/pdf). The stopping criterion can be defined as the ratio of prediction accuracy (e.g., MSE or C-Index) between the teacher model and KDDT on the observed data, evaluated through cross-validation.

3.2 Hybrid Induction Algorithm and Hybrid KDDT

We assume that the teacher model is well-defined, meaning it is well-fitted to the observed data without overfitting. Since KDDT aims to optimize its approximation to the teacher model, we can ignore any overfitting concerns for KDDT in relation to the observed data. Thus, we can focus solely on achieving a balance between the degree of approximation and computational efficiency during KDDT construction. KDDT induction algorithm requires repetitive sampling and fitting, performed N_i times to identify the best split. Each time, the pseudo-data need to have a sufficient size, leading to high computational load. Furthermore, to achieve a high-quality approximation, the tree needs to grow to a large size. Consequently, growing a large tree solely using KDDT induction algorithm is often computationally infeasible.

Typically, in most real-world applications, only a small set of splits is needed for interpretation purposes. We refer to these splits as interpretable nodes (splits), while all other nodes, i.e., terminal nodes, are named predictive nodes. Since the ODT induction algorithm is much more efficient in constructing large trees than KDDT's, it is reasonable to combine these two algorithms to a hybrid induction algorithm. Specifically, we apply the KDDT induction algorithm to the interpretable nodes, ensuring their two-level stability, which is crucial for interpretation. Then, we employ the ODT induction algorithm to construct large sub-trees at the predictive nodes, maintaining a good approximation to the teacher model and increasing the computation efficiency. We refer to this tree as a hybrid KDDT. For instance, in Figure 4, the interpretable nodes are $\{1, 2, 3, 4, 7, 14\}$. We use the KDDT induction algorithm to identify the stable splits for these nodes. Then, we employ the ODT induction algorithm to grow the large sub-trees $\{T_5, T_6, T_8, T_9, T_{15}, T_{28}, T_{29}\}$ at the respective predictive nodes.

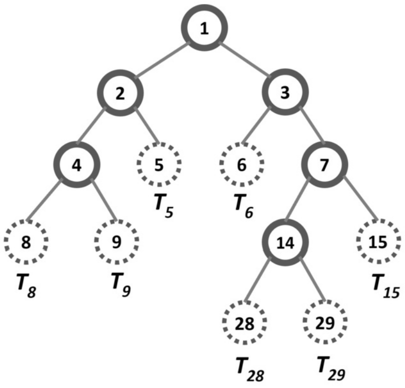


Figure 4: An example of the hybrid knowledge distillation decision tree: the nodes $\{1, 2, 3, 4, 7, 14\}$ are interpretable nodes, while all other nodes are predictive nodes.

The small set of interpretable nodes enhances the simplicity of model interpretation. Meanwhile, the complexity necessary to ensure a prediction accuracy comparable to that of the teacher model is achieved through the construction of large sub-trees at the predictive nodes. This decoupling between interpretability and complexity offers the potential for hybrid KDDT to strike a balance between prediction accuracy and interpretability. For the sake of simplicity, we use the term "KDDT" to refer to hybrid KDDT in the remainder of this paper.

The informativeness of the interpretation can vary across different interpretable nodes. Measuring and reporting these differences is crucial for the interpretation according to these nodes. To address this issue, we introduce the concept of explanation index (XI). A similar index is calculated and referred to as the potential explanation index (PXI) for the predictive nodes.

Definition 6 (Explanation Index and Potential Explanation Index). The explanation index of interpretable node (split) i, denoted as XI_i, and the potential explanation index of predictive node j, denoted as PXI_j, are defined as follows:

$$XI_i = \frac{\frac{n_i}{n} * \Delta_{S_i}}{\Delta_{KDDT}} * 100\%, \quad PXI_j = \frac{\frac{n_j}{n} * \Delta_{T_j}}{\Delta_{KDDT}} * 100\% \quad (3.1)$$

where n_i, n_j, and n denote the number of observations in node i, j, and the entire dataset. Δ_{S_i} and Δ_{T_j} represent the change in the measure defined by the split criterion (e.g., impurity or MSE reduction) after fitting split i or the subtree at node j, respectively. $\Delta_{KDDT} = \sum_i \frac{n_i}{n} * \Delta_{S_i} + \sum_j \frac{n_j}{n} * \Delta_{T_j}$.

Based on Definition 6, it is straightforward to verify that $\sum_i XI_i + \sum_j PXI_j = 1$. XI_i and PXI_j can be considered as information contained in the interpretable node i and

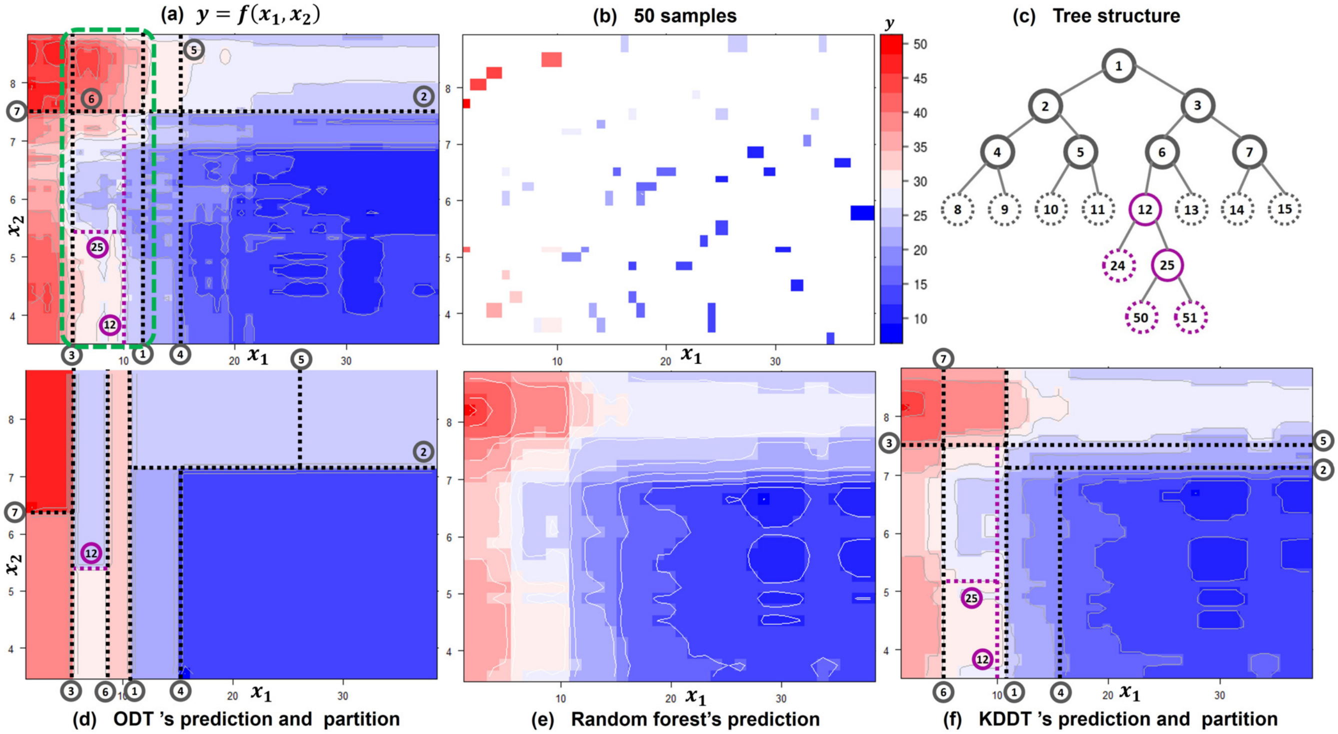


Figure 5: The effectiveness of KDDT in revealing and explaining complex data structures. (a) The true function $y = f(x_1, x_2)$ and its partition with 9 splits. (b) 50 random samples from the true function. (c) The tree structure and splits for defining the partitions in (a), (d), and (f). (d) ODT is fitted based on 50 samples. (e) RF is fitted based on the 50 samples. (f) KDDT is constructed based on the RF.

predictive node j, respectively. Furthermore, we can extend the concept of XI to apply to a path in KDDT as follows.

Definition 7 (Path Explanation Index).

$$XI_{ij} = \sum_{k \in S_{ij}} XI_k \tag{3.2}$$

where node i is an interpretable node, node j is a descendant of node i, and S_{ij} is a set of node IDs that includes the nodes in the path from node i to the parent of node j.

With the above indices, we can identify the desired hybrid KDDT with an appropriate number of interpretable nodes. For instance, if we want to achieve more than 70% of the information in the data explained by the interpretable nodes, the stopping criterion is $\sum_i XI_i > 70\%$, i.e., $\sum_j PXI_j < 30\%$. Examples demonstrating their applications can be found in Section 5. The process of constructing a desired hybrid KDDT with appropriate number of interpretable nodes is illustrated by an example shown in the Figure B.14 in Appendix B. The potential explanation index of a predictive node can also be used to determine the size of the pseudo data in its sampling region which could be proportional to its PXI. Because, a higher PXI indicates greater unexplained information, requiring a larger pseudo data size.

4. SIMULATION STUDY

The simulation study has three primary objectives: (1) to demonstrate the effectiveness of KDDT in revealing intricate structures of the data, (2) to validate the interpretability of KDDT, and (3) to illustrate the stability of interpretable splits (nodes).

To facilitate a clear and intuitive discussion, we introduce a two-dimensional function $y = f(x_1, x_2)$, consisting of 2601 generated data points, as illustrated in panel (a) of Figure 5. This function exhibits high non-linearity and intricate interactions, making it well-suited for our purposes. Let us assume that $y = f(x_1, x_2)$ is unknown. We can gain insights about it by analyzing the sampled observations. In panel (b), we have 50 observations randomly sampled from the true function. We want to compare KDDT with other interpretable models. The well-known ones include linear regression and decision tree (ODT). As linear regression is not suitable for this data, we opt for ODT. The ODT fitted from 50 observations is presented in panel (d). In comparison to the true function, the ODT estimation is coarse and unable to capture the interaction structure within the area marked by the green rectangle. In contrast, the random forest model provides a refined and precise estimation, as shown in panel (e). The KDDT presented in panel (f), as a close approximation of its teacher, maintains a high-quality (resolution) estimation. This highlights the ability of the KDDT to re-

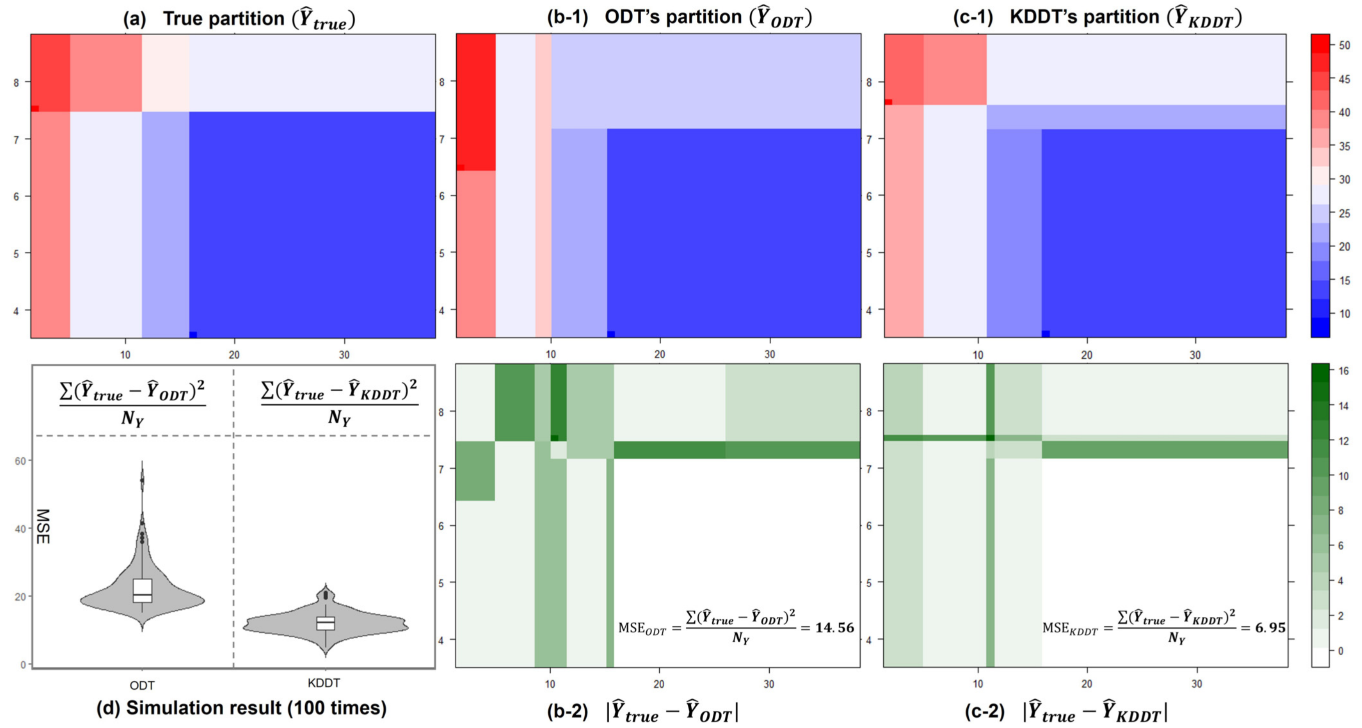


Figure 6: The comparison of interpretations and simulation result. (a) The true partition is obtained from the ODT that is fitted based on the entire data of the true function. (b-1) The ODT is fitted based on 50 samples. (b-2) The result of $|\hat{Y}_{true} - \hat{Y}_{ODT}|$. (c-1) The KDDT is built from RF. (c-2) The result of $|\hat{Y}_{true} - \hat{Y}_{KDDT}|$. (d) MSE comparison of ODT and KDDT with 100 times simulations.

veal intricate structures in the data.

For the function $y = f(x_1, x_2)$, effective interpretation is visually demonstrated through a suitable partition of the response values y based on the covariates x_1 and x_2, as shown in panel (a) of Figure 5. This partition comprises nine splits generated by the ODT in panel (c), which is fitted using the entire dataset of 2601 data points. We refer to this partition as the true partition, representing the optimal interpretation. Although the random forest model provides an accurate estimation of the true function, it cannot generate a partition for interpretation. The ODT (fitted with 50 samples) is interpretable, but its interpretation (partition) is not accurate. In contrast, the KDDT's interpretation closely approximates the optimal one (true partition), which is better than the ODT's. This claim relies on visual inspection, which is a qualitative approach. Figure 6 presents a quantitative method for comparing the quality of interpretation between ODT and KDDT. Panel (a) displays the true partition (optimal interpretation). The partitions of ODT and KDDT are depicted in panels (b-1) and (c-1), respectively. Panels (b-2) and (c-2) illustrate the absolute errors of ODT and KDDT compared to the truth. Clearly, visual inspection still leads to the same conclusion that KDDT's interpretation (partition) is superior to ODT's. More importantly, we can quantify this difference using MSE. In this example, KDDT's MSE is 6.95, significantly smaller than ODT's MSE of 14.56. Furthermore, we repeat this comparison 100 times. The result in panel (d) demonstrates that, in general, KDDT outperforms ODT in terms of interpretation quality measured by MSE. Note that the medians of MSE are 20.32 and 12.14 for ODT and KDDT, respectively. The corresponding means of MSE are 22.39 and 12.17. KDDT results in a 40.3% reduction in the median and a 45.6% reduction in the mean compared to ODT. The maximum MSE of KDDT is 20.93, corresponding to 53 percentile of ODT's. The maximum MSE of ODT is 54.06, which is more than 2.5 times higher than the KDDT's.

To examine the KDDT in panel (c) of Figure 5 in more detail, it contains nine interpretable nodes (splits). The first-level and second-level stability can be found in Figure 7. Except for split 12, which maintains a still impressive first-level stability of 97%, all other splits exhibit a first-level stability of 100%. Regarding second-level stability, each density function is tightly concentrated within a narrow interval and displays a sharp peak. Consequently, we can confidently assert that the interpretable splits within the KDDT are stable.

5. APPLICATIONS

When should we use KDDT? Two fundamental conditions should be met.

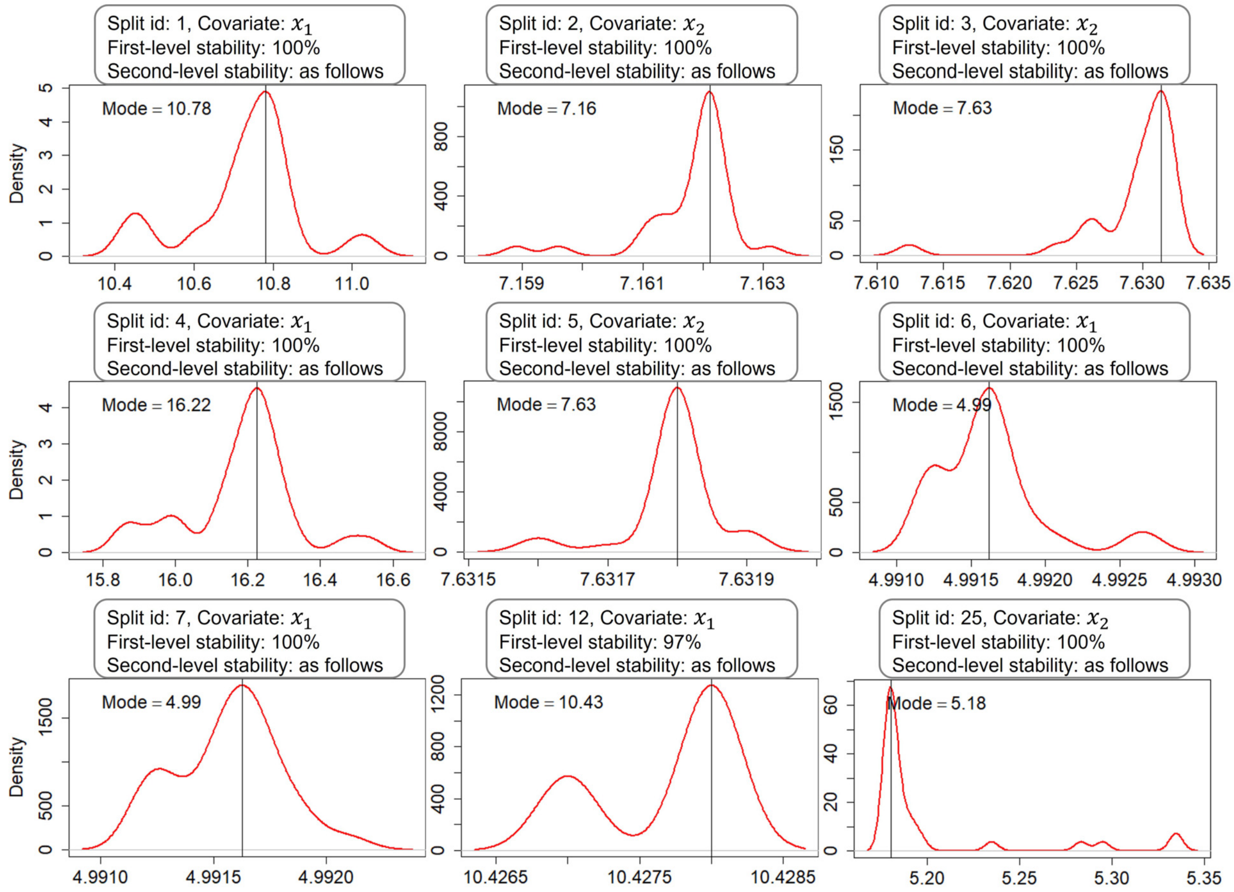


Figure 7: The two-level stability of interpretable splits in panel (f) of Figure 5.

- **Demands for understanding or explanation:** We need to understand or explain the data, either to gain personal insight or to communicate findings to others.
- **Possess good prediction accuracy:** The black-box ML model, which KDDT aims to approximate, should outperform simple interpretable models, such as linear regression or ODT, in predicting the data. This suggests that the black-box model may have a better understanding of the data and the potential to offer a more accurate interpretation compared to the simple models.

Considering these conditions, we discuss two real applications of KDDT in this section.

5.1 Example for Model Interpretation

In the application of model interpretation, we use the Boston Housing dataset, which comprises a total of 506 observations with 14 variables. The description of variables can be found in Table B.2 in Appendix B. Our goal is to understand the effects of covariates on the price of houses in Boston (in 1970). To check the second condition, we select the linear regression model (LM) and ODT as simple interpretable models while choosing the random forest (RF) and SVM as two candidate black-box models. A five-fold cross-validation was conducted to compare their prediction accuracy. The MSE (mean square error) on testing data are LM: 23.2, ODT: 24.9, RF: 10.9, SVM: 13.4, and KDDT(RF): 14.9. More details of comparison can be found in Figure B.13 in Appendix B. From the results, the ML models outperform the simple interpretable models, and RF performs better than SVM. Hence, we can choose RF as the teacher model. The student model KDDT(RF) outperforms the simple interpretable models and exhibits similar performance to SVM. It indicates that KDDT(RF) may offer a more accurate interpretation than the simple interpretable models.

The panel (a) of Figure 8 illustrates the interpretations of KDDT(RF) for its teacher model RF. Since KDDT(RF) is essentially a decision tree, identifying the variables of importance is straightforward. The three most important variables are lstat, rm, and nox, related to social status, house size, and the natural environment, respectively. This is consistent with the corresponding results of the teacher model RF (see Figure B.15 in Appendix B), which is evidence that KDDT(RF) can provide accurate interpretation for its teacher model. More detailed and specific interpretations can be obtained by examining the interpretable splits (nodes) featured in panel (a). For example, if a house has

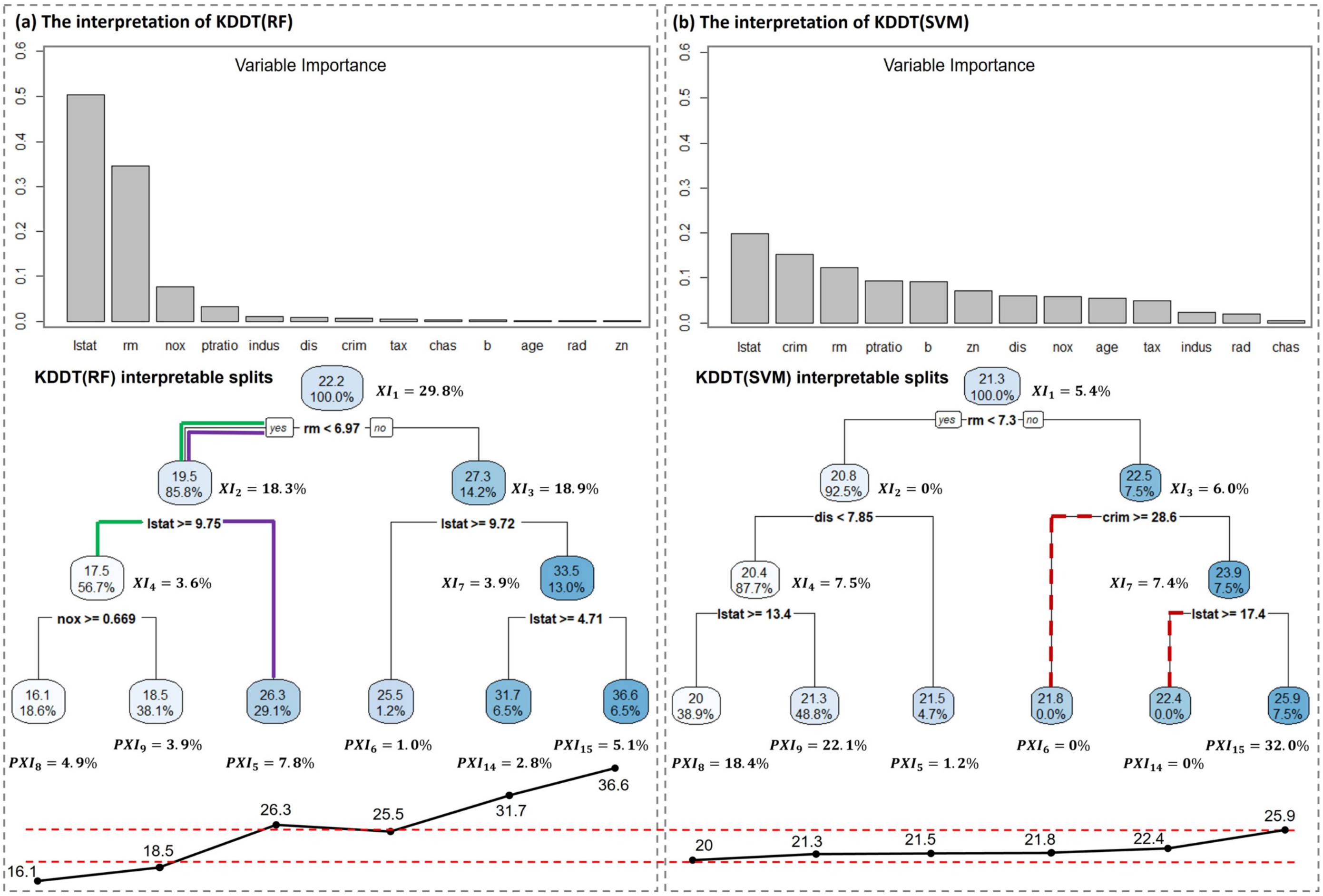
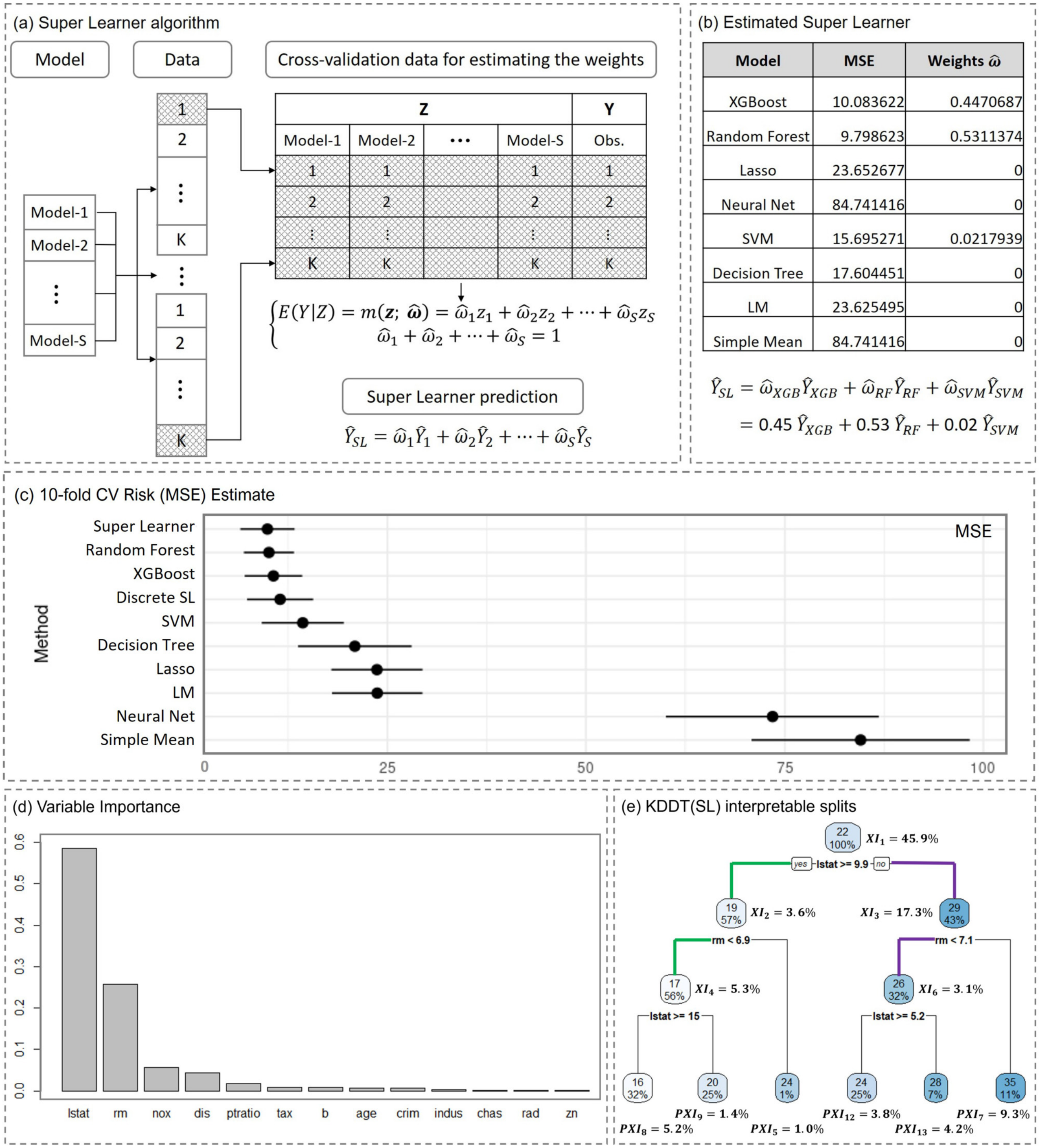


Figure 8: Model interpretation through KDDT. (a) The interpretation of RF using KDDT(RF). (b) The interpretation of SVM using KDDT(SVM). Note that the left node (yes) and the right node (no) indicate whether the split condition is met or not, respectively. Note: the process of constructing KDDT(RF) in panel (a) can be found in the Figure B.14 in Appendix B.

seven or more rooms and is situated in an affluent community where the percentage of the population with lower social status (lstat) is less than 4.71%, it is likely to have a high value, averaging $36,600. Additionally, for potential buyers, an intriguing insight emerges: they might acquire a larger house with seven or more rooms in a less affluent community with lstat $\geq$ 9.8%, priced around $25,500, which is cheaper than a smaller house that could cost around $26,300 in a community with lstat $\leq$ 9.7%. These specific insights are exemplified by nodes 5 and 6 in the tree. The stability of the interpretable splits shown in Figure B.16 in Appendix B ensures the credibility of interpretations.

In panel (a) of Figure 8, the XI and PXI associated with the interpretable and predictive nodes provide the relative importance information for their interpretation. For instance, $XI_1 = 29.8\%$ for the split rm<6.97 indicates whether a house has seven or more rooms is crucial for assessing its value. Moreover, these indices could serve as stop criterion for identifying the interpretable nodes set. For example, we can identify the KDDT(RF) interpretable nodes by the criterion that the sum of PXI is less than 30%. This criterion ensures that predictive nodes do not contain substantial information. Furthermore, we can interpret any prediction of KDDT by using the concept of the path explanation index in Definition 7. For example, if a prediction is made through the predictive node 9 (see panel (a)), its XI can be calculated as $XI_{1,9} = XI_1 + XI_2 + XI_4 = 51.7\%$. Then, with the $PXI_9 = 3.9\%$, we can obtain that $(\frac{XI_{1,9}}{XI_{1,9}+PXI_9}, \frac{PXI_9}{XI_{1,9}+PXI_9}) = (93\%, 7\%)$. It indicates that the prediction can be interpreted with a degree of 93% using the chain of decision rules $\{rm < 6.97 \rightarrow lstat \geq 9.75 \rightarrow nox \geq 0.669\}$.

Last but not least, the percentage of observed data of each node also plays a pivotal role in comprehending the interpretation of KDDT. This percentage serves as crucial evidence of how strongly the interpretation of a particular node is supported by the observed data. Given that KDDT is not a direct interpretation of the observed data but rather



Model	MSE	Weights $\hat{\omega}$
XGBoost	10.083622	0.4470687
Random Forest	9.798623	0.5311374
Lasso	23.652677	0
Neural Net	84.741416	0
SVM	15.695271	0.0217939
Decision Tree	17.604451	0
LM	23.625495	0
Simple Mean	84.741416	0

Figure 9: The interpretation of super learner through KDDT. (a) The framework of super learner algorithm/model. The number 1, ..., K refer to different cross-validation folds. The gray folds refers testing data. (b) The estimated weights and super learner. (c) The comparison of prediction accuracy between super learner and base models. (d) The variable importance of KDDT(SL). (e) The tree structure of KDDT(SL).

of the teacher model, the support from the observed data is pivotal for the interpretation's practical significance. Even a node (split) with a high XI may lack practical relevance if the percentage of observed data associated with it (or its children) is exceedingly low. For instance, consider node 6 (split 3). Although it has $XI_{1,6} = XI_1 + XI_3 = 48.7\%$ ($XI_3 = 18.9\%$), it (left child) comprises a mere 1.2% of observed data. This suggests that the interpretation of this node (split) might not carry much practical importance. In other words, the chance of purchasing a larger house at a lower price is not zero, but it is very low in practice. Consequently, it is imperative to take into account both the XI and the percentage of observed data when interpreting KDDT. As an example, we are confident in the interpretation of predictions made through node 9. Because this node not only has a high path XI of $XI_{1,9} = 51.7\%$ that can be interpreted with a degree of 93% but also enjoys strong practical support from a large number (38.1%) of observed data.

As demonstrated in panel (b) of Figure 8, KDDT can also provide an interpretation for SVM, which differs from the one for RF. In KDDT(SVM), the top three important variables are lstat, crim, and rm, related to social status, security, and house size, respectively. It indicates that, except for social status and house size, the SVM's explanation focuses on security, in contrast to RF emphasis on natural environment. Regarding the interpretable splits, the sum of their XIs in KDDT(SVM) is 26.3%, which is smaller than the 74.5% in KDDT(RF). This suggests that the interpretable nodes set of KDDT(SVM) has less interpretability compared to its counterpart in KDDT(RF). Their comparison shown at the bottom of Figure 8 provides an intuitive illustration supporting this assertion, demonstrating that more variation in the data is explained by KDDT(RF) than by KDDT(SVM). Another issue of KDDT(SVM) is that splits 3 and 7 have child nodes 6 and 14, respectively, which do not include any observed data. To address this, we can omit these two branches (red dashed lines) and focus solely on node 15. The path explanation index from node 1 to 15 can then be calculated as $XI_{1,15} = XI_1 + XI_3 + XI_7 = 18.8\%$. In sum, through KDDT, SVM can offer a different interpretation compared to RF. But, the interpretable splits in KDDT(SVM) do not perform as effectively as their counterparts in KDDT(RF).

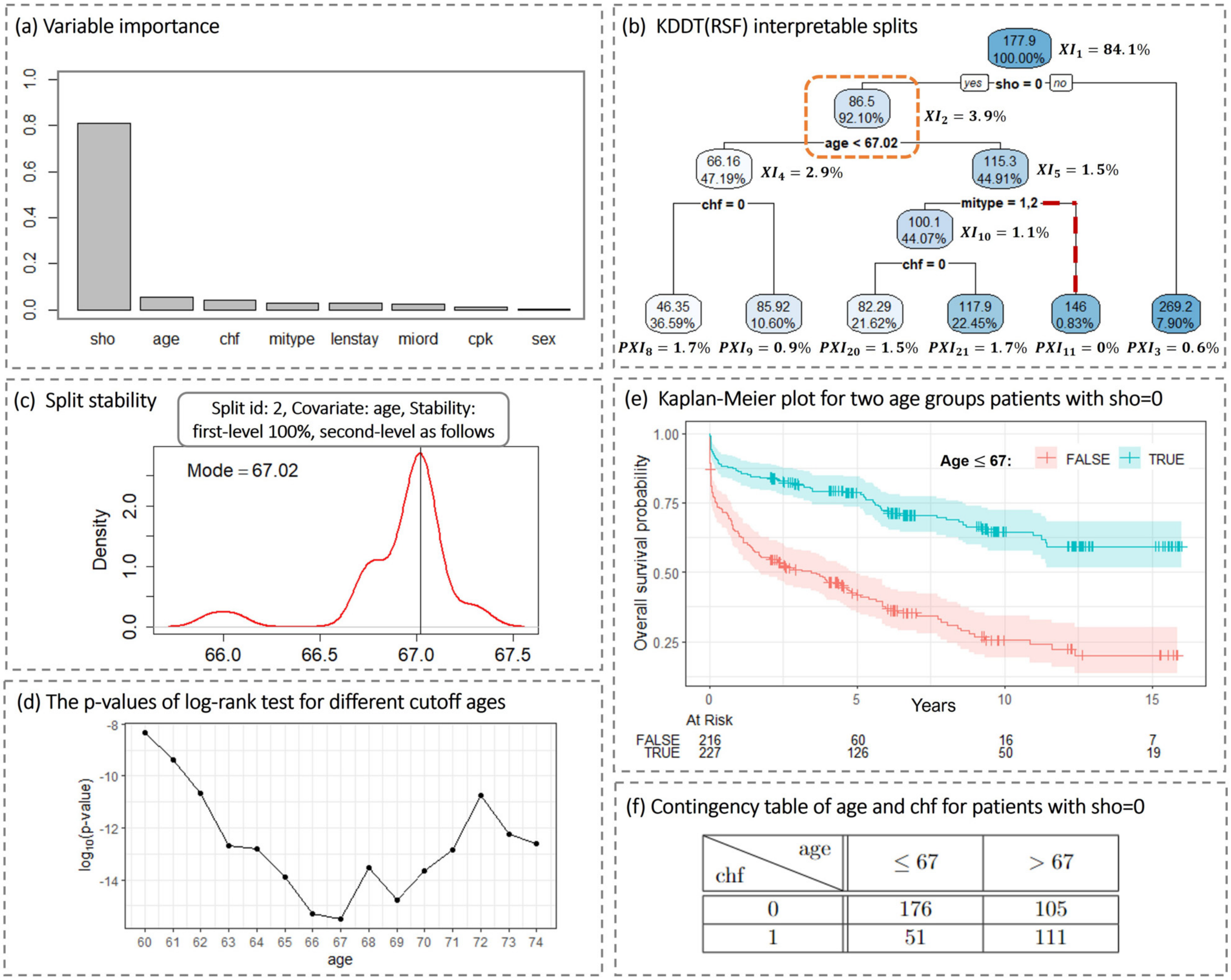
KDDT can also be valuable in interpreting the model that is ensembled from other models. One typical example is the Super Learner introduced by [13]. As depicted in panel (a) of Figure 9, the Super Learner employs cross-validation to estimate the performance of multiple base models. Subsequently, it constructs an optimal weighted average of these models based on their testing performance. This approach has been proven to yield predictions that are asymptotically as good as or even better than any single model within the ensemble. In this example, we introduced eight base models and estimated their weights in the Super Learner, as shown in panel (b). Evaluated through a 10-fold cross-validation, the result presented in panel (c) demonstrates that the Super Learner outperforms all its base models in terms of prediction accuracy, which satisfies the second condition for applying KDDT.

Compared to the base models, the ensemble nature of the Super Learner renders it a more opaque black-box model, which makes the interpretation more challenging. KDDT can provide a solution. Panel (d) of Figure 9 presents the variable importance of KDDT(SL), which remarkably resemble those of the RF model shown in panel (a) of Figure 8. In panel (e) of Figure 9, interpretable splits (nodes) were selected based on the criterion that the sum of PXI is less than 30%. The sum of XIs is 75.2%, indicating that the interpretable nodes of KDDT(SL) offer substantial interpretability. An interesting observation emerges when comparing KDDT(RF) and KDDT(SL): the predictions and interpretations of nodes 4 and 5 in KDDT(RF) closely resemble those of nodes 4 and 6 in KDDT(SL). In panel (a) of Figure 8 and panel (e) of Figure 9, these corresponding paths are highlighted in green and purple, respectively. Notably, all of the paths exhibit both high path XI and substantial percentages of observed data. This suggests a strong similarity in interpretation between RF and the Super Learner.

We have three KDDT interpretations associated with RF, SVM, and Super Learner. It is important to be aware that all of these interpretations are reasonable and valid. All roads lead to Rome. Choosing which one depends on the application requirements. For example, consider a real estate consultant whose client is interested in the natural environment of the house, KDDT(RF)'s explanation would be a good choice. If the client's main concern is the safety of the neighborhood, KDDT(SVM)'s interpretation may be a better choice. Furthermore, if significant splits or paths consistently appear in different KDDT interpretations, it serves as an indicator of their critical roles in the data. These interpretations have the potential to provide valuable insights or knowledge about the data or application. For example, as discussed in the comparison of KDDT(RF) and KDDT(SL), we can derive the valuable insight that 10% lower status of the population and 7 rooms are two critical thresholds shaping people's evaluations of house prices in Boston.

5.2 Example for Subgroup Discovery

With the ability to uncover patterns in complex data and explore non-linear relationships, ML models have gained popularity in data-driven precision medicine, fueled by the rapid expansion in the availability of a wide variety of patient data. In precision medicine, identifying heterogeneity plays a central role, where subgroups of patients are defined based on baseline values of demographic, clinical, genomic, and other covariates, known as biomarkers. Understanding the effects of biomarkers in data analysis models is crucial for subgroup discovery. KDDT can bridge the gap between understanding the role of biomarkers and the lack



chf \ age	≤ 67	> 67
0	176	105
1	51	111

Figure 10: Subgroup discovery and optimal cutoff identification. (a) Variable importance provides information for selecting split variables. (b) Interpretable splits for identifying optimal cutoff and subgroups. (c) The two-level stability of split 2. (d) Log10(p-value) of log-rank tests for validating the optimality of cutoff value. (e) The Kaplan-Meier plot for the identified subgroups. (f) Contingency table depicts the association between age and chf.

of interpretability in black-box ML data analysis models. Particularly, as a tree-based approach, KDDT can incorporate information on higher-order interaction effects and be applied to define subgroups based on multiple biomarkers. Moreover, cutoff values do not need to be pre-specified for continuous/ordinal biomarkers. They are automatically estimated from the process of constructing KDDT.

In this example, we select the time-to-event dataset WHAS (Worcester Heart Attack Study), whose aim is to describe factors associated with trends in incidence and survival over time after admission for acute myocardial infarction. This dataset is available in the R package "mlr3proba", and includes 481 observations and 14 variables. Four variables, id (Patient ID), year (Cohort year), yrgrp (Grouped cohort year), and dstat (Discharge status from the hospital: 1 = Dead, 0 = Alive), were excluded as they were not pertinent to the goal of study. The description of the remaining variables can be found in Table B.3 in Appendix B. We choose Cox Proportional Hazard (CoxPH) model as the interpretable model and Random Survival Forest (RSF) as the black-box teacher model. Similar to Section 5.1, the comparison of prediction accuracy was conducted with a five-fold cross-validation. Instead of MSE, the C-index serves as the criterion, with higher C-index indicating higher accuracy. The result on testing data is CoxPH: 0.766, RSF: 0.797, and KDDT(RSF): 0.797. More details of the comparison can be found in Figure B.17 in Appendix B. This result demonstrates the superiority of RSF in prediction and suggests that it is worth trying to take advantage of KDDT(RSF) in the application of subgroup discovery.

The structure of KDDT(RSF) is depicted in panel (b) of Figure 10. As the first split, sho=0, $XI_1 = 84.1\%$ suggests a great practical significance for the identified subgroups. Actually, it is widely recognized that cardiogenic shock is positively associated with an increased risk of death. It is not a surprising discovery. The researcher's interest may lie more in the subgroups identified from the patients who didn't experience cardiogenic shock. The second split, age< 67.02, reveals two subgroups. Although $XI_2 = 3.9\%$ is not high compared to $XI_1 = 84.1\%$, it is relatively high in the rest of interpretable nodes, $\frac{XI_2}{XI_2+XI_4+XI_5+XI_{10}} = \frac{3.9\%}{3.9\%+2.9\%+1.5\%+1.1\%} = 41.5\%$. Moreover, the observed data in its child nodes are substantial and well-balanced, indicating strong support from the observed data. They are evidence that indicates the importance of the subgroups identified by the second split. The split stability of the optimal cutoff value is displayed in panel (c). The greedy search algorithm ensures its optimality which is substantiated by the p-values from log-rank tests across different values in panel (d). Consequently, there is no need to explore multiple cutoff values, thus alleviating the multiplicity issues. Panel (e) displays the Kaplan-Meier plot for the two subgroups, illustrating the varying risks associated with each subgroup. Finer subgroups and covariates interactions can be explored by considering deeper splits. Since node 11 just contains 4 observations (0.83% of the data), we can remove it and its parent node 5 (see the red dashed line). This can be achieved by redistributing these 4 observations to nodes 20 and 21 based on their chf values. As a result, four subgroups with the number of patients can be identified in the table in panel (f). Analyzing this table reveals a clear interaction (dependency) between the risk of left heart failure (chf=1) and the age of patients. This relationship can be statistically confirmed through a χ^2 test, which yields a p-value of 2.655e-10.

6. DISCUSSION

KDDT offers a general method for interpreting black-box ML models, enabling the exploration of intricate data structures captured by these models for more precise and detailed interpretations. Essential attributes for good interpretable models include simplicity, stability, and predictivity. Stability is the central focus of this study. The primary challenge lies in constructing a stable KDDT while handling the randomness of the pseudo-data (knowledge) sampled in the knowledge distillation process. We propose a comprehensive theory for split stability and develop efficient algorithms for constructing stable KDDTs. To ensure simplicity, KDDT efficiently decouples the tasks of interpretation and prediction, maintaining a concise set of interpretable nodes for the purpose of interpretation. Regarding predictivity, KDDT, as a closed approximation of black-box ML models, retains strong predictive performance comparable to the original black-box models. In conclusion, KDDT is an excellent interpretable model with great potential for practical applications.

In our theory and algorithms, we employed the random sampling method to generate pseudo-data for constructing KDDT. This approach performed well in simulation and real data studies. Specifically, when the sample size is less than 60000, the time required to fit an interpretable node was under one minute. In general, for cases where the number of continuous covariates (n_{con}) is relatively small, typically less than 20, the sample size of 60000 is sufficient. However, when dealing with larger n_{con}, a larger sample size is necessary. In such cases, random sampling will be less efficient, and non-uniform random sampling strategies may be more attractive. Two promising strategies are MCMC sampling, which leverages information from the teacher model to enhance sampling efficiency, and PCA sampling, which uses dimension reduction to improve sampling efficiency. They are interesting directions for future study.

APPENDIX A. THEOREMS AND PROOFS

Although the teacher model $f(\mathbf{x})$ may have p-dimensional covariates $\mathbf{x} = (x_1, \dots x_p)$, only one covariate is used at each split. Therefore, we need to marginalize over all other covariates to eliminate their influence. The result is a unary function $f(x)$ defined as follows.

$$f(x) = f_k(x_k) = \int \dots \int f(\mathbf{x})\, d\mathbf{x}_{-k}, \quad k = 1, \dots, p, \tag{A.1}$$

where $d\mathbf{x}_{-k} = \prod_{i \neq k} dx_i$. If x_j is categorical variables takes values in $\{1, \dots, C\}$, we set $\int f(\mathbf{x}_{-j}, x_j)\, dx_j = \sum_{l=1}^{C} f(\mathbf{x}_{-j}, x_j) I(x_j = l)$, where $\mathbf{x}_{-j}$ is the vector $\{x_i\}_{i \neq j}$, $I(\cdot)$ is an indicator function. We don't need to explicitly perform the integral for the univariate projection. It is implicitly handled in the greedy search algorithm by assessing the relationship between the univariate x and the response y at each split.

Lemma 1. *Assume $x \in [a, b]$, where $a, b \in \mathbb{R}$, be a continuous variable in the teacher model $y = f(x)$, and y can be a continuous or categorical variable. Let $z_c^l(x) = \int_a^x h_c^l(t)\, dt$, $z_c^r(x) = \int_x^b h_c^r(t)\, dt$, where $h_c^l(\cdot)$ and $h_c^r(\cdot)$ are integrable functions in $[a, b]$, $c \in \{1, \dots, C\}$ and $C \in \mathbb{N}^+$. The function $g(\cdot) : \mathbb{R}^{\mathbb{C}} \to \mathbb{R}$ is defined in Definition 1. Let x_s be the unique optimal split in (a, b) that is defined by (2.4).*

Consider $\{x_1, \dots, x_{n-1}\}$ as $n-1$ points drawn uniformly at random from the interval (a, b), and arrange them in ascending order. Let $x_0 = a$ and $x_n = b$, and include them in $\{x_1, \dots, x_{n-1}\}$ to form the set $\{x_0, x_1, \dots, x_{n-1}, x_n\}$. Utilizing the teacher model, we can generate pseudo-data as $\{(x_0, f(x_0)), (x_1, f(x_1)), \dots, (x_{n-1}, f(x_{n-1})), (x_n, f(x_n))\}$. Subsequently, we can fit a stump to the pseudo-data by employing the greedy split search algorithm. The split criterion is defined as follows.

$$x_s^n = \underset{x_k, k \in \{1, \dots, n-1\}}{\operatorname{argmin}} \big[g\big(z_1^{l(n)}(x_k), \dots, z_C^{l(n)}(x_k)\big) + g\big(z_1^{r(n)}(x_{k+1}), \dots, z_C^{r(n)}(x_{k+1})\big) \big], \tag{A.2}$$

where, $z_c^{l(n)}(x_k) = \sum_{i=1}^{k} h_c^l(x_i) * \Delta_i$*,* $z_j^{r(n)}(x_{k+1}) = \sum_{j=k+1}^{n} h_c^r(x_j) * \Delta_j$ *and* $\Delta_i = x_i - x_{i-1}$*,* $i = 1, \ldots, n$*.*

Let k_s^n *denote the optimal integer* k *that minimized* (A.2)*, in other worlds,* $x_s^n = x_{k_s^n}$*. Then, the following holds:*

$$x_s^n \xrightarrow{p} x_s, \quad \textit{as} \;\; n \to \infty.$$

The rate of convergence is $O(n^{-1})$*.*

Proof. There must exist a point x_m such that,

$$|x_s - x_m| = min\{|x_s - x_i|\}, \;\; i = 1, \ldots, n-1.$$

Because $x_i, \;\; i = 1, \ldots, n-1$ are uniformly distributed in (a, b). For a constant $\epsilon, \;\; 0 < \epsilon < b - a$, by the theory of order statistics, it is easy to prove the following:

$$P\Big(|x_s - x_m| > \frac{\epsilon}{2}\Big) = \Big(1 - \frac{\epsilon}{b-a}\Big)^{n-1}. \tag{A.3}$$

For any $\epsilon, \;\; 0 < \epsilon < b - a$,

$$\lim_{n\to\infty} P\Big(|x_s - x_m| > \frac{\epsilon}{2}\Big) = \lim_{n\to\infty}\Big(1 - \frac{\epsilon}{b-a}\Big)^{n-1} = 0. \tag{A.4}$$

In other words, $x_m \xrightarrow{p} x_s$ as $n \to \infty$.

For any two consecutive points x_{i-1} and x_i, $i = 1, \ldots, n$, it is easy to know

$$\begin{aligned} |x_i - x_{i-1}| =& min\{|x_0 - x_i|, \ldots, |x_{i-2} - x_i|, \\ & |x_{i-1} - x_i|, |x_{i-1} - x_{i+1}|, \ldots, |x_{i-1} - x_n|\}. \end{aligned}$$

Thus, for a constant $\epsilon, \;\; 0 < \epsilon < b - a$, by the theory of order statistics, we have that

$$P\Big(|x_i - x_{i-1}| > \frac{\epsilon}{2}\Big) = \Big(1 - \frac{\epsilon}{b-a}\Big)^{n-1}, \quad i = 1, \ldots, n.$$

Let $\Delta_i = |x_i - x_{i-1}|$. Similar to (A.4), we can prove that $\Delta_i \xrightarrow{p} 0$ as $n \to \infty$ for $i = 1, \ldots, n$.

Because $h(\cdot)$ is integrable in $[a, b]$ and $x_m \xrightarrow{p} x_s$, $\Delta_i \xrightarrow{p} 0$, as $n \to \infty$, we can get that

$$\begin{aligned} & \lim_{n\to\infty} z_c^{l(n)}(x_m) \\ &= \lim_{n\to\infty} \sum_{i=1}^{m} h_c^l(x_i) * \Delta_i \xrightarrow{p} \int_a^{x_s} h_c^l(u)\,du \\ &= z_c^l(x_s). \end{aligned} \tag{A.5}$$

Similarly, by $x_{m+1} \xrightarrow{p} x_s$ as $n \to \infty$ (because $\Delta_{m+1} \xrightarrow{p} 0$ as $n \to \infty$), we can prove that,

$$\lim_{n\to\infty} z_c^{r(n)}(x_{m+1}) \xrightarrow{p} z_c^r(x_s).$$

Then, by the continuity of function g, we can obtain that

$$\begin{aligned} & \lim_{n\to\infty} \big[g\big(z_1^{l(n)}(x_m), \ldots, z_C^{l(n)}(x_m)\big) \\ & + g\big(z_1^{r(n)}(x_{m+1}), \ldots, z_C^{r(n)}(x_{m+1})\big)\big] \\ & \xrightarrow{p} g\big(z_1^l(x_s), \ldots, z_C^l(x_s)\big) + g\big(z_1^r(x_s), \ldots, z_C^r(x_s)\big). \end{aligned} \tag{A.6}$$

Because of $\Delta_i = |x_i - x_{i-1}| \xrightarrow{p} 0$ as $n \to \infty$, for any two consecutive points, we have $|x_{k_s^n} - x_{k_s^n+1}| \xrightarrow{p} 0$. Therefore, there must exist a point x_s^*, such that $x_{k_s^n} \xrightarrow{p} x_s^*$ and $x_{k_s^n+1} \xrightarrow{p} x_s^*$. Consequently, recall equation (A.2), we have $x_{k_s^n} = x_s^n \xrightarrow{p} x_s^*$ as $n \to \infty$. Now, let us assume that the sequence $\{x_s^n\}$ does not converge solely to x_s^*. This would imply the existence of at least two distinct values, say x_{s1}^* and x_{s2}^* (with $x_{s1}^* \neq x_{s2}^*$), such that both $x_s^n \xrightarrow{p} x_{s1}^*$ and $x_s^n \xrightarrow{p} x_{s2}^*$ hold. This means that both x_{s1}^* and x_{s2}^* satisfied equation (A.2), which is equivalent to the definition of an optimal split in equation (2.4). This implies the existence of two optimal splits, contradicting the assumption of a unique optimal split. So, we have

$$x_s^n \xrightarrow{p} x_s^*, \quad \textit{as} \;\; n \to \infty.$$

Because $x_s^n = x_{k_s^n}$ and $x_{k_s^n+1} \xrightarrow{p} x_s^n$ as $n \to \infty$, we know

$$x_{k_s^n} \xrightarrow{p} x_s^* \quad , \quad x_{k_s^n+1} \xrightarrow{p} x_s^*.$$

Using the integrability of $h(\cdot)$ and the same proof procedures of (A.5) and (A.6), we can obtain the following:

$$\begin{aligned} & \lim_{n\to\infty} \big[g\big(z_1^{l(n)}(x_{k_s^n}), \ldots, z_C^{l(n)}(x_{k_s^n})\big) \\ & + g\big(z_1^{r(n)}(x_{k_s^n+1}), \ldots, z_C^{r(n)}(x_{k_s^n+1})\big)\big] \\ & \xrightarrow{p} g\big(z_1^l(x_s^*), \ldots, z_C^l(x_s^*)\big) + g\big(z_1^r(x_s^*), \ldots, z_C^r(x_s^*)\big). \end{aligned} \tag{A.7}$$

According to the greedy search algorithm and split criterion in (A.2), for all $n \in \mathbb{N}$, we know that

$$\begin{aligned} & g\big(z_1^{l(n)}(x_m), \ldots, z_C^{l(n)}(x_m)\big) \\ & + g\big(z_1^{r(n)}(x_{m+1}), \ldots, z_C^{r(n)}(x_{m+1})\big) \\ & \geq g\big(z_1^{l(n)}(x_{k_s^n}), \ldots, z_C^{l(n)}(x_{k_s^n})\big) \\ & + g\big(z_1^{r(n)}(x_{k_s^n+1}), \ldots, z_C^{r(n)}(x_{k_s^n+1})\big). \end{aligned} \tag{A.8}$$

The equal sign holds if and only if $m = k_s^n$.

From (A.6), (A.7), and (A.8) we can get,

$$\begin{aligned} & g\big(z_1^l(x_s), \ldots, z_C^l(x_s)\big) + g\big(z_1^r(x_s), \ldots, z_C^r(x_s)\big) \\ & \geq g\big(z_1^l(x_s^*), \ldots, z_C^l(x_s^*)\big) + g\big(z_1^r(x_s^*), \ldots, z_C^r(x_s^*)\big). \end{aligned}$$

Since x_s is the unique and optimal split that satisfies the split criterion (2.4), the following must hold:

$$\begin{aligned} & g\big(z_1^l(x_s), \ldots, z_C^l(x_s)\big) + g\big(z_1^r(x_s), \ldots, z_C^r(x_s)\big) \\ & \leq g\big(z_1^l(x_s^*), \ldots, z_C^l(x_s^*)\big) + g\big(z_1^r(x_s^*), \ldots, z_C^r(x_s^*)\big). \end{aligned}$$

So,

$$g\big(z_1^l(x_s),\dots,z_C^l(x_s)\big)+g\big(z_1^r(x_s),\dots,z_C^r(x_s)\big)$$
$$=g\big(z_1^l(x_s^*),\dots,z_C^l(x_s^*)\big)+g\big(z_1^r(x_s^*),\dots,z_C^r(x_s^*)\big),$$

and

$$x_s^n \xrightarrow{p} x_s^* = x_s, \quad as \quad n\to\infty.$$

For the rate of convergence, let us recall (A.4) and change ϵ to $\frac{\epsilon'}{n}$, where ϵ' is a constant in $(0, b-a)$.

$$\lim_{n\to\infty} P\left(|x_s - x_m| > \frac{\epsilon'}{2n}\right)$$
$$=\lim_{n\to\infty}\left(1+\frac{-\epsilon'/(b-a)}{n}\right)^n = e^{-\epsilon'/(b-a)}$$

Because $e^{-\epsilon'/(b-a)}$ is a constant, the convergence rate of x_m is $O(n^{-1})$.

Since $|x_s - x_s^n| \ge |x_s - x_m|$ always holds x_s^n converges to x_s slower or equal to x_m. However, by (A.8), we know that

$$g\big(z_1^{l(n)}(x_{k_s^n}),\dots,z_C^{l(n)}(x_{k_s^n})\big)$$
$$+\,g\big(z_1^{r(n)}(x_{k_s^n+1}),\dots,z_C^{r(n)}(x_{k_s^n+1})\big)$$

converges to

$$g\big(z_1^l(x_s),\dots,z_C^l(x_s)\big)+g\big(z_1^r(x_s),\dots,z_C^r(x_s)\big)$$

faster or equal than

$$g\big(z_1^{l(n)}(x_m),\dots,z_C^{l(n)}(x_m)\big)+g\big(z_1^{r(n)}(x_{m+1}),\dots,z_C^{r(n)}(x_{m+1})\big)$$

in all instances. This implies x_s^n converges to x_s faster or equal than x_m.

So, the convergence rate of x_s^n is exactly the same as x_m, and it is at the level $O(n^{-1})$ too. □

Corollary 1. *Let $d=\frac{3(b-a)}{2(n-1)}$. The interval $[x_s^n - d, x_s^n + d]$ approximates to the 95% confidence interval of the true optimal split x_s for large n.*

Proof. According to the proof of Lemma 1, for a constant $d \in (0, \frac{b-a}{2}]$, $P(|x_s - x_m| > d)$ represents the probability that all random samples in $\{x_1,\dots,x_{n-1}\}$ fall outside the interval $[x_s - d, x_s + d]$. Similar to equation (A.3), we have

$$P(|x_s - x_m| > d) = \left(1-\frac{2d}{b-a}\right)^{n-1}.$$

Thus, $P(|x_s - x_m| \le d) = 1-(1-\frac{2d}{b-a})^{n-1}$ is the probability that the true optimal split x_s lies within the interval $[x_m - d, x_m + d]$. In other words, $[x_m - d, x_m + d]$ is the confidence interval of the true optimal split x_s at a significance level of $1-\alpha = 1-(1-\frac{2d}{b-a})^{n-1}$.

Because neither x_s nor x_m are known. Let us recall the proof of Lemma 1, where we know that $x_s^n \xrightarrow{p} x_s$ and $x_m \xrightarrow{p} x_s$ as $n\to\infty$, and both x_m and x_s^n converge to x_s at the same rate. Therefore, we have $(x_s^n \pm d) \xrightarrow{p} (x_s \pm d)$, $(x_m \pm d) \xrightarrow{p} (x_s \pm d)$, and $|x_s - x_s^n| \xrightarrow{p} |x_s - x_m|$ as $n\to\infty$. Such that

$$\lim_{n\to\infty} P\big(|x_s - x_s^n| \le d\big)$$
$$=\lim_{n\to\infty} P(|x_s - x_m| \le d)$$
$$=\lim_{n\to\infty}\left[1-\left(1-\frac{2d}{b-a}\right)^{n-1}\right]$$
$$=\lim_{n\to\infty}\left[1-\left(1+\frac{-3}{n-1}\right)^{n-1}\right], \quad by \quad d=\frac{3(b-a)}{2(n-1)}$$
$$=1-e^{-3}\approx 0.950.$$

So, the interval

$$[x_s^n - d, x_s^n + d] \equiv \left[x_s^n - \frac{3(b-a)}{2(n-1)}, x_s^n + \frac{3(b-a)}{2(n-1)}\right] \quad (A.9)$$

is a good approximation of the 95% confidence interval of the true optimal split x_s for large n. □

To empirically validate the theoretical results about the confidence interval (A.9), we conducted an experiment using a step function, denoted as $f(x)$, with a single optimal split at $x = 1$, as illustrated in panel (a) of Figure A.11. The experiment proceeded as follows:

(1) At each sample size, we performed the following steps 1000 times:
   - Calculated the theoretical 95% confidence interval using (A.9).
   - Determined whether the true optimal split $x = 1$ falls within this interval.

   We then calculated the coverage rate, which is the proportion of times the true optimal split was covered by the confidence interval.

(2) We repeated step (1) a total of 100 times to obtain the empirical distribution of the coverage rate.

The results of this experiment are presented in panel (b) of Figure A.11. Notably, the empirical expectation of the coverage rate is approximately 95% when the sample size is 500 or larger. This empirical finding aligns well with the theoretical 95% confidence interval and supports the validity of confidence interval (A.9) and Lemma 1 in a practical context.

Theorem 1 (Continuous split convergence under the criterion of SSE). *Let X be a continuous random variable that takes values in $[a,b]$, where $a, b \in \mathbb{R}$. The teacher model $f(x)$ is integrable in $[a,b]$. We assume the existence of an unknown unique optimal split x_s in (a,b), which is defined as follows:*

$$x_s = \underset{x\in(a,b)}{\operatorname{argmin}}\left[\int_a^x \big(f(t)-\mu_l(x)\big)^2\,dt + \int_x^b \big(f(t)-\mu_r(x)\big)^2\,dt\right], \quad (A.10)$$

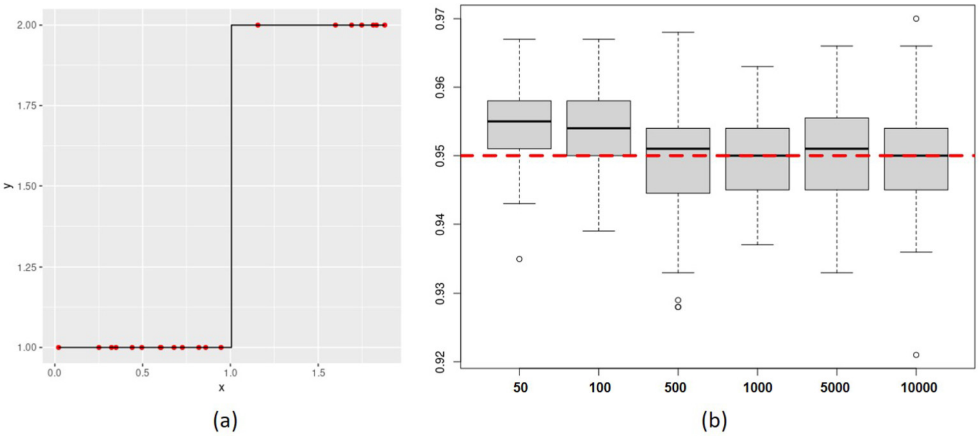


Figure A.11: (a) The true optimal split $x = 1$. (b) The coverage rate of 95% confidence intervals converges to the theoretical value (95%).

where,

$$\mu_l(x) = \frac{1}{x-a}\int_a^x f(u)\,du \quad , \quad \mu_r(x) = \frac{1}{b-x}\int_x^b f(u)\,du.$$

Consider $\{x_1, \ldots, x_{n-1}\}$ as $n-1$ points drawn uniformly at random from the interval (a, b), and arrange them in ascending order. Let $x_0 = a$ and $x_n = b$, and include them in $\{x_1, \ldots, x_{n-1}\}$ to form the set $\{x_0, x_1, \ldots, x_{n-1}, x_n\}$. Utilizing the teacher model, we can generate pseudo-data as $\{(x_0, f(x_0)), (x_1, f(x_1)), \ldots, (x_{n-1}, f(x_{n-1})), (x_n, f(x_n))\}$. Subsequently, we can fit a stump to the pseudo-data by employing the greedy split search algorithm and splitting criterion SSE in (2.1). Let x_s^n denote the split of the stump.

Then, 1) x_s^n converges to x_s in probability as $n \to \infty$. 2) The values of two fitted nodes converge to $\mu_l(x_s)$ and $\mu_r(x_s)$ in probability respectively as $n \to \infty$. 3) The rate of convergence is $O(n^{-1})$.

Proof. Construct

$$g\big(z_1^l(x)\big) = z_1^l(x), \;\; g\big(z_1^r(x)\big) = z_1^r(x),$$

$$z_1^l(x) = \int_a^x h^l(t)\,dt, \;\; z_1^r(x) = \int_x^b h^r(t)\,dt,$$

$$h^l(t) = \big(f(t) - \mu_l(x)\big)^2, \;\; h^r(t) = \big(f(t) - \mu_r(x)\big)^2,$$

$$z_1^{l(n)}(x_k) = \sum_{i=1}^{k} h^l(x_i) * \Delta_i \; , \; z_1^{r(n)}(x_{k+1}) = \sum_{j=k+1}^{n} h^r(x_j) * \Delta_j \; ,$$

where $\Delta_i = x_i - x_{i-1}$, $i = 1, \ldots, n$.

Under this construction, the optimal split x_s defined in (A.10) follows (2.4) in Definition 1. Obviously, $h^l(\cdot)$ and $h^r(\cdot)$ are integrable in $[a, b]$, because $f(\cdot)$ is integrable. So, by applying Lemma 1, x_s^n converges to x_s in probability and the rate of convergence is $O(n^{-1})$.

Since $f(\cdot)$ is integrable in $[a, b]$ and $x_{k_s^n} = x_s^n$, $x_{k_s^n+1} \xrightarrow{p} x_s^n$, $x_s^n \xrightarrow{p} x_s$, as $n \to \infty$, we can prove that

$$\lim_{n\to\infty} \frac{1}{x_{k_s^n} - a} \sum_{i=1}^{k_s^n} f(x_i)\Delta_i \xrightarrow{p} \frac{1}{x_s - a}\int_a^{x_s} f(u)\,du$$
$$= \mu_l(x_s),$$

$$\lim_{n\to\infty} \frac{1}{b - x_{k_s^n+1}} \sum_{j=k_s^n+1}^{n} f(x_j)\Delta_j \xrightarrow{p} \frac{1}{b - x_s}\int_{x_s}^{b} f(u)\,du$$
$$= \mu_r(x_s),$$

and the rate of convergence is $O(n^{-1})$. □

Theorem 2 (Continuous split convergence under the Tsallis entropy criterion)**.** *Let X be a continuous random variable that takes values in $[a, b]$, where $a, b \in \mathbb{R}$. $y = f(x)$ is the teacher model. $Y = f(X)$ is a discrete random variable taking values $y \in \{y_1, \ldots, y_C\}$, where $C \in \mathbb{N}^+$. Let $S_i = \{x | f(x) = y_i, x \in [a, b]\}$, $i = 1, \ldots, C$. The probability mass function of Y in $[a, b]$ is that:*

$$p(y_i) = \int_{S_i} \frac{1}{b-a}\,dx, \quad i = 1, \ldots, C.$$

And, the probability mass function of Y in $[a, x]$ is that:

$$p_{[a,x]}(y_i) = \int_{S_i \cap [a,x]} \frac{1}{x-a}\,dt.$$

Then, a Tsallis entropy can be calculated in $[a, x]$,

$$S_q\big([a, x]\big) = \frac{1}{1-q}\left(\sum_{i=1}^{C} p_{[a,x]}(y_i)^q - 1\right), \quad q \in \mathbb{R}.$$

Assume the existence of an unknown unique optimal split x_s in (a,b), which is defined as follows:

$$x_s = \operatorname*{argmin}_{x\in(a,b)} \big[S_q\big([a,x]\big) + S_q\big([x,b]\big)\big]. \tag{A.11}$$

Consider $\{x_1,\dots,x_{n-1}\}$ as $n-1$ points drawn uniformly at random from the interval (a,b), and arrange them in ascending order. Let $x_0 = a$ and $x_n = b$, and include them in $\{x_1,\dots,x_{n-1}\}$ to form the set $\{x_0,x_1,\dots,x_{n-1},x_n\}$. Utilizing the teacher model, we can generate pseudo-data as $\{(x_0,f(x_0)),(x_1,f(x_1)),\dots,(x_{n-1},f(x_{n-1})),(x_n,f(x_n))\}$. Subsequently, we can fit a stump to the pseudo-data by employing the greedy split search algorithm and Tsallis entropy splitting criterion in (2.2) and (2.3). x_s^n denotes the split of the stump.

Then, 1) x_s^n converges to x_s in probability as $n\to\infty$. 2) The rate of convergence is $O(n^{-1})$.

Proof. Construct

$$g\big(z_1^l(x),\dots,z_C^l(x)\big) = \frac{1}{1-q}\left(\sum_{c=1}^{C} z_c^l(x)^q - 1\right),$$

$$g\big(z_1^r(x),\dots,z_C^r(x)\big) = \frac{1}{1-q}\left(\sum_{c=1}^{C} z_c^r(x)^q - 1\right),$$

$$\begin{aligned} z_c^l(x) &= \int_a^x h_c^l(t)\,dt \\ &= \int_a^x \frac{1}{x-a} * I_{y_c}\big(f(t)\big)\,dt \\ &= \int_{S_i\cap[a,x]} \frac{1}{x-a}\,dt = p_{[a,x]}(y_c), \end{aligned}$$

$$\begin{aligned} z_c^r(x) &= \int_x^b h_c^r(t)\,dt \\ &= \int_x^b \frac{1}{b-x} * I_{y_c}\big(f(t)\big)\,dt \\ &= \int_{S_i\cap[x,b]} \frac{1}{b-x}\,dt = p_{[x,b]}(y_c), \end{aligned}$$

$$h_c^l(t) = \frac{1}{x-a} * I_{y_c}\big(f(t)\big),\quad h_c^r(t) = \frac{1}{b-x} * I_{y_c}\big(f(t)\big),$$

$$z_c^{l(n)}(x_k) = \sum_{i=1}^{k} h_c^l(x_i) * \Delta_i = \frac{1}{x-a}\sum_{i=1}^{k} \Delta_i * I_{y_c}\big(f(x_i)\big),$$

$$z_c^{r(n)}(x_{k+1}) = \sum_{j=k+1}^{n} h_c^r(x_j)*\Delta_j = \frac{1}{b-x}\sum_{j=k+1}^{n} \Delta_j*I_{y_c}\big(f(x_j)\big),$$

where $c = 1,\dots,C$, $\Delta_i = x_i - x_{i-1}$, $i = 1,\dots,n$ and $I_{y_c}(f(x))$ is an indicator function that is equal to 1 at $f(x) = y_c$ and 0 elsewhere.

Under this construction, the optimal split x_s defined in (A.11) follows (2.4) in Definition 1. Obviously, $h_c^l(\cdot)$ and $h_c^r(\cdot)$ are integrable in $[a,b]$. So, by applying Lemma 1, x_s^n converges to x_s in probability and the rate of convergence is $O(n^{-1})$. □

Theorem 3 (Categorical split convergence under MSE criterion)**.** *X is a discrete random variable taking values $x\in\{1,\dots,C_x\}$, where $C_x\in\mathbb{N}^+$. Y is a continuous random variable taking values $y\in[c,d]$, where $c,d\in\mathbb{R}$. Y has a finite mean μ. The conditional distribution of $Y|X=k$ is defined through the teacher model $y=f(k)$. The expectation of $Y|X=k$ is given by:*

$$\begin{aligned} &E(Y|X=k) = \mu_k,\ \mu = \frac{1}{C_x}\sum_{k=1}^{C_x}\mu_k \\ &k = 1,\dots,C_x. \end{aligned} \tag{A.12}$$

Let us randomly sample n instances of X, denoted as $\{x_1,\dots,x_n\}$, from $\{1,\dots,C_x\}$. Corresponding samples of Y, denoted as $\{y_1,\dots,y_n\}$, are generated through the conditional distribution of $Y|X=k$. The uniform sampling assumption indicates $\lim_{n\to\infty}\frac{n_k}{n} = \frac{1}{C_x}$, where $n_k = \sum_{i=1}^{n} I(x_i = k)$, $k = 1,\dots,C_x$.

Assume the existence of an unknown unique optimal split x_s in $\{1,\dots,C_x\}$, which is defined as follows:

$$\begin{aligned} x_s = \operatorname*{argmin}_{k\in\{1,\dots,C_x\}} \lim_{n\to\infty}\frac{1}{n}\Bigg[&\sum_{i=1}^{n_k}(y_{ki}-\mu_k)^2 \\ &+ \sum_{l\neq k}\left(\sum_{j=1}^{n_l}(y_{lj}-\mu_{-k})^2\right)\Bigg], \end{aligned} \tag{A.13}$$

where $\mu_{-k} = E(Y|X\neq k)$.

A stump can be fitted on the pseudo-data $\{(x_1,y_1),\dots,(x_n,y_n)\}$ by using the greedy search algorithm and splitting MSE criterion in (2.1). Let x_s^n denote the split of the stump.

Then, 1) x_s^n converges to x_s in probability as $n\to\infty$. 2) The rate of convergence is $O(n^{-1})$.

Proof. Let us construct

$$\begin{aligned} &\lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n_k}(y_{ki}-\mu_k)^2 = z^l(k), \\ &\lim_{n\to\infty}\sum_{l\neq k}\left(\sum_{j=1}^{n_l}(y_{lj}-\mu_{-k})^2\right) = z^r(k), \\ &g(z) = z. \end{aligned}$$

Obviously, (A.13) follows (2.4), so x_s follows Definition 1.

By the splitting criterion (2.1), the optimal split of the stump can be found that

$$x_s^n = \underset{k\in\{1,\ldots,C_x\}}{\operatorname{argmin}} \frac{1}{n}\left[\sum_{i=1}^{n_k}(y_{ki}-\bar{y}_k)^2 + \sum_{l\neq k}\left(\sum_{j=1}^{n_l}(y_{lj}-\bar{y}_{-k})^2\right)\right],$$

where

$$\bar{y}_k = \frac{1}{n_k}\sum_{i=1}^{n_k} y_{ki} \quad \textit{and} \quad \bar{y}_{-k} = \frac{1}{\sum_{l\neq k} n_l}\sum_{l\neq k}\sum_{j=1}^{n_l} y_{lj}.$$

Since we know that $n_k = \frac{1}{C_x}n, \quad k = 1,\ldots,C_x$. By the weak law of large numbers, we can prove that

$$\bar{y}_k \xrightarrow{p} \mu_k, \quad \bar{y}_{-k} \xrightarrow{p} \mu_{-k} \quad \textit{as} \quad n\to\infty.$$

So, with probability one,

$$\begin{aligned}\lim_{n\to\infty} x_s^n = & \underset{k\in\{1,\ldots,C_x\}}{\operatorname{argmin}} \lim_{n\to\infty}\frac{1}{n}\left[\sum_{i=1}^{n_k}(y_{ki}-\bar{y}_k)^2\right. \\ & \left. + \sum_{l\neq k}\left(\sum_{j=1}^{n_l}(y_{lj}-\bar{y}_{-k})^2\right)\right] \\ \xrightarrow{p} & \underset{k\in\{1,\ldots,C_x\}}{\operatorname{argmin}} \lim_{n\to\infty}\frac{1}{n}\left[\sum_{i=1}^{n_k}(y_{ki}-\mu_k)^2\right. \\ & \left. + \sum_{l\neq k}\left(\sum_{j=1}^{n_l}(y_{lj}-\mu_{-k})^2\right)\right] = x_s.\end{aligned}$$

x_s^n converges to x_s in probability and the rate of convergence is $O(n^{-1})$. □

Theorem 4 (Categorical split convergence under Tsallis entropy criterion)**.** *X is a discrete random variable taking values $x \in \{1,\ldots,C_x\}$, where $C_x \in \mathbb{N}^+$. Y is a discrete random variable taking values $y \in \{1,\ldots,C_y\}$, where $C_y \in \mathbb{N}^+$. A joint distribution (X,Y) can be defined through the teacher model $y = f(x)$. Its probability mass function can be denoted as $p(x=i,y=j) = p_{ij}$, where $i = 1,\ldots,C_x, \quad j = 1,\ldots,C_y$.*

Let us randomly sample n instances of X, denoted as $\{x_1,\ldots,x_n\}$, from $\{1,\ldots,C_x\}$. Corresponding samples of Y, denoted as $\{y_1,\ldots,y_n\}$, are generated through the joint distribution (X,Y). The uniform sampling assumption indicates $\lim_{n\to\infty}\frac{n_k}{n} = \frac{1}{C_x}$, where $n_k = \sum_{i=1}^n I(x_i = k)$, $k = 1,\ldots,C_x$.

Assume the existence of an unknown unique optimal split x_s in $\{1,\ldots,C_x\}$, which is defined as follows:

$$x_s = \underset{k\in\{1,\ldots,C_x\}}{\operatorname{argmin}} \big[S_q(k) + S_q(-k)\big], \tag{A.14}$$

where, $S_q(\cdot)$ is the Tsallis entropy,

$$S_q(k) = \frac{1}{1-q}\left(\sum_{j=1}^{C_y}(p_{kj})^q - 1\right),$$

$$S_q(-k) = \frac{1}{1-q}\left(\sum_{j=1}^{C_y}\left(\sum_{i\neq k}p_{ij}\right)^q - 1\right),$$

$$k, i \in \{1,\ldots,C_x\}, \quad q\in\mathbb{R}.$$

A stump can be fitted with the pseudo-data $\{(x_1,y_1),\ldots,(x_n,y_n)\}$ by using the greedy search algorithm and the Tsallis entropy splitting criterion in (2.2) and (2.3). Let x_s^n denote the split of the stump.

Then, 1) x_s^n converges to x_s in probability as $n\to\infty$. 2) The rate of convergence is $O(n^{-1})$.

Proof. Let us construct

$$p_{kj} = z^l(k), \quad \sum_{i\neq k} p_{ij} = z^r(k) \quad \textit{and} \quad g\big(z(k)\big) = S_q(k).$$

Obviously, (A.14) follows (2.4), so x_s follows Definition 1.

By the splitting criterion (2.2), the optimal split of the stump can be found that

$$x_s^n = \underset{k\in\{1,\ldots,C_x\}}{\operatorname{argmin}} \big[S_q^n(k) + S_q^n(-k)\big],$$

where

$$S_q^n(k) = \frac{1}{1-q}\left(\sum_{j=1}^{C_y}\left(\frac{1}{n_k}\sum_{l=1}^{n_k} I(y_l = j)\right)^q - 1\right),$$

$$S_q^n(-k) = \frac{1}{1-q}\left(\sum_{j=1}^{C_y}\left(\sum_{i\neq k}\left(\frac{1}{n_i}\sum_{m=1}^{n_i} I(y_m = j)\right)\right)^q - 1\right),$$

$$i \in \{1,\ldots,C_x\}, \quad q\in\mathbb{R}.$$

Since we know that $n_k = \frac{1}{C_x}n, \quad k = 1,\ldots,C_x$. By Borel's law of large numbers, with probability one,

$$\lim_{n\to\infty}\frac{1}{n_k}\sum_{m=1}^{n_k} I(y_m = j) = p_{kj}, \quad k = 1,\ldots,C_x, \quad j = 1,\ldots,C_y.$$

So, with probability one,

$$\begin{aligned}\lim_{n\to\infty} x_s^n &= \underset{k\in\{1,\ldots,C_x\}}{\operatorname{argmin}} \lim_{n\to\infty}\big[S_q^n(k) + S_q^n(-k)\big] \\ &= \underset{k\in\{1,\ldots,C_x\}}{\operatorname{argmin}} \big[S_q(k) + S_q(-k)\big] = x_s.\end{aligned}$$

x_s^n converges to x_s in probability and the rate of convergence is $O(n^{-1})$. □

APPENDIX B. SUPPLEMENTARY MATERIALS

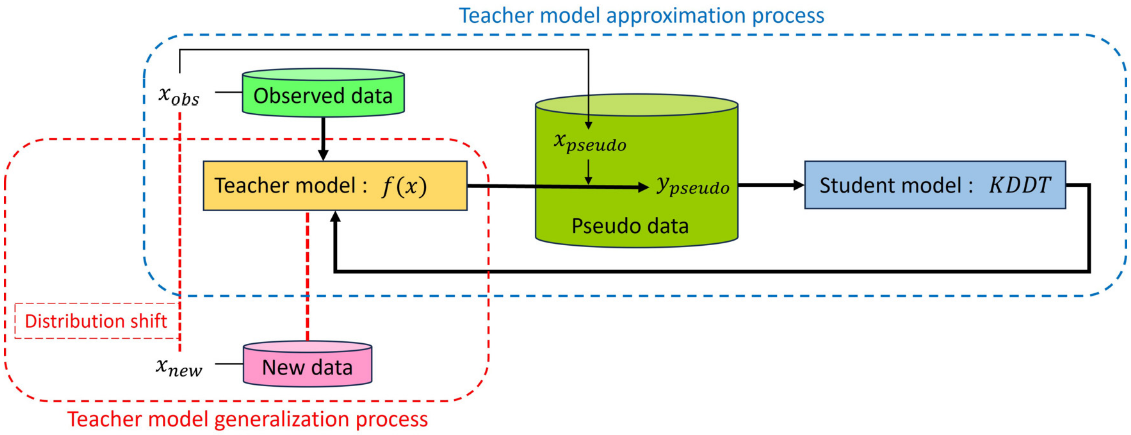


Figure B.12: The teacher model approximation and generalization process. The model generalization process may encounter the challenge of distribution shift, whereas the approximation process does not.

Table B.2. Variables and short descriptions.

Variable	Short descriptions
medv	median value of owner-occupied homes in USD 1000's.
crim	per capita crime rate by town.
zn	proportion of residential land zoned for lots over 25,000 sq.ft.
indus	proportion of non-retail business acres per town.
chas	Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).
nox	nitric oxides concentration (parts per 10 million).
rm	average number of rooms per dwelling.
age	proportion of owner-occupied units built prior to 1940.
dis	weighted distances to five Boston employment centers.
rad	index of accessibility to radial highways.
tax	full-value property-tax rate per USD 10,000.
ptratio	pupil-teacher ratio by town.
b	$1000(B - 0.63)^2$ where B is the proportion of blacks by town.
lstat	percentage of lower status of the population.

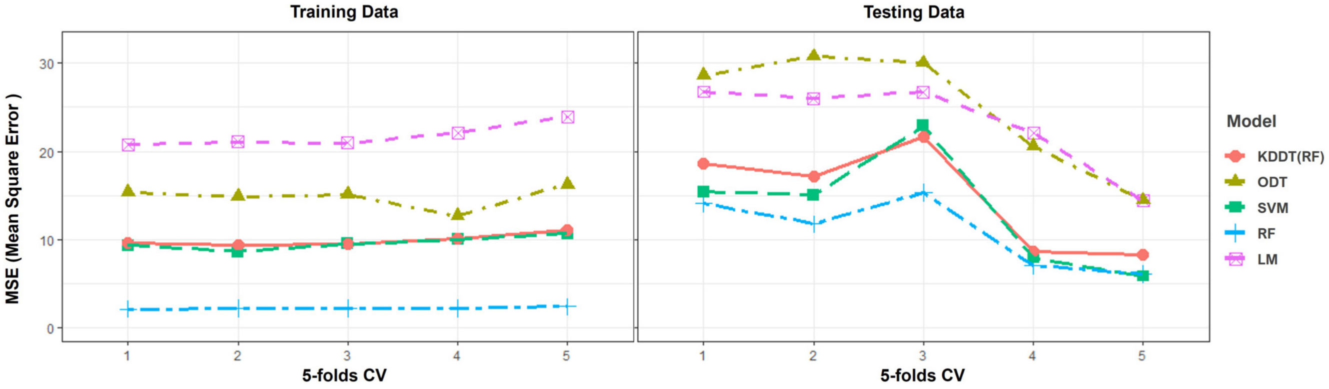
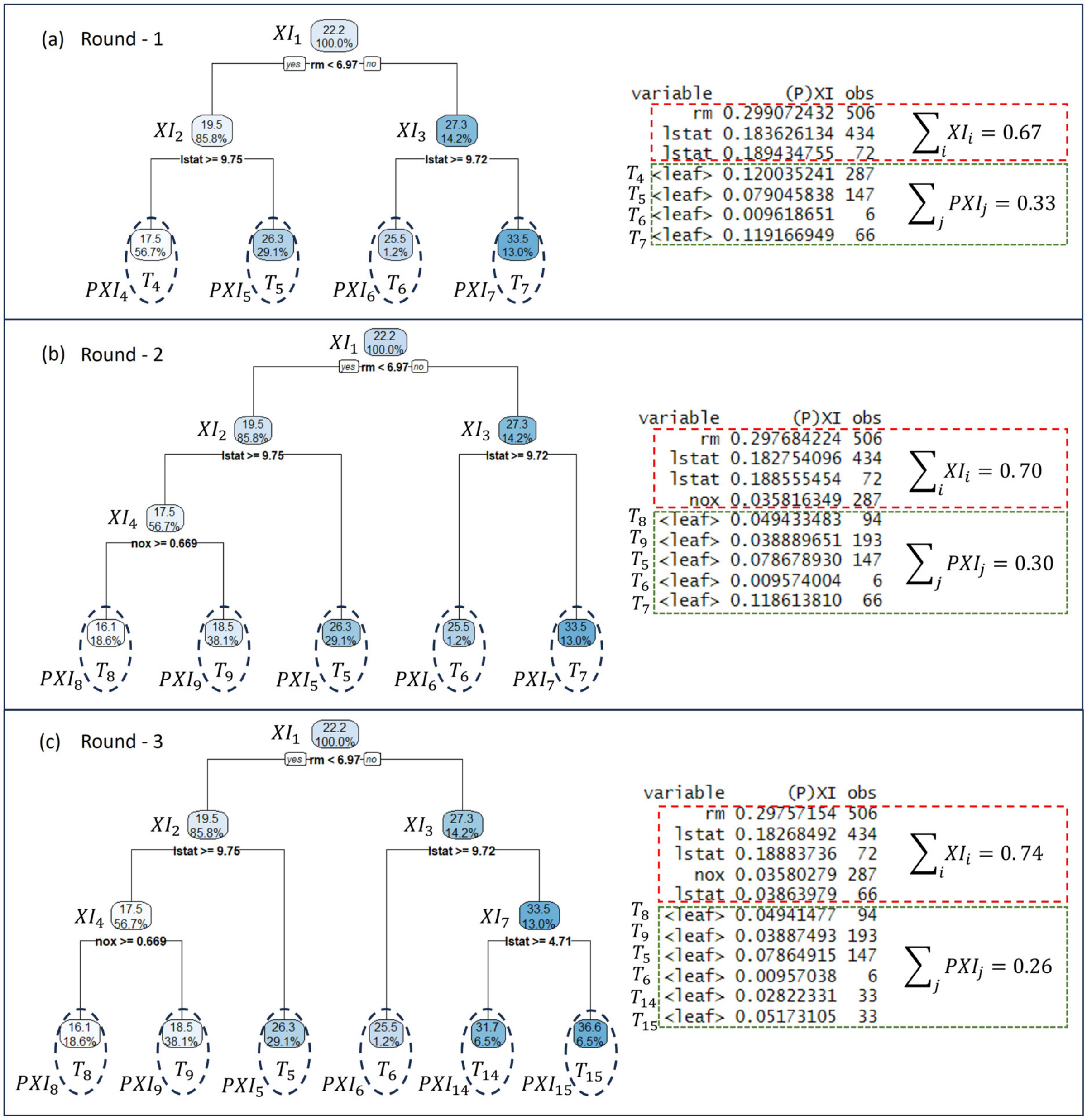


Figure B.13: Comparison of prediction accuracy among ODT, LM, SVM, RF, and KDDT(RF) on the training dataset (left) and testing dataset (right). Note that RF is the teacher model and KDDT(RF) is the student model. We included SVM to demonstrate that any black-box ML model can serve as the teacher model, and we opted for the one with higher prediction accuracy.



```
variable         (P)XI obs
      rm 0.299072432 506
   lstat 0.183626134 434
   lstat 0.189434755  72
  <leaf> 0.120035241 287
  <leaf> 0.079045838 147
  <leaf> 0.009618651   6
  <leaf> 0.119166949  66
```

```
variable         (P)XI obs
      rm 0.297684224 506
   lstat 0.182754096 434
   lstat 0.188555454  72
     nox 0.035816349 287
  <leaf> 0.049433483  94
  <leaf> 0.038889651 193
  <leaf> 0.078678930 147
  <leaf> 0.009574004   6
  <leaf> 0.118613810  66
```

```
variable        (P)XI obs
      rm 0.29757154 506
   lstat 0.18268492 434
   lstat 0.18883736  72
     nox 0.03580279 287
   lstat 0.03863979  66
  <leaf> 0.04941477  94
  <leaf> 0.03887493 193
  <leaf> 0.07864915 147
  <leaf> 0.00957038   6
  <leaf> 0.02822331  33
  <leaf> 0.05173105  33
```

Figure B.14: An example to find the desired hybrid KDDT under the criterion of $\sum_i XI_i > 70\%$ (or $\sum_j PXI_j < 30\%$). The final hybrid KDDT in panel (c) is same with the one in the panel (a) of Figure 8.

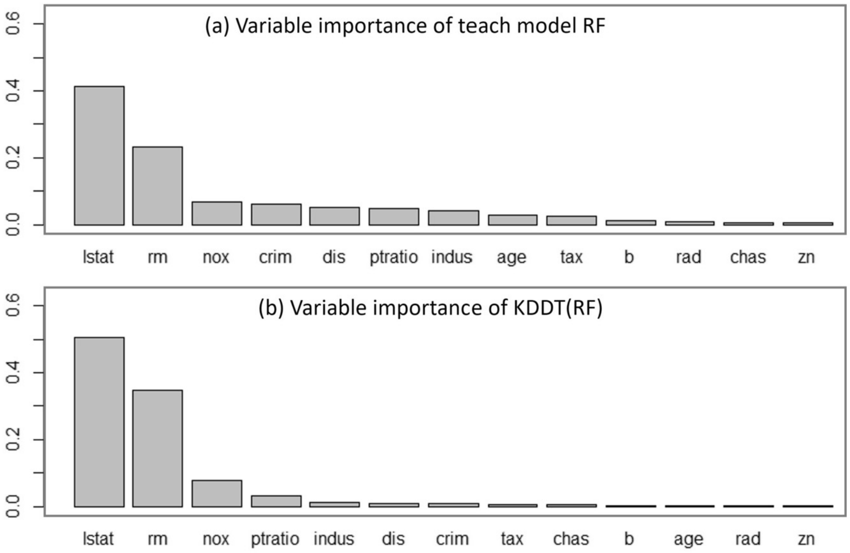


Figure B.15: Comparison of variable importance between the teacher Random Forest and the student KDDT(RF).

Table B.3. Variables and short descriptions.

Variable	Short descriptions
age	Age (per chart) (years).
sex	Sex. 0 = Male. 1 = Female.
cpk	Peak cardiac enzyme (iu).
sho	Cardiogenic shock complications. 1 = Yes. 0 = No.
chf	Left heart failure complications. 1 = Yes. 0 = No.
miord	MI Order. 1 = Recurrent. 0 = First.
mitype	MI Type. 1 = Q-wave. 2 = Not Q-wave. 3 = Indeterminate.
lenstay	Days in hospital.
lenfol	Total length of follow-up from hospital admission (days).
fstat	Status as of last follow-up. 1 = Dead. 0 = Alive.

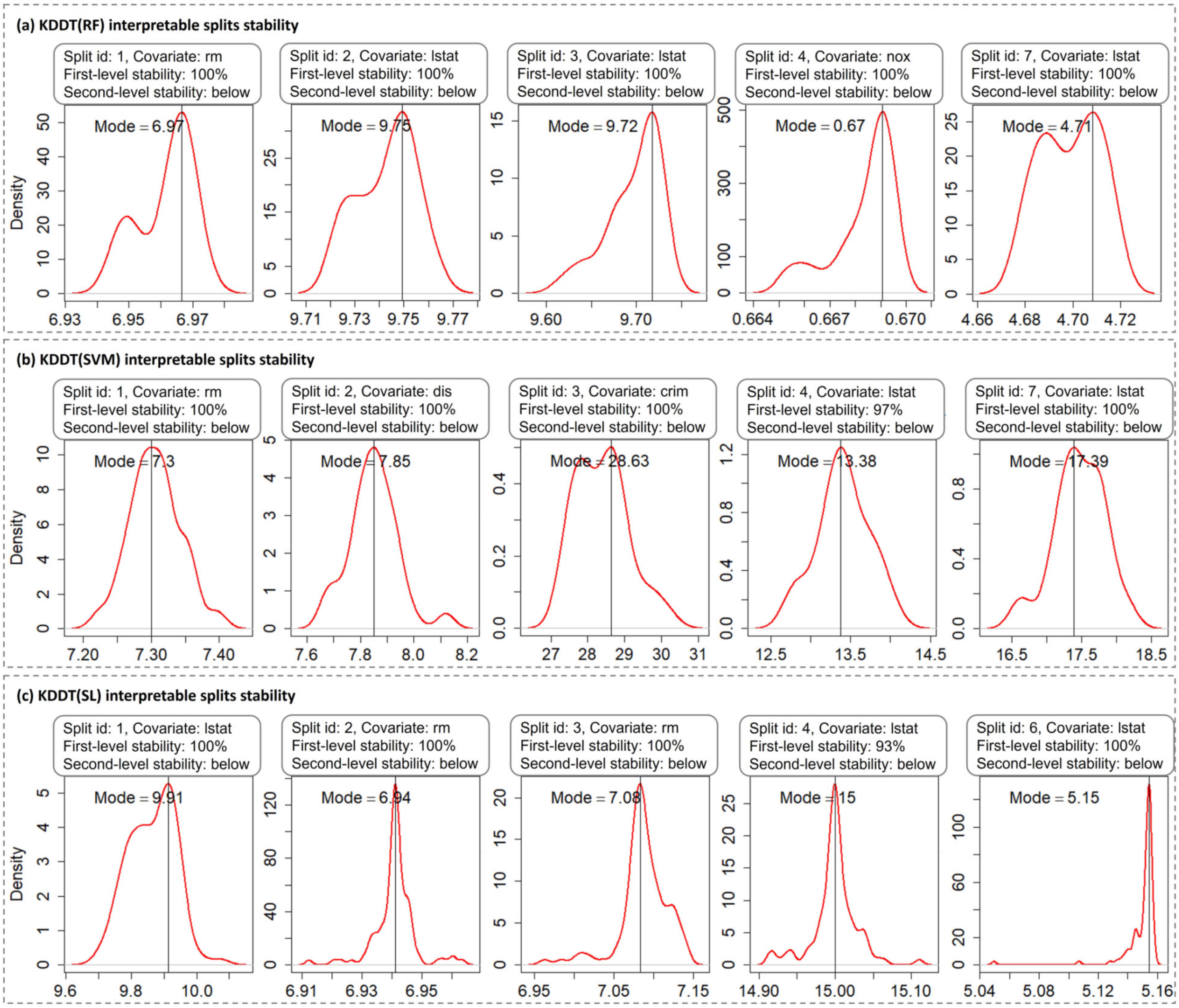


Figure B.16: Two-level stability of the interpretable splits in KDDT(RF), KDDT(SVM) and KDDT(SL).

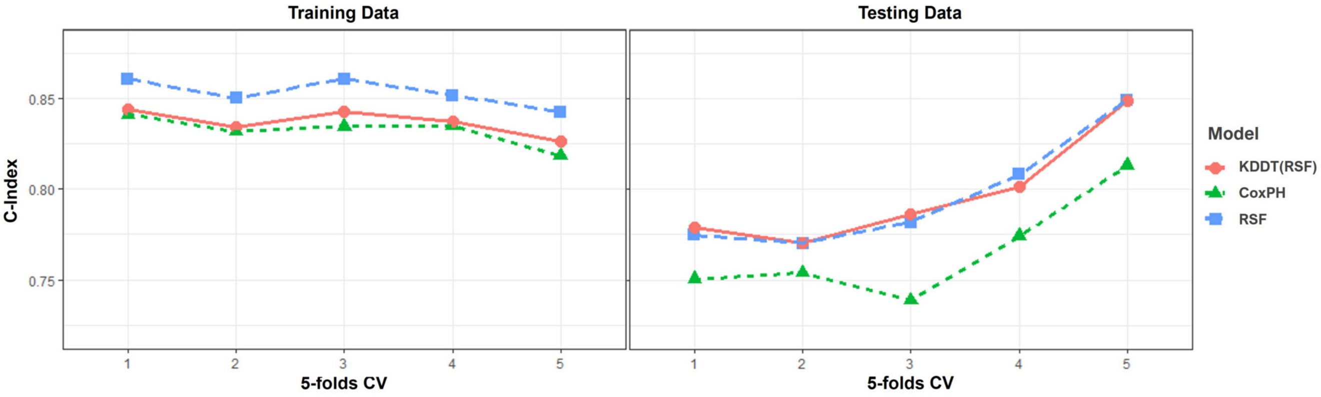


Figure B.17: Comparison of prediction accuracy among CoxPH, RSF, and KDDT(RSF) on the training dataset (left) and testing dataset (right). Note that RSF is the teacher model and KDDT(RF) is the student model. A higher C-index indicates superior performance in prediction. Notably, KDDT(RSF) surpasses its teacher model RSF on the first and third folds of the testing data. This is because KDDT(RSF), being an approximation of RSF, might relieve overfitting on the testing data.

ACKNOWLEDGEMENTS

The authors appreciate the helpful discussions with Dr. Wei-Ying Lou and the editorial assistance from Mrs. Jessica Swann.

FUNDING

J. Jack Lee's research was supported in part by the grants P30CA016672, P50CA221703, U24CA224285, and U24CA274274 from the National Cancer Institute.

Accepted 18 February 2028

REFERENCES

[1] Allen-Zhu, Z. and Li, Y. (2021). *Towards Understanding Ensemble, Knowledge Distillation and Self-Distillation in Deep Learning*. arXiv:2012.09816.

[2] Ba, J. and Caruana, R. (2014). Do Deep Nets Really Need to be Deep? In *Advances in Neural Information Processing Systems* (Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence and K. Q. Weinberger, eds.) **27**. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2014/file/ea8fcd92d59581717e06eb187f10666d-Paper.pdf.

[3] Breiman, L., Friedman, J., Stone, C. J. and Olshen, R. A. (1984) *Classification and Regression Trees*. Chapman and Hall/CRC. MR0726392

[4] Coppens, Y., Efthymiadis, K., Lenaerts, T. and Nowé, A. (2019). Distilling Deep Reinforcement Learning Policies in Soft Decision Trees. In *IJCAI 2019*.

[5] Ding, Z., Hernandez-Leal, P., Ding, G. W., Li, C. and Huang, R. (2021). *CDT: Cascading Decision Trees for Explainable Reinforcement Learning*. arXiv:2011.07553.

[6] Frosst, N. and Hinton, G. (2017). *Distilling a Neural Network Into a Soft Decision Tree*. arXiv:1711.09784.

[7] Gou, J., Yu, B., Maybank, S. J. and Tao, D. (2021). Knowledge Distillation: A Survey. *International Journal of Computer Vision* **129**(6) 1789–1819. https://doi.org/10.1007/s11263-021-01453-z.

[8] Hinton, G., Vinyals, O. and Dean, J. (2015). *Distilling the knowledge in a neural network*. arXiv:1503.02531.

[9] Hu, C., Li, X., Liu, D., Wu, H., Chen, X., Wang, J. and Liu, X. (2023). *Teacher-Student Architecture for Knowledge Distillation: A Survey*. arXiv:abs/2308.04268.

[10] Hyafil, L. and Rivest, R. L. (1976). Constructing optimal binary decision trees is NP-complete. *Information Processing Letters* **5**(1) 15–17. https://doi.org/10.1016/0020-0190(76)90095-8. MR0413598

[11] Johansson, U., Sönströd, C. and Löfström, T. (2011). One tree to explain them all. In *2011 IEEE Congress of Evolutionary Computation (CEC)* 1444–1451. https://doi.org/10.1109/CEC.2011.5949785.

[12] Kaleem, S. M., Rouf, T., Habib, G., Saleem, T. J. and Lall, B. (2024). *A Comprehensive Review of Knowledge Distillation in Computer Vision*. arXiv:abs/2404.00936.

[13] Laan, M. J. V. D., Polley, E. C. and Hubbard, A. E. (2007). Super Learner. *Statistical Applications in Genetics and Molecular Biology* **6**(1) 25. https://doi.org/10.2202/1544-6115.1309. MR2349918

[14] Li, J., Li, Y., Xiang, X., Xia, S.-T., Dong, S. and Cai, Y. (2020). TNT: An Interpretable Tree-Network-Tree Learning Framework using Knowledge Distillation. *Entropy* **22**(11). https://doi.org/10.3390/e22111203. MR4222006

[15] Lundberg, S. M. and Lee, S. -I. (2017). A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan and R. Garnett, eds.) **30**. Curran Associates, Inc.

[16] Quinlan, J. R. (1993) *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA. http://portal.acm.org/citation.cfm?id=152181.

[17] Ribeiro, M. T., Singh, S. and Guestrin, C. (2016). "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD'16* 1135–1144. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/2939672.2939778.

[18] Rokach, L. and Maimon, O. (2014) *Data Mining With Decision Trees: Theory and Applications*, 2nd ed. World Scientific Publishing Co., Inc., USA.

[19] Shen, Y., Xu, X.and Cao, J. (2020). Reconciling predictive and interpretable performance in repeat buyer prediction via model distillation and heterogeneous classifiers fusion. *Neural Comput. Appl.*

[20] Shi, Y., Hwang, M. -Y., Lei, X. and Sheng, H. (2019). *Knowledge Distillation For Recurrent Neural Network Language Modeling With Trust Regularization*. arXiv:1904.04163.

[21] Song, J., Zhang, H., Wang, X., Xue, M., Chen, Y., Sun, L., Tao, D. and Song, M. (2021). Tree-Like Decision Distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* 13488–13497.

[22] Stanton, S., Izmailov, P., Kirichenko, P., Alemi, A. A. and Wilson, A. G. (2021). *Does Knowledge Distillation Really Work?* arXiv:2106.05945.

[23] Urban, G., Geras, K. J., Kahou, S. E., Aslan, O., Wang, S., Caruana, R., Mohamed, A., Philipose, M. and Richardson, M. (2017). *Do Deep Convolutional Nets Really Need to be Deep and Convolutional?* arXiv:1603.05691.

[24] Wang, Y. and Xia, S.-T. (2017). Unifying attribute splitting criteria of decision trees by Tsallis entropy. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* 2507–2511. https://doi.org/10.1109/ICASSP.2017.7952608.

[25] Yang, C., Zhu, Y., Lu, W., Wang, Y., Chen, Q., Gao, C., Yan, B. and Chen, Y. (2024). Survey on Knowledge Distillation for Large Language Models: Methods, Evaluation, and Application. *ACM Trans. Intell. Syst. Technol.* Just Accepted. https://doi.org/10.1145/3699518.

[26] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A. and Torralba, A. (2016). Learning Deep Features for Discriminative Localization. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 2921–2929. https://doi.org/10.1109/CVPR.2016.319.

[27] Zhou, Y., Zhou, Z. and Hooker, G. (2018). *Approximation Trees: Statistical Stability in Model Distillation*. arXiv:1808.07573.

Xuetao Lu. Department of Biostatistics, The University of Texas MD Anderson Cancer Center, USA. E-mail address: xlu7@mdanderson.org

J. Jack Lee. Department of Biostatistics, The University of Texas MD Anderson Cancer Center, USA. E-mail address: jjlee@mdanderson.org