# Learning Hybrid Interpretable Models: Theory, Taxonomy, and Methods

**Anonymous authors**
**Paper under double-blind review**

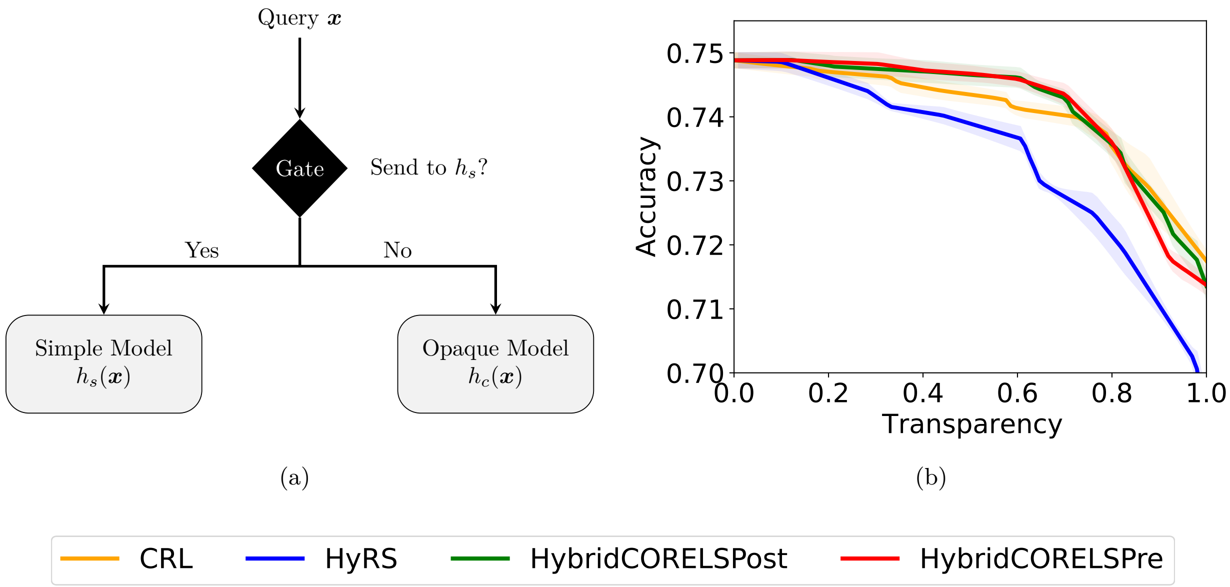
## Abstract

A hybrid model involves the cooperation of an interpretable model and a complex black box. At inference, any input of the hybrid model is assigned to either its interpretable or complex component based on a gating mechanism. The advantages of such models over classical ones are two-fold: 1) They grant users precise control over the level of transparency of the system and 2) They can potentially perform better than a standalone black box since redirecting some of the inputs to an interpretable model implicitly acts as regularization. Still, despite their high potential, hybrid models remain under-studied in the interpretability/explainability literature. In this paper, we remedy this fact by presenting a thorough investigation of such models from three perspectives: Theory, Taxonomy, and Methods. First, we explore the theory behind the generalization of hybrid models from the Probably-Approximately-Correct (PAC) perspective. A consequence of our PAC guarantee is the existence of a *sweet spot* for the optimal transparency of the system. When such a sweet spot is attained, a hybrid model can potentially perform better than a standalone black box. Secondly, we provide a general taxonomy for the different ways of training hybrid models: the *Post-Black-Box* and *Pre-Black-Box* paradigms. These approaches differ in the order in which the interpretable and complex components are trained. We show where the state-of-the-art hybrid models Hybrid-Rule-Set and Companion-Rule-List fall in this taxonomy. Thirdly, we implement the two paradigms in a single method: HybridCORELS, which extends the CORELS algorithm to hybrid modeling. By leveraging CORELS, HybridCORELS provides a certificate of optimality of its interpretable component and precise control over transparency. We finally show empirically that HybridCORELS is competitive with existing hybrid models, and performs just as well as a standalone black box (or even better) while being partly transparent.

## 1 Introduction

The ever-increasing integration of machine learning models in high-stakes decision-making contexts (e.g., healthcare, justice, finance) has fostered a growing demand for transparency in recent years. Current workhorses to address transparency concerns in machine learning include black-box explanation and transparent design techniques (Guidotti et al., 2018). Black-box explanation techniques aim at explaining complex machine learning models in a post-hoc fashion with global explanations such as `Trepan` (Craven & Shavlik, 1995) and `BETA` (Lakkaraju et al., 2017) or local explanations such as `LIME` (Ribeiro et al., 2016) and `SHAP` (Lundberg & Lee, 2017). On the other hand, transparent design concerns the development of inherently interpretable models such as rule lists (Rivest, 1987; Angelino et al., 2017), rule sets (Rijnbeek & Kors, 2010), decision trees (Breiman, 2017), and scoring systems (Ustun & Rudin, 2016).

However, both black-box explanations and transparent design face performance and trustworthiness challenges that can prevent their wide adoption. On the one hand, while inherently interpretable models can be more easily understood and adopted by non-domain experts, their out-of-the-box performance can be worst than non-transparent models. Moreover, training such models to optimality is often NP-hard due to their discrete nature. On the other hand, black boxes can effortlessly attain high performance but their decision



(a)

(b)

Figure 1: Overview of Hybrid Interpretable Modeling. (a) General schematic of a Hybrid Model where, at inference time, a gating mechanism determines whether to send the instance to the interpretable component $h_s$ or to the complex one $h_c$. (b) Letting transparency be the ratio of samples sent to the interpretable component $h_s$, the trade-off between accuracy and transparency can be measured and compared across different Hybrid Models.

mechanisms are opaque and hard to understand by both experts and non-experts. Also, post-hoc explanations of these complex models have been shown to be unreliable and highly manipulable by ill-intentioned entities (Aïvodji et al., 2019; Slack et al., 2020; Dimanov et al., 2020; Laberge et al., 2022; Aïvodji et al., 2021). This conundrum between black-box or transparent designs is colloquially referred to as the "accuracy-transparency trade-off", that is, one has to choose between transparent models with lower performance or opaque models that perform well but whose explanations are not trustworthy. Still, this trade-off is not a quantitative measure but rather a part of the collective imagination of researchers in interpretable machine learning. For this reason, the accuracy-transparency trade-off has been heavily criticized and even labeled a myth (Rudin, 2019). But the question remains, does such a trade-off exist? And if it does, is there a way to quantitatively measure it? Or even optimize it?

To explore such questions, we will not treat black-box and transparent designs as dichotomies. Rather, we will embrace both and explore the continuum between the two philosophies. More specifically, we will study Hybrid Interpretable Models (Wang, 2019; Pan et al., 2020; Wang & Lin, 2021), which are systems that involve the cooperation of an interpretable model and a complex black box. At inference time, any input of the hybrid model is assigned to either its interpretable or complex component based on a gating mechanism, see Figure 1 (a). The intuition behind this type of modeling is that not all examples in a dataset are hard to classify.

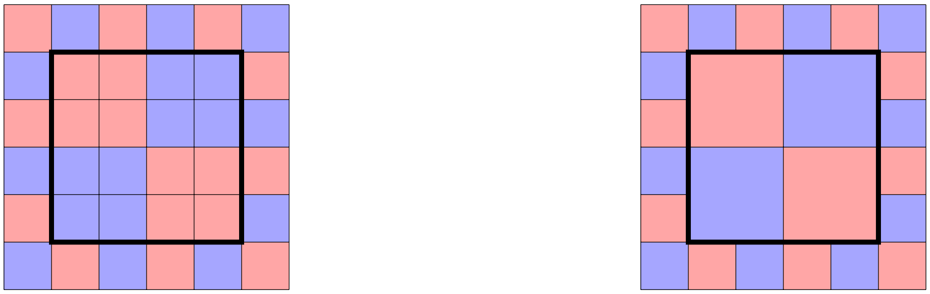
In line with the literature (Wang, 2019; Pan et al., 2020; Wang & Lin, 2021), we define the system's transparency as the ratio of samples that are sent to the interpretable part. The higher the transparency, the more model predictions one can actually understand and possibly certify. However, it is possible that the interpretable component makes more errors on average meaning that the overall system suffers a performance loss. Therefore, an integral part of hybrid modeling is to empirically explore the accuracy-transparency trade-off and find the best compromises, see Figure 1 (b). We note that the accuracy-transparency trade-off becomes something we measure and optimize. This is why we believe Hybrid Models are a very interesting research direction in the quest for interpretable machine learning.

Still, despite their high potential, hybrid models remain under-studied and under-used in the interpretability/explainability literature. One of the reasons for this under-exploration could be that learning interpretable models is very hard (often NP-Hard), and fitting a Hybrid Model on top can only be harder. To address this issue, past studies have optimized such models using local search heuristics (Wang, 2019; Pan et al., 2020). Nevertheless, we show in this study that the inherent stochasticity of these local search algorithms hinders the ability of practitioners to consistently attain a target level of transparency. Simply put, hybrid models are currently not user-friendly enough to promote widespread study and application.

Given the recent development of highly efficient libraries for training interpretable models to optimality ( *e.g.*, CORELS for rule-lists (Angelino et al., 2017), GOSDT for decision trees (Hu et al., 2019))), we believe it is now possible to practically train Hybrid Models to optimality, even when adding a hard constraint on transparency level.

To sensitize the community to the immediate potential of hybrid models and to encourage additional research, we offer a fundamental investigation of such models from three perspectives: Theory, Taxonomy, and Methods. From the theory point of view, we explore Probably-Approximately-Correct (PAC) generalization guarantees of hybrid models. A consequence of our PAC guarantee is the existence of a *sweet spot* for the optimal transparency of the system. When such a sweet spot is attained, a hybrid model can potentially perform better than a standalone black box. Secondly, we provide a general taxonomy for the different ways of training hybrid models: the *Post-Black-Box* and *Pre-Black-Box* paradigms. These approaches differ in the order in which the interpretable and complex components are trained. We show where state-of-the-art hybrid models fall in this taxonomy. Thirdly, we implement the two paradigms in a single method: HybridCORELS, which extends the library CORELS to hybrid modeling. By leveraging CORELS, HybridCORELS provides a certificate of optimality of its interpretable component and precise control over transparency. We finally show empirically that HybridCORELS is competitive with existing hybrid models, and performs just as well as a standalone black box (or even better) while being partly transparent. To resume, our contributions are:

- We theoretically study hybrid models under the PAC-Learning framework and derive generalization bounds. We show that such bounds depend on the amount of data classified by each part of the hybrid model and that an optimal transparency value exists.
- We introduce a taxonomy of hybrid models' learning methods, identifying two main families: the *Pre-Black-Box* paradigm and the *Post-Black-Box* paradigm. We instantiate the proposed *Pre-Black-Box* paradigm with a generic framework, using a key notion of *black-box specialization via re-weighting.* In a nutshell, it leverages the fact that in the *Pre-Black-Box* paradigm, the part of the input space handled by the black-box component is already defined before it is trained. One can then specialize it in such region by assigning larger weights to the appropriate training examples.
- We review state-of-the-art methods for learning hybrid models, and show that they all fall into the *Post-Black-Box* category.
- We modify a state-of-the-art algorithm for learning optimal sparse rule lists, named CORELS. More precisely, we propose a method for learning rule-based hybrid models with the *Post-Black-Box* paradigm. Our method, called HybridCORELS$_{\text{Post}}$, is the first to provide optimality guarantees and explicit control of the model *transparency.*
- We propose another modified version of CORELS for learning rule-based hybrid models with the *Pre-Black-Box* paradigm. This method, named HybridCORELS$_{\text{Pre}}$, is the first one using the proposed framework for the *Pre-Black-Box* paradigm. Again, it provides optimality guarantees and explicit control of the model *transparency.*
- We empirically show, using the proposed HybridCORELS$_{\text{Pre}}$ algorithm, that the *Pre-Black-Box* paradigm is suitable for learning accurate hybrid models with transparency constraints.
- We empirically compare both HybridCORELS$_{\text{Pre}}$ and HybridCORELS$_{\text{Post}}$ with state-of-the-art methods for learning rule-based hybrid models. We show that both methods offer competitive trade-offs between *accuracy* and *transparency*, while also providing facilitated control over the latter, and optimality guarantees.



(a) Example of a region $\Omega$ (shown as a thick square) where a complex model $h_c \in \mathcal{H}_c$ (with $|\mathcal{H}_c| = 2^{36}$) is overly complex.

(b) The complex model $h_c$ can be replaced by a simpler one $h_s \in \mathcal{H}_s$ (with $|H_s| = 2^4$). Overall, this hybrid model space has size $|\mathsf{Hyb}| = 2^{24}$.

Figure 2: Toy example with $\mathcal{X} = [0,1] \times [0,1]$. Here the complex models $\mathcal{H}_c$ are all the ways to color the 36 width-1 squares. The simpler models $\mathcal{H}_s$ are all the ways to color the 4 width-2 squares in the middle.

## 2 Hybrid Interpretable Models: a Theoretical Analysis

In this section, we formally introduce hybrid interpretable models and analyze them under the PAC-Learning framework. We derive generalization bounds and show that an optimal trade-off between accuracy and transparency (the proportion of data classified by the interpretable component) exists, leveraging the advantages of both parts of the model.

### 2.1 Definitions

Let $\mathcal{X}$ be the input space and let $\mathcal{H}_c, \mathcal{H}_s$ be two sets of binary classifiers $h : \mathcal{X} \to \{0,1\}$. We shall impose that $|\mathcal{H}_s| < |\mathcal{H}_c| < \infty$ so that $\mathcal{H}_s$ represents a simple set of models while $\mathcal{H}_c$ represents a complex set of models. Finally, we let $\mathcal{P}$ be a set of subsets of $\mathcal{X}$ (for instance, $\mathcal{P}$ may be the power set of $\mathcal{X}$, or the set of linear half-spaces). The intuition behind hybrid modeling is that there may exist a region $\Omega \in \mathcal{P}$ where a complex model $h_c \in \mathcal{H}_c$ is overkill and hence it could be replaced by a simpler model $h_s \in \mathcal{H}_s$ on that region without significant loss in terms of classification performance. Formally, a hybrid model is a triplet $\langle h_c, h_s, \Omega \rangle \in \mathsf{Hyb} := \mathcal{H}_c \times \mathcal{H}_s \times \mathcal{P}$ which instantiates a function of the form

$$\forall \boldsymbol{x} \in \mathcal{X}, \quad \langle h_c, h_s, \Omega \rangle(\boldsymbol{x}) = \begin{cases} h_s(\boldsymbol{x}) & \text{if } \boldsymbol{x} \in \Omega, \\ h_c(\boldsymbol{x}) & \text{otherwise.} \end{cases}$$

Figure 2 presents an informal argument favoring this modeling choice. We will additionally assume that the smaller hypothesis space $\mathcal{H}_s$ involves models that are *interpretable by design* such as rule lists, sparse decision trees, scoring systems, etc. This assumption will not affect the theoretical analysis, which will just rely on $|\mathcal{H}_s|$ being small, but it will specify the desiderata of the hybrid model. Indeed, if $h_s$ is interpretable, then we would like the region $\Omega$ on which it operates to be as big as possible without hindering performance. Letting $\mathcal{D}$ be a distribution over $\mathcal{X} \times \{0,1\}$ that represents a specific binary classification task, we want the *transparency* $C_\Omega := \mathbb{P}_{\boldsymbol{x} \sim \mathcal{D}}[\boldsymbol{x} \in \Omega]$ to be as large as possible.

The rest of this section is structured as follows: in **Section 2.2** we prove that finite hybrid models (*i.e.,* $|\mathsf{Hyb}| < \infty$) are PAC-Learnable. That is if we learn a hybrid model on a finite dataset with sufficiently many examples, then we can guarantee that the model will generalize to new unseen samples. This is an important first step in the fundamental understanding of hybrid models. Afterward, in **Section 2.3**, we study the impact of transparency on the looseness of the bound and show that a "sweet spot" for transparency exists.

### 2.2 Finite Hybrid Models are PAC-Learnable

We are going to study distributions $\mathcal{D}$ where a perfect model $\langle h_c^\star, h_s^\star, \Omega^\star\rangle \in \mathsf{Hyb}$ exists:

$$\mathcal{L}_\mathcal{D}(\langle h_c^\star, h_s^\star, \Omega^\star\rangle) := \underset{(\boldsymbol{x},y)\sim\mathcal{D}}{\mathbb{P}}[\langle h_c^\star, h_s^\star, \Omega^\star\rangle(\boldsymbol{x}) \neq y] = 0. \tag{1}$$

Intuitively, the predictions of the optimal hybrid interpretable model $\langle h_c^\star, h_s^\star, \Omega^\star\rangle$ match the true label $y$ of every example $(\boldsymbol{x}, y) \in (\mathcal{X} \times \{0,1\})$ drawn from distribution $\mathcal{D}$. To learn such a model, we can employ the Empirical Risk Minimization (ERM) principle, which consists of sampling a dataset of $M$ iid examples $S := \{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^M \sim \mathcal{D}^M$, defining the empirical risk

$$\widehat{\mathcal{L}}_S(\langle h_c, h_s, \Omega\rangle) := \sum_{i=1}^M \mathbb{1}[\langle h_c, h_s, \Omega\rangle(\boldsymbol{x}^{(i)}) \neq y^{(i)}], \tag{2}$$

and minimizing it across $\mathsf{Hyb}$

$$\langle h_c, h_s, \Omega\rangle_S := \mathrm{ERM}_{\mathsf{Hyb}}(S) = \underset{\langle h_c, h_s, \Omega\rangle\in\mathsf{Hyb}}{\arg\min}\ \widehat{\mathcal{L}}_S(\langle h_c, h_s, \Omega\rangle).$$

One can notice that we do not scale the empirical risk by a factor $\frac{1}{M}$ as multiplication by a constant factor does not affect ERM. The following theoretical result characterizes the generalization of hybrid models learned with ERM.

**Theorem 1** *Given a finite hybrid model space* ($|\mathsf{Hyb}| < \infty$) *and some* $\epsilon > 0$*, letting* $C_\Omega := \mathbb{P}_{\boldsymbol{x}\sim\mathcal{D}}[\boldsymbol{x} \in \Omega]$ *be the transparency of* $\Omega$*, then for any distribution* $\mathcal{D}$ *where there exists a triplet* $\langle h_c^\star, h_s^\star, \Omega^\star\rangle$ *with zero generalization error (as defined in (1)), the following holds for a training set of size* $M$*:*

$$\underset{S\sim\mathcal{D}^M}{\mathbb{P}}[\mathcal{L}_\mathcal{D}(\langle h_c, h_s, \Omega\rangle_S) > \epsilon] \leq \sum_{\Omega\in\mathcal{P}} \mathcal{B}(\epsilon, C_\Omega, \mathcal{H}_c, \mathcal{H}_s, M),$$

*where*

$$\mathcal{B}(\epsilon, C_\Omega, \mathcal{H}_c, \mathcal{H}_s, M) := (1 - |\mathcal{H}_c|)C_\Omega^M + (1 - |\mathcal{H}_s|)C_{\overline{\Omega}}^M + |\mathcal{H}_c|(C_{\overline{\Omega}}e^{-\epsilon} + C_\Omega)^M + |\mathcal{H}_s|(C_\Omega e^{-\epsilon} + C_{\overline{\Omega}})^M.$$

*If we assume that the optimal subset* $\Omega \equiv \Omega^\star$ *is known in advance, then the bound tightens*

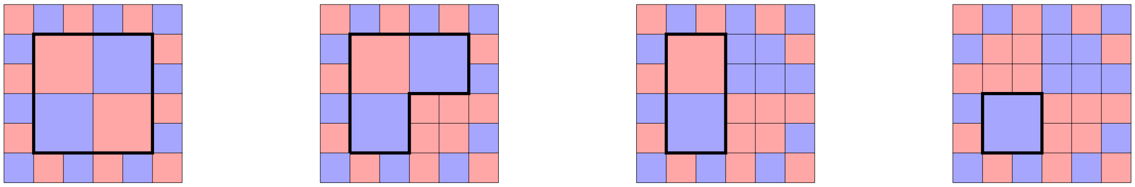
$$\underset{S\sim\mathcal{D}^M}{\mathbb{P}}[\mathcal{L}_\mathcal{D}(\langle h_c, h_s, \Omega\rangle_S) > \epsilon] \leq \mathcal{B}(\epsilon, C_\Omega, \mathcal{H}_c, \mathcal{H}_s, M). \tag{3}$$

*Important point:* $\mathcal{B}(\epsilon, C_\Omega, \mathcal{H}_c, \mathcal{H}_s, M)$ *is positive despite the presence of the negative terms* $(1 - |\mathcal{H}_c|)C_\Omega^M + (1 - |\mathcal{H}_s|)C_{\overline{\Omega}}^M$*. Indeed, these terms are outweighed by the remaining two terms, which are larger and positive.*

**Proof** The complete proof is provided in Appendix A. ■

This generalization bound involves several key quantities: the amount of data $M$, the transparency $C_\Omega$ and its complement $C_{\overline{\Omega}}$ as well as the complexities of the hypothesis spaces $|\mathcal{H}_s|$ and $|\mathcal{H}_c|$. We will see in the coming subsection how these various parameters impact the looseness of the bound.

We note some of the limitations of these theoretical bounds. First, taking $C_\Omega = 0$, we obtain a trivial bound $1 + |\mathcal{H}_c|e^{-\epsilon M}$. The same thing occurs when setting $C_\Omega = 1$. Basically, the bound is trivial unless input samples are shared between the complex and simple models. Secondly, the bound requires the knowledge of transparency $C_\Omega := \mathbb{P}_{\boldsymbol{x}\sim\mathcal{D}}[\boldsymbol{x} \in \Omega]$ which cannot be computed exactly in practice since the data-generating distribution $\mathcal{D}$ is unknown. The only way to practically estimate this quantity is to count how many data instances land in the region $\Omega$. Thirdly, the bound is loose as its computation relies on applying the union bound repeatedly over $\mathcal{P}$, $\mathcal{H}_c$, and $\mathcal{H}_s$. Still, for $C_\Omega \in ]0,1[$, and any $\epsilon \in ]0,1]$ the bound decreases as $M$ increases which implies that learning hybrid models is possible in theory.



(a) Region $\Omega_1$ (b) Region $\Omega_2$ (c) Region $\Omega_3$ (d) Region $\Omega_4$

Figure 3: Four hybrid models that are functionally equivalent but have different regions $\Omega$.

### 2.3 Fine-Tuning the Transparency

A particular property of hybrid models is that the optimal model $\langle h_c^\star, h_s^\star, \Omega^\star \rangle$ from Equation (1) need not be unique. Indeed, given the flexibility of choosing the region $\Omega$ on which the simple model is applied, we could have two hybrid models with the same functional output. Figure 3 presents a toy example of four hybrid models that are all functionally equivalent but with different regions $\Omega$.

Now the hypothesis that the optimal region $\Omega^\star$ is known in advance could be replaced with the knowledge of a set of optimal regions $\{\Omega_i^\star\}_{i=1}^R$. If such is the case, which region should be returned by the learning algorithm? Using the empirical error as a criterion would not work since any ERM fitted using these optimal regions would return an error of 0. We propose to leverage the theoretical bound to decide which region to employ. Specifically, if we fix some region $\Omega_i^\star$ for the ERM algorithm, then Equation (3) provides a bound $\mathcal{B}(\epsilon, C_\Omega, \mathcal{H}_c, \mathcal{H}_s, M)$ on the probability of having an error that exceeds $\epsilon$ for any $\epsilon \in ]0, 1]$. Taking the average value of the bound highlights its looseness

$$\overline{\mathcal{B}}(C_{\Omega_i^\star}, \mathcal{H}_c, \mathcal{H}_s, M) := \int_0^1 \mathcal{B}(\epsilon, C_{\Omega_i^\star}, \mathcal{H}_c, \mathcal{H}_s, M) d\epsilon \quad \forall i = 1, 2, \ldots, R.$$

Hence, by studying $\overline{\mathcal{B}}(C_{\Omega_i^\star}, \mathcal{H}_c, \mathcal{H}_s, M)$ as a function of $C_{\Omega_i^\star}$, one can theoretically decide which region to use in the final model.

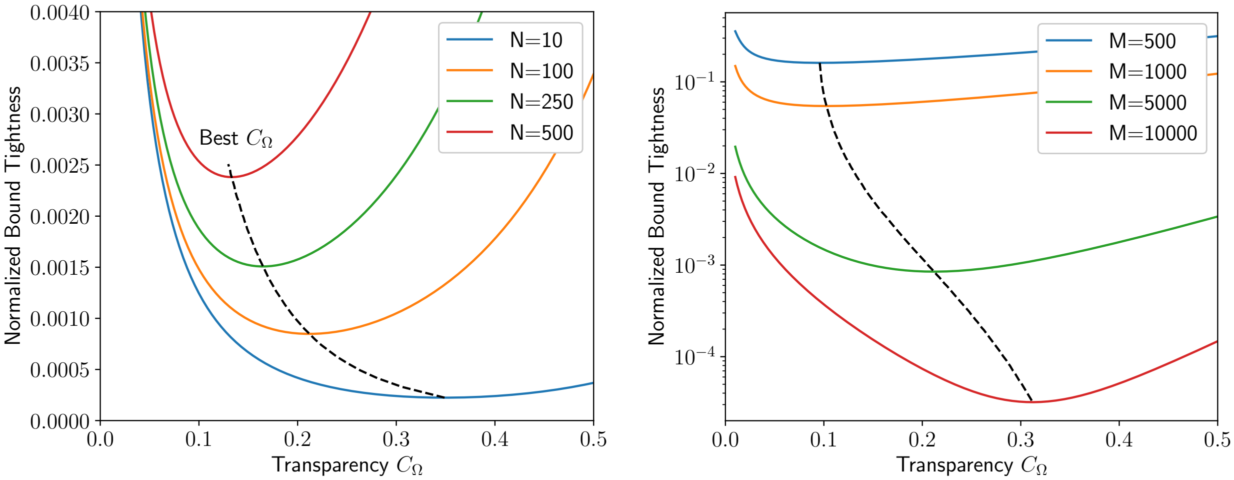
In the following example, we have defined $\mathcal{H}_s$ as the set of all binary depth-3 decision trees (7 internal nodes and 8 leaves with binary outcomes) fitted on 200 binary features ($\mathcal{X} = \{0, 1\}^{200}$). This hypothesis space has a size $|\mathcal{H}_s| = 2^8 \times 200 \times 199^2 \times 198^4 \approx 3.11 \times 10^{18}$. We have defined $\mathcal{H}_c$ to be any hypothesis space that is larger than $\mathcal{H}_s$ by some factor $|\mathcal{H}_c| = N \times |\mathcal{H}_s|$.

Figure 4 presents the looseness of the generalization bound as a function of the transparency for this hypothetical example. We observe that, given $\mathcal{H}_c$, $\mathcal{H}_s$, and $M$, there is a "sweet spot" where the bound on error is the tightest

$$\Omega^{\star\star}(\mathcal{H}_c, \mathcal{H}_s, M) = \underset{i=1,2,\ldots,R}{\arg\min}\, \overline{\mathcal{B}}(C_{\Omega_i^\star}, \mathcal{H}_c, \mathcal{H}_s, M).$$

Looking more specifically at Figure 4 (a), increasing $N$ reduces the transparency that reaches the minimal looseness. This means that the more complex $\mathcal{H}_c$ is, the more input samples must be sent to train $h_c$ so it does not overfit. Inspecting Figure 4 (b), the optimal transparency increases as $M$ increases. This means that as we reach large values of $M$, we can afford to train the black box on a smaller ratio of the data without over-fitting.

We conclude this example by emphasizing that Figure 4 is mostly of theoretical interest so practitioners must take it with a grain of salt. More precisely, the exact values of the "sweet spot" for transparency are not indicative of the values one would obtain in real-life experiments. This is because our analysis is performed on a loose upper bound which we hope still captures the generalization dynamics of hybrid models. In real-life applications, the existence of an optimal transparency must be assessed experimentally. Still, the theory suggests the existence of such a "sweet spot", which in itself is an interesting result.



(a) Varying $N$ with $M = 5000$.

(b) Varying $M$ with $N = 100$.

Figure 4: Normalized Bound Looseness (*i.e.*, $\overline{\mathcal{B}}/|\mathcal{H}_s|$) of the theoretical upper bound as a function of transparency $C_\Omega$. We observe a "sweet spot" with minimal Looseness which depends on $N$ (ratio of the hypothesis spaces' sizes $\frac{|\mathcal{H}_c|}{|\mathcal{H}_s|}$) and $M$ (size of the training dataset).

### 2.4 Takeaways

Although the bound makes strong assumptions that may not hold in practical applications, our theoretical analysis leads to fundamental insights into training hybrid models:

1. Training hybrid interpretable models is theoretically possible given enough data.
2. Important parameters that influence generalization are the complexities $|\mathcal{H}_s|$ and $|\mathcal{H}_c|$, the transparency $C_\Omega$, and the number of data points $M$.
3. There exists a "sweet spot" of the bound in terms of transparency which varies with $\mathcal{H}_c$, $\mathcal{H}_s$, and $M$. Henceforth, in practical applications, we should sweep over possible values of transparency. Some of the resulting hybrid models may attain better generalization.

## 3 Learning Hybrid Interpretable Models: Taxonomy and Methods

We now introduce our proposed taxonomy of hybrid models learning frameworks. We then show how rule-based classifiers can be used to implement hybrid interpretable models. Finally, we position state-of-the-art methods within the proposed taxonomy.

### 3.1 Taxonomy of Hybrid Models Learning Frameworks

A major challenge in training hybrid models is that two models must be trained instead of one. Given the proliferation of out-of-the-box implementations of complex model $h_c$, such as `Scikit-Learn` and `XGBoost` classifiers, it would be simpler to rely on them via their pre-existing `fit` and `predict` methods. Henceforth, we encourage hybrid model training procedures to be *agnostic* to the type of black box $h_c$. This makes hybrid models a lot more versatile and user-friendly because any practitioner could just plug in their favorite black box implementation.

Given the technical constraint of black box agnosticism, we now leverage the previous PAC generalization bound to derive a learning objective. As a reminder, in previous section defined the data-generating distribution $\mathcal{D}$, the training set $S := \{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^M \sim \mathcal{D}^M$, and the transparency $C_\Omega := \mathbb{P}_{\boldsymbol{x}\sim\mathcal{D}}[\boldsymbol{x} \in \Omega] \approx |S \cap \Omega|/|S|$.

It was also demonstrated that two important quantities guaranteeing generalization are the complexity of the simple hypothesis space $\mathcal{H}_s$ and the transparency $C_\Omega$. Since "smaller is better" in any learning objective, we should actually minimize the complement of transparency $C_{\overline{\Omega}} = (1 - C_\Omega) \approx |S \cap \overline{\Omega}|/|S|$. A general regularized learning objective would then be

$$\mathsf{obj}(\langle h_c, h_s, \Omega\rangle, S) = \frac{\widehat{\mathcal{L}}_S(\langle h_c, h_s, \Omega\rangle)}{|S|} + \lambda \cdot K_{\mathcal{H}_s} + \beta \cdot \frac{|S \cap \overline{\Omega}|}{|S|}, \tag{4}$$

where $K_{\mathcal{H}_s}$ is a complexity measure of $\mathcal{H}_s$ and $\lambda, \beta \geq 0$ are regularization hyper-parameters that respectively control the cost of increasing the complexity of $\mathcal{H}_s$ (for instance considering depth), and that of increasing the black box part coverage $C_{\overline{\Omega}}$ (equivalently decreasing the interpretable part coverage $C_\Omega$, which constitutes the model's transparency).

Equation (4) presents the learning of hybrid models in its most abstract form and we shall make it more specific shortly. We first present several ways to minimize the objective over the space $\mathsf{Hyb} = \mathcal{H}_c \times \mathcal{H}_s \times \mathcal{P}$ that differ on the order in which the simple $h_s$ and the black box $h_c$ parts are trained.

### 3.1.1 The *Post-Black-Box* Paradigm: Wrapping an Interpretable Model around the Complex One

A common approach encompassing all state-of-the-art methods for learning hybrid models consists in training a black box first and then wrapping an interpretable model on top of it. We coin this strategy as the *Post-Black-Box* paradigm. In this setting, the interpretable components $h_s$ and $\Omega$ can be seen as a way to simplify the model in regions where it is overkill. A key advantage of this paradigm is that a user owning a pre-trained black box with high performance can easily wrap an interpretable model on top of it to get an increase of transparency (and possibly a generalization improvement as suggested by our theoretical analysis of **Section 2.3**). Furthermore, the interpretable part of the hybrid model is able to correct the mistakes made by the black box, as its predictions are known in advance. We illustrate the *Post-Black-Box* paradigm in Figure 5 (Top).

### 3.1.2 The *Pre-Black-Box* Paradigm: black box Specialization by Reweighting

Another possibility for learning hybrid models consists in first learning the interpretable part of the model before training a black box model on the remaining examples. We label this approach *Pre-Black-Box*. The objective of the initial training of the interpretable part is to identify the easiest examples from the data and train a simple model on them. Then, the black box part will only have to classify the examples not sent to the simple part ($\boldsymbol{x} \notin \Omega$). Leveraging the black box complexity to specialize it on such part of the input space could hence lead to enhanced performances. However, it could also cause overfitting, especially when the interpretable part transparency is high (and the black box only deals with a small portion of the input space/a reduced number of examples). In our proposed framework, this issue is tackled by training the black box on a reweighted version of the entire training set, with weights

$$\forall i \in \{1, 2, \ldots, M\}, \quad w_i = \frac{e^{\alpha\, \mathbb{1}[\boldsymbol{x}^{(i)} \in \overline{\Omega}]}}{\sum_{j=1}^{M} e^{\alpha\, \mathbb{1}[\boldsymbol{x}^{(j)} \in \overline{\Omega}]}}, \tag{5}$$

that are higher for instances not classified by $h_s$. The non-uniform weights rely on a **specialization coefficient** $\alpha \geq 0$. The higher $\alpha$, the more the black box focuses on the data not captured by the interpretable part of the model. On the other hand, low values of $\alpha$ (*e.g.,* for $\alpha = 0$, all examples' weights are equal) lead to a more generalist black box model. Since this trade-off is non-trivial, the hyperparameter $\alpha$ will need to be fine-tuned in practice. Figure 5 (Bottom) illustrates the *Pre-Black-Box* paradigm pipeline. We note that many classifiers in the `Scikit-Learn` and `XGBoost` packages support non-uniform data weights in their training procedure. Hence, the *Pre-Black-Box* paradigm is also black box-agnostic.

This paradigm intrinsically comes with several drawbacks and advantages. On the one side, the *Pre-Black-Box* paradigm limits the possible collaboration between both parts of the hybrid model. Indeed, the interpretable part (characterized by $h_s$ and $\Omega$) is trained first, defining the data split with the black box part. Then, there is no possibility to redefine the data split between the two parts of the hybrid model in the

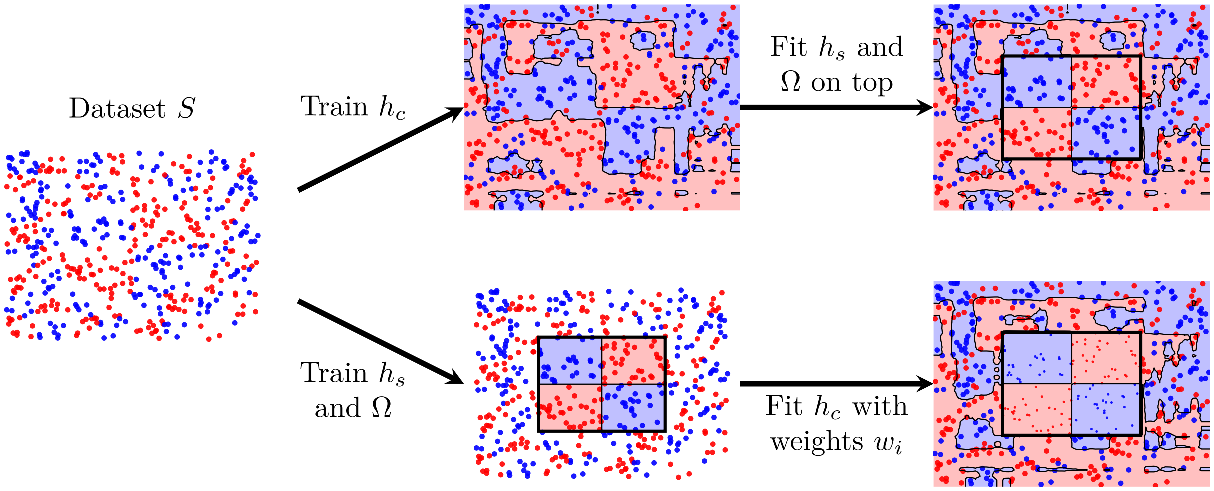


Figure 5: Two paradigms for learning hybrid models. (Top) In the *Post-Black-Box* paradigm, a black box is first trained on the whole dataset. Then, the interpretable components are fitted on top of the black box to simplify it in regions where it is overkill. (Bottom) In the *Pre-Black-Box* paradigm, the interpretable part of the model is trained to identify a region where the task is simple. Afterward, the black box model is fitted on the data with specialization weights $w_i$ to encourage high performance on instances outside of $\Omega$. Here the weights are visualized as the markers' size.

second phase of the learning (black box training). Consequently, there can be no correction of one part of the model's errors by the other, as was done in the *Post-Black-Box* paradigm. On the other side, because the data split is perfectly defined while training the black box, it is possible to adapt the black box training procedure to leverage its complexity and *specialize* it on its support region $\overline{\Omega}$.

### 3.1.3 Another Perspective: End-to-end Approach

Finally, a last possible strategy consists in training both parts of a hybrid model *simultaneously*. While this approach could theoretically provide the best performances (as it allows for a global optimality guarantee), it is also very challenging, as it requires encoding both the simple and black box parts of the hybrid model within a unified framework.

One key applicability advantage of both *Pre-Black-Box* and *Post-Black-Box* paradigms is their black box-agnostic nature: there is no limitation over the type of black box used nor its training procedure. This would not hold anymore in an end-to-end paradigm, and we let such an approach as an interesting avenue for future works.

In the coming subsection, we discuss how rule-based models can be used for implementing the triplet $\langle h_c, h_s, \Omega \rangle$.

## 3.2 Rule-Based Modeling

One of the important design choices of a hybrid model is the space $\mathcal{P}$ of possible subsets $\Omega$ where the interpretable model will operate. An example from previous work is to model these sets via thresholded linear models (Wang & Lin, 2021). An alternative way to encode the input subsets $\Omega$ is by employing a rule-based model $r$ (*e.g.,* a rule list or a rule set) and defining $\Omega_r$ as

$$\Omega_r := \{\boldsymbol{x} \in \mathcal{X} : \text{cover}(r, \boldsymbol{x}) = 1\},$$

where $\text{cover}(r, \boldsymbol{x}) = 1$ if $\boldsymbol{x}$ respects the condition in at least one of the rules in $r$ (we say that $\boldsymbol{x}$ is *captured* by $r$). The advantage of using rule-based models to partition the input space is that they are interpretable by design, hence they can also serve as the simple hypothesis space $\mathcal{H}_s$. That is, we can assign a label to an input depending on which rule captures it. Hereafter is an example of a hybrid model involving a rule list $r$ containing two rules.

```
if $18 \leq \text{Age} \leq 22$ and gender=male then
    return $y = 1$
else if Prior-Crimes $> 3$ then
    return $y = 1$
else
    return $h_c(\boldsymbol{x})$
```

Since a rule-based model encodes both the region $\Omega$ and the simple function $h_s$ on this region, we can think of rule-based hybrid models as a tuple $\langle h_c, r\rangle \in \mathcal{H}_c \times \mathcal{H}_s$ instead of a triplet $\langle h_c, h_s, \Omega\rangle$. The learning objective on the training set $S$ becomes

$$\mathsf{obj}(\langle h_c, r\rangle, S) = \frac{\widehat{\mathcal{L}}_S(\langle h_c, r\rangle)}{|S|} + \lambda \cdot |r| + \beta \cdot \frac{|S \cap \overline{\Omega}_r|}{|S|}, \tag{6}$$

where we measure the complexity of a rule-list (rule-set) $r$ by its length $|r|$.

### 3.3 Rule-Based *Post-Black-Box* Hybrid Models

Now that we have introduced several learning paradigms as well as a modeling choice for the hybrid model based on rules, we can describe two approaches in the literature that apply the *Post-Black-Box* paradigm with rule-sets and rule-lists.

#### 3.3.1 Hybrid Rule-Set (HyRS)

This hybrid model has been introduced by Wang (2019) and considers a rule set $r = r_+ \cup r_-$ that combines a set of positive rules $r_+$ and a set of negative rules $r_-$. The resulting hybrid model $\langle h_c, r\rangle$ takes the form of Figure 6.

```
if cover($r_+, \boldsymbol{x}$) then
    return 1
else if cover($r_-, \boldsymbol{x}$) then
    return 0
else
    return $h_c(\boldsymbol{x})$
```

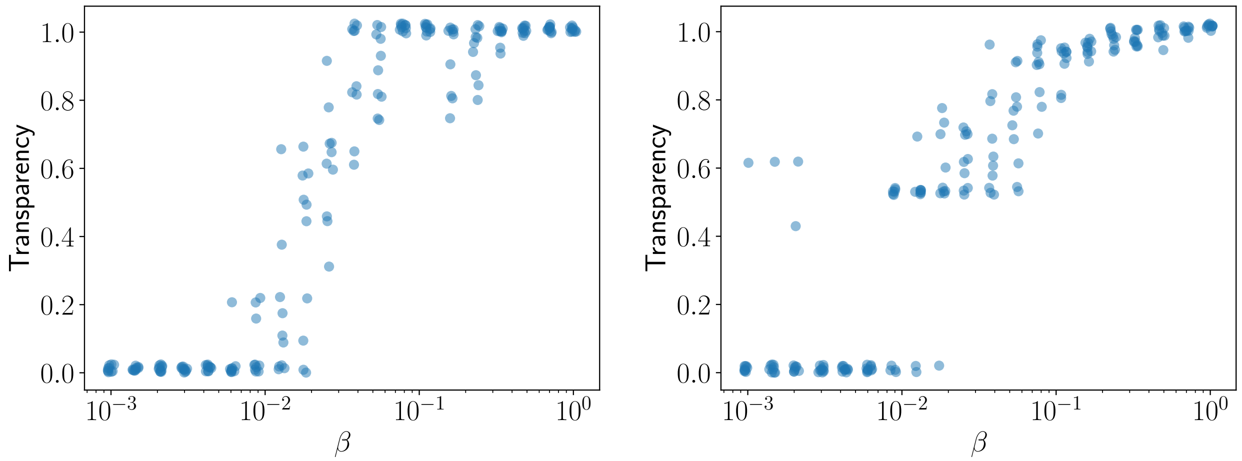
Figure 6: Hybrid Rule-Set.

The complexity of the interpretable model is the total number of rules $|r|$ and so the learning objective of Equation 6 is used. The minimization of this combinatorial problem is tackled by a local search algorithm where neighborhoods are defined as random perturbations of the rule-sets $r_+$ and $r_-$.

One of the drawbacks of HyRS is that the user does not have precise control over the transparency $C_\Omega$ of the resulting hybrid model. There are two design choices in HyRS that lead to this issue. First of all, the only way to control the desired transparency is to increase the hyper-parameter $\beta$ which will incentivize the rule sets to cover more examples. Still, because the objective is extremely complex, it is not clear what $\beta$ is high enough to ensure a certain level of transparency. Secondly, since the local search algorithm employed to find the rules is inherently stochastic, several runs of the training procedure with the same hyperparameters can lead to very different models and, by extension, different transparencies. Figure 7 shows different reruns of HyRS on two datasets for 20 different values of $\beta$ that span four orders of magnitude. We see that the relation between transparency and $\beta$ is hardly monotonic because of the variance between reruns. Moreover, the transparency does not vary smoothly w.r.t $\beta$ as seen in the UCI Adult Income dataset, where the transparency jumps from 0 to 0.5 at around $\beta = 10^{-2}$.

#### 3.3.2 Companion Rule-List (CRL)

An alternative method called Companion-Rule-List (CRL) has later been developed in order to address previous limitations (Pan et al., 2020). Notably, CLR streamlines the accuracy-transparency tradeoff by returning multiple hybrid models with increasing transparency instead of a single model. To encode multiple



(a) COMPAS dataset

(b) UCI Adult Income dataset

Figure 7: Instability of the transparency of HyRS for different random seeds. A small jitter was applied to the points to remove juxtapositions.

hybrid models, note that, given a rule list, one can insert the black box at any level of the **else-if** statements. For instance, Figure 8 presents three hybrid models $\langle h_c, r\rangle$ that are derived from the same list of three rules $r = [r_1, r_2, r_3]$.

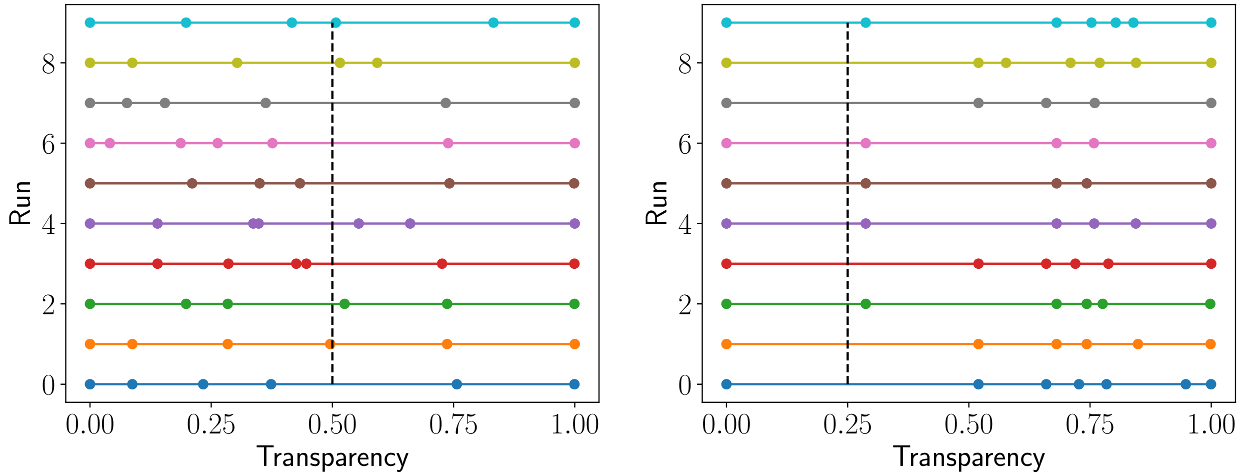
```
if cover($r_1, \boldsymbol{x}$) then
    return 1
else
    return $h_c(\boldsymbol{x})$
```

```
if cover($r_1, \boldsymbol{x}$) then
    return 1
else if cover($r_2, \boldsymbol{x}$) then
    return 0
else
    return $h_c(\boldsymbol{x})$
```

```
if cover($r_1, \boldsymbol{x}$) then
    return 1
else if cover($r_2, \boldsymbol{x}$) then
    return 0
else if cover($r_3, \boldsymbol{x}$) then
    return 1
else
    return $h_c(\boldsymbol{x})$
```

Figure 8: A rule list $r = [r_1, r_2, r_3]$ encodes three hybrid models $\langle h_c, r\rangle$ with increasing transparency (from left to right).

Returning multiple hybrid models allows users to decide what hybrid model to use based on their desired transparency. The training objective of CRL is no longer the accuracy but rather the Area-Under-the-Curve (AUC) of the accuracy-transparency curve of the different hybrid models. A regularization $\lambda \cdot |r|$ is also added to the objective to avoid long rule-lists. Similarly to HyRS, CRL is trained with a local search algorithm where neighbourhoods are defined as random perturbations of the rule-list $r$. Although CRL offers more possibilities for transparency, we find that the inherent stochasticity of the learning procedure still hinders the ability to consistently reach target transparency. Figure 9 presents simple experiments conducted on the COMPAS and UCI Adult Income where a CRL model was fitted for 10 different random seeds. We present the different levels of transparency attained by each run. For the COMPAS dataset, if a user wishes for a transparency of at least 0.5, then on half of the runs, they would need to go up to about 0.75 transparency using the CRL framework (which may excessively conflict with predictive accuracy). For the UCI Adult Income dataset, if an end-user requires transparency of at least 0.25, then on half of the runs, they would need to go up to 0.5 transparency. These experiments highlight that CRL does not provide full control over the desired level of transparency of the hybrid models.



(a) COMPAS dataset

(b) UCI Adult Income dataset

Figure 9: Instability of the transparency of CRL for different random seeds each indicated by a different color. The dots represent the transparency attained by the many hybrid models returned from a single run of CRL.

## 4 HybridCORELS: Learning Optimal Hybrid Interpretable Models

We now present our methods for learning optimal hybrid models. First, we introduce the CORELS algorithm, which was proposed to learn optimal rule lists. Then, we describe the integration of the transparency requirements within our proposed methods. Finally, we propose HybridCORELS$_{\text{Post}}$ (resp. HybridCORELS$_{\text{Pre}}$), a modified version of CORELS to learn optimal hybrid models following the *Post-Black-Box* (respectively, *Pre-Black-Box*) framework.

### 4.1 Learning Optimal Rule Lists: the CORELS Algorithm

Rule lists are interpretable classifiers formed by an ordered list of if-then rules $r$, followed by a default prediction $q_0$ (Rivest, 1987). The set of ordered rules preceding the default prediction is called a prefix. One can observe that any rule list $d = (r, q_0)$ represents a classification function, while any prefix $r$ defines a *partial* classification function, defined within its support $\Omega_r$ (examples matching at least one of the rules within $r$) .

To learn Certifiable Optimal RulE ListS, Angelino et al. (2017) proposed CORELS, a branch-and-bound algorithm. It represents the search space of rule lists using a prefix tree, in which each node corresponds to a prefix $r$. Adding a default prediction $q_0$ to $r$ allows the building of a rule list $d = (r, q_0)$. In CORELS' prefix tree, the children nodes of $r$ correspond to prefixes formed by adding exactly one rule at the end of $r$. Thus, the $r$-rooted sub-tree corresponds to all possible extensions of $r$. CORELS' objective function for rule list $d = (r, q_0)$ on dataset $S$ is a weighted sum of classification error and sparsity:

$$\mathsf{obj}(d, S) = \frac{\widehat{\mathcal{L}}_S(d)}{|S|} + \lambda \cdot |r| \tag{7}$$

where $\widehat{\mathcal{L}}_S(d)$ measures the number of errors (incorrect classifications) made by $d$ on $S$ (as defined in (2)), and $|r|$ is the length (number of rules) of rule list $d$'s prefix $r$.

Let $S_r = S \cap \Omega_r$ be the subset of $S$ made of all examples of $S$ captured by some prefix $r$ (*i.e.*, the examples classified by $r$'s partial classification function). Just like any branch-and-bound algorithm, CORELS uses an objective lower bound to prune the prefix tree, and eventually guide the search in a best-first search fashion. For each node of the prefix tree (corresponding to a prefix $r$), it measures the best objective function value that may be reached by extending prefix $r$. If this value is worse than the best solution (rule list) known so far, then the $r$-rooted sub-tree can be pruned safely. Let $\widehat{\mathcal{L}}_{S_r}(r)$ counts the number of mistakes made by prefix $r$ (measured on its support set $S_r$), and $\mathsf{incons}(S)$ denote the minimum number of examples of $S$ that can never be classified correctly, because they have the exact same features vector as some other examples, but with a different label (due to potential dataset inconsistencies). CORELS' objective lower bound for prefix $r$ on dataset $S$ is then computed as follows:

$$\mathsf{lb}(r, S) = \frac{\widehat{\mathcal{L}}_{S_r}(r) + \mathsf{incons}(S \setminus S_r)}{|S|} + (|r| + 1) \cdot \lambda \tag{8}$$

Intuitively, $\widehat{\mathcal{L}}_{S_r}(r) + \mathsf{incons}(S \setminus S_r)$ corresponds to the minimum number of errors that any extension of $r$ can make, given the errors made by $r$ and the errors that can not be avoided due to data inconsistency.

CORELS uses several efficient data structures to speed up the computation by breaking down symmetries (Angelino et al., 2017). For instance, a prefix permutation map ensures that only the most accurate permutation of every set of rules is kept. These data structures are still valid in our setup. Finally, we can leverage the efficiency of the CORELS' machinery to learn optimal hybrid models, by only modifying CORELS' objective function and providing a valid lower bound on the new objective function. For reference, we provide the pseudo-code of the branch-and-bound underlying CORELS within Algorithm 1 in Appendix B.2. In particular, our modified algorithms will only learn prefixes (which will constitute the interpretable parts of our hybrid models), and hence will never care about the default prediction. In sections 4.3 and 4.4, we show how the objective function (7) and its lower bound (8) can be modified to learn hybrid models implementing the *Post-Black-Box* and *Pre-Black-Box* paradigms (respectively).

### 4.2 Ensuring a User-Defined Transparency Level

State-of-the-art methods for learning hybrid models integrate transparency requirements using a regularization term, as described in sections 3.3.1 and 3.3.2. However, this approach does not allow the user to have a precise control over the desired transparency level, and several runs with the exact same hyperparameters but different random seeds can lead to hybrid models with significantly different transparency levels. To address this issue, we build on the flexibility of the branch and bound algorithm underlying CORELS and integrate transparency as a hard constraint, stating that the learnt prefix $r$ must capture at least a proportion of $min\_transp \in [0, 1]$ of the examples within dataset $S$:

$$\frac{|S_r|}{|S|} \geq min\_transp \tag{9}$$

where, as aforementioned, $S_r = S \cap \Omega_r$ is the subset of $S$ made of all examples of $S$ captured by prefix $r$. Both our proposed approaches implement this hard-constraint approach. It allows for the building of hybrid models whose transparency (on the training set) is guaranteed to be at least $min\_transp$. To the best of our knowledge, our approach is the first to implement such direct control of the transparency level. Compared to state-of-the-art hybrid learning methods (which use a regularization term to encourage transparency), this approach allows for a tight control of the desired transparency, which can help build denser sets of tradeoffs between transparency and utility using $\epsilon$-constrained methods. To enforce constraint (9) using the CORELS branch-and-bound algorithm, we simply modify the best solution update subroutine, to only perform the update operation if the candidate prefix satisfies the transparency requirement. This guarantees that any returned solution will satisfy (9) while maintaining optimality as the exploration and bounds are not modified.

Even if constraint (9) ensures the strict respect of a user-defined transparency level, we also integrate transparency using a regularization term. This allows to break ties: if two models exhibit the same accuracy and sparsity levels, then this regularization term will favor the one with higher transparency. In practice, we set the associated regularization coefficient $\beta$ to a value small enough to only break ties. Indeed, because it is already enforced through hard constraint (9), we do not want transparency to trade-off with accuracy nor sparsity in the objective function (*i.e.,* we will always prefer any non-zero improvement on the accuracy or sparsity term over any improvement on the transparency term). Just like in constraint (9), transparency is measured using $\frac{|S_r|}{|S|} \in [0,1]$ (as $S_r \subseteq S$). Thus, we penalize (un)transparency as $\frac{|S \setminus S_r|}{|S|} \in [0,1]$ in the objective function, and set $\beta < \frac{1}{|S|} \leq \lambda$ for both approaches. Finally, our objective functions (10) and (11) both add this $(\beta \cdot \frac{|S \setminus S_r|}{|S|})$ term.

### 4.3 *Post-Black-Box* framework: HybridCORELS$_{\text{Post}}$

We now introduce HybridCORELS$_{\text{Post}}$, a modified version of the CORELS algorithm to produce optimal hybrid models using the state-of-the-art *Post-Black-Box* paradigm. More precisely, HybridCORELS$_{\text{Post}}$ first trains a black-box model (or takes as input a pre-trained black-box model). This first step is totally agnostic to the type of black-box and its training algorithm. Then, given a minimum transparency constraint (9), it builds a prefix optimizing the overall model's accuracy and sparsity.

**Objective** Given a black-box $h_c$'s training set predictions, HybridCORELS$_{\text{Post}}$ builds a prefix $r$ capturing at least a proportion of *min_transp* of the training data (transparency constraint (9)), and minimizing the following objective function:

$$\begin{aligned}\mathsf{obj}_{\text{post}}(r,S) &= \frac{\widehat{\mathcal{L}}_S(\langle h_c, r\rangle)}{|S|} + \lambda \cdot |r| + \beta \cdot \frac{|S \setminus S_r|}{|S|} \\ &= \frac{\widehat{\mathcal{L}}_{S_r}(r) + \widehat{\mathcal{L}}_{S\setminus S_r}(h_c)}{|S|} + \lambda \cdot |r| + \beta \cdot \frac{|S \setminus S_r|}{|S|}.\end{aligned} \tag{10}$$

Here, $\frac{\widehat{\mathcal{L}}_{S_r}(r)+\widehat{\mathcal{L}}_{S\setminus S_r}(h_c)}{|S|}$ is the exact accuracy of the overall hybrid model. Indeed, because the black-box predictions are fixed, the interpretable part is able to correct the mistakes made by the pre-trained black-box model.

**Objective lower bound** CORELS' original objective lower bound (8) (leveraging both the prefix's errors and the inconsistent examples among the uncaptured ones) is still valid and tight in this setup, so we do not need to modify it. Indeed, the error term lower bound $\widehat{\mathcal{L}}_{S_r}(r) + \mathsf{incons}(S \setminus S_r)$ is unchanged, as all remaining black-box errors $\widehat{\mathcal{L}}_{S\setminus S_r}(h_c)$ may potentially be corrected by extending $r$, but the errors already made by prefix $r$ and those related to remaining inconsistencies can not be avoided. Then, the transparency regularization term can not be used within the objective lower bound, as this term can always reach 0.0 by sufficiently extending prefix $r$. Finally, the lower bound over the sparsity regularization term still holds: any extension of prefix $r$ must have at least $|r| + 1$ rules.

Finally, HybridCORELS$_{\text{Post}}$ is an exact method: it provably returns a prefix $r$ for which $\mathsf{obj}_{post}(r,S)$ (10) is the smallest among those satisfying the transparency constraint (9). This means that, given fixed black-box predictions and desired transparency level, it produces an optimal hybrid interpretable model in terms of accuracy/sparsity. We provide a detailed pseudo-code of HybridCORELS$_{\text{Post}}$ in Algorithm 2 in the Appendix B.3.

### 4.4 *Pre-Black-Box* framework: HybridCORELS$_{\text{Pre}}$

HybridCORELS$_{\text{Pre}}$ is the first algorithm to implement our proposed *Pre-Black-Box* paradigm for learning hybrid interpretable models. It first builds a prefix optimizing accuracy and sparsity, given a minimum transparency constraint (9). Then, it trains the black-box part of the hybrid model, specializing it on the uncaptured examples, using the weighting scheme (5). As aforementioned in **Section 3.1.2**, the *Pre-Black-Box* paradigm intrinsically limits the possible collaboration between both parts of the hybrid model, as it is not possible for the black-box part to correct the mistakes made by the interpretable part. However, it is possible to consider the inconsistencies left to the black-box part while training the interpretable part, which implements a form of collaboration.

**Objective** HybridCORELS$_{\text{Pre}}$ builds a prefix $r$ capturing at least *min_transp* of the training data (transparency constraint (9)), and minimizing the overall hybrid model's classification error lower bound (based on both prefix $r$'s errors and the inconsistencies let to the black-box part) and sparsity:

$$\mathsf{obj}_{\text{pre}}(r, S) = \frac{\widehat{\mathcal{L}}_{S_r}(r) + \mathsf{incons}(S \setminus S_r)}{|S|} + \lambda \cdot |r| + \beta \cdot \frac{|S \setminus S_r|}{|S|} \tag{11}$$

where the error term $\frac{\widehat{\mathcal{L}}_{S_r}(r) + \mathsf{incons}(S \setminus S_r)}{|S|}$ corresponds to the entire hybrid model accuracy if the black-box performs perfectly (*i.e.,* correctly classifies all examples except the inconsistent ones). It hence provides a tight upper-bound on the entire hybrid model accuracy.

**Objective lower bound** CORELS' original objective lower bound $\mathsf{lb}(r, S)$ (8) (leveraging both the prefix's errors and the inconsistent examples among the uncaptured ones) is still valid and tight in this setup, so we do not need to modify it. Indeed, the error term is tight: it is not possible for any extension of $r$ to avoid the errors already made by $r$ nor the inconsistencies within the remaining examples. The sparsity term is also tight as any extension of $r$ must have a length of at least $|r| + 1$. As for HybridCORELS$_{\text{Post}}$, the (un)transparency term can not be used within the objective lower bound, as it can always reach 0.0 (if the built prefix captures all the training examples). An interesting observation is that $\mathsf{lb}(r, S) > \mathsf{obj}_{\text{pre}}(r, S)$ for any prefix $r$ (since $\beta < \lambda$ as indicated in **Section 4.2**). This means that, for any prefix $r$ with sufficient transparency (*i.e.,* satisfying the transparency constraint (9)), the algorithm will always discard any of its extensions as they increase the sparsity term and can not lower the error term (they can only equal it if they add no additional error). In fact, extending a prefix $r$ can only worsen its objective function (again, assuming that $\beta < \lambda$), and so an extension will only be performed in order to meet the transparency constraint (9).

Finally, HybridCORELS$_{\text{Pre}}$ is an exact method: it provably returns a prefix $r$ for which $\mathsf{obj}_{\text{pre}}(r, S)$ (11) is the smallest among those satisfying the transparency constraint (9). This means that, given desired transparency level, it produces an optimal prefix (interpretable part of the final hybrid model) in terms of overall hybrid model accuracy upper bound and sparsity. If the black-box performs perfectly, then the overall model is certifiably optimal. We provide a detailed pseudo-code of HybridCORELS$_{\text{Pre}}$ in Algorithm 3 in the Appendix B.3.

We additionally introduce in the Appendix C another possible implementation of the *Pre-Black-Box* paradigm based on the CORELS algorithm but optimizing an objective function different from that of HybridCORELS$_{\text{Pre}}$. This new variant HybridCORELS$_{\text{Pre,NoCollab}}$ learns a prefix by maximizing its accuracy on the subset $S_r$, without accounting for the task left to the black-box part. Appendix C.1 provides a description of this algorithm and Appendix C.2 empirically compares it with HybridCORELS$_{\text{Pre}}$. The experiments confirm that HybridCORELS$_{\text{Pre,NoCollab}}$ is not competitive with HybridCORELS$_{\text{Pre}}$ in medium to high transparency regimes, due to the lack of collaboration between both parts of the hybrid model.

## 5 Experiments

In this section, we empirically evaluate our proposed algorithms. We first introduce our experimental setup. Then, we use HybridCORELS$_{\text{Pre}}$ to show that the *Pre-Black-Box* paradigm is suitable to learn hybrid interpretable models exhibiting interesting trade-offs between accuracy and transparency. We explore the parameters of this paradigm, such as the specialization coefficient, to assess their effect and utility. Afterwards, we compare HybridCORELS$_{\text{Pre}}$ and HybridCORELS$_{\text{Post}}$ with two state-of-the-art methods for learning hybrid interpretable models: Hybrid-Rule-Set (HyRS) and Companion-Rule-List (CRL). Our thorough experimental study demonstrates that our proposed approaches are strongly competitive with the state of the art, while also providing optimality guarantees and allowing tight control of the desired transparency.

### 5.1 Setup

**Datasets** In our experiments, we consider several datasets with various prediction tasks and sizes:

- The **COMPAS** dataset[1](analyzed by Angwin et al. (2016)) contains 6,150 records from criminal offenders in the Broward County of Florida collected from 2013 and 2014. The corresponding binary classification task is to predict whether a person will re-offend within two years.
- The **UCI Adult Income** dataset (Dua & Graff, 2017) stores demographic attributes of 48,842 individuals from the 1994 U.S. census. Its binary classification task is to predict whether or not a particular person makes more than 50K USD per year.
- The **ACS Employment** dataset (Ding et al., 2021) is an extension of the UCI Adult Income dataset that includes more recent Census data (2014-2018). The goal is to predict if a person is employed/unemployed based on 10 socioeconomic factors. The specific dataset contained information on 203,358 constituents of the Texas state in 2018.

**Rules mining** To ensure a fair comparison between hybrid models, we pre-mined a set of rules $\Upsilon$ for each dataset. The various hybrid models were then restricted to select rules $r \in \Upsilon$ and, therefore, any difference in performance between models is solely attributable to the learning algorithms and not the quality of the rules. To mine the rules, the datasets were first binarized using quantile for numerical features and one-hot encoding for categorical features. Afterwards, the FP-Growth algorithm (Han et al., 2000) was applied to identify rules of cardinality 1-2 and support of at least 1%. To these sets of rules, we also added the negation of each rule in the original binarized dataset. Finally, the 300 rules with the largest support were kept to generate $\Upsilon$. We ended up with $|\Upsilon| = 230$ rules on COMPAS and $|\Upsilon| = 300$ on the UCI Adult Income and ACS Employment datasets.

**Black-boxes** In all experiments we used the following `Scikit-learn` (Pedregosa et al., 2011) classifiers as black-boxes: a `RandomForestClassifier`, an `AdaBoostClassifier`, and a `GradientBoostingClassifier`. Such black-boxes are in line with the setup considered in the literature (Wang, 2019). We further detail the hyper-parameters tuning of these models in sections 5.2 and 5.3. We note that the Hybrid models studied (HyRS, CRL, and HybridCORELS) are not tied to any specific black-box, nor to a specific implementation. Indeed they are black-box-agnostic by design.

**Implementation details** Our algorithms HybridCORELS$_{\text{Post}}$ and HybridCORELS$_{\text{Pre}}$ (as well as its HybridCORELS$_{\text{Pre,NoCollab}}$ variant discussed in the Appendix C) are integrated into a user-friendly Python module, provided in the supplementary material. They build upon the original CORELS (Angelino et al., 2017) C++ implementation[2] and its Python wrapper[3]. All experiments are run on a computing grid over a set of homogeneous nodes using Intel Platinum 8260 Cascade Lake @2.4Ghz CPU.

**HybridCORELS transparency regularization coefficient $\beta$ setting** In all our experiments using HybridCORELS$_{\text{Pre}}$ or HybridCORELS$_{\text{Post}}$, we set the transparency regularization coefficient $\beta = min(\frac{1}{2 \cdot |S|}, \frac{\lambda}{2})$ to only break ties but ensure that no accuracy nor sparsity will be traded-off for transparency, which is already enforced as a hard constraint (as discussed in **Section 4.2**).

### 5.2 Exploring the *Pre-Black-Box* Paradigm

**Objective** The objective of this subsection is to assess the appropriateness of the proposed *Pre-Black-Box* paradigm for learning accurate hybrid interpretable models. To this end, we use our proposed algorithm implementing this framework: HybridCORELS$_{\text{Pre}}$, depicted in **Section 4.4**. More precisely, we aim to explore the effect of the *specialization coefficient* on the performances of the produced models.

**Setup** For the three datasets presented in **Section 5.1**, we use HybridCORELS$_{\text{Pre}}$ to produce hybrid interpretable models for several *transparency* levels: low (0.25), medium (0.5), high (0.75, 0.85) and very high (0.95). For the prefix building part, we optimize the hyperparameters of HybridCORELS$_{\text{Pre}}$ using grid search

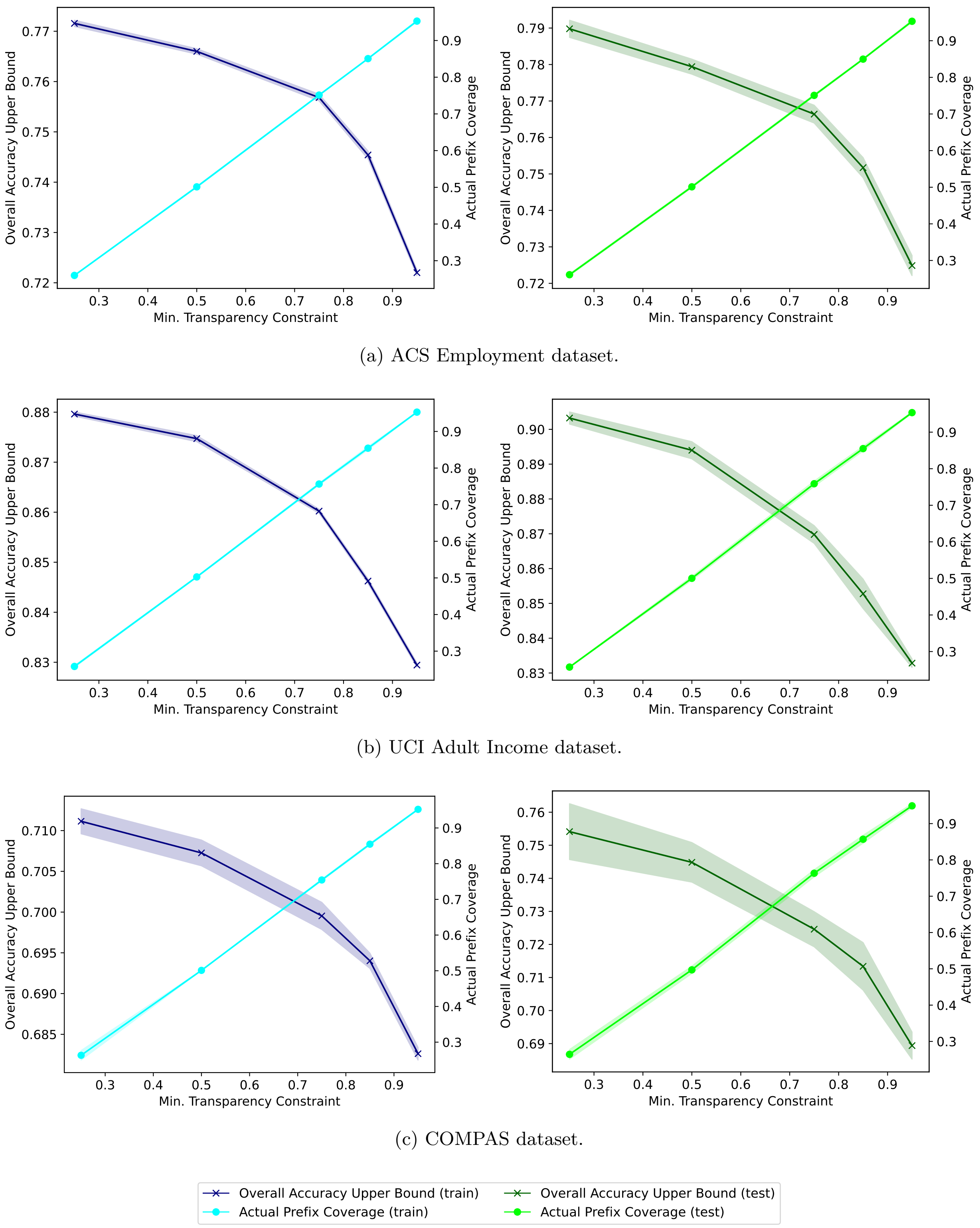
[1] `https://raw.githubusercontent.com/propublica/compas-analysis/master/compas-scores-two-years.csv`
[2] `https://github.com/corels/corels`
[3] `https://github.com/corels/pycorels`

over the following values: $\lambda \in \{10^{-2}, 10^{-3}, 10^{-4}\}$, $min_{support} \in \{0.01, 0.05, 0.10\}$, and the *objective-guided*, *lower-bound-guided*, and *BFS* search policies. For each experiment, the prefix yielding the best (training) accuracy upper-bound (considering the prefix's errors as well as the inconsistencies left to the black-box part, as depicted in (11)) is retained. Indeed, as discussed, this upper-bound precisely quantifies the accuracy that the overall hybrid model can reach if the black-box part makes no mistakes (other than those which can not be avoided due to inconsistencies): there exists a classification function for the black-box part which allows reaching such accuracy. The black-box part of the final hybrid model is then trained, and experiments are run for three different `Scikit-learn` (Pedregosa et al., 2011) black-boxes: an `AdaBoostClassifier` with default parameters, a `GradientBoostingClassifier` with default parameters and a `RandomForestClassifier` with $min\_samples\_split = 10$ and $max\_depth = 10$. Each black-box is retrained using different values for the specialization coefficient $\alpha$, ranging from 0 (no specialization) to 10 (highly specialized). Experiments are run for five different train/test splits, with 80% of the data used for training and the remaining 20% for testing.

**Results** The train and test performances of the learned prefixes are presented in Figure 10. As expected, when the enforced transparency level increases, the number of errors made by the interpretable part increases, and so does the overall hybrid model error lower bound (computed by the objective function (11)). We note that the actual prefix transparency on the training set is very close to the enforced constraint, with very small standard deviations. This illustrates the conflict between accuracy and transparency. Indeed, if a prefix with very high accuracy and transparency were available, the learning algorithm would systematically select it irrespective of the transparency constraint. However, the fact that transparencies are very close to their enforced constraint means that increasing the coverage of the prefix hinders the performance. This empirical observation meets the theoretical discussion of **Section 4.4** (Objective lower bound paragraph). We also observe that transparency generalizes well: the test set transparency levels are very close to the training set ones. Again, the standard deviation across dataset splits is very small.

We report results for the `AdaBoostClassifier` black-box in Figure 11 for the three datasets. Results for the two other black-boxes are provided in the supplementary material[4] and show the same trends. As expected, higher values of the specialization coefficient $\alpha$ lead to higher training accuracy of the black-box part. Indeed, the black-box component is evaluated on the subset of the data that is not captured by the interpretable part. Hence, specializing it on this subset is expected to raise its performances *on these samples*. Note that small variations exist, which can be explained by the heuristic nature of the considered black-box training algorithms. Overall, a *reasonable* specialization is usually beneficial. For low transparency values, the improvements brought by specialization are relatively modest (check the y-axis scales). This is explained by the fact that, in such contexts, the black-box subset of the data already represents most of the dataset. For very high transparency values, the black-box subset is relatively small, and an excessive specialization may not always pay off due to overfitting (as is the case with the UCI Adult Income experiment). For medium to high transparency values, specialization (with carefully chosen specialization coefficient $\alpha$) is always beneficial in these experiments. Here, specialization allows for black-box test accuracy absolute improvements up to 2.27 pps (experiment using the ACS Employment dataset, with minimum transparency 0.95). Considering all the experiments run with the `AdaBoostClassifier` black-box, the improvement rate (proportion of experiments for which specialization allowed improvements of the black-box test accuracy) is the highest for $\alpha = 2$, with a 93.33% improvement rate. Considering all the run experiments (including runs for the three datasets and the different transparency levels), the improvement rate values are the highest for $\alpha \in \{1, 2\}$. This confirms the usefulness of specialization but highlights the need to use reasonable specialization coefficient values $\alpha$. Observe that, when $\alpha = 1$, misclassifying an example belonging to the (training) black-box subset costs $\frac{e^1}{e^0} \approx 2.72$ times more than misclassifying a training example outside this set (in the optimized loss function). When $\alpha = 2$, it costs $\frac{e^2}{e^0} \approx 7.39$ times more.

[4] `HybridCORELS-code/paper/paper_5.2_results.zip`

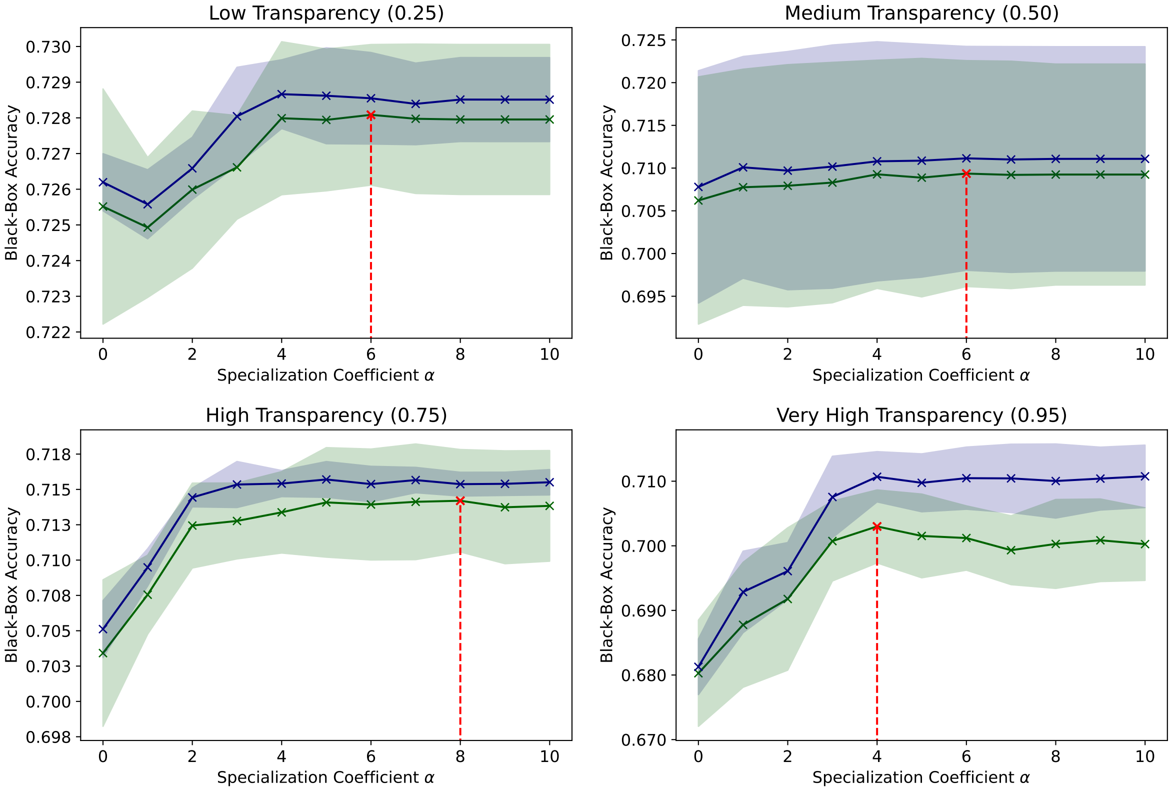


(a) ACS Employment dataset.

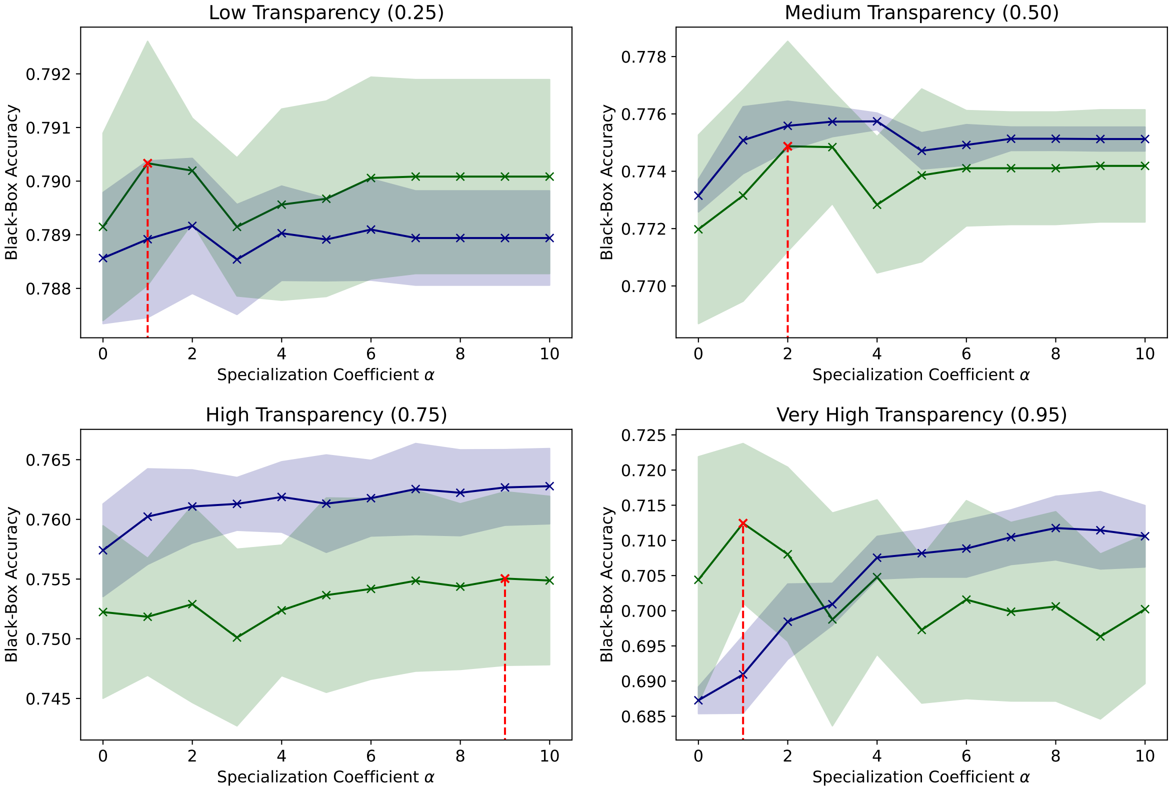
(b) UCI Adult Income dataset.

(c) COMPAS dataset.

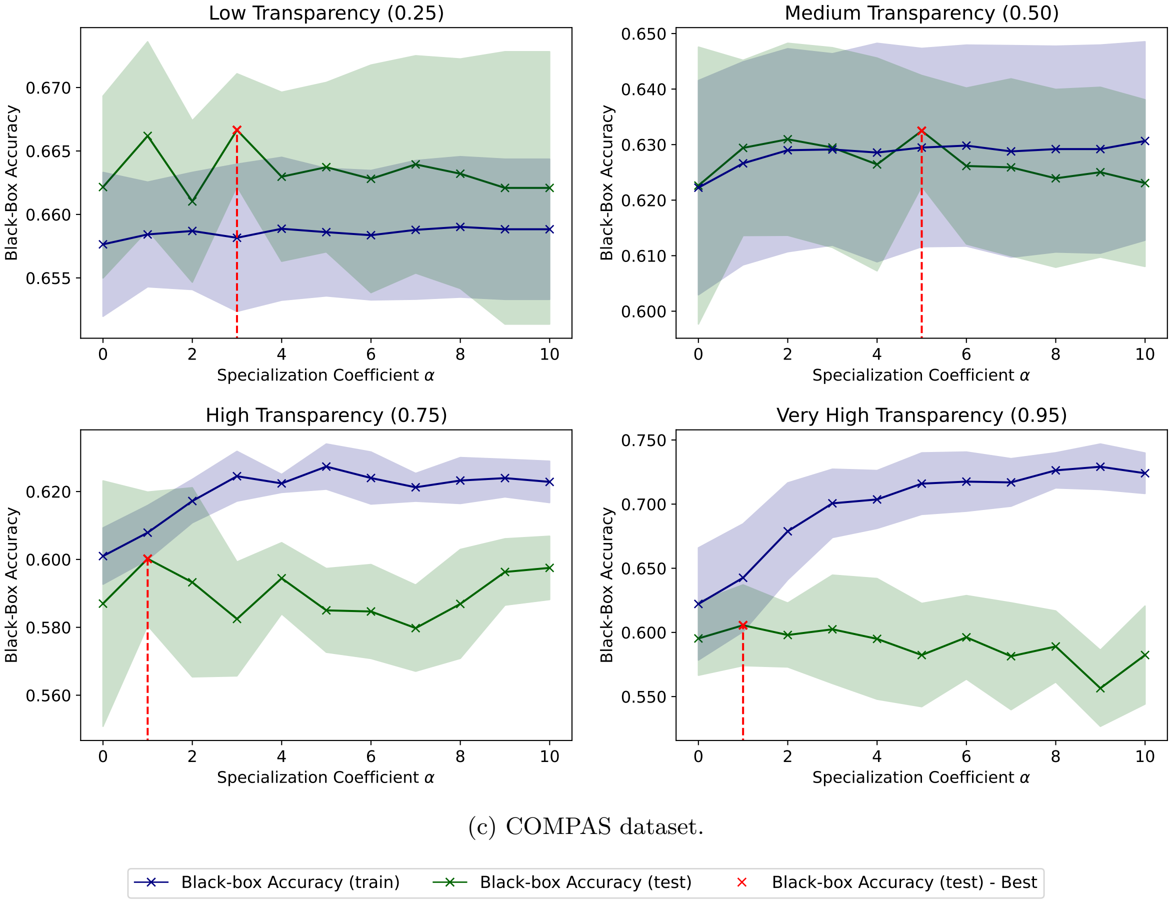
Figure 10: Training and test performances of the prefixes learnt using HybridCORELS$_{\text{Pre}}$. We report the actual transparency (prefix coverage) and overall accuracy upper bound (considering both the prefix's errors and the remaining inconsistencies) - which corresponds to the hybrid model accuracy if the black-box classifies correctly all consistent examples. The plots show both average values and standard deviation.



(a) ACS Employment dataset.



(b) UCI Adult Income dataset.



(c) COMPAS dataset.

Figure 11: Training and test performances of the black-box parts (`AdaBoostClassifier`) of the hybrid interpretable models learnt using HybridCORELS$_{\text{Pre}}$ on different datasets, for different transparency levels. The plots show both average values and standard deviation.

### 5.3 Tradeoffs and Comparison with the State-of-the-Art

**Objective** The aim of this subsection is to explore the trade-offs between the accuracy and transparency of several hybrid interpretable models learning frameworks: the state-of-the-art HyRS and CRL methods, as well as our proposed HybridCORELS$_{\text{Post}}$ and HybridCORELS$_{\text{Pre}}$ algorithms. These experiments serve the dual purpose of quantitatively comparing the various methods, but also to advertise the considerable amounts of transparency that can be attained while maintaining high performance.

**Setup** For these experiments, each dataset was split into training (60%), validation (20%), and test (20%) sets. We randomly generate five such splits and average the results over them. More precisely, for each split, the training set is used to train the models (both the black-box and the interpretable parts). The models' hyperparameters are optimized using the (separate) validation set. This allows a fair comparison between the methods while also encouraging generalization. Finally, the resulting hybrid models are evaluated on the (separate) test set. Note that while we hereafter focus on such test performances, the training and validation curves show similar trends, and the associated results are all provided within the supplementary material[5]. Hereafter, we detail the training and hyper-parameters optimization procedures for both the black-boxes and the hybrid interpretable models themselves.

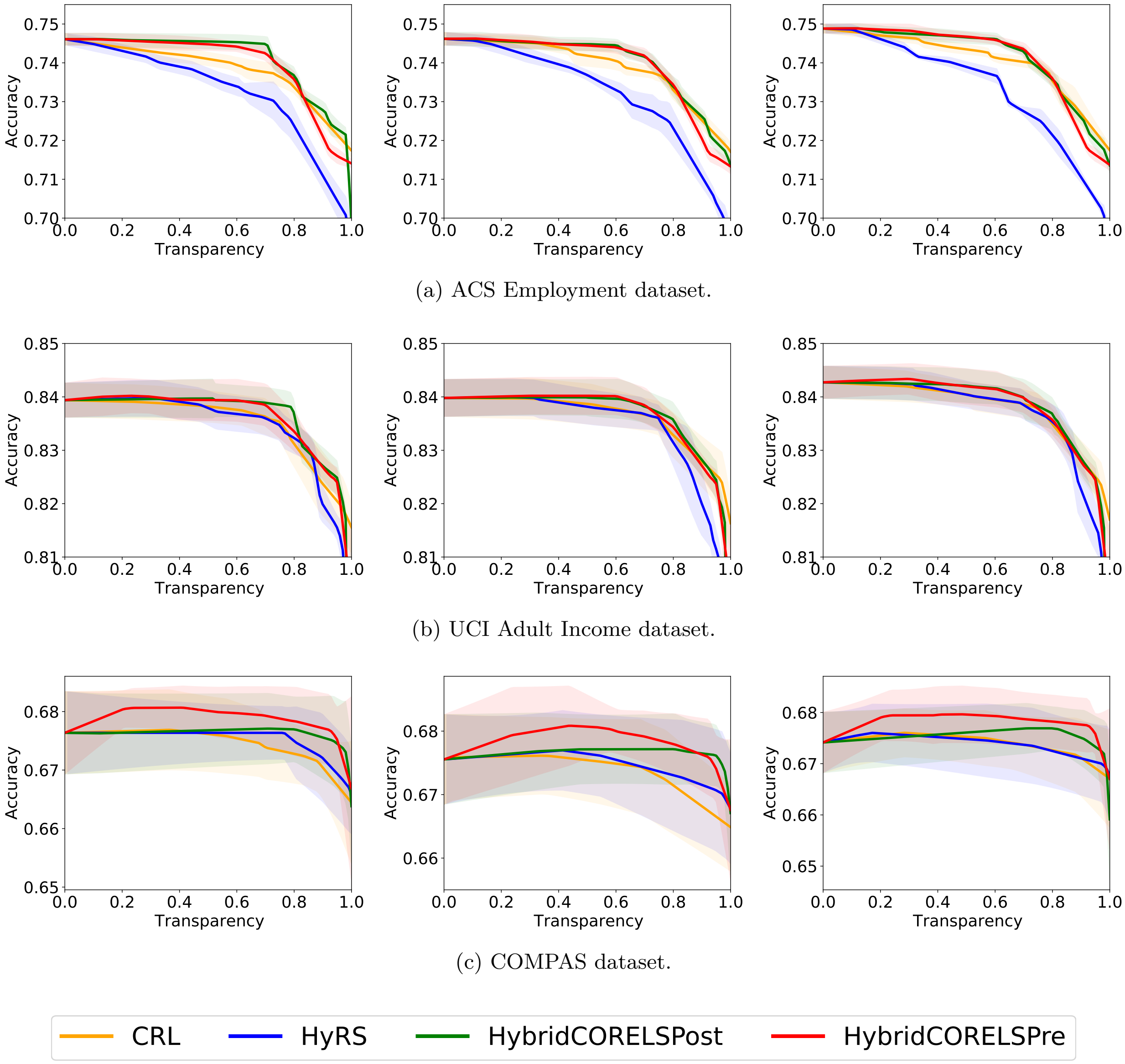
[5] `HybridCORELS-code/paper/results_part_4.zip`

***Pre-Black-Box* method setup** The experiments using the HybridCORELS$_{\text{Pre}}$ algorithm are divided into two phases. First, for each dataset (out of 3) and each random split (out of 5), we learn prefixes for 12 different minimum transparency constraints (0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 0.925, 0.95, 0.975) trying the following hyperparameters values: $\lambda \in \{10^{-2}, 10^{-3}, 10^{-4}\}$, $min_{support} \in \{0.01, 0.05, 0.10\}$, and the *objective-guided*, *lower-bound-guided*, and *BFS* search policies for HybridCORELS$_{\text{Pre}}$. Each prefix learning is limited to a maximum CPU time of 1 hour and a maximum memory use of 8 GB. For each experiment (dataset - random split - minimum transparency), the prefix yielding the best validation accuracy is retained. In a second phase, for each retained prefix, we try three different `Scikit-learn` (Pedregosa et al., 2011) black-boxes: a `RandomForestClassifier`, an `AdaBoostClassifier`, and a `GradientBoostingClassifier`. The black-box hyperparameters are tuned using the `Hyperopt` (Bergstra et al., 2013) Python library and its Tree of Parzen Estimators (TPE) algorithm, with 100 iterations. Just like the prefixes in the first phase, the black-boxes are trained using the training split (60%) and the hyperparameters are selected based on the validation split (20%) performances. Note that, as for the training set, the validation set loss is weighted to encourage the black-box to accurately classify the examples belonging to its assigned part of the input space (which is fixed as the prefix was trained first - *which allows specialization*, as previously discussed). Based on the observations from **Section 5.2**, we set the specialization coefficient $\alpha = 1$, which corresponds to a moderate black-box specialization.

***Post-Black-Box* methods setup** Three methods correspond to the *Post-Black-Box* paradigm: HybridCORELS$_{\text{Post}}$, along with the two state-of-the-art HyRS (Wang, 2019) and CRL (Pan et al., 2020) methods. The experiments using these methods are divided into two phases. First, for each dataset (out of 3) and each random split (out of 5), we train three different `Scikit-learn` (Pedregosa et al., 2011) black-boxes: a `RandomForestClassifier`, an `AdaBoostClassifier`, and a `GradientBoostingClassifier`. The black-box hyperparameters are tuned using the `Hyperopt` (Bergstra et al., 2013) Python library and its Tree of Parzen Estimators (TPE) algorithm, with 100 iterations. The black-boxes are trained using the training split (60%) and their hyperparameters are selected based on the validation split (20%) performances. In the second phase of the experiments, we train the interpretable parts of the hybrid models for the three compared methods, using the black-boxes learned in the previous phase. For each of the three methods, we try different hyperparameter values. Again, the training is performed on the training split (60%), while the hyperparameters values are selected based on the validation split (20%) performances. For HybridCORELS$_{\text{Post}}$, we consider 12 different minimum transparency constraints (0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 0.925, 0.95, 0.975), and the following hyperparameters values: $\lambda \in \{10^{-2}, 10^{-3}, 10^{-4}\}$, $min_{support} \in \{0.01, 0.05, 0.10\}$, and the *objective-guided*, *lower-bound-guided*, and *BFS* search policies. For the HyRS method, similarly to what was done in (Wang, 2019), we use 10 different values for its $\lambda$ hyperparameter (ranging logarithmically between $10^{-3}$ and $10^{-2}$) and 10 different values for its $\beta$ hyperparameter (ranging logarithmically between $10^{-3}$ and $10^{0}$). For CRL, we consider 10 different values for its *temperature* hyperparameter (ranging linearly between $10^{-3}$ and $10^{-2}$) and 10 different values for its $\lambda$ hyperparameter (ranging logarithmically between $10^{-3}$ and $10^{-1}$). For all three methods HybridCORELS$_{\text{Post}}$, HyRS, and CRL, the hyperparameter grid is roughly of size 100. As in the HybridCORELS$_{\text{Pre}}$ experiments, the interpretable parts building is limited to a maximum CPU time of 1 hour and a maximum memory use of 8 GB.

**Final results computation** After tuning the hyper-parameters, we are left with a Pareto front representing the hybrid models that are not dominated in terms of both validations set accuracy and transparency. Still, since the black box and hybrid models were fine-tuned on the validation set, we argue that this Pareto front will be an over-optimistic description of the true generalisation of our hybrid models. For this reason, we decided to take the Pareto-optimal models on validation, and compute their accuracy and transparency on the test set, which has not been used yet in this experiment. Hence, we can obtain unbiased measures of the accuracy and transparency for these models. These final measures of accuracy/transparency are used as a means to compare the different approaches and assess if increasing transparency can lead to equivalent/better generalization.

**Results** The test set accuracy/transparency trade-offs of the different hybrid models learning frameworks are shown in Figure 12 for each dataset and black-box type. We highlight three main insights from these results.



(a) ACS Employment dataset.

(b) UCI Adult Income dataset.

(c) COMPAS dataset.

Figure 12: Test set accuracy/transparency trade-offs for various hybrid models learning frameworks and datasets. The Pareto front for each method is represented as a line and the filled bands encode the std across the five data split reruns. Results are provided for several black-boxes: (Left) AdaBoost, (Middle) Random Forests, (Right) Gradient Boosted Trees.

```
if ["age_medium" and "No Cognitive difficulty"] then 1
else if ["age_high"] then 0
else
    AdaBoost()
```

(a) HyRS: Test Accuracy 72.8%, Transparency 64.3%

```
if  ["age_high" and "Female"] then 0
else if ["age_high" and "Native"] then 0
else if ["Reference person" and "No disability"] then 1
else if ["Husband/wife" and "No disability"] then 1
else if ["Cognitive difficulty" and "not own child of householder"] then 0
else
    AdaBoost()
```

(b) CRL: Test Accuracy 73.7%, Transparency 75.8%.

```
if ["Disability" and "age_high"] then 0
else if ["Husband/wife" and "Male"] then 1
else if ["age_high" and "Native"] then 0
else if ["age_high" and "Female"] then 0
else if ["Reference person" and "No disability"] then 1
else if ["Bachelor degree"] then 1
else
    AdaBoost()
```

(c) HybridCORELS$_{\text{Pre}}$: Test Accuracy 74.0%, Transparency 70.1%.

```
if ["age_high" and "Female"] then 0
else if ["Husband/wife" and "No disability"] then 1
else if ["age_high" and "Native"] then 0
else if ["Reference person" and "No disability"] then 1
else
    AdaBoost()
```

(d) HybridCORELS$_{\text{Post}}$ : Test Accuracy 73.7%, Transparency 73.0%.

Figure 13: Example hybrid interpretable models obtained by the different methods on the same data split of the ACS Employment dataset with a AdaBoost black-box.

First, on almost all datasets and black-box types, the methods HybridCORELS$_{\text{Pre}}$ and HybridCORELS$_{\text{Post}}$ are better or equivalent to HyRS and CRL. The only exception is HybridCORELS$_{\text{Pre}}$ in high transparency regimes (0.85-1.0) on the ACS Employment dataset. The reason HybridCORELS is so competitive with state-of-the-art methods is that it solves its objective to optimality, exploring the whole search space of prefixes (which methods based on local search can hardly achieve). Hence, given a learning paradigm and a transparency constraint, it builds the prefix that maximizes accuracy. Furthermore, unlike other approaches, HybridCORELS offers precise control over the desired level of transparency. In Figure 13, we show example hybrid models for each of the four methods fitted on the same data split (train/validation/test) of the ACS Employment dataset with an AdaBoost black-box. These models were selected on the basis of having the highest test accuracies for test transparencies restricted between 0.6 and 0.8. We note that HybridCORELS$_{\text{Pre}}$ and HybridCORELS$_{\text{Post}}$ are competitive with CRL and even employ similar rules, for example, `["age_high" and "Female"]`, `["Reference person" and "No disability"]`, and `["age_high" and "Native"]`. HyRS on the other hand, performs worst than the other three since it has a lesser accuracy and transparency.

Secondly, using HybridCORELS on the ACS Employment and UCI Adult Income datasets, one can reach high transparency values (0.7) while retaining the same performance as the black-box (0.0 transparency). This observation is consistent across all black-box types, which suggests that complex models are often overkill in certain regions of the input space and can safely be replaced by a simpler model on those inputs. From the point of view of certification/maintenance of a machine learning model, being able to assign a

```
if ["Prior-Crimes=0" and "Age>=30"] then 0
else if ["Prior-Crimes>5" and "Age=24-30"] then 1
else if ["Prior-Crimes=1-3" and "Age>=30"] then 0
else
    RandomForest()
```

(a) HybridCORELS$_{\mathrm{Pre}}$: Test Accuracy 68.1%, Transparency 42.7%

Figure 14: Example hybrid interpretable model obtained by HybridCORELS$_{\mathrm{Pre}}$ on the COMPAS dataset with a Random Forest black-box. Consistent with Figure 12, this model generalizes better than the black-box alone.

majority of inputs to an interpretable component is a tremendous step forward. For instance, since rule lists are interpretable, one might be able to certify that the hybrid model works properly/safely on the region $\Omega_r$ that will contain the majority of examples seen in deployment. For the minority of instances that fall outside the region, certification might require the verification of the opaque decisions by a committee of domain experts. Such verification might be time-consuming but, the higher the transparency, the fewer examples this committee would need to verify regularly.

Thirdly, when studying HybridCORELS$_{\mathrm{Pre}}$ fitted on COMPAS, one can consistently observe a “sweet spot” for transparency where the generalization is maximal and even better than the standalone black-box. The existence of such a “sweet spot” is predicted by our theory of **Section 2.3** and highlights the regularization effect of the hybrid modeling. Although retaining the same level of performance while increasing the transparency is enough to argue in favor of hybrid modeling (as was the case with the ACS Employment and UCI Adult Income datasets), it is interesting to see that hybrid models can also improve the generalization performance. This generalization improvement is mainly observed with the HybridCORELS$_{\mathrm{Pre}}$ method, which constitutes an argument in favor of the *Pre-Black-Box* paradigm. We report in Figure 14 an example model learned with HybridCORELS$_{\mathrm{Pre}}$ on COMPAS, which generalizes better than a standalone black-box. As we observe, just adding three simple rules before the black-box model allows for test accuracy improvements.

## 6 Conclusion

In this paper, we laid the foundations for a promising line of work that was initiated some years ago: hybridizing interpretable and black-box models to “take the best of both worlds". More precisely, we first provided theoretical evidence that such models have generalization advantages, while also being easier to certify and understand. We then proposed a taxonomy of learning algorithms aimed at producing such models, along with a generic framework implementing the (new) *Pre-Black-Box* paradigm. We introduced algorithms belonging to two identified paradigms, namely *Pre-Black-Box* and *Post-Black-Box*. Compared to state-of-the-art methods, our proposed approaches, coined HybridCORELS$_{\mathrm{Pre}}$ and HybridCORELS$_{\mathrm{Post}}$, certify the optimality of the learned models and provide direct control over the desired transparency level. Our thorough experiments demonstrated the ability of the proposed *Pre-Black-Box* paradigm and the high competitivity of our algorithms with the state-of-the-art. Furthermore, empirical findings suggest that this new paradigm may lead to better-generalizing models. Investigating the reasons for this observation is an interesting future work. Adapting other optimal search-based learning algorithms (as was done with CORELS in this work) - for instance those producing optimal sparse decision trees - to produce new forms of hybrid interpretable models (beyond rule-based ones) is also a promising research avenue. Finally, designing fully end-to-end and certifiably optimal hybrid interpretable models’ learning algorithms is an open challenge, whose main difficulty consists in encoding both parts of the model within a unified framework to train them jointly.

## References

Ulrich Aïvodji, Hiromi Arai, Olivier Fortineau, Sébastien Gambs, Satoshi Hara, and Alain Tapp. Fairwashing: the risk of rationalization. In *International Conference on Machine Learning*, pp. 161–170. PMLR, 2019.

Ulrich Aïvodji, Hiromi Arai, Sébastien Gambs, and Satoshi Hara. Characterizing the risk of fairwashing. *Advances in Neural Information Processing Systems*, 34:14822–14834, 2021.

Elaine Angelino, Nicholas Larus-Stone, Daniel Alabi, Margo I. Seltzer, and Cynthia Rudin. Learning certifiably optimal rule lists for categorical data. *J. Mach. Learn. Res.*, 18:234:1–234:78, 2017. URL `http://jmlr.org/papers/v18/17-716.html`.

Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias: There's software used across the country to predict future criminals. and it's biased against blacks. propublica (2016). *ProPublica, May*, 23, 2016.

James Bergstra, Daniel Yamins, and David Cox. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *International conference on machine learning*, pp. 115–123. PMLR, 2013.

Leo Breiman. *Classification and regression trees*. Routledge, 2017.

Mark Craven and Jude Shavlik. Extracting tree-structured representations of trained networks. *Advances in neural information processing systems*, 8, 1995.

Botty Dimanov, Umang Bhatt, Mateja Jamnik, and Adrian Weller. You shouldn't trust me: Learning models which conceal unfairness from multiple explanation methods. In Giuseppe De Giacomo, Alejandro Catalá, Bistra Dilkina, Michela Milano, Senén Barro, Alberto Bugarín, and Jérôme Lang (eds.), *ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, August 29 - September 8, 2020 - Including 10th Conference on Prestigious Applications of Artificial Intelligence (PAIS 2020)*, volume 325 of *Frontiers in Artificial Intelligence and Applications*, pp. 2473–2480. IOS Press, 2020. doi: 10.3233/FAIA200380. URL `https://doi.org/10.3233/FAIA200380`.

Frances Ding, Moritz Hardt, John Miller, and Ludwig Schmidt. Retiring adult: New datasets for fair machine learning. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 6478–6490, 2021.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`.

Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5):1–42, 2018.

Jiawei Han, Jian Pei, and Yiwen Yin. Mining frequent patterns without candidate generation. *ACM sigmod record*, 29(2):1–12, 2000.

Xiyang Hu, Cynthia Rudin, and Margo Seltzer. Optimal sparse decision trees. *Advances in Neural Information Processing Systems*, 32, 2019.

Gabriel Laberge, Ulrich Aïvodji, and Satoshi Hara. Fooling shap with stealthily biased sampling. *arXiv preprint arXiv:2205.15419*, 2022.

Himabindu Lakkaraju, Ece Kamar, Rich Caruana, and Jure Leskovec. Interpretable & explorable approximations of black box models. *arXiv preprint arXiv:1707.01154*, 2017.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

Danqing Pan, Tong Wang, and Satoshi Hara. Interpretable companions for black-box models. In *International conference on artificial intelligence and statistics*, pp. 2444–2454. PMLR, 2020.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " 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*, pp. 1135–1144, 2016.

Peter R Rijnbeek and Jan A Kors. Finding a short and accurate decision rule in disjunctive normal form by exhaustive search. *Machine learning*, 80(1):33–62, 2010.

Ronald L. Rivest. Learning decision lists. *Mach. Learn.*, 2(3):229–246, 1987. doi: 10.1007/BF00058680. URL `https://doi.org/10.1007/BF00058680`.

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

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Dylan Slack, Sophie Hilgard, Emily Jia, Sameer Singh, and Himabindu Lakkaraju. Fooling lime and shap: Adversarial attacks on post hoc explanation methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pp. 180–186, 2020.

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

Tong Wang. Gaining free or low-cost interpretability with interpretable partial substitute. In *International Conference on Machine Learning*, pp. 6505–6514. PMLR, 2019.

Tong Wang and Qihang Lin. Hybrid predictive models: When an interpretable model collaborates with a black-box model. *J. Mach. Learn. Res.*, 22:137–1, 2021.

# A Proof of Theorem 1

**Theorem 1** *Given a finite hybrid model space ($|\mathsf{Hyb}| < \infty$) and some $\epsilon > 0$, letting $C_\Omega := \mathbb{P}_{\boldsymbol{x}\sim\mathcal{D}}[\boldsymbol{x} \in \Omega]$ be the transparency of $\Omega$, then for any distribution $\mathcal{D}$ where there exists a triplet $\langle h_c^\star, h_s^\star, \Omega^\star\rangle$ with zero generalization error (as defined in (1)), the following holds for a training set of size $M$:*

$$\mathop{\mathbb{P}}_{S\sim\mathcal{D}^M}[\mathcal{L}_\mathcal{D}(\langle h_c, h_s, \Omega\rangle_S) > \epsilon] \leq \sum_{\Omega\in\mathcal{P}} \mathcal{B}(\epsilon, C_\Omega, \mathcal{H}_c, \mathcal{H}_s, M),$$

*where*

$$\mathcal{B}(\epsilon, C_\Omega, \mathcal{H}_c, \mathcal{H}_s, M) := (1-|\mathcal{H}_c|)C_\Omega^M + (1-|\mathcal{H}_s|)C_{\overline{\Omega}}^M + |\mathcal{H}_c|(C_{\overline{\Omega}}e^{-\epsilon} + C_\Omega)^M + |\mathcal{H}_s|(C_\Omega e^{-\epsilon} + C_{\overline{\Omega}})^M.$$

*If we assume that the optimal subset $\Omega \equiv \Omega^\star$ is known in advance, then the bound tightens*

$$\mathop{\mathbb{P}}_{S\sim\mathcal{D}^M}[\mathcal{L}_\mathcal{D}(\langle h_c, h_s, \Omega\rangle_S) > \epsilon] \leq \mathcal{B}(\epsilon, C_\Omega, \mathcal{H}_c, \mathcal{H}_s, M).$$

**Proof** The distribution $\mathcal{D}$ is fixed apriori and our only assumption is that it can be perfectly solved by a hybrid model in $\mathsf{Hyb}$. Since we assume a perfect model exists in $\mathsf{Hyb}$, we must have $\widehat{\mathcal{L}}_S(\langle h_c, h_s, \Omega\rangle_S) = 0$. Given $\epsilon > 0$, our main objective is to upper bound the probability $\mathbb{P}_{S\sim\mathcal{D}^M}[\mathcal{L}_\mathcal{D}(\langle h_c, h_s, \Omega\rangle_S) > \epsilon]$ which corresponds to the probability of "failure" by the model. Letting $\mathsf{Hyb}_\epsilon := \{\langle h_c, h_s, \Omega\rangle \in \mathsf{Hyb} : \mathcal{L}_\mathcal{D}(\langle h_c, h_s, \Omega\rangle) > \epsilon\}$ be the set of all "failing" hybrid models, we have that

$$\begin{aligned}\mathop{\mathbb{P}}_{S\sim\mathcal{D}^M}[\mathcal{L}_\mathcal{D}(\langle h_c, h_s, \Omega\rangle_S) > \epsilon] &\leq \mathop{\mathbb{P}}_{S\sim\mathcal{D}^M}[\exists\langle h_c, h_s, \Omega\rangle \in \mathsf{Hyb}_\epsilon \text{ with } \widehat{\mathcal{L}}_S(\langle h_c, h_s, \Omega\rangle) = 0]\\ &\leq \sum_{\Omega\in\mathcal{P}} \mathop{\mathbb{P}}_{S\sim\mathcal{D}^M}[\exists\langle h_c, h_s, \Omega\rangle \in \mathsf{Hyb}_\epsilon \text{ with } \widehat{\mathcal{L}}_S(\langle h_c, h_s, \Omega\rangle) = 0],\end{aligned} \tag{12}$$

where we have used the union bound over all $\Omega \in \mathcal{P}$. From this point on, we will assume that the domain $\Omega$ is fixed. Letting $C_\Omega := \mathbb{P}_{\boldsymbol{x}\sim\mathcal{D}}[\boldsymbol{x} \in \Omega]$ and $\overline{\Omega} := \mathcal{X} \setminus \Omega$, we can see the distribution $\mathcal{D}$ as a mixture of two distributions $\mathcal{D}_c, \mathcal{D}_s$ with disjoint supports $\overline{\Omega}$ and $\Omega$. Formally, we have $\mathcal{D} = C_{\overline{\Omega}}\mathcal{D}_c + C_\Omega\mathcal{D}_s$. The edge cases $\text{supp}(\mathcal{D}) \subset \Omega$ and $\text{supp}(\mathcal{D}) \subset \overline{\Omega}$ are covered by setting $C_\Omega{=}1, C_{\overline{\Omega}}{=}0$ and $C_\Omega{=}0, C_{\overline{\Omega}}{=}1$ respectivelly. Sampling from such a mixture distribution $\mathcal{D}$ can be seen as a two-step process. First, we choose a number of instances $m \sim \text{Bin}(C_\Omega, M)$ from a binomial law of $M$ trials and probability $C_\Omega$ of success. Then we sample $m$ simple examples $S_s \sim \mathcal{D}_s^m$, and sample $M - m$ hard examples $S_c \sim \mathcal{D}_c^{M-m}$. We get

$$\begin{aligned}&\mathop{\mathbb{P}}_{S\sim\mathcal{D}^M}[\exists\langle h_c, h_s, \Omega\rangle \in \mathsf{Hyb}_\epsilon \text{ with } \widehat{\mathcal{L}}_S(\langle h_c, h_s, \Omega\rangle) = 0]\\ &\quad= \mathop{\mathbb{P}}_{\substack{m\sim\text{Bin}(C_\Omega,M)\\ S_s\sim\mathcal{D}_s^m\\ S_c\sim\mathcal{D}_c^{M-m}}}[\exists\langle h_c, h_s, \Omega\rangle \in \mathsf{Hyb}_\epsilon \text{ with } \widehat{\mathcal{L}}_{S_c\cup S_s}(\langle h_c, h_s, \Omega\rangle) = 0]\\ &\quad= \sum_{m=0}^M b(m; C_\Omega, M) \mathop{\mathbb{P}}_{\substack{S_s\sim\mathcal{D}_s^m\\ S_c\sim\mathcal{D}_c^{M-m}}}[\exists\langle h_c, h_s, \Omega\rangle \in \mathsf{Hyb}_\epsilon \text{ with } \widehat{\mathcal{L}}_{S_c\cup S_s}(\langle h_c, h_s, \Omega\rangle) = 0],\end{aligned} \tag{13}$$

where we have introduced $b(m; C_\Omega, M) := \binom{M}{m} C_\Omega^m (1 - C_\Omega)^{M-m}$ as the binomial coefficients. In this formula, there are two extreme edges cases $m = 0$ and $m = M$ which occur with probability $C_{\overline{\Omega}}^M$ and $C_\Omega^M$ respectively. The issue with both of these extreme cases is that we are meant to bound the population loss of the whole hybrid model while only one of its sub-models is evaluated on empirical data. We decide to employ trivial bounds which will become less and less dominant when the probability of these extreme cases goes to zero as $M \to \infty$, assuming $C_\Omega \in ]0, 1[$.

**Case $\mathbf{m = 0}$**

$$\mathop{\mathbb{P}}_{S_c\sim\mathcal{D}_c^M}[\exists\langle h_c, h_s, \Omega\rangle \in \mathsf{Hyb}_\epsilon \text{ with } \widehat{\mathcal{L}}_{S_c}(h_c) = 0] \leq 1$$

**Case $\mathbf{m = M}$**

$$\mathop{\mathbb{P}}_{S_s \sim \mathcal{D}_s^M}[\exists \langle h_c, h_s, \Omega\rangle \in \mathsf{Hyb}_\epsilon \text{ with } \widehat{\mathcal{L}}_{S_s}(h_s) = 0] \leq 1$$

**Case $\mathbf{0 < m < M}$** Since the expected loss can be rewritten

$$\mathcal{L}_{\mathcal{D}}(\langle h_c, h_s, \Omega\rangle) = C_{\overline{\Omega}}\mathcal{L}_{\mathcal{D}_c}(h_c) + C_\Omega \mathcal{L}_{\mathcal{D}_s}(h_s),$$

we have that

$$\mathcal{L}_{\mathcal{D}_c}(h_c) \leq \epsilon \text{ and } \mathcal{L}_{\mathcal{D}_s}(h_s) \leq \epsilon \Rightarrow \mathcal{L}_{\mathcal{D}}(\langle h_c, h_s, \Omega\rangle) \leq \epsilon,$$

which implies

$$\langle h_c, h_s, \Omega\rangle \in \mathsf{Hyb}_\epsilon \Rightarrow h_c \in \mathcal{H}_{c,\epsilon} \text{ or } h_s \in \mathcal{H}_{s,\epsilon}, \tag{14}$$

where $\mathcal{H}_{c,\epsilon} := \{h_c \in \mathcal{H}_c : \mathcal{L}_{\mathcal{D}_c}(h_c) > \epsilon\}$ and $\mathcal{H}_{s,\epsilon} := \{h_s \in \mathcal{H}_s : \mathcal{L}_{\mathcal{D}_s}(h_s) > \epsilon\}$ are the sets of complex and simple models "failing" on the distributions $\mathcal{D}_c$ and $\mathcal{D}_s$. Note that the "or" in (14) is not exclusive and both parts of the model may fail simultaneously, although it is not necessary. Therefore the following holds

$$\begin{aligned}
&\mathop{\mathbb{P}}_{\substack{S_s \sim \mathcal{D}_s^m \\ S_c \sim \mathcal{D}_c^{M-m}}}[\exists \langle h_c, h_s, \Omega\rangle \in \mathsf{Hyb}_\epsilon \text{ with } \widehat{\mathcal{L}}_{S_c \cup S_s}(\langle h_c, h_s, \Omega\rangle) = 0] \\
&\qquad \leq \mathop{\mathbb{P}}_{\substack{S_s \sim \mathcal{D}_s^m \\ S_c \sim \mathcal{D}_c^{M-m}}}[\{\exists h_c \in \mathcal{H}_{c,\epsilon} \text{ s.t. } \widehat{\mathcal{L}}_{S_c}(h_c) = 0\} \text{ or } \{\exists h_s \in \mathcal{H}_{s,\epsilon} \text{ s.t. } \widehat{\mathcal{L}}_{S_s}(h_s) = 0\}] \\
&\qquad \leq \mathop{\mathbb{P}}_{S \sim \mathcal{D}_c^{M-m}}[\exists h_c \in \mathcal{H}_{c,\epsilon} \text{ s.t. } \widehat{\mathcal{L}}_S(h_c) = 0] + \mathop{\mathbb{P}}_{S \sim \mathcal{D}_s^m}[\exists h_s \in \mathcal{H}_{s,\epsilon} \text{ s.t. } \widehat{\mathcal{L}}_S(h_s) = 0] \\
&\qquad \leq |\mathcal{H}_c| e^{-\epsilon(M-m)} + |\mathcal{H}_s| e^{-\epsilon m},
\end{aligned}$$

where we have used the inequality $\mathbb{P}_{S \sim \mathcal{D}_s^m}[\exists h_s \in \mathcal{H}_{s,\epsilon} \text{ s.t. } \widehat{\mathcal{L}}_S(h_s) = 0] \leq |\mathcal{H}_s| e^{-\epsilon m}$ (Equation 2.9 of Shalev-Shwartz & Ben-David (2014)), and a similar one for $\mathcal{H}_c$. Going back to Equation (13), we get

$$\begin{aligned}
&\mathop{\mathbb{P}}_{S \sim \mathcal{D}^M}[\exists \langle h_c, h_s, \Omega\rangle \in \mathsf{Hyb}_\epsilon \text{ with } \widehat{\mathcal{L}}_S(\langle h_c, h_s, \Omega\rangle) = 0] \\
&\quad = \sum_{m=0}^{M} b(m; C_\Omega, M) \mathop{\mathbb{P}}_{\substack{S_s \sim \mathcal{D}_s^m \\ S_c \sim \mathcal{D}_c^{M-m}}}[\exists \langle h_c, h_s, \Omega\rangle \in \mathsf{Hyb}_\epsilon \text{ with } \widehat{\mathcal{L}}_{S_c \cup S_s}(\langle h_c, h_s, \Omega\rangle) = 0] \\
&\quad \leq C_{\overline{\Omega}}^M + C_\Omega^M + \sum_{m=1}^{M-1} b(m; C_\Omega, M)\big(|\mathcal{H}_c| e^{-\epsilon(M-m)} + |\mathcal{H}_s| e^{-\epsilon m}\big) \\
&\quad = C_{\overline{\Omega}}^M + C_\Omega^M + |\mathcal{H}_c| \sum_{m=1}^{M-1} b(m; C_\Omega, M) e^{-\epsilon(M-m)} + |\mathcal{H}_s| \sum_{m=1}^{M-1} b(m; C_\Omega, M) e^{-\epsilon m} \\
&\quad = C_{\overline{\Omega}}^M + C_\Omega^M + |\mathcal{H}_c| \sum_{m=1}^{M-1} b(m; C_{\overline{\Omega}}, M) e^{-\epsilon m} + |\mathcal{H}_s| \sum_{m=1}^{M-1} b(m; C_\Omega, M) e^{-\epsilon m} \\
&\quad \leq C_{\overline{\Omega}}^M + C_\Omega^M + |\mathcal{H}_c| \sum_{m=1}^{M} b(m; C_{\overline{\Omega}}, M) e^{-\epsilon m} + |\mathcal{H}_s| \sum_{m=1}^{M} b(m; C_\Omega, M) e^{-\epsilon m} \\
&\quad = (1 - |\mathcal{H}_c|) C_\Omega^M + (1 - |\mathcal{H}_s|) C_{\overline{\Omega}}^M + |\mathcal{H}_c| \sum_{m=0}^{M} b(m; C_{\overline{\Omega}}, M) e^{-\epsilon m} + |\mathcal{H}_s| \sum_{m=0}^{M} b(m; C_\Omega, M) e^{-\epsilon m} \\
&\quad = (1 - |\mathcal{H}_c|) C_\Omega^M + (1 - |\mathcal{H}_s|) C_{\overline{\Omega}}^M + |\mathcal{H}_c| (C_{\overline{\Omega}} e^{-\epsilon} + C_\Omega)^M + |\mathcal{H}_s| (C_\Omega e^{-\epsilon} + C_{\overline{\Omega}})^M \\
&\quad := \mathcal{B}(\epsilon, C_\Omega, \mathcal{H}_c, \mathcal{H}_s, M)
\end{aligned}$$

where for the second-to-last step we have used the identity

$$\sum_{m=0}^{M} b(m; C_\Omega, M) e^{-\epsilon m} = (C_\Omega e^{-\epsilon} + C_{\overline{\Omega}})^M.$$

Finally, combining this with Equation (12),

$$\underset{S\sim\mathcal{D}^M}{\mathbb{P}}[\mathcal{L}_\mathcal{D}(\langle h_c, h_s, \Omega\rangle_S) > \epsilon] \leq \sum_{\Omega\in\mathcal{P}} \mathcal{B}(\epsilon, C_\Omega, \mathcal{H}_c, \mathcal{H}_s, M),$$

which is the first desired result.

Now assuming that the optimal region $\Omega \equiv \Omega^\star$ is known in advance, then the logic of the proof is identical except that we do not employ a union bound over all $\Omega \in \mathcal{P}$ as in Equation (12). ■

## B Pseudo-Codes of the HybridCORELS algorithms

While the CORELS algorithm and our proposed HybridCORELS variants were already introduced in **Section 4**, we describe them in more detail in this appendix section. We first introduce some necessary notation that we later use to provide a detailed pseudo-code and description of the CORELS algorithm. We then depict our proposed variants $\text{HybridCORELS}_{\text{Post}}$ and $\text{HybridCORELS}_{\text{Pre}}$ for learning hybrid interpretable models.

### B.1 Notations

To formally describe the pseudo-code of the CORELS algorithm and those of our modified HybridCORELS variants, we first need to introduce some more detailed notation. As mentioned in **Section 4.1**, a rule list $d$ consists in an ordered set of rules $r$, called a prefix, followed by a default decision $q_0$. Then, we note: $d = (r, q_0)$. Each individual rule $r_i$ involved within prefix $r$ consists of an *antecedent* $a_i$ (“if" part of the rule, consisting in a Boolean assertion over the features' values) and a consequent $q_i$ (“then" part of the rule, consisting in a prediction). We note: $r_i = a_i \rightarrow q_i$, and $r = (r_1, r_2, \ldots, r_{|r|})$ with $|r|$ the length of prefix $r$.

### B.2 CORELS

The pseudo-code of the CORELS algorithm is provided within Algorithm 1. As mentioned in **Section 4.1**, CORELS is a branch-and-bound algorithm exploring a prefix tree, in which each node corresponds to a prefix $r$ and its children are prefixes formed by extending $r$. At each step of the exploration, the nodes belonging to the exploration frontier are sorted within a priority queue $Q$, ordered according to a given search policy. CORELS implements several such policies, including Breadth First Search, Depth First Search, and several Best First Searches. While these policies define the order in which the nodes of the prefix tree are ordered (and may affect the convergence speed), note that they do not affect optimality, and must all lead to the same optimal objective function value given sufficient time and memory. At each step of the exploration, the most promising prefix $r$ is popped from the priority queue $Q$ (line 4). If its lower bound is greater than the best objective found so far (*i.e., r* can not lead to a rule list improving the current best objective function), it is discarded. Otherwise, it is used to build a rule list by appending a default prediction $q_0$ (line 6). If the resulting rule list $d$ has a better objective function than the best one reached so far, the current best solution is updated at line 9. Finally, each possible extension of $r$ formed by adding a new rule at the end of $r$ gives a new node which is pushed into the priority queue at line 12. The exploration is completed (and optimality is proved) once the priority queue is empty. Note that efficient data structures are used to cut the prefix tree symmetries: for instance, a prefix permutation map ensures that only the best permutation of every set of rules is kept.

### B.3 HybridCORELS

A key difference between our proposed HybridCORELS algorithms and the original CORELS is that our methods aim at learning prefixes (expressing partial classification functions) while CORELS' purpose is to learn rule lists (classification functions). Both $\text{HybridCORELS}_{\text{Post}}$ and $\text{HybridCORELS}_{\text{Pre}}$ return prefixes (and not rule lists) and take as input an initial best known prefix $r^0$ satisfying the transparency constraint (while the original CORELS takes as input an initial rule list $d^0$). A simple choice for the initial prefix

**Algorithm 1** CORELS

**Input**: Training data $S$ with set of pre-mined antecedents $\Upsilon$; initial best known rule list $d^0$ such that $\mathsf{obj}(d^0, S) = z^0$

**Output**: $(d^*, z^*)$ in which $d^*$ is a rule list with the minimum objective function value $z^*$

```
 1: $(d^c, z^c) \leftarrow (d^0, z^0)$
 2: $Q \leftarrow queue(())$                                   ▷ Initially the queue contains the empty prefix ()
 3: while $Q$ not empty do                                     ▷ Stop when the queue is empty
 4:     $r \leftarrow Q.pop()$
 5:     if $\mathsf{lb}(r, S) < z^c$ then
 6:         $d \leftarrow (r, q_0)$                            ▷ Set default prediction $q_0$ to minimize training error
 7:         $z \leftarrow \mathsf{obj}(d, S) = \frac{\widehat{\mathcal{L}}_S(d)}{|S|} + \lambda \cdot |r|$      ▷ Compute objective $\mathsf{obj}(d, S)$
 8:         if $z < z^c$ then
 9:             $(d^c, z^c) \leftarrow (d, z)$                 ▷ Update best rule list and objective
10:         for $a$ in $\Upsilon \setminus \{a_i \mid \exists r_i \in r, r_i = a_i \to q_i\}$ do      ▷ Antecedent $a$ not involved in $r$
11:             $r_{new} \leftarrow (a \to q)$                 ▷ Set $a$'s consequent $q$ to minimize training error
12:             $Q.push(r \cup r_{new})$                       ▷ Enqueue extension of $r$ with new rule $r_{new}$
13: $(d^*, z^*) \leftarrow (d^c, z^c)$
```

$r^0$ satisfying the transparency constraint is a constant majority prediction: $r^0 \leftarrow [(True \to q_0)]$ (whose transparency is 1.0). In practice, we use such trivial initial solution for all our experiments.

**Algorithm 2** HybridCORELS$_{\text{Post}}$

**Input**: Training data $S$ with set of pre-mined antecedents $\Upsilon$; minimum transparency value $min\_transp$; initial prefix $r^0$ such that $\frac{|S_{r^0}|}{|S|} \geq min\_transp$; pre-trained black-box model $h_c$

**Output**: $(r^*, z^*)$ in which $r^*$ is a prefix with the minimum objective function value $z^*$

```
 1: $(r^c, z^c) \leftarrow (r^0, z^0)$
 2: $Q \leftarrow queue(())$                                   ▷ Initially the queue contains the empty prefix ()
 3: while $Q$ not empty do                                     ▷ Stop when the queue is empty
 4:     $r \leftarrow Q.pop()$
 5:     if $\mathsf{lb}(r, S) < z^c$ then
 6:         $z \leftarrow \frac{\widehat{\mathcal{L}}_{S_r}(r) + \widehat{\mathcal{L}}_{S \setminus S_r}(h_c)}{|S|} + \lambda \cdot |r| + \beta \cdot \frac{|S \setminus S_r|}{|S|}$      ▷ Compute objective $\mathsf{obj}_{\text{post}}(r, S)$
 7:         if $z < z^c$ and $\frac{|S_r|}{|S|} \geq min\_transp$ then
 8:             $(r^c, z^c) \leftarrow (r, z)$                 ▷ Update best prefix and objective
 9:         for $a$ in $\Upsilon \setminus \{a_i \mid \exists r_i \in r, r_i = a_i \to q_i\}$ do      ▷ Antecedent $a$ not involved in $r$
10:             $r_{new} \leftarrow (a \to q)$                 ▷ Set $a$'s consequent $q$ to minimize training error
11:             $Q.push(r \cup r_{new})$                       ▷ Enqueue extension of $r$ with new rule $r_{new}$
12: $(r^*, z^*) \leftarrow (r^c, z^c)$
```

The pseudo-code of HybridCORELS$_{\text{Post}}$ is provided in Algorithm 2. Key modifications include the use of a different objective function (10) at line 6, aimed at evaluating the overall hybrid interpretable model's performances. One can note that the computation of the new objective function $\mathsf{obj}_{\text{post}}(r, S)$ requires access to the pre-trained black-box $h_c$, which is part of the algorithm's inputs. The original CORELS' lower bound is valid and tight for our new objective (as discussed in **Section 4.3**) so we keep this computation unchanged at line 5. Finally, to ensure that the built prefix satisfies a given transparency constraint (9), this condition is verified at line 7 before updating the current best solution at line 8.

The pseudo-code of HybridCORELS$_{\text{Pre}}$ is provided in Algorithm 3. Again, the objective function computation is modified at line 6 to use our proposed $\mathsf{obj}_{\text{pre}}(r, S)$ objective (11). As before, the original lower bound is still valid (as discussed in **Section 4.4**) so we leave it unchanged at line 5. Just like for HybridCORELS$_{\text{Post}}$, the transparency constraint (9) is checked line 7, right before the current best solution update (line 8).

**Algorithm 3** HybridCORELS$_{\text{Pre}}$

**Input**: Training data $S$ with set of pre-mined antecedents $\Upsilon$; minimum transparency value *min_transp*; initial prefix $r^0$ such that $\frac{|S_{r^0}|}{|S|} \geq min\_transp$

**Output**: $(r^*, z^*)$ in which $r^*$ is a prefix with the minimum objective function value $z^*$

```
$(r^c, z^c) \leftarrow (r^0, z^0)$
$Q \leftarrow queue(())$    ▷ Initially the queue contains the empty prefix ()
while $Q$ not empty do    ▷ Stop when the queue is empty
    $r \leftarrow Q.pop()$
    if $\mathsf{lb}(r, S) < z^c$ then
        $z \leftarrow \frac{\widehat{\mathcal{L}}_{S_r}(r) + \mathsf{incons}(S \setminus S_r)}{|S|} + \lambda \cdot |r| + \beta \cdot \frac{|S \setminus S_r|}{|S|}$    ▷ Compute objective $\mathsf{obj}_{\text{pre}}(r, S)$
        if $z < z^c$ and $\frac{|S_r|}{|S|} \geq min\_transp$ then
            $(r^c, z^c) \leftarrow (r, z)$    ▷ Update best prefix and objective
        for $a$ in $\Upsilon \setminus \{a_i \mid \exists r_i \in r, r_i = a_i \rightarrow q_i\}$ do    ▷ Antecedent $a$ not involved in $r$
            $r_{new} \leftarrow (a \rightarrow q)$    ▷ Set $a$'s consequent $q$ to minimize training error
            $Q.push(r \cup r_{new})$    ▷ Enqueue extension of $r$ with new rule $r_{new}$
$(r^*, z^*) \leftarrow (r^c, z^c)$
```

Once the optimal prefix $r^*$ is known, the black-box part can be trained (which is not represented in the pseudo-code) using our proposed specialization scheme as described in **Section 3.1.2**.

Finally, both our proposed approaches are *anytime*: the user can specify any desired running time and memory limits, and the algorithm returns the current best solution and objective value $(r^c, z^c)$ if one of the limits is hit and the priority queue is not empty. Even if optimality is not guaranteed in such case, the ability to precisely bound running times and memory footprints is a very practical feature for real-life applications.

## C Another *Pre-Black-Box* Implementation for HybridCORELS

In this appendix section, we describe another possible implementation of the *Pre-Black-Box* paradigm based on the CORELS algorithm but optimizing a different objective function. We discuss the theoretical differences with the HybridCORELS$_{\text{Pre}}$ algorithm introduced in **Section 4.4** and empirically compare the two methods.

### C.1 HybridCORELS$_{\text{Pre,NoCollab}}$: Theoretical Presentation

We now introduce another possible variant of CORELS implementing the *Pre-Black-Box* paradigm. We coin it HybridCORELS$_{\text{Pre,NoCollab}}$, because contrary to the HybridCORELS$_{\text{Pre}}$ algorithm introduced in **Section 4.4**, the prefix learning phase of HybridCORELS$_{\text{Pre,NoCollab}}$ does not account for the task left to the black-box part. Instead, the prefix is learned to maximize its own accuracy, which results in the remaining examples (that will be handled by the black-box model) being the hardest ones to classify. While black-box specialization could be helpful to deal with such difficult tasks, we observe that, in practice, it has to deal with many inconsistent examples, which considerably lowers its performances.

**Objective** HybridCORELS$_{\text{Pre,NoCollab}}$ builds a prefix $r$ capturing at least a proportion of *min_transp* of the training data (transparency constraint (9)), and minimizing the weighted sum of $r$'s classification error and sparsity:

$$\mathsf{obj}_{\text{pre,nocollab}}(r, S) = \frac{\widehat{\mathcal{L}}_{S_r}(r)}{|S_r|} + \lambda \cdot |r| + \beta \cdot \frac{|S \setminus S_r|}{|S|} \tag{15}$$

**Objective lower bound** CORELS' original lower bound (8) does not hold for objective function (15). Indeed, the difficulty here is that $\mathsf{obj}_{\text{pre,nocollab}}$ quantifies a prefix's error only on the subset of examples that it classifies ($S_r$), hence it is not possible to directly consider the inconsistent examples $\mathsf{incons}(S \setminus S_r)$ as in $\mathsf{lb}$ (8): an extension of $r$ may not capture them at all. To obtain a tight lower bound $\mathsf{lb}_{\text{pre,nocollab}}$, one needs to consider simultaneously the support $S_r$ and errors $\widehat{\mathcal{L}}_{S_r}(r)$ of prefix $r$, as well as the labels cardinalities among each group of inconsistent examples (also called *set of equivalent points* in the context of CORELS (Angelino et al., 2017)). A pre-processing step computes a list $\mathcal{G}$ of *inconsistent groups of examples*. Each group $g \in \mathcal{G}, g \subset S$ is defined by its number of minority examples $min_g$ (those with the least frequent label among group $g$), and its number of majority examples $maj_g$. In fact, our previously introduced count of unavoidable errors uses such groups for its computation: $\mathsf{incons}(S) = \sum_{g \in \mathcal{G}} (min_g)$. For each group $g \in \mathcal{G}$ not captured by prefix $r$ ($g \not\subset S_r$), we verify whether capturing its examples could lower the current prefix's error rate: $c_{p,g} = \mathbb{1}\left[\frac{min_g}{min_g + maj_g} \leq \frac{\widehat{\mathcal{L}}_{S_r}(r)}{|S_r|}\right]$. Then:

$$\begin{aligned}\mathsf{lb}_{\text{pre,nocollab}}(r, S) = & \frac{\widehat{\mathcal{L}}_{S_r}(r) + \sum_{g \in \mathcal{G}, g \not\subset S_r} c_{p,g} \cdot min_g}{|S_r| + \sum_{g \in \mathcal{G}, g \not\subset S_r} c_{p,g} \cdot (min_g + maj_g)} \\ & + (K_r + 1) \cdot \lambda \end{aligned} \tag{16}$$

Finally, $\mathsf{lb}_{\text{pre,nocollab}}$ precisely quantifies the best objective function that can be reached based on prefix $r$, by only capturing inconsistent groups of examples that improve the objective function (lowering the error rate). The definition of $c_{p,g}$ uses a less or equal operator because in case the error rate is unchanged after capturing an additional group of inconsistent examples, the operation should be performed as it would increase the coverage (and the associated regularisation term). There exists a (partial) classification function whose error rate is exactly the one computed in $\mathsf{lb}_{\text{pre,nocollab}}$, so this bound is tight.

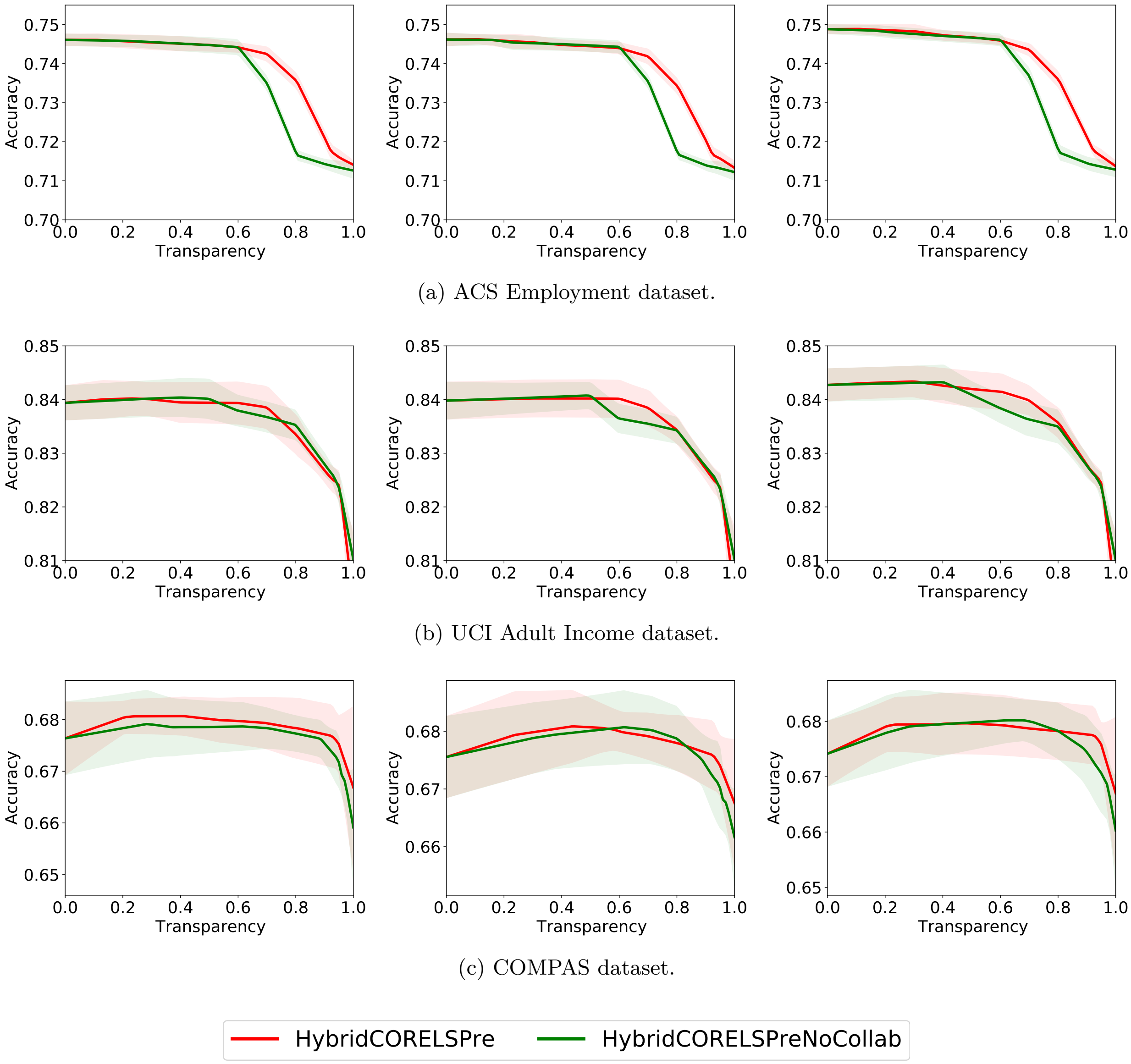
Finally, HybridCORELS$_{\text{Pre,NoCollab}}$ is an exact method: it provably returns a prefix $r$ for which $\mathsf{obj}_{\text{pre,nocollab}}(r, S)$ (15) is the smallest among those satisfying the transparency constraint (9). This means that, given desired transparency level, it produces an optimal prefix (interpretable part of the final hybrid model) in terms of accuracy/sparsity. The pseudo-code of HybridCORELS$_{\text{Pre,NoCollab}}$ is similar to that of HybridCORELS$_{\text{Pre}}$ presented in Algorithm 3, except that the objective function $\mathsf{obj}_{\text{pre}}(r, S)$ and lower bound $\mathsf{lb}(r, S)$ on lines 6 and 5 are replaced by $\mathsf{obj}_{\text{pre,nocollab}}(r, S)$ and $\mathsf{lb}_{\text{pre,nocollab}}(r, S)$, as introduced in equations (15) and (16).

Again, note that within this proposed implementation, the prefix learning phase does not consider the difficulty of the task let to the black-box learning part. For datasets containing inconsistent examples, this could result in sub-optimal overall accuracy in regimes of medium to high transparency, when collaboration between both parts of the hybrid interpretable model is required.

### C.2 HybridCORELS$_{\text{Pre,NoCollab}}$: Empirical Evaluation

We ran the experiments of **Section 5.3** using HybridCORELS$_{\text{Pre,NoCollab}}$ (with the same setup as HybridCORELS$_{\text{Pre}}$), and provide a comparison with HybridCORELS$_{\text{Pre}}$ within Figure 15. The results show that for very low transparency values, HybridCORELS$_{\text{Pre,NoCollab}}$ and HybridCORELS$_{\text{Pre}}$ have very close performances. Indeed, in such regimes, most of the classification task is handled by the black-box part of the model and the absence of collaboration with the interpretable part does not really matter. We observe the same phenomenon in regimes of very high transparency, where most of the examples are classified by the interpretable part. However, in regimes of medium to high transparency, we observe a significant drop of HybridCORELS$_{\text{Pre,NoCollab}}$'s performances. This trend is particularly visible with the ACS Employment dataset. It can be explained by the absence of collaboration between both parts of the model: the prefix learning sacrifices the black-box performances (sending it most of the inconsistent examples) to obtain the most accurate prefix possible. While this policy leads to slightly more accurate interpretable parts compared to the prefixes learned by HybridCORELS$_{\text{Pre}}$, it also harms the overall model accuracy considerably, and the obtained trade-offs are not competitive with those produced by HybridCORELS$_{\text{Pre}}$. As observed in **Section 5.3** with HybridCORELS$_{\text{Pre}}$, on the COMPAS dataset, hybrid models with intermediate transparency values exhibit better test accuracies than the standalone black-box, due to better generalization. Again, this constitutes an argument in favor of the *Pre-Black-Box* paradigm, as this trend was not observed with the other *Post-Black-Box* methods.

We provide in Figure 16 examples of hybrid models found with HybridCORELS$_{\text{Pre}}$ and HybridCORELS$_{\text{Pre,NoCollab}}$ on the same data splits of the ACS Employment dataset and transparencies roughly 80%. We observe, as aforementioned, that the black-boxes trained after the HybridCORELS$_{\text{Pre,NoCollab}}$ prefixes exhibit considerably lower performances. On the other hand, the prefix and black-box parts of the models trained using HybridCORELS$_{\text{Pre}}$ have comparable classification performances, as the former was trained while accounting for the inconsistent samples that would be left for the later to classify.



(a) ACS Employment dataset.

(b) UCI Adult Income dataset.

(c) COMPAS dataset.

Figure 15: Test set accuracy/transparency trade-offs for our two *Pre-Black-Box* variants of HybridCORELS. The Pareto front for each method is represented as a line and the filled bands encode the std across the five data split reruns. Results are provided for several black-boxes: (Left) AdaBoost, (Middle) Random Forests, (Right) Gradient Boosted Trees.

```
if ["age_high" and "Female"] then 0 (acc 71.3%)
else if ["Husband/wife" and "No disability"] then 1 (acc 71.9%)
else if ["age_high" and "Native"] then 0 (acc 64.5%)
else if ["Bachelor's degree" and "No disability"] then 1 (acc 84.8%)
else if ["Reference person" and "No disability"] then 1 (acc 82.3%)
else if ["high school diploma" and "No disability"] then (acc 60.0%)
else
    AdaBoost() (acc 71.9%)
```

(a) HybridCORELS$_{\text{Pre}}$: Test Accuracy 73.7%, Transparency 80.2%.

```
if ["age_low" and "Reference person"] then 1 (acc 82.2%)
else if ["Disability"] then 0 (acc 77.7%)
else if ["age_medium"] then 1 (acc 79.2%)
else if ["age_low" and "Married"] then 1 (acc 69.1%)
else if ["Husband/wife" and "Female"] then 0 (acc 68.6%)
else if ["age_low" and "not own child of householder"] then (acc 59.0%)
else
    AdaBoost() (acc 60.4%)
```

(b) HybridCORELS$_{\text{Pre,NoCollab}}$ : Test Accuracy 71.7%, Transparency 80.3%.

Figure 16: Examples of hybrid interpretable models obtained on the ACS Employment dataset with AdaBoost black-boxes and the same train/validation/test split. The models with transparency closest to 80% were selected. We note that the black-box has worst performance in HybridCORELS$_{\text{Pre,NoCollab}}$ than HybridCORELS$_{\text{Pre}}$ seeing as the prefix sent it the inconsistent examples.