Hybrid Decision Making: When Interpretable Models Collaborate With Black-Box Models

Tong Wang
University of Iowa
Iowa City, IA 52245
tong-wang@uiowa.edu

ABSTRACT

Interpretable machine learning models have received increasing interest in recent years, especially in domains where humans are involved in the decision-making process. However, the possible loss of the task performance for gaining interpretability is often inevitable. This performance downgrade puts practitioners in a dilemma of choosing between a top-performing black-box model with no explanations and an interpretable model with unsatisfying task performance.

In this work, we propose a novel framework for building a Hybrid Decision Model that integrates an interpretable model with ***any*** black-box model to introduce explanations in the decision making process while preserving or possibly improving the predictive accuracy. We propose a novel metric, explainability, to measure the percentage of data that are sent to the interpretable model for decision. We also design a principled objective function that considers predictive accuracy, model interpretability, and data explainability. Under this framework, we develop Collaborative Black-box and RUle Set Hybrid (CoBRUSH) model that combines logic rules and *any* black-box model into a joint decision model. An input instance is first sent to the rules for decision. If a rule is satisfied, a decision will be directly generated. Otherwise, the black-box model is activated to decide on the instance. To train a hybrid model, we design an efficient search algorithm that exploits theoretically grounded strategies to reduce computation. Experiments show that CoBRUSH models are able to achieve same or better accuracy than their black-box collaborator working alone while gaining explainability. They also have smaller model complexity than interpretable baselines.

KEYWORDS

interpretable machine learning, black-box model, hybrid decision model, association rules

ACM Reference format:
Tong Wang. 2018. Hybrid Decision Making: When Interpretable Models Collaborate With Black-Box Models. In *Proceedings of , , ,* 9 pages.
DOI: 10.475/123_4

DOI: 10.475/123_4

1 INTRODUCTION

The deployment of machine learning in real-world applications has led to a surge of interest in systems optimized not only for task performance but also model interpretability. The desire for interpretability is especially motivated by domains where human experts are involved in the decision-making process. In many heavily regulated industries such as judiciary and healthcare, understanding the decision-making process of an analytical model is not just a preference but often a matter of legal and ethic compliance [6]. While interpretable machine learning is blossoming, black-box machine learning, in parallel, has also undertaken unprecedented development in recent years, especially with the advancement in deep learning [19]. Unlike interpretable models, black-box models generate decisions through a complicated process that can hardly be comprehended by humans.

Despite their rapid advance, neither of the two types of models alone suffice to serve the need of many real-world applications. Interpretable models have to use only a small number of cognitive chunks and simple (shallow) structures, in order to be understandable to humans. Due to this constraint on model dimensions, predictive performance may be sacrificed to gain interpretability under certain circumstances, especially when dealing with high-dimensional data. Black-box models, on the other hand, do not have these constraints but are unable to provide explanations for their results. When solving a real-world problem where both task performance and interpretability is a necessity, neither of a pure interpretable nor a pure black-box model will be satisfying.

One recent solution to bridging this gap is to provide interpretable explanations or approximations for a black-box model, either locally [27] or globally [2, 18]. The explanations serve as post hoc analysis which provides some insights into the decision-making process. But the interpretable model does not interfere or assist the black-box in the decision making and the output depends solely on the black-box. Therefore, the interpretable model does not help improve the performance.

In this paper, we introduce a new solution from a different perspective. We design a framework that integrates an interpretable model and **any** black-box model into a joint decision-making process, leveraging the strength of both. The goal is to preserve or possibly improve the predictive performance of the black-box model while providing interpretable decisions on (at least a subset of) the data. We call such a model a *Hybrid Decision Model.*

This form of the model is motivated by how humans make decisions in many real world situations. For example, when a doctor diagnoses a patient, if the patient demonstrates easy "textbook" symptoms, a diagnosis can be made right away via simple reasoning and symptom matching. However, if the patient is a much harder case with symptoms that do not perfectly match documented descriptions of any disease, then a more experienced doctor or a consultation of several experts will be summoned, representing a more complicated decision model being activated. A hybrid decision model works in a similar way. It assigns an interpretable model to a subset of data, providing explanations, and activates the black-box model on the rest of the data where the interpretable model does not suffice. We call the subset of data sent to the interpretable model "explained", and define a new concept, *explainability*, that refers to the percentage of data in this subset.

The trade-off in accuracy to gain interpretability is often criticized in interpretable machine learning. However, the proposed hybrid decision models do not have this concern. On the contrary, integrating an interpretable model with a black-box model can often boost the predictive performance compared to using the black-box alone. This counter-intuitive phenomenon can be explained by an observation that while a black-box model can be more accurate than an interpretable model globally, there may exist a subspace of data for which an interpretable model suffices to make *as or more accurate* decisions compared to the black-box model. (This is proven nicely by our experiments in this paper.) This subset of data could contain instances that are "easy" to decide, for example, those that are far from the decision boundary, so a simple, interpretable model makes no difference than a more complicated model. This subset may also happen to be better captured by the interpretable model which obtains higher accuracy locally. Therefore, if combined strategically, a hybrid decision model is capable of obtaining better predictive accuracy than using any of the two models alone while gaining explainability for the decision-making process.

In this paper, we choose rules as the interpretable component for their symbolic presentation and simple logics. Following the proposed framework, we develop a classifier, **Collaborative Black-box and Rule Set Hybrid (CoBRUSH)** model which consists of a black-box model and a small set of short rules. We define an objective that considers three fundamental aspects of a hybrid decision model, predictive accuracy, model interpretability, and data explainability. We design an efficient training algorithm that exploits theoretically grounded strategies for fast computation.

An important strength of the proposed framework is that the hybrid decision model only needs the black-box's predictions on the training data. It does not need to know any implementation detail or even what type of model it is, largely concealing information of the black-box model from the interpretable collaborator. This property is critical in building collaborations among different systems. When some systems are working with other systems, they may not wish to disclose all information, which may contain confidential features or techniques utilized to build their model.

Our work in this paper unifies interpretable models and black-box models into the same learning framework. Both are extremes of hybrid decision models. Interpretable models are hybrid models with explainability = 1, and black-box models are hybrid models with explainability = 0. A hybrid framework connects the two discrete points into a continuous regime defined by accuracy and explainability, providing more flexible choices of models for different applications.

2 RELATED WORK

Our work is broadly related to new methods for interpretable machine learning. There have been two lines of research in interpretable machine learning in recent years. The first is developing decision models that are interpretable. Previous work in this category include rule-based models (Rule Sets, e.g., [17, 22, 28, 34], Rule Lists [3, 36]), scoring systems [14, 32, 37], case based models [1, 4, 13], and etc. The second line of research is on developing diagnostic or probing models that interpret black-boxes by providing human understandable explanations or approximations, locally [27] or globally [2, 18]. One representative work is LIME [27] that explains the predictions of any classifier in an interpretable and faithful manner, by learning an interpretable model locally around the prediction. More recently, developments in deep learning have been connected strongly with interpretable machine learning and have contributed novel insights into representational issues. So far these representations have been low level, and have not been integrated with the high-level symbolic representations used in knowledge representation [8].

Our work is fundamentally different from the research above. A hybrid decision model is not a pure interpretable model. It utilizes the predictive power of black-box models to preserve the predictive accuracy. It is also not a diagnostic model where the interpretable component only observes but does not participate in the decision process. Here, a hybrid decision model uses an interpretable and a black-box model simultaneously in decision making and yields better predictive performance than the black-box model alone.

The idea of combining two models can date back to more than two decades ago [15, 31]. One of the earliest works [15] proposed NBTree which induces a hybrid of decision-tree classifiers and Naive-Bayes classifiers, [29] proposed a system combining neural network and memory-based learning, [12] combined SVM and logistic regression to forecast intermittent demand of spare parts, etc. A recent work [33] divides feature spaces into regions with sparse oblique tree splitting and assign local sparse additive experts to individual regions. Aside from the singletons, there has been a large body of continuous work on neural-symbolic or neural-expert systems [8] pursued by a relatively small research community over the last two decades and has yielded several significant results [9, 23, 30, 31]. This line of research has been carried on to combine deep neural networks with expert systems to improve the predictive performance [11].

Compared to the collaborative models discussed above, our model is distinct in that the proposed framework can work with ***any*** black-box model. This black-box model could be a carefully calibrated, advanced model using confidential features or techniques. Our model only needs predictions from the black-box model and do not need to alter the black-box during training or know any other information from it. This minimal requirement of information from the black-box collaborator renders much more flexibility in creating collaboration between different models, largely preserving confidential information from the more advanced partner.

3 HYBRID DECISION MODEL

We present a general framework for building a hybrid decision model and formulate a principled objective function combining basic properties. We start with a set of training examples $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ where $\mathbf{x}_i \in \mathcal{X}$ is a tuple of attributes and $y_i \in \{1, 0\}$ is the corresponding class label. Let $f = \langle f_\iota, f_b \rangle$ represent a hybrid decision model that consists of an interpretable model f_ι, and a black-box model f_b. The black-box model is given as an input, which can be *any* model, and we only need its prediction on the training data as an input of the proposed model, denoted as $\mathcal{Y}_b = \{\hat{y_b}_i\}_i^N$.

Our goal is to construct an interpretable model f_ι to be combined with f_b, with the objective to preserve or possibly improve the overall predictive accuracy while achieving high model interpretability and data explainability. A critical issue in designing such a hybrid decision process is how to automatically distribute data to f_ι and f_b. This is equivalent to creating a partition of the dataset $\mathcal{D}$ to $\mathcal{D}_\iota$ and $\mathcal{D}_b$, corresponding to training examples sent to f_ι and f_b, respectively. We design the decision process as below: an input instance $\mathbf{x}_k$ is first sent to the interpretable model f_ι to try to generate a decision. If a decision can be made, an output $\hat{y}_{\iota k}$ is directly generated. Otherwise, it is sent to f_b to generate a decision $\hat{y_b}_k$. $\hat{y}_{\iota k} \in \mathcal{D}_\iota$ and $\hat{y_b}_k \in \mathcal{D}_b$. See the decision process in Figure 1.

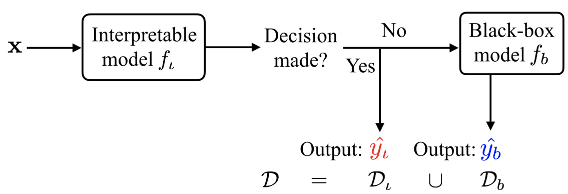


Figure 1: A decision process of a hybrid decision model.

We consider three properties critical when building such a hybrid model. 1) The **predictive accuracy**. Since f_b is pre-given, the accuracy is determined by two factors, the overall predictive accuracy of f_ι independently and the collaboration of f_ι and f_b, i.e., the partition of $\mathcal{D}$ to $\mathcal{D}_\iota$ and $\mathcal{D}_b$. f_ι and f_b being completely different models give an opportunity to the hybrid model to exploit the strengths of both models if the training examples are partitioned strategically, sending examples to the model which can predict them correctly. In some circumstance, the combination of a weak and a strong model can yield performance better than the strong model alone. 2) The **model interpretability** of f_ι. Bringing interpretability into the decision process is one of the motivations of building a hybrid model. Therefore, small size and low complexity are much-desired properties of f_ι. The definition of interpretability is model specific and usually refers to using a small number of cognitive chunks [6]. 3) The **data explainability** of the hybrid model. This is a novel metric we propose for the hybrid framework to capture the percentage of data that can be “explained” by f_ι, i.e., the percentage of $\mathcal{D}_\iota$ in $\mathcal{D}$.

Definition 3.1. The explainability of a hybrid model $f = \langle f_\iota, f_b \rangle$ on $\mathcal{D}$ is the percentage of data processed by f_ι, i.e., $\frac{\mathcal{D}_\iota}{\mathcal{D}}$, denoted as $\mathcal{E}(f, \mathcal{D})$.

We formulate the learning objective for building a hybrid decision model as a linear combination of the metrics described above. This framework unifies interpretable models and black-box models: interpretable models have explainability of 1, and black-box models have explainability of 0.

4 COLLABORATIVE BLACK-BOX AND RULE SET HYBRID MODEL

Under the proposed framework, we instantiate a hybrid decision model in this section. Here we take a significant step towards interpretability by choosing rules for f_ι. Rules are easy to understand for their simple logic and symbolic presentation. They also naturally handle the partition of data by separating examples according to if they satisfy the rules.

Now we present the Collaborative Black-box and Rule Set (CoBRUSH) model. A CoBRUSH model consists of two sets of rules. The first is a set of rules capturing positive instances, called the *positive rule set* and denoted as $\mathcal{R}^+$. The second is a set of rules capturing negative instances, called the *negative rule set* and denoted as $\mathcal{R}^-$. Let $\mathcal{R}$ represent the union of $\mathcal{R}^+$ and $\mathcal{R}^-$. If $\mathbf{x}_k$ satisfies any positive rules, it is classified as positive. Otherwise, if it satisfies any negative rules, it is classified as negative. A decision produced from rules is denoted as $\hat{y}_{\iota k}$. If $\mathbf{x}_k$ does not satisfy any rules in $\mathcal{R}^+$ or $\mathcal{R}^-$, it means f_ι fails to decide on $\mathbf{x}_k$. Then $\mathbf{x}_k$ is sent to the black-box model f_b to generate a decision $\hat{y_b}_k$. $\mathcal{D}_\iota$ is a set of instances sent to f_ι. In the context of a CoBRUSH model, we use f_ι and $\mathcal{R}$ interchangeably when we refer to the interpretable model.

We show an example of a CoBRUSH model in Table 1 learned from a heart disease data set from UCI Machine Learning Repository [21]. In this model, there are two rules in the positive rule set and one rule in the negative rule set.

Table 1: An example of a CoBRUSH model

	Rules	Model
if	age< 35 AND maximum heart rate ≥ 178 OR serum cholestorol ≥ 234 AND thal ≠ 3 AND the number of vessels ≥ 1 → $Y = 1$ (heart disease)	Positive rule set
else if	chest pain type ≠ 4 AND age > 40 → $Y = 0$ (no heart disease)	Negative rule set
else	→ $Y = f_b(\mathbf{x})$	Black-box model

Before we proceed to formulate the model, we introduce necessary notations and definitions for rule-based models.

Definition 4.1. size(R) is the number of rules in R.

Definition 4.2. A rule r *covers* an example $\mathbf{x}_i$ if $\mathbf{x}_i$ obeys the rule, denoted as covers$(r, \mathbf{x}_i) = 1$.

Definition 4.3. A rule set R *covers* an example $\mathbf{x}_i$ if $\mathbf{x}_i$ obeys at least one rule in R, i.e.

$$\text{covers}(R, \mathbf{x}_i) = \mathbb{1}\left(\sum_{r \in R} \text{covers}(r, \mathbf{x}_i) \geq 1\right). \tag{1}$$

Definition 4.4. Given a data set $\mathcal{D}$, the *coverage* of a rule set R in $\mathcal{D}$ is a set of observations covered by R, i.e., coverage$(\mathcal{R}) = \{i | \text{covers}(R, \mathbf{x}_i) = 1\}$.

Given the definitions, the classifier f built from $\mathcal{R}$ and f_b can be formulated as below.

Definition 4.5. A CoBRUSH model consists of a positive rule set $\mathcal{R}^+$, a negative rule set $\mathcal{R}^-$, and a model f_b, such that

$$f(\mathbf{x}) = \begin{cases} 1 & \text{if covers}(\mathcal{R}^+, \mathbf{x}) = 1 \\ 0 & \text{if covers}(\mathcal{R}^+, \mathbf{x}) = 0 \text{ and covers}(\mathcal{R}^-, \mathbf{x}) = 1 \\ f_b(\mathbf{x}) & \text{if covers}(\mathcal{R}^+, \mathbf{x}) = 0 \text{ and covers}(\mathcal{R}^-, \mathbf{x}) = 0 \end{cases}$$

We formulate the objective function for CoBRUSH following the objective described in Section 3. First, we measure the misclassification error to represent the predictive performance. Given rules $\mathcal{R}$, a black-box model f_b and data $\mathcal{D}$, the loss function is

$$\begin{aligned} \ell(\langle \mathcal{R}, f_b \rangle, \mathcal{D}) = \sum_{i=1}^{N} \Big(&(1 - y_i)\text{covers}(\mathcal{R}^+, \mathbf{x}_i) && \triangleright \text{ errors from } \mathcal{R}^+ \\ &+ y_i \Big(1 - \text{covers}(\mathcal{R}^+, \mathbf{x}_i)\Big)\text{covers}(\mathcal{R}^-, \mathbf{x}_i) && \triangleright \text{ errors from } \mathcal{R}^- \\ &+ \big((1 - \text{covers}(\mathcal{R}^+, \mathbf{x}_i)(1 - \text{covers}(\mathcal{R}^-, \mathbf{x}_i)\big) && \triangleright \text{ not covered by } \mathcal{R}^+ \text{ or } \mathcal{R}^- \\ &\times \big(y_i(1 - \hat{y_b}_i) + (1 - y_i)\hat{y_b}_i\big)\Big)/N. && \triangleright \text{ errors from } f_b \end{aligned} \tag{2}$$

Second, we associate the interpretability with the number of rules, size($\mathcal{R}$), a commonly used criteria for rule-based models [17, 34, 35]. Last, the explainability of a CoBRUSH model follows definition 3.1.

$$\mathcal{E}(\langle \mathcal{R}, f_b \rangle, \mathcal{D}) = \frac{|\text{coverage}(\mathcal{R})|}{|\mathcal{D}|}. \tag{3}$$

We will write $\ell(\langle \mathcal{R}, f_b \rangle, \mathcal{D})$ and $\mathcal{E}(\langle \mathcal{R}, f_b \rangle, \mathcal{D})$ as $\ell(\mathcal{R})$ and $\mathcal{E}(\mathcal{R})$, respectively, ignoring the dependence on $\mathcal{D}$ and f_b for simpler notations since f_b and $\mathcal{D}$ are both given as input.

Combining definition 4.1, formula (2) and (3), we formulate the objective function combining predictive accuracy, model interpretability and data explainability.

$$\Lambda(\mathcal{R}) = \ell(\mathcal{R}) + \theta_1 \text{size}(\mathcal{R}) - \theta_2 \mathcal{E}(\mathcal{R}), \tag{4}$$

and our goal is to find an optimal model $\mathcal{R}^*$ such that

$$\mathcal{R}^* \in \arg\min_{\mathcal{R}} \Lambda(\mathcal{R}). \tag{5}$$

Here, θ_1 and θ_2 are non-negative coefficients. Tuning the parameters will produce models at different operating point of accuracy, interpretability, and explainability. For example, in an extreme case when $\theta_2 >> \theta_1$, the output will be a model that sends all data to f_ι, producing a pure interpretable model. When $\theta_1 >> \theta_2$ and $\theta_1 >> 1$, the the model will force f_ι to have complexity 0, i.e., producing a pure black-box model. We will show in experiments the trade-offs between the three metrics when varying the parameters.

5 THEORETICAL PROPERTIES

We would like to investigate the performance of an optimal CoBRUSH model that consists of $\mathcal{R}^*$ and an input black-box f_b in terms of accuracy, interpretability, and explainability. Let $\mathcal{D}_\iota^*$ and $\mathcal{D}_b^*$ represent the two joint sets that $\mathcal{D}$ is partitioned to by $\langle \mathcal{R}^*, f_b \rangle$.

Bounds on Error. We start with the predictive performance. If only pursuing accuracy, then an interpretable model should only be used when the rules are more accurate than the black-box. In some situations when we pursue explainability more, f_ι has to include data points which it cannot predict as accurately as f_b. We would like to investigate how much the predictive performance needs to be sacrificed to gain explainability. We derive an upper bound on the error. The misclassification error on $\mathcal{D}_b$ is fixed given f_b. The predictive performance on $\mathcal{D}_\iota^*$, on the other hand, is unknown. So first we derive an upper bound on the error on $\mathcal{D}_\iota^*$. Let $\ell(\mathcal{R}^*, \mathcal{D}_\iota^*)$ represent the error on $\mathcal{D}_\iota^*$ produced by $\mathcal{R}^*$. We claim

Lemma 1 (Upper Bound on Subset Error). *If $\mathcal{R}^* \neq \emptyset$, $\ell(\mathcal{R}^*, \mathcal{D}_\iota^*) \leq \ell(\emptyset, \mathcal{D}_\iota^*) - \theta_1 + \theta_2$.*[1]

$\ell(\emptyset, \mathcal{D}_\iota^*)$ represents the error on $\mathcal{D}_\iota^*$ by applying f_b alone. $\emptyset$ represents an empty rule set. This lemma says that the error on $\mathcal{D}_\iota$ produced by $\mathcal{R}^*$ is upper bounded by the error of using f_b alone. It means the interpretable model cannot lose too much in predictive accuracy if replacing the black-box to predict $\mathcal{D}_\iota$. Otherwise, the hybrid model will not choose to use interpretable models at all, since the gain in explainability is not enough to make up for the loss in predictive accuracy. Moreover, the error bound decreases as θ_1 increases. Decreasing θ_1 places a stricter constraint on the number of rules. Therefore, only highly accurate rules qualify to hold a spot in $\mathcal{R}$. The error bound also increases with θ_2. If θ_2 increases, more emphasis is placed on expanding the coverage, so the model becomes more tolerant to errors brought by $\mathcal{R}$.

Given Lemma 1, it is straightforward to derive Theorem 1 by adding the classification errors on $\mathcal{D}_b$ produced by f_b.

Theorem 1 (Upper Bound On Error). *If $\mathcal{R}^* \neq \emptyset$, $\ell(\mathcal{R}^*) \leq \ell_b - \theta_1 + \theta_2$.*

ℓ_b is the classification error on $\mathcal{D}$ by applying f_b alone, which can be pre-computed. This theorem guarantees a good predictive performance of a hybrid model. Deriving the theorem, we claim

Corollary 5.1. *If $\ell_b \leq \theta_1 - \theta_2$, $\mathcal{R}^* = \emptyset$.*

This corollary means that if f_b is too good such that the misclassification error is less than $\theta_1 - \theta_2$, then it does not need any interpretable collaborator and there's no room for improvement in the objective function, whatever the performance of the rules are. This corollary can guide us towards choosing appropriate parameters. If we would like to get a non-empty interpretable collaborator, we need to choose θ_1, θ_2 such that $\theta_1 - \theta_2 < \ell_b$.

Bounds on Interpretability. To proceed, we investigate the bound on interpretability. We claim

Theorem 2 (Upper Bound on Size). *$size(\mathcal{R}^*) \leq \frac{\ell_b + \theta_2}{\theta_1}$.*

The upper bound decreases as θ_1 increases, consistent with the intuition from the objective function. θ_1 represents the penalty for adding a rule to the model. Therefore the higher θ_1, smaller models are favored. The bound increases with θ_2 since expanding the coverage needs more rules. Then we drive a lower bound on the support of any rule in $\mathcal{R}^*$.

Theorem 3 (Lower Bound on Support). *$\forall r \in \mathcal{R}^*, |coverage(r)| \geq \frac{N\theta_1}{1+\theta_2}$.*

This means that the optimal model does not contain rules with a support lower than the bound. This bound increases as θ_1 increases, since θ_1 represents the penalty of adding a rule. This theorem is important in the search procedure discussed later as it confines

[1]For all the proofs in the paper, please see the supplementary material on the author's Researchgate page at https://www.researchgate.net/profile/Tong_Wang57

the search chain to a subset of the rule space, excluding unqualified rules (rules with support lower than $\frac{N\theta_1}{1+\theta_2}$) from consideration, which greatly reduces computation.

Bounds on Explainability. Finally, we study the explainability performance of optimal CoBRUSH models.

Theorem 4 (Lower Bound On Explainability). *If $\mathcal{R}^* \neq \emptyset$, then $\mathcal{E}(\mathcal{R}^*) \geq \frac{\theta_1 - \ell_b}{\theta_2}$.*

This theorem shows that the explainability is lower bounded. This lower bound is binding when the CoBRUSH model is perfect and has exactly one rule, giving $\ell(\mathcal{R}^*) = 0$ and $\text{size}(\mathcal{R}^*) = 1$.

6 MODEL TRAINING

We describe a training algorithm to find an optimal solution $\mathcal{R}^*$ via minimizing the objective (5). Learning rule-based models is challenging because it involves a search over exponentially many possible sets of rules. Since each rule is a conjunction of conditions, the number of rules increases exponentially with the number of features in a data set, and the solution space (all possible rule sets) is a power set of the rule space. Fortunately, our objective has a nice structure that can be exploited for reducing computation. Here we propose an efficient search algorithm and derive theoretical bounds for faster computation.

Algorithm structure: The algorithm is presented in Algorithm 1. Given training examples $\mathcal{D}$, a black-box model f_b, parameters θ_1, θ_2, base temperature C_0, and the total number of iterations T, the search procedure follows the main structure of simulated annealing. Each state corresponds to a rule set model, indexed by the time stamp t, denoted as $\mathcal{R}_{[t]}$. The temperature is a function of the time stamp, $C_0^{1-\frac{t}{T}}$ that gradually cools down. The neighboring states are defined as rule sets that are obtained via adding or removing a rule from the current set. A proposed neighbor is accepted with probability $\exp(\frac{\Lambda(\mathcal{R}_{[t]})-\Lambda(\mathcal{R}_{[t+1]})}{C_0^{1-\frac{t}{T}}})$. The starting state $\mathcal{R}_{[0]}$ is initialized with an empty set.

Dynamic Bounds Throughout the algorithm, we utilize different bounds to improve the search efficiency. First, we use Theorem 3 to prune the search space. We use FP-Growth to pre-mine a set of rules Υ that satisfy the minimum support of $\frac{N\theta_1}{1+\theta_2}$ and restrict the search only within Υ. Therefore, instead of searching the entire rule space, we only examine a subset of rules. This reduces computation without hurting the optimality. Then, we would like to use bounds on the size of the model and explainability to confine the Markov Chain within promising solution space, preventing it from going too far and wasting too much search time. In Theorem 2 and Theorem 4, we derived bounds by comparing the optimal solution with the pure black-box f_b as the benchmark. As the model gets better with each iteration, the algorithm will likely find solutions better than f_b. Therefore, we compare the optimal solution with the best solution found so far, to get tighter bounds.

Let $\lambda^*_{[t]}$ represent the best objective value found till time t, i.e.

$$\lambda^*_{[t]} = \min_{\tau \leq t} \Lambda(\mathcal{R}_{[\tau]}). \tag{6}$$

Algorithm 1 Stochastic Local Search Algorithm

Input: f_b, $\mathcal{D}$, θ_1, θ_2, C_0
Initialize:
$\mathcal{R}^* = \mathcal{R}^{[0]} \leftarrow \emptyset$ ▷ start with a pure black-box model
$\Upsilon \leftarrow$ FPGrowth($\mathcal{D}$, minsupp $= \frac{N\theta_1}{1+\theta_2}$) ▷ mine candidate rules from $\mathcal{D}$ using minimum support from Theorem 3

1: **for** $t = 0 \to T$ **do**
2: $\quad\delta$ = random()
3: $\quad$**if** $\delta \leq \frac{1}{3}$ or size($\mathcal{R}_{[t]}$) $\geq \frac{\lambda^*_{[t]}+\theta_2}{\theta_1}$ **then** ▷ Using Corollary 6.1
4: $\quad\quad\mathcal{R}_{[t+1]} \leftarrow$ remove a rule from $\mathcal{R}_{[t]}$ ▷ decrease the size of $\mathcal{R}_{[t]}$
5: $\quad$**else if** $\delta \leq \frac{2}{3}$ or $\mathcal{E}^* \leq \frac{\theta_1 - \lambda^*_{[t]}}{\theta_2}$ **then** ▷ Using Corollary 6.2
6: $\quad\quad\mathcal{R}_{[t+1]} \leftarrow$ add a rule to $\mathcal{R}$ ▷ increase the explainability
7: $\quad$**else** ▷ decrease classification error
8: $\quad\quad\epsilon \leftarrow \{k | f_{[t]}(\mathbf{x}_k) \neq y_k\}$ ▷ indices of misclassified examples.
9: $\quad\quad$**if** $\epsilon \in$ coverage($\mathcal{R}_{[t]}$) **then** ▷ an error from $\mathcal{R}_{[t]}$
10: $\quad\quad\quad\mathcal{R}_{[t+1]} \leftarrow$ remove a rule from $\mathcal{R}_{[t]}$ that covers ϵ.
11: $\quad\quad$**else** ▷ an error from f_b
12: $\quad\quad\quad\mathcal{R}_{[t+1]} \leftarrow$ add a rule that covers ϵ to $\mathcal{R}_{[t]}$
13: $\quad\quad$**end if**
14: $\quad$**end if**
15: $\quad\mathcal{R}_{[t+1]}$ is accepted with probability $\exp(\frac{\Lambda(\mathcal{R}_{[t]})-\Lambda(\mathcal{R}_{[t+1]})}{C_0^{\frac{t}{T}}})$ ▷ accept the proposal with certain probability
16: $\quad\mathcal{R}^* = \arg\min_{\mathcal{R}_{[t+1]}, \mathcal{R}^*} \Lambda(\mathcal{R})$ ▷ update the best soslution
17: **end for**
18: **Output:** $\mathcal{R}^*$

We update Theorem 2 and Theorem 4 with $\lambda^*_{[t]}$. Instead of comparing with $\Lambda(\emptyset)$, one needs to use $\lambda^*_{[t]}$ which is updated whenever a better solution is found.

Corollary 6.1 (Dynamic Upper Bound on Size). *$size(\mathcal{R}^*) \leq \frac{\lambda^*_{[t]}+\theta_1}{\theta_2}$.*

Corollary 6.2 (Dynamic Lower Bound on Explainability). *$\mathcal{E}(\mathcal{R}^*) \leq \frac{\theta_1 - \lambda^*_{[t]}}{\theta_1}$.*

Both bounds become tighter as $\lambda^*_{[t]}$ continuously gets smaller. Then we detail the proposing step at each iteration (line 2-15).

Proposing step: To propose a neighbor, at each iteration, we choose to improve one of the three terms (accuracy, interpretability, and explainability) with approximately equal probabilities. With probability $\frac{1}{3}$, or when the upper bound of the model in Corollary 6.1 is violated, we aim to decrease the size of $\mathcal{R}_{[t]}$ (improve interpretability) by removing a rule from $\mathcal{R}_{[t]}$ (line 3 - 4). With probability $\frac{1}{3}$ or when the lower bound on explainability is violated, we aim to increase coverage of $\mathcal{R}_{[t]}$ (improve explainability) by adding a rule to $\mathcal{R}_{[t]}$ (line 5-6). Finally, with another probability $\frac{1}{3}$, we aim to decrease the classification error (improve accuracy).

To decrease the misclassification error, we follow similar steps used in [34]. At each iteration, we draw an example from examples misclassified by the current model. If the example is not covered by $\mathcal{R}_{[t]}$, it means it was previously sent to the black-box model, since we cannot alter the black-box model, we add a rule to the positive or negative rule set (consistent with the label of the instance) to cover the example, re-routing it to f_l. If the example is covered by $\mathcal{R}_{[t]}$, it means it was sent to the interpretable model but was covered by the wrong rule set. Then there are two ways to correct the mistake. With probability 1/2 or if the instance is covered by both positive and negative rule sets, we swap the two rule sets, i.e., change the order of the rule sets that an instance goes through. Otherwise, we remove a rule from the positive (if the instance is negative) or negative (if the instance is positive) rule set that covers the example, re-routing it to f_b. When choosing a rule to add or remove, we first evaluate the rules using precision, which is the percentage of correctly classified examples of the rule. Then we balance between exploitation, choosing the best rule, and exploration, choosing a random rule, to avoid getting in a local minimum.

7 EXPERIMENTS

We perform a detailed experimental evaluation of CoBRUSH on synthetic and real-world datasets and compare the performance with state-of-the-art interpretable and black-box baselines.

7.1 Effect of Parameters and Black-Box Models

The first goal of the analysis in this section is to investigate the trade-offs between accuracy, explainability, and interpretability of CoBRUSH models. Specifically, we want to study how these three metrics change with parameters θ_1 and θ_2, and how varying one metric affects the others. The second goal is to analyze the effect of f_b with various predictive performance. In order to have full control of data and have a ground truth to compare with, we choose synthetic datasets for this analysis.

We generate 10 datasets $\{\mathbf{x}^m\}^{m=1,\cdots 10}$ of size 10,000 (instances) $\times$10 (features) where each feature value $\mathbf{x}^m_{i,d}$ is a number randomly drawn from 0 to 1. Then we used logistic functions with $\{w_i\}_{i=0}^{10}$ randomly drawn from -10 to 10 to generate the target variables. Then $y_i^m = 1$ if $\frac{1}{1+\exp(-w_0-\mathbf{w}^T\mathbf{x}_i^m)} \geq 0.5$. Thus we obtain 10 independent datasets for binary classification. We split each dataset into 75% of training set and 25% of test set and run the proposed model on these datasets. The maximum length of rules is set to 4.

Our goal is to study the effect of f_b, θ_1, θ_2 on the output model. First, we vary the performance of black-box models. For each dataset, we create three black-box models with different predictive performance. The first black-box is a perfect model with accuracy of 1, i.e., $\ell_b = 0$, with the input labels set as $\hat{y_b}_i = y_i^m$. Then we create two imperfect black-box models with error rates $\ell_b = 0.2$ and 0.4. To do that we set a random 20% and 40% of labels to $1 - \mathrm{Y}_i^m$ and the rest to Y_i^m. Then, we choose θ_1, θ_2 from $\{0, 0.2, 0.4, 0.6, 0.8, 1, 10, 100\}$, obtaining 64 pairs of parameters. Therefore, for each dataset, we run CoBRUSH with a combination of black-box models and parameters, generating a total of 192 models. We evaluate the size of the models and test their accuracy and explainability on the test set. Figure 2 shows the results.

The explainability and interpretability behaved consistently for different ℓ_b: as θ_1 increases (along the Y-axis), the model pursues fewer rules, decreasing the size of rule sets. Meanwhile, as θ_2 increases (along the X-axis), CoBRUSH pursues outputs with higher explainability by including more rules to cover more data. Explainability and the size of rules act in cohort that higher explainability generally indicates more rules in the model.

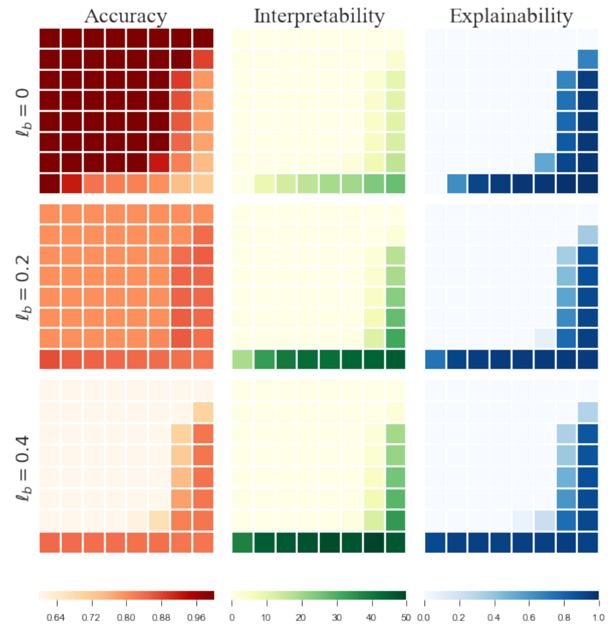
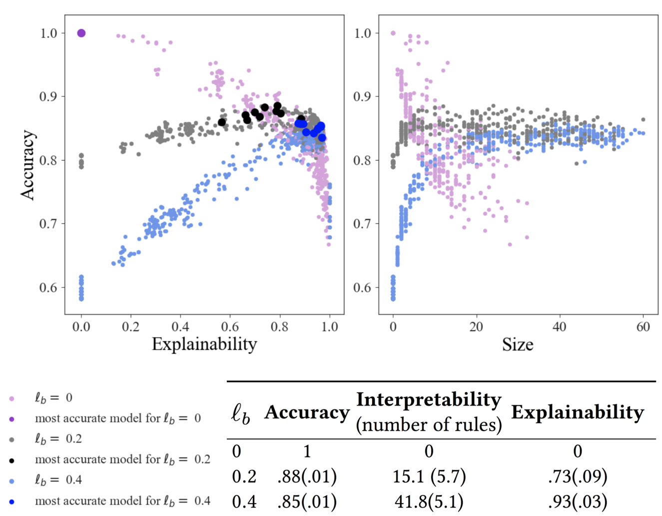


Figure 2: Accuracy, interpretability, and explainability of models trained from synthetic data with θ_1 and θ_2 chosen from $\{0, 0.2, 0.4, 0.6, 0.8, 1, 10, 100\}$, and ℓ_b from $\{0, 0.2, 0.4\}$. The y-axis represents different values of θ_1 and the x-axis represents different values of θ_2. θ_1 and θ_2 increase along the direction of the axes.



ℓ_b	Accuracy	Interpretability (number of rules)	Explainability
0	1	0	0
0.2	.88(.01)	15.1 (5.7)	.73(.09)
0.4	.85(.01)	41.8(5.1)	.93(.03)

Figure 3: Accuracy vs. explainability and size of the most accurate CoBRUSH models from 10 datasets $\times$ 3 bb models.

To better visualize the relationship between accuracy and the other two metrics, we plot in Figure 3 the accuracy of all models on the y-axis versus the explainability and model size on the x-axis. There are three sets of points marked with different colors, representing black-box models with $\ell_b = 0, 0.2, 0.4$, respectively.

Each set has 640 of points generated from 10 datasets × 64 pairs of parameters. In the accuracy-explainability plot, each set of points starts from explainability = 0, representing using a pure black-box model and achieving a mean accuracy equal to ℓ_b and end at explainability = 1 when all data are classified by rules. Note that the accuracy at explainability = 1 is independent of which black-box model the rules collaborate with, therefore the three clusters all end with a mean accuracy of 0.74 at explainability =1.

Then we examine each set of points. For the perfect black-box model with $\ell_b = 0$, as the explainability and size increase, accuracy decreases. The intuition is straightforward: as more data are sent to the non-perfect model, the overall accuracy decreases accordingly. If the black-box model is not perfect ($\ell_b = 0.2$ or 0.4), it is interesting to observe that the accuracy-explainability and the accuracy-size appear convex, with the most accurate model located in the middle of the curve with a non-negligible explainability, represented with darker points in Figure 3. We find 10 most accurate models from 10 datasets and then report their accuracy, model size, and explainability in the table below Figure 3. The accuracy increases 10% compared to the black-box model alone for $\ell_b = 0.2$ and 25% for $\ell_b = 0.25$. It means the **collaboration of a black-box model and an interpretable model can improve the predictive accuracy while gaining considerable explainability in the data.**

This behavior of hybrid decision models is not surprising considering similar phenomenon in ensemble methods where a collection of weak learners can form a model that's more accurate than any individual base learner. Ensemble methods rely on the combined decision of multiple leaners, often referred to as "collective wisdom". Each data point is decided by multiple learners at the same time. The hybrid decision model, however, relies on a strategic partitioning of data to utilize the strength of two models in different data subspace. A data point is sent to one model for decision, and if the model is trained well, it is sent to the model that can predict it correctly. This yields a performance boost in the overall accuracy.

This result is important since it shows that introducing interpretability into a decision-making process doesn't necessarily need to sacrifice predictive performance, but can improve the predictive accuracy while gaining explainability to the data. This big advantage will give an immense opportunity for hybrid decision models to be used in real-world applications.

7.2 Experiments on Public Datasets

We analyze five real-world datasets from domains where interpretability is highly desired, including healthcare, judiciaries, demography and customer analysis. A summary of the datasets is shown in Table 2. These datasets vary in size and types of variables. All datasets are publicly available at UCI Repository [21] or ICPSR. We processe each dataset by binarizing all categorical features and discretizing real-valued features into five intervals.

Table 2: A summary of Datasets

Dataset	N	d	$Y = 1$
Hospital Readmission	100,000	55	readmitted
Juvenile Delinquency	4,023	69	committed delinquency
Recidivism	11,645	106	recidivism
Credit card	30,000	24	credit card default
Census	48,842	14	income$\geq 50k$

Baselines We benchmark the performance of CoBRUSH against state-of-the-art interpretable and black-box methods. For interpretable models, we choose classic decision trees, C4.5 [25] and C5.0 [16], and two state-of-the-art rule-based classifiers, Scalable Bayesian Rule Lists (SBRL) [36] and Bayesian Rule Sets (BRS) [35]. BRS and SBRL are two representative recent methods which have proved to achieve simpler models with competitive predictive accuracy compared to the older rule-based classifiers. For black-box models, we choose three top performing models, Random Forests [20], AdaBoost [7] and extreme gradient boosting trees (XGBoost) [5]. Specifically, XGBoost has been the leading model for working with standard tabular data in recent years.

Implementation We use R and python packages [10, 24, 26] for baseline methods except for BRS which has the code publicly available[2]. For C5.0, we tune the minimum number of samples in at least two of the splits. For BRS, we set the maximum length of rules to 4. There are parameters $\alpha_+, \beta_+, \alpha_-, \beta_-$ that govern the likelihood of the data. We set β_+, β_- to 1 and vary α_+, α_- from $\{100, 1000, 10000\}$. For SBRL, we set the maximum length of rules to 2 since the computer will have a memory overflow using longer rules (the computer used for this experiments has 3.5 GHz 6-Core Intel Xeon E5 processor and 64 GB 1866 MHz DDR3 RAM). There are hyperparameters λ for the expected length of the rule list and η for the expected cardinality of the rules in the optimal rule list. We vary λ from $\{5, 10, 15, 20\}$ and η from $\{1, 2, 3, 4, 5\}$. For random forests, we tune the number of trees and the number of features picked at each tree. For Adaboost, we tune the number of estimators. For XGBoost, we choose the booster type to be gbtree and tuned gamma, maximum depth of tree and minimum child weight. We use nested cross-validation and grid search to tune parameters for each method on each fold, obtaining five best models for each method. For C4.5, we tune the minimum size of leaves.

Then we use the three black-box methods as an input to CoBRUSH to build three hybrid decision models, represented as $\langle \cdot, \text{RF} \rangle$, $\langle \cdot, \text{AdaBoost} \rangle$ and $\langle \cdot, \text{XGBoost} \rangle$. We set the maximum length of rules to 4. For each model, we tune θ_1 and θ_2 via nested CV and choose the pair that achieved the highest mean CV error.

7.2.1 Accuracy and explainability. We report the accuracy and explainability on the test set in in Table 3and 4, respectively. It is clear from the table that when a black-box model collaborated with rules, the CoBRUSH model was able to achieve the same or better predictive accuracy. This is consistent with our results from the synthetic data. We underlined the results when the CoBRUSH model outperformed its black-box, for example, all CoBRUSH models on dataset juvenile and recidivism. Especially for Adaboost on recidivism, introducing rules increased the over predictive accuracy from .690 to .743. Combining the two tables, we observe that the better accuracy were achieved by CoBRUSH models when the rules captured a considerable portion of the data, i.e., high explainability. One the other hand, hybrid model failed to do better when the explainability was 0 or very small. This is because the predictive performance can be better only when rules are exploited and the data space is partitioned strategically to combine the strength of

[2] https://github.com/wangtongada/BOA

Table 3: Mean accuracy and standard deviation of accuracy of all methods via 5-CV.

Models \ Task		Juvenile	Credit card	Census	Recidivism	Diabetes
Interpret-able	SBRL	.875(.006)	.820(.010)	.828(.005)	.740(.008)	.613(.012)
	BRS	.881(.009)	.815(.011)	.747(.006)	.786(.004)	.610(.014)
	C4.5	.878(.013)	.820(0.01)	.848(.004)	.692(.055)	.613(.012)
	C5.0	.885(.008)	.819(.009)	.865(.003)	.685(.050)	.612(.012)
Black-box	RF	.894(.004)	.819(.009)	.863(.006)	.725(.027)	.615(.012)
	AdaBoost	.892(.007)	.818(.010)	.837(.004)	.690(.052)	.617(.011)
	XGBoost	.886(.006)	.818(.010)	.853(.004)	.738(.016)	.619(.011)
Hybrid **CoBRUSH**	$\langle\cdot$, RF$\rangle$	.898(.006)	.820(.010)	.863(.006)	.732(.025)	.615(.012)
	$\langle\cdot$, AdaBoost$\rangle$	.899(.005)	.820(.010)	.837(.004)	.743(.018)	.617(.012)
	$\langle\cdot$, XGBoost$\rangle$	.889(.003)	.818(.010)	.853(.004)	.747(.006)	.619(.011)

Table 4: Mean accuracy and standard deviation of explainability of models reported in Table 3.

	Juvenile	Credit Card	Census	Recidivism	Diabetes
$\langle\cdot$, RF$\rangle$	.884(.114)	.902(.044)	.000(.000)	.797(.119)	.000(.000)
$\langle\cdot$, Adaboost$\rangle$	.802(.071)	.878(.052)	.132(.264)	.688(.354)	.256(.313)
$\langle\cdot$, XGBoost$\rangle$	.558(.456)	.900(.005)	.000(.000)	.614(.323)	.200(.400)

the two models. When the explainability is 0 or very low, the contribution from the interpretable model is limited, therefore could not improve the predictive accuracy.

7.2.2 Why a hybrid model is more accurate. We investigate why a CoBRUSH model was able to achieve higher accuracy than an individual black-box model. Since $\mathcal{D} = \mathcal{D}_\iota \cup \mathcal{D}_b$, the predictive performance of a hybrid model $\langle f_\iota, f_b\rangle$ depends on the performance of f_ι on $\mathcal{D}_\iota$ and the performance of f_b on $\mathcal{D}_b$ (pre-given). Therefore we break down the accuracy and report partial accuracy on $\mathcal{D}_\iota$ and $\mathcal{D}_b$. See Table 5. The predictive performance of f_b alone can also be broken down into $\mathcal{D}_b$ and $\mathcal{D}_\iota$. Comparing f_b and $\langle f_\iota, f_b\rangle$, the predictions on $\mathcal{D}_b$ are the same. Therefore, the difference in their accuracy only comes from $\mathcal{D}_\iota$, represented by f_ι on $\mathcal{D}_\iota$ and f_b on $\mathcal{D}_\iota$ in Table 5. We also report the accuracy of f_ι alone when forcing explainability to 1 by increasing θ_2 to a very large number. This accuracy is independent of which black-box the rules work with.

Table 5: Accuracy break down for CoBRUSH models

		Juvenile	Credit Card	Census	Recidivism	Diabetes
$\langle\cdot$, RF$\rangle$	f_b on $\mathcal{D}_b$	.512(.280)	.642(.020)	.863(.006)	.579(.063)	.615(.012)
	f_ι on $\mathcal{D}_\iota$	.925(.025)	.839(.013)	–	.768(.042)	–
	f_b on $\mathcal{D}_\iota$	.921(.026)	.838(.014)	–	.760(.045)	–
$\langle\cdot$, Ada-boost$\rangle$	f_b on $\mathcal{D}_b$	.727(.038)	.655(.020)	.779(.000)	.595(.117)	.615(.002)
	f_ι on $\mathcal{D}_\iota$	.942(.019)	.842(.015)	.864(.000)	.753(.016)	.618(.009)
	f_b on $\mathcal{D}_\iota$	.932(.017)	.841(.015)	.863(.000)	.676(.047)	.617(.014)
$\langle\cdot$, XG-Boost$\rangle$	f_b on $\mathcal{D}_b$	.712(.119)	.694(.045)	.853(.004)	.747(.072)	.616(.011)
	f_ι on $\mathcal{D}_\iota$	.909(.007)	.832(.006)	–	.753(.008)	.620(.010)
	f_b on $\mathcal{D}_\iota$	.907(.009)	.832(.007)	–	.736(.015)	.618(.007)
f_ι on $\mathcal{D}$(expla=1)		.880(.008)	.812(.010)	.756(.012)	.725(.004)	.610(.009)

As shown in the table, the accuracy of f_ι on $\mathcal{D}_\iota$ is always higher than that of f_b on $\mathcal{D}_\iota$. This again shows that our fundamental assumption is true - there exists a subspace where the interpretable model is more accurate than the black-box model, even if the black-box model is better globally. This subspace is $\mathcal{D}_\iota$ and our CoBRUSH was successful in finding it and assigning it to rules, gaining an improved accuracy and explainability at the same time. Therefore, the process of building a hybrid decision model is to equivalent to finding a subspace where the interpretable component can outperform the black-box model.

Next, we investigate what kind of data is in $\mathcal{D}_\iota$ compared to those in $\mathcal{D}_b$. So we compare the partial accuracy of f_b on $\mathcal{D}_\iota$ and f_b on $\mathcal{D}_b$ and find that the former is always lower than the latter. This result illuminates how the data space is partitioned: data that are "easy" to predict are sent to $\mathcal{D}_\iota$ and those that are harder to predict remain in $\mathcal{D}_b$ for f_b to decide. Therefore $\mathcal{D}_\iota$ is a subspace that is relatively far away from the decision boundary. This is consistent with our intuition: if an instance is easy to predict than a simple model can suffice or even outperform a complicated model.

7.2.3 Interpretability. Finally, we analyze the interpretability of hybrid decision models, measured by the total number of rules in the positive and negative rule set. We compare the size of CoBRUSH models with the number of rules in the interpretable baselines. Among the four baseline models, we can directly measure the number of rules in SBRL models and BRS models since, like CoBRUSH, they consist of a list or a set of rules. To compare with decision trees, we count the number of leaves. Figure 4 shows the results for the seven methods on five datasets. The x-axis represents the sizes, and the y-axis represents the 5-CV test accuracy. We use error bars to represent the standard deviation. As shown in Figure 4, CoBRUSH always uses a much smaller number of rules. This is not surprising since the model is responsible for only part of the data including a few times when there is no rule in the model.

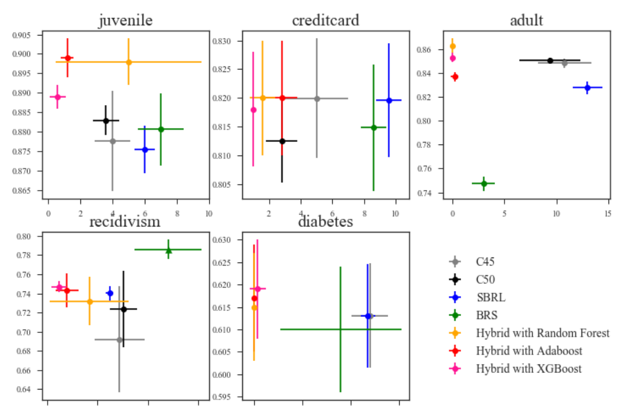


Figure 4: Comparison of model size for CoBRUSH, BRS, SBRL, C4.5 and C5.0.

We show an example of CoBRUSH model in Table 6. This model is trained with Adaboost on one fold of census dataset. The overall accuracy on the test set is 0.826, and the explainability is 0.97. The positive rule set has a partial predictive accuracy of 0.739, and it covers 13% of the data. The negative rule set consists of two rules, and the partial accuracy is 0.85. This part explains 84% of the data. The last 3% of data cannot be determined by rules and are sent to the Adaboost model which achieved an accuracy of 0.64.

Table 6: An example of CoBRUSH learned from census dataset

	Rules	Model	Accuracy	Support
if	number of years of education $\geq$ 13 AND marital status = married civil spouse AND relationship $\neq$ not in family $\rightarrow Y = 1$ (Income $\geq$ 50k)	Positive rule set (one rule)	.73	.13
else if	number of years of education $<$ 13 AND occupation $\neq$ executive manager AND relationship $\neq$ wife OR relationship $\neq$ husband $\rightarrow Y = 0$ (Income $<$ 50k)	Negative rule set (two rules)	.85	.84
else	$\rightarrow Y = f_b(\mathbf{x})$	Black-box Model	.64	.03

8 CONCLUSIONS

In this paper, we proposed a general framework for learning a Hybrid Decision Model that integrates an interpretable model with *any* black-box model to preserve or possibly improve the predictive accuracy while introducing explanations in the decision process. We instantiated this framework with Collaborative Black-box and Rule Set Hybrid (CoBRUSH) model using logic rules as the interpretable component. We investigated theoretical properties of CoBRUSH models and derived performance bounds on predictive accuracy, model interpretability, and data explainability. To train the model, we designed an efficient search algorithm that exploits theoretical bounds to improve search efficiency. Experiments demonstrated that hybrid decision models were able to achieve higher accuracy than black-boxes alone while gaining explainability.

The CoBRUSH model is just one example of the proposed hybrid decision models. A major contribution of this work is that we proposed a general framework of combining interpretable models with black-box models. Specifically, we proposed a new concept of explainability, which is a fundamental property for any hybrid models. Our framework can support the exploration of a variety of interpretable models, such as linear models, decision trees, etc.

The proposed framework provides a new solution to bridging the gap between interpretable and black-box models, especially when one wishes not to give up the high predictive accuracy or interpretability. Hybrid decision models connect the two discrete points of pure interpretable models and pure black-box models into a continuous regime defined by accuracy and explainability. A well-trained hybrid model can strike a nice balance between the two extremes and provide a better combination of both. We envision the hybrid machine learning will open up many new research opportunities in machine learning for decision-making.

Code The code for this paper is available at https://github.com/wangtongada/CoBRUSH

REFERENCES

[1] A. Aamodt and E. Plaza. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI comm*, 7(1):39–59, 1994.

[2] P. Adler, C. Falk, S. A. Friedler, G. Rybeck, C. Scheidegger, B. Smith, and S. Venkatasubramanian. Auditing black-box models for indirect influence. In *ICDM*, pages 1–10. IEEE, 2016.

[3] E. Angelino, N. Larus-Stone, D. Alabi, M. Seltzer, and C. Rudin. Learning certifiably optimal rule lists. In *SIGKDD*, pages 35–44. ACM, 2017.

[4] I. Bichindaritz and C. Marling. Case-based reasoning in the health sciences: What's next? *Artificial intelligence in medicine*, 36(2):127–135, 2006.

[5] T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. In *SIGKDD*, pages 785–794. ACM, 2016.

[6] F. Doshi-Velez and B. Kim. A roadmap for a rigorous science of interpretability. *arXiv preprint arXiv:1702.08608*, 2017.

[7] Y. Freund and R. E. Schapire. A desicion-theoretic generalization of on-line learning and an application to boosting. In *European conference on computational learning theory*, pages 23–37. Springer, 1995.

[8] A. d. Garcez, T. R. Besold, L. De Raedt, P. Földiak, P. Hitzler, T. Icard, K.-U. Kühnberger, L. C. Lamb, R. Miikkulainen, and D. L. Silver. Neural-symbolic learning and reasoning: contributions and challenges. In *Proceedings of the AAAI Spring Symposium on Knowledge Representation and Reasoning: Integrating Symbolic and Neural Approaches, Stanford*, 2015.

[9] A. S. d. Garcez, K. B. Broda, and D. M. Gabbay. *Neural-symbolic learning systems: foundations and applications*. Springer Science & Business Media, 2012.

[10] K. Hornik, A. Zeileis, T. Hothorn, and C. Buchta. Rweka: an r interface to weka. *R package version*, pages 03–4, 2007.

[11] Z. Hu, X. Ma, Z. Liu, E. Hovy, and E. Xing. Harnessing deep neural networks with logic rules. *arXiv preprint arXiv:1603.06318*, 2016.

[12] Z. Hua and B. Zhang. A hybrid support vector machines and logistic regression approach for forecasting intermittent demand of spare parts. *Applied Mathematics and Computation*, 181(2):1035–1048, 2006.

[13] B. Kim, C. Rudin, and J. Shah. Bayesian case model: A generative approach for case-based reasoning and prototype classification. In *NIPS*, 2014.

[14] H. C. Koh, W. C. Tan, and C. P. Goh. A two-step method to construct credit scoring models with data mining techniques. *International Journal of Business and Information*, 1(1), 2015.

[15] R. Kohavi. Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. In *KDD*, volume 96, pages 202–207, 1996.

[16] M. Kuhn, S. Weston, N. Coulter, and R. Quinlan. C50: C5. 0 decision trees and rule-based models. *R package version 0.1. 0-21*, 50, 2014.

[17] H. Lakkaraju, S. H. Bach, and J. Leskovec. Interpretable decision sets: A joint framework for description and prediction. In *SIGKDD*. ACM, 2016.

[18] H. Lakkaraju, E. Kamar, R. Caruana, and J. Leskovec. Interpretable & explorable approximations of black box models. *arXiv preprint arXiv:1707.01154*, 2017.

[19] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[20] A. Liaw, M. Wiener, et al. Classification and regression by randomforest. *R news*, 2(3):18–22, 2002.

[21] M. Lichman. Uci machine learning repository, 2013.

[22] T. McCormick, C. Rudin, and D. Madigan. A hierarchical model for association rule mining of sequential events: An approach to automated medical symptom prediction. 2011.

[23] K. McGarry, S. Wermter, and J. MacIntyre. Hybrid neural systems: from simple coupling to fully integrated neural networks. *Neural Computing Surveys*, 2(1):62–93, 1999.

[24] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.

[25] J. R. Quinlan. *C4. 5: programs for machine learning*. Elsevier, 2014.

[26] R. Quinlan. Data mining tools see5 and c5. 0. 2004.

[27] M. T. Ribeiro, S. Singh, and C. Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *SIGKDD*, pages 1135–1144. ACM, 2016.

[28] P. R. Rijnbeek and J. A. Kors. Finding a short and accurate decision rule in disjunctive normal form by exhaustive search. *Machine learning*, 80(1), 2010.

[29] C.-K. Shin, U. T. Yun, H. K. Kim, and S. C. Park. A hybrid approach of neural network and memory-based learning to data mining. *IEEE Transactions on Neural Networks*, 11(3):637–646, 2000.

[30] I. A. Taha and J. Ghosh. Symbolic interpretation of artificial neural networks. *IEEE Transactions on knowledge and data engineering*, 11(3):448–463, 1999.

[31] G. G. Towell and J. W. Shavlik. Knowledge-based artificial neural networks. *Artificial intelligence*, 70(1-2):119–165, 1994.

[32] B. Ustun and C. Rudin. Supersparse linear integer models for optimized medical scoring systems. *Machine Learning*, 102(3):349–391, 2016.

[33] J. Wang, R. Fujimaki, and Y. Motohashi. Trading interpretability for accuracy: Oblique treed sparse additive models. In *SIGKDD*, pages 1245–1254. ACM, 2015.

[34] T. Wang, C. Rudin, F. Doshi, Y. Liu, E. Klampfl, and P. MacNeille. A bayesian framework for learning rule set for interpretable classification. *Journal of Machine Learning Research*, 2017.

[35] T. Wang, C. Rudin, F. Velez-Doshi, Y. Liu, E. Klampfl, and P. MacNeille. Bayesian rule sets for interpretable classification. *ICDM*, 2016.

[36] H. Yang, C. Rudin, and M. Seltzer. Scalable bayesian rule lists. *ICML*, 2017.

[37] J. Zeng, B. Ustun, and C. Rudin. Interpretable classification models for recidivism prediction.