Amazing Things Come From Having Many Good Models

Cynthia Rudin [1 *] **Chudi Zhong** [1] **Lesia Semenova** [1] **Margo Seltzer** [2] **Ronald Parr** [1] **Jiachang Liu** [1]
Srikar Katta [1] **Jon Donnelly** [1] **Harry Chen** [1] **Zachery Boner** [1]

Abstract

The *Rashomon Effect*, coined by Leo Breiman, describes the phenomenon that there exist many equally good predictive models for the same dataset. This phenomenon happens for many real datasets and when it does, it sparks both magic and consternation, but mostly magic. In light of the Rashomon Effect, this perspective piece proposes reshaping the way we think about machine learning, particularly for tabular data problems in the nondeterministic (noisy) setting. We address how the Rashomon Effect impacts (1) the existence of simple-yet-accurate models, (2) flexibility to address user preferences, such as fairness and monotonicity, without losing performance, (3) uncertainty in predictions, fairness, and explanations, (4) reliable variable importance, (5) algorithm choice, specifically, providing advanced knowledge of which algorithms might be suitable for a given problem, and (6) public policy. We also discuss a theory of when the Rashomon Effect occurs and why. Our goal is to illustrate how the Rashomon Effect can have a massive impact on the use of machine learning for complex problems in society.

1. Introduction

Real-world datasets often admit many approximately-equally-good models. Leo Breiman called this phenomenon the *Rashomon Effect*, naming it after a Japanese movie in which four different views of the same murder show no single truth, just many reasonable explanations, for what happened (Breiman, 2001b; Kurosawa, 1950). One might think of the Rashomon Effect as a nuisance that prevents us from getting at a single "true" understanding of the data due to uncertainty, but from another perspective, the Rashomon Effect unlocks a treasure trove of information about the relationship of real datasets to families of predictive models. Harnessing this knowledge has massive practical implications, providing answers to some of the most fundamental questions in machine learning, such as: Is there an accuracy-interpretability trade-off? Which algorithm(s) are suitable for a given dataset? How can we easily (i.e., without solving a difficult optimization problem) find a model that incorporates multiple objectives such as fairness or monotonicity? How can we get stable variable importance estimates? The Rashomon Effect provides surprising insight into all of these questions – and more.

The Rashomon Effect is often present in datasets generated by processes that are nondeterministic, i.e., *noisy* or *uncertain*, including data used for bail and parole decisions, healthcare, and financial loan decisions – high stakes applications. In fact, as we will discuss, it has been proven in specific cases that datasets drawn from noisy processes tend to exhibit a large Rashomon Effect in that there are many approximately-equally accurate models (Semenova et al., 2023). Furthermore, a large Rashomon Effect correlates with the existence of simple-yet-accurate models (Semenova et al., 2022). Hence, there is no accuracy/interpretability trade-off in many domains. The lack of a trade-off has been well-established empirically for tabular data (e.g., see Holte, 1993; Lou et al., 2013; Angelino et al., 2018; McTavish et al., 2022; Liu et al., 2022b; McElfresh et al., 2023). This knowledge has important policy implications, because it explains that black box models have no performance advantage over interpretable models that are easier to administer and use.

Knowing that the Rashomon Effect exists – and the extent to which it exists – *changes the lens through which we view just about everything in machine learning*. The current machine learning paradigm solves problems by finding a single good model. However, understanding that many good models differ dramatically – in terms of variable importance, predictions on individual points, complexity, fairness, etc. – changes how we approach the problem. For instance, we cannot generally assume that any of the variables used heav-

[*] Authors, except the first, are listed in reverse alphabetical order, since there are many good ways to list equally contributing authors. [1]Department of Computer Science, Duke University, Durham, North Carolina, USA [2]Department of Computer Science, University of British Columbia, Vancouver, Canada. Correspondence to: Cynthia Rudin <cynthia@cs.duke.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

ily by one model are important to every well-performing model. We cannot assume that because an algorithm finds a complex model with good test performance that this level of complexity is necessarily needed for obtaining that test performance; similarly, we cannot assume that a complex model has discovered secrets in the dataset that a simpler model could not also find. Knowledge that many equally good models could exist might make us unhappy with the status quo of algorithms that optimize for only one machine learning model, when we could select from many. That is, just *knowing about existence of the Rashomon Effect* shows us that the standard machine learning paradigm that returns only one model is too narrow, and new methods and insights are needed.

We define the *Rashomon set* as the set of models that perform approximately-equally to the best models from a given function class. The first algorithms that quantify the Rashomon Effect by capturing and storing the Rashomon set for nontrivial function classes have been developed only recently (Xin et al., 2022; Zhong et al., 2023; Liu et al., 2022a; Zhu et al., 2023). These algorithms allow users to interact with the Rashomon set to address user preferences, such as fairness concerns and monotonicity constraints. They also allow us to study variable importance in a holistic way, including all of the good models.

We elaborate on how the Rashomon Effect has implications for simplicity, specifically the existence of simple-yet-accurate models (Section 3), flexibility to address user preferences without losing performance (Section 5), uncertainty in predictions, models, and explanations (Section 7), stable variable importance (Section 8), algorithm choice, specifically advance knowledge of which algorithms might be suitable for a given problem (Section 9), and public policy (Section 10). We question the relevance of the classical machine learning paradigm in light of the Rashomon Effect in Section 4 and discuss an alternative paradigm in Section 5. This perspective piece distills work from several technical papers to make them more widely accessible and discusses their link to policy.

2. The Rashomon Effect is Everywhere

The Rashomon Effect occurs when there are many different well-performing models for the same dataset. Standard machine learning (ML) analysis reveals it, but most researchers might not recognize it, because they are not looking for it.

Let us work with a dataset – the FICO dataset from the Explainable ML Challenge (FICO et al., 2018) – though extremely similar results hold for an astounding number of other datasets (Semenova et al., 2022). We will examine the Rashomon Effect for a large model class, large enough to encompass the usual function classes used in machine learning, such as combinations of trees, neural networks, kernel machines, and so on. We applied a variety of machine learning methods to the data, including boosted decision trees, random forest, multi-layer perceptrons, support vector machines, logistic regression, and a 2-layer additive risk model. All of these models have completely different functional forms, from linear models to kernel-based nonparametric models with smooth decision boundaries, to tree-based nonparametric models with sharp decision boundaries, yet most of these models perform comparably, as shown in Table 1. This means all of these models are in the Rashomon set of the large class of functions.

Classifier	Test Accuracy	Test AUC
Random Forest	0.697±0.017	0.757±0.017
Boosted trees	0.723±0.024	0.789±0.028
SVM (linear kernel)	0.720±0.029	0.795±0.027
SVM (RBF kernel)	0.727±0.023	0.799±0.022
8-layer neural network	0.722±0.022	0.792±0.026
Logistic regression	0.731±0.023	0.801±0.028
2-layer additive risk model	0.738±0.020	0.806±0.025

Table 1. Performance of different machine learning models on the 23-feature FICO dataset (Chen et al., 2022) over 10 test folds. They perform similarly. Some of these models (specifically, the 2-layer additive risk model) are interpretable.

Table 2 shows that these different models depend on variables differently. This exemplifies the Rashomon Effect – when the best models that can be (practically) constructed for a given dataset are numerous and diverse. In Table 2, we used simple permutation importance (e.g., see Fisher et al., 2019) to estimate variable importance of each variable to the model's predictions. This type of analysis can be conducted with similar results on many tabular datasets. We will see another way to directly observe the Rashomon Effect in Section 5: by enumerating all of the good models from a given function class. Appendix A shows other ways to measure the Rashomon Effect.

Because there are many good functions, some of these functions are simple.

3. The Rashomon Effect Gives Rise to Simpler-Yet-Accurate Models

When large portions of the function space contain many good models, the Rashomon set is likely large enough to include simpler models. A mathematical explanation for why this is true is illustrated in Figure 1 (Semenova et al., 2022). We have two function classes: a class of complex functions with a large Rashomon set (blue region) and a simpler function class contained in the complex function class (orange dots). Assume that every model in the complex

Classifier	Reliance on	Reliance on	Reliance on
	ExternalRiskEstimate	NumInqLast6M	NetFractionRevolvingBurden
Boosted trees	1.18 ± 0.02	1.01 ± 0.00	1.03 ± 0.01
SVM (RBF kernel)	1.15 ± 0.01	1.01 ± 0.00	1.12 ± 0.00
Logistic regression	1.02 ± 0.00	1.02 ± 0.00	1.23 ± 0.02
2-layer additive risk model	1.01 ± 0.00	1.08 ± 0.01	1.00 ± 0.00

Table 2. Model reliance, calculated by permuting feature values, (the multiplicative version, from Fisher et al., 2019) on three important features across top performing model classes, with standard error across 5 train-test splits. Here, e.g., 1.18 means the loss increases by 18% when a variable is scrambled; in that case, the variable is quite important to the model. Value 1.00 means the variable is not important at all. The ordering of variable importance is inconsistent among these model classes; boosted trees rely most heavily on "ExternalRiskEstimate," SVM relies heavily on both "ExternalRiskEstimate" and "NetFractionRevolvingBurden," logistic regression relies most on "NetFractionRevolingBurden," and the model from Chen et al. (2022) relies most on "NumInqLast6M." The variable "NetFractionRevolingBurden" is not important to the 2-layer additive risk models, but very important to logistic regression.

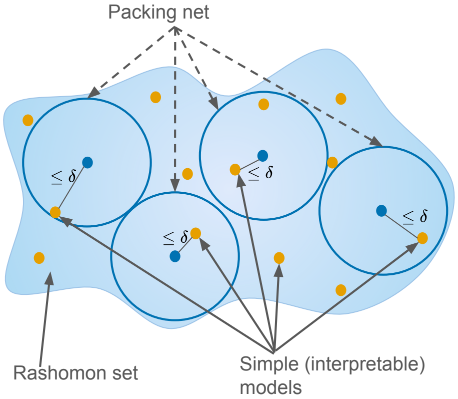


Figure 1. Illustration showing that for hypothesis spaces with good approximating properties, larger Rashomon sets tend to contain multiple simpler models. For every model in the more complex space (blue shaded region), there exists a δ-close model from the simpler space (orange dots). In this illustration, the Rashomon set contains at least four simpler models, which is its 2δ-packing number, where blue dots correspond to the centers of the balls in the packing.

class is "close to" a simpler one, meaning that they are within a radius δ of each other in function space (i.e., the simpler class is a *cover* of the more complex set). Then, the Rashomon set in the complex class contains at least as many functions from the simpler class as its 2δ-packing number, where the packing number is the maximal number of balls in the Rashomon set, the centers of which are at least 2δ apart. From the "closeness" assumption, every ball in the packing contains a simpler model, therefore, the larger the Rashomon set, the more simpler models it contains.

The "closeness" assumption is reasonable. For instance, sparse decision trees serve as a cover for deeper, more complex decision trees, and trees are universal function approximators (Barron, 1993). If the more complex trees are optimized so that they are not so deep that they overfit, shallow trees serve as an even better cover. For instance, Theorem 4.2 in McTavish et al. (2022) shows that any boosted decision tree is equivalent to a single tree with a greater depth, so the set of boosted trees is naturally covered by single trees. Because the closeness assumption generally holds, and because we often have large Rashomon sets, we often find that for tabular data, a single sparse tree (of depth ∼5) can achieve performance similar to that of a boosted decision tree. Figure 2 shows a single tree for the FICO dataset that achieves the accuracy of the black box models shown in Table 1. Thus, as we stated, **for many problems, simple models can perform as well as much more complex models**, and there is no accuracy/interpretability trade-off.

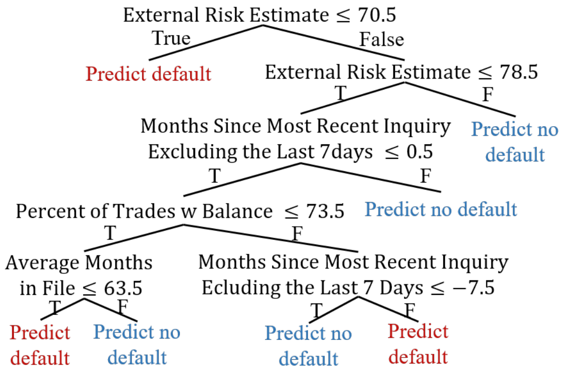

External Risk Estimate ≤ 70.5
True
False
Predict default
External Risk Estimate ≤ 78.5
T
F
Months Since Most Recent Inquiry Excluding the Last 7days ≤ 0.5
Predict no default
T
F
Percent of Trades w Balance ≤ 73.5
Predict no default
T
F
Average Months in File ≤ 63.5
Months Since Most Recent Inquiry Ecluding the Last 7 Days ≤ −7.5
T
F
T
F
Predict default
Predict no default
Predict no default
Predict default

Figure 2. Decision tree: train and test accuracy are both approximately 72%, which is comparable to the best black box algorithms (deep learning and boosted decision trees). This tree has 7 leaves and was obtained in 8.1 seconds by the GOSDT algorithm (Lin et al., 2020; McTavish et al., 2022).

A second simple model, of a different functional form, is shown in Figure 3. This is a sparse generalized additive model (GAM). Interestingly, the GAM has no feature interaction terms yet achieves accuracy that is extremely similar to that of the decision trees, which rely only on feature interactions. Again, this illustrates a case of similar performance from completely different models.

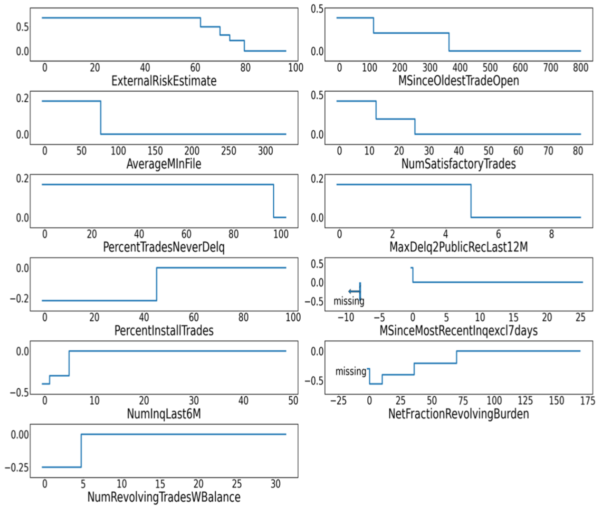


Figure 3. Sparse GAM: the user gets a score for each of the 11 features in this model, from the FastSparse algorithm (Liu et al., 2022b). The sum of scores translates into a risk of defaulting on a loan. The model was obtained in 3.15 seconds. Its test AUC and accuracy are .790 and .723. The cross-validation AUC and accuracy of FastSparse are 0.791 ± 0.010 and $72.4\% \pm 1.2\%$.

This dataset was provided by FICO, the major credit scoring agency in the United States, for a data science competition to discover post-hoc explanations to black box models. It is a particularly interesting dataset, because the competition organizers thought interpretable-yet-accurate models did not exist, but they do. The dataset's variables represent credit history, and its label is whether an individual will default on a loan. The dataset appears to represent a particularly difficult (balanced) subpopulation that has a high risk of loan default relative to the population. If interpretable models demonstrate strong performance on this dataset, which was specifically designed for a benchmarking challenge, we could only imagine the potential success of interpretable models in addressing a much broader range of high-stakes problems.

Interpretable and simple models are easier to verify, easier to debug, and easier to use. However, they are not easier to design, as we discuss shortly.

4. The Standard ML Paradigm is Too Narrow

Among models with equal complexity on a static dataset, statistical learning theory says that it should not matter which model in the Rashomon set we choose – all models with equal complexity should generalize equally well to the test data (e.g., Rudin, 2020, chapter on learning theory). This standard paradigm, where any model that has good cross-validation performance on a static dataset can be trusted, assumes the test data come from the same distribution as the training set, the data need no troubleshooting, and no additional domain knowledge is needed. This is why machine learning methods need only choose one model from the Rashomon set in the standard paradigm.

However, the real world is not an ML benchmark. In the real world, data are messy, the model inputs must be easy to check, domain knowledge can substitute for lack of data or problems with data, and models often need to be easy to understand and/or obey additional domain-specific constraints (Wagstaff, 2012; Rudin & Wagstaff, 2014). The standard ML paradigm makes it difficult to do just about anything except what it was designed for – to produce accurate models on a static dataset.

Black box models have hidden flaws, whereas users trying to design interpretable machine learning models realize quickly that **understandable models have understandable flaws**. This is a key reason why the standard machine learning paradigm often fails in the real world. Selecting just a single, understandable model – ignoring the Rashomon Effect – comes with problems. In our experience, the user is rarely satisfied with the first model that is output by a standard interpretable machine learning algorithm – because they see flaws that can be fixed. Since standard ML algorithms are not interactive, the feedback loop can be fraught and frustrating. This is what we call the **interaction bottleneck**, where users cannot effectively interact with algorithms to improve machine learning models. Fortunately, there is a fix: finding Rashomon sets.

5. A New Paradigm: Finding Rashomon Sets

Because the Rashomon set is large, it often includes *many* simple-yet-good models. Being able to **find all of the good models** from a given simple function class has benefits that we spend the rest of this article discussing, the most important ones appearing in this section.

The first algorithm that finds all good models for a nontrivial function class is the TreeFARMS algorithm, which finds all decision trees with a low regularized risk value (Xin et al., 2022). The second such algorithm is the GAM Rashomon set algorithm, which finds accurate sparse generalized additive models (Zhong et al., 2023). Third is the FasterRisk algorithm, which finds accurate sparse scoring systems (Liu et al., 2022a; Zhu et al., 2023). Figure 4 shows several sparse trees for the COMPAS dataset (Larson et al., 2016) found using TreeFARMS.

The opportunity to search through the Rashomon set means we can optimize multiple objectives simultaneously, which has implications for **constraint handling** and **alignment** with domain knowledge. Typical constraints that one might wish to include are *monotonicity* constraints, where the predicted outcomes increase with a specified set of variables,

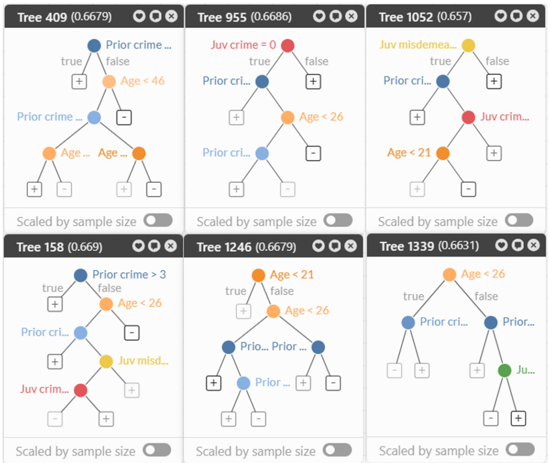
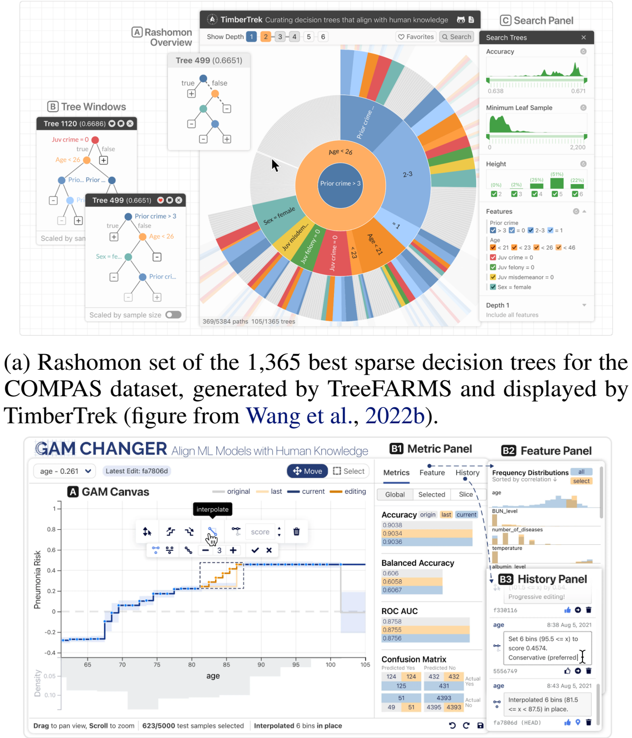


Figure 4. Example sparse decision trees in the Rashomon set of the COMPAS dataset (Larson et al., 2016), found by TreeFARMS.

and *algorithmic fairness* constraints. A simple loop over the Rashomon set will suffice to find all possible answers to a constrained or multi-objective optimization problem. Accordingly, the new paradigm allows us to easily align the model with multiple fairness objectives. **Among equally-good models, the user can choose one that optimally satisfies the criteria**.

Having the Rashomon set at one's fingertips **resolves the interaction bottleneck**. As discussed in the previous section, in the standard ML paradigm, if a user wants to add domain knowledge or constraints to the model, they need to formulate and solve new optimization problems each time they get new feedback from the user. This process is time-consuming, requires possibly many reformulations of optimization problems, and can be exceedingly frustrating. Access to the full Rashomon set resolves this. Using interactive tools, such as TimberTrek and GAMChanger (Wang et al., 2022b; 2021), to explore the Rashomon set, users can find a model that aligns with their domain knowledge in real time, even when that knowledge was not specified in advance. Figure 5 shows screenshots of these interactive tools. Even if the Rashomon set contains hundreds of millions of models, when they are organized effectively, humans can easily explore them.

The computational cost of finding the Rashomon set is much higher than that of finding a single optimal model. Luckily, the TreeFARMS, GAM Rashomon set, and FasterRisk algorithms can handle the computation for most reasonably sized problems for sparse trees, generalized additive models, and scoring systems in minutes (see timing tables in Xin et al., 2022; Zhong et al., 2023; Liu et al., 2022a), which is often acceptable to users. If the Rashomon set is exceedingly large, these algorithms have mechanisms to sample from it or otherwise represent it.



(a) Rashomon set of the 1,365 best sparse decision trees for the COMPAS dataset, generated by TreeFARMS and displayed by TimberTrek (figure from Wang et al., 2022b).

(b) GAM Changer empowers domain experts and data scientists to easily and responsibly align model behaviors with their domain knowledge, via direct manipulation of GAM model weights (figure from Wang et al., 2021).

Figure 5. Interactive tools.

Rashomon sets containing all accurate models contain Rashomon sets for a variety of other objectives. Many of the objectives we consider in machine learning are related to each other. For instance, a highly accurate model probably also has high AUC, low loss, high F1-score, etc. We can take advantage of the relationship between these objectives. If we create a Rashomon set that includes all models with misclassification error below a threshold, we can often prove that it also includes all models with a *different* objective below a different threshold. For example, Xin et al. (2022) showed how all models with high F1-score could be calculated without ever optimizing for F1-score directly. These models are easy to find, because they are contained in a high-accuracy Rashomon set.

Thus, Rashomon sets place substantially more control into the hands of human data analysts.

6. Why Does the Rashomon Effect Occur?

One theory as to why the Rashomon Effect occurs for so many real-world datasets comes from "A Path to Simpler

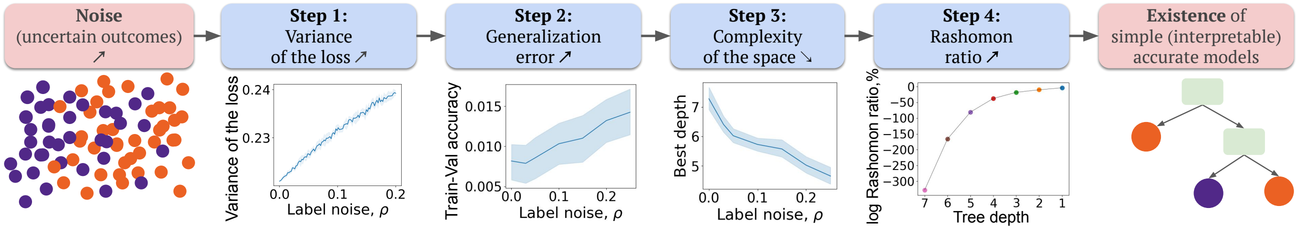


Figure 6. Description of the "path" that starts with noise and leads to larger Rashomon ratios and the existence of simpler models. The plots are created for the COMPAS dataset (Larson et al., 2016).

Models Starts with Noise," by Semenova et al. (2023). This work shows that uncertainty in the outcomes (noise) is one of the causes of the Rashomon Effect. This work provides a 4-step "path" (see Figure 6) that outlines how the uncertainty in the outcomes leads to simpler-yet-accurate models.

In Step 1 of the path, uncertainty in the outcomes (label noise) increases the variance of the loss function with respect to random draws of the data. This means the loss function's values are more uncertain, thus, the user cannot tell with certainty what the value of the loss might be on the test set by looking only at the training set. This leads to Step 2, where the generalization error, i.e., the difference between train and test performance, increases with the variance. Thus, it is easier to overfit the training set.

In Step 3, the user realizes that they are overfitting, perhaps through conducting cross-validation, and simplifies the function class. Even stopping the path here, we arrive at simpler functions. The user had to simplify the function class, because they were not able to generalize, and the test performance would be poor if they did not do it. In other words, even if the data were generated from a complex process, as long as there is label noise, the user would still need to simplify the function class to avoid overfitting to that noise and to reduce cross-validation error.

In Step 4, when the function class is simpler, the *Rashomon Ratio* is large. The Rashomon Ratio is the fraction of the model class that has close-to-optimal loss. It is the fraction of the function class within the Rashomon set. Thinking of these simpler functions as (diverse) representatives of the original function class, a large Rashomon Ratio for the simpler function class equates to a large number of well-performing representatives from the larger function class; i.e., a large Rashomon Effect. Thus, when the data generation process has noise, models with a wide range of complexity can all be part of the same large Rashomon set, leading to a large Rashomon Effect.

Tying this back to the existence of simpler-yet-accurate models from Section 3, the packing argument from Section 3 can be used to show that models from an even simpler class are likely to exist that are approximately as accurate as functions within the user's current function class.

A separate situation in which there is generally a large Rashomon Effect is when the margins between classes (distance between classes in feature space) are large. For example, the MNIST dataset is not a noisy dataset, yet almost any machine learning method – with functions from any type of function class – performs well on it.

Although most real tabular datasets seem to have a large Rashomon Effect, it is easy to construct cases where Rashomon sets are extremely small. An example is where the labels are generated deterministically (no noise) from a complicated function. Incidentally, this is why *approximating* or *explaining* an *already-selected* machine learning model will generally have an accuracy-interpretability trade-off, whereas working with real (noisy) labels will not. In other words, if we try to approximate a fixed, already-selected function f with a simpler function g, then there will likely be a trade-off between the complexity of g and how well it can fit f. This is discussed, for instance, by Kleinberg & Mullainathan (2019) (note that they reversed the terms "interpretability" and "explainability" from us).

7. Uncertainty in Predictions, Fairness, and Explanations

Knowledge of the Rashomon set can illuminate uncertainty that causes problems with ML systems. This includes *underspecification* – where the model development process does not have enough information to learn generalizable domain knowledge – and *predictive multiplicity*, where there are many different predictions made by models within the Rashomon set. If we can calculate the degree of predictive multiplicity in the Rashomon set (how many different predictions are possible), we gain insight into underspecification (many different conclusions). Researchers have recently started to quantify these effects (Marx et al., 2020; Coker et al., 2021; Hsu & Calmon, 2022; Watson-Daniels et al., 2023). Again, analysts typically minimize the *loss* without considering the variation in other quantities, such as *predictions*, *variable importance*, or *fairness*, which is where problems arise.

D’Amour et al. (2020) demonstrate the impact of underspecification for several industry-scale learning tasks, such as medical imaging with eye and skin images, clinical risk prediction with electronic healthcare records, and pronoun affiliation with large language models; they observe that even the *choice of random seed* can dramatically impact the behavior of the final model – a serious consequence of overlooking the powerful Rashomon Effect.

Consider benchmarking challenges in algorithmic fairness. Here, one would use a special “biased” dataset, compute a model for it, find unfairness in this model, and fix the problem. However, Cooper et al. (2024) points out that on these special datasets, if one averages over bootstrap samples to find a stable model, then such a model *is already fair*. In other words, prior to Cooper et al. (2024), the improvement that fairness researchers were seeing could be a mirage due to the Rashomon Effect – they were only considering one model from the Rashomon set that happened to be biased. This follows the work of Rodolfa et al. (2021); Coston et al. (2021); Black et al. (2022; 2024) showing that although there can exist an accuracy-fairness trade-off in theory (Kleinberg, 2018), it may not exist in practice due to the Rashomon Effect.

Rather than ignoring the Rashomon Effect, we should make use of it. We could, for instance, answer questions such as “How important can this variable possibly be for all good models?” (Fisher et al., 2019), or “Can I create a model with sparser counterfactual explanations?” (Sun et al., 2024), or perhaps “What is the largest and smallest my prediction could be from all of the good models?” (Coker et al., 2021). We can also visualize the Rashomon Effect by projecting the Rashomon set into variable importance space (Dong & Rudin, 2020), where we can see how much every model in the Rashomon set depends on each variable. Or, we could leverage the Rashomon set to create stable variable importance values, discussed next.

8. Stable Variable Importance

Measuring the global significance of a variable in predicting an outcome holds paramount importance in scientific exploration and critical decision-making. Two important examples are genetics (e.g., Wang et al., 2020; Novakovsky et al., 2022), where the goal is to figure out which genes have unique information for predicting traits, and criminal justice, where mistakes in variable importance analysis have led to confusion and accusations of racial bias (see Larson et al., 2016; Rudin et al., 2020). Traditional approaches generally assess variable importance based on a *single* model trained on a specific dataset, but this framework does not account for the Rashomon Effect. As we know, failing to consider it can lead different researchers to draw divergent conclusions from identical data, based on identifying different variables as important. Along with the Rashomon Effect, another issue in variable importance is the lack of reproducibility: a variable importance estimate can change amid reasonable data perturbations (e.g., swapping out on observation). One solution is provided by the Rashomon Importance Distribution (RID), which quantifies the importance of a variable across the set of all good models and across perturbations to the original dataset using almost any variable importance metric of interest (Donnelly et al., 2023). By considering variations of the data through bootstrapping, and considering the Rashomon set for each bootstrap sample to produce a distribution of variable importance values for each variable, RID can obtain much more stable variable importance calculations. It is able to recover variables that are important to complex data generation processes more accurately than other approaches, demonstrating how leveraging knowledge of the Rashomon Effect can be helpful for scientific discovery.

9. Which Algorithm Should I Use?

A perennial question in machine learning is about the match of algorithms to problems: which machine learning algorithm is likely to work for my data? For image and text data there are clear current answers that take advantage of the structure in these data types (e.g., CNNs and transformers), but for tabular data there is not – most machine learning algorithms perform equally well. (In fact, researchers have had to compile special “hard” datasets because it is uncommon to find cases where different ML algorithms perform differently, see McElfresh et al., 2023). From what we have discussed above, the answer depends on the level of noise in the outcomes.

For predicting criminal recidivism, where we predict months or years in advance whether someone will commit a crime, the randomness in this process means that simpler models will tend to perform as well as complex models. Thus, we would expect that boosted decision trees (Freund & Schapire, 1997), random forest (Breiman, 2001a), or neural networks provide no performance advantage over sparse additive models or the type of simple scoring systems that are often used for this purpose by the criminal justice system. Empirical evidence on recidivism prediction supports this (see e.g., Wang et al., 2022a; Zeng et al., 2017; Tollenaar & van der Heijden, 2013). Results of this flavor would be expected to hold for loan default predictions, and we presented empirical evidence based on the FICO dataset earlier. This same reasoning process holds for many healthcare prediction problems (readmission, mortality, e.g., see Zhu et al., 2023). In other words, simply by knowing the type of data and the level of noise in the outcome, we can determine whether methods that produce optimized simpler models are likely to be sufficiently accurate.

It is important to note that there is a performance difference between algorithms such as CART (Breiman et al., 1984) and C4.5 (Quinlan, 1993) from the 1980s and 1990s and more modern algorithms. As shown by Xin et al. (2022), CART rarely produces models within the Rashomon set of a dataset, even when subsampling data numerous times and rerunning CART for each subsample. This means these older algorithms do not achieve an accuracy/sparsity balance that is as good as more modern algorithms. (Additional experiments for CART vs. more modern tree algorithms appear in Lin et al., 2020; McTavish et al., 2022).

Thus, in terms of best practices for standard ("noisy") tabular data in cases where interpretability is important, one would typically first *find baseline performance using black box models.* Then, one would *try to match baseline performance using modern interpretable modeling algorithms* (from the 2020s rather than the 1980s). The easiest interpretable ML algorithm to start with is probably FastSparse (Liu et al., 2022b), which produces sparse generalized additive models. Decision trees are a much harder class to optimize, so when working with them, it is useful to use a reduction in the search space provided by GOSDT+Guesses (McTavish et al., 2022; Lin et al., 2020); here, the splits from boosted decision trees are used for constructing single high-accuracy trees. There are quite a lot of interpretable ML algorithms available that produce models of a variety of functional forms. Scoring systems (e.g., as produced by the FasterRisk algorithm of Liu et al., 2022a), are extremely popular in medicine and criminal justice. For linear models, OKRidge (Liu et al., 2023) can find sparse solutions with provable optimality. Rule sets (e.g., Wang et al., 2017) are simple logical models. For datasets requiring more complex models, try GA2M models (Lou et al., 2013), which are additive models with pairwise interactions. Or, as is typically done in credit risk scoring, one could create several smaller models (subscales), and combine them using a small model, as in the 2-layer additive risk model from Chen et al. (2022) that was used in Table 1. Usually each subscale represents a category of features (e.g., credit delinquency features, or satisfactory trade features).

Finally, assuming these "first try" interpretable models are flawed in some way that a user can identify, we suggest *allowing the user to explore the Rashomon set using an algorithm and interface* such as TreeFARMS & TimberTrek or GAM Rashomon set & GAM Changer. This should yield a model suitable for further consideration.

10. Policy Implications

Knowledge of the Rashomon Effect can be used to deliver significant positive impacts to society, including the development of fairer and more interpretable models.

Currently, policy makers have started to govern the "right to explanation" for certain algorithmic decisions (Wikipedia, 2024). However, companies often do not want to provide models that could provide an advantage to competitors. This tension between a desire to preserve secrecy and mandated explanations leads to them providing narrow explanations that can be both misleading and incomplete, rather then genuinely transparent. Explanations are generally *post hoc*, which introduces several possible problems. First, they might be unfaithful to the underlying reasoning process, e.g., "You were denied a loan due to factors A and B," when, in fact, the loan denial was due to different factors. Second, the explanations might be so incomplete as to be practically useless, e.g., "Factors A and B are important in our decision," with no further explanation of how they were used and whether other factors might also be important. A person receiving an explanation has no way to determine the quality of that explanation. Problems with explanations have been discussed at length (e.g., Adebayo et al., 2018; Rudin, 2019; Yanagawa & Sato, 2024; Han et al., 2022). Essentially, black box models, even when supplemented with explanations, create barriers for individuals to examine and question the models, effectively allowing model designers to hide their flaws.

Interpretable models do not have any of these issues. Their explanations must be faithful and complete by design. They are much easier to troubleshoot and use in practice. And, as we discussed in Sections 3 and 6, the Rashomon Effect theoretically explains why and when interpretable models perform as well as their black box counterparts. For these reasons, **interpretable models should be used by default for many high-stakes decisions using machine learning**. Thus, for applications such as criminal recidivism, we should default to interpretable models when we know the outcomes are noisy and where empirical evidence on similar problems has confirmed that interpretable models perform well (see, e.g., Wang et al., 2022a; Zeng et al., 2017). Exceptions can be made in cases where models are 100% accurate (e.g., lesion detection in medical images), in cases where no reason is needed (e.g., medical image segmentation), or in cases where there is no practical way to create an interpretable model. However, since we can now find the Rashomon set, as discussed in Section 5, making it easier to build interpretable models, there is often no excuse to continue the use of black boxes.

Another policy implication involves other types of fairness besides simplicity. We can find the "most fair" model within the Rashomon set, according to any fairness metric, including recourse (e.g., Black et al., 2022), and thus **can verify claims about whether there exists a fairer-yet-accurate model for a given dataset.**

Even though interpretability, uncertainty, and fairness are essential to AI in practice and policy – with the Rashomon Effect being central to all of them – these topics are touched upon only superficially in most of today's academic courses. With respect to interpretability, most courses introduce no techniques more modern than CART (Breiman et al., 1984) and C4.5 (Quinlan, 1993). Information on post hoc explanations is much more widespread, sometimes (unfortunately) using the terminology "interpretable" to describe them. A review of interpretable machine learning appears in Rudin et al. (2022), and course material is available at Rudin (2020). Policy makers can fund **ethical AI education, which will inevitably involve the Rashomon Effect** since it determines whether trade-offs can exist between performance and ethical AI objectives.

11. Conclusion

The Rashomon Effect shows us that among models with similar *loss*, there are a multitude of models with different *properties*, including various levels of simplicity, fairness, and explanations/variable importance values.

The ability to capture Rashomon sets and display them to users addresses what is arguably the hardest open problem in interpretable machine learning – incorporating human interaction. Solving the interaction bottleneck can have a major impact on our ability to troubleshoot and add constraints, which, in turn, could have a major impact on whether machine learning models can be used in high-stakes decisions.

We do not believe that we have truly grasped the full extent of the Rashomon Effect yet, but we can already see that its impact on practical machine learning will be enormous. It forces us to change the way we think – even back to the fundamentals of ML. Since we formulate ML algorithms in terms of trade-offs between objectives, we tend to think that trade-offs among these objectives must then exist in the models they create. This is – surprisingly – wrong.

Acknowledgements

We acknowledge support from DOE DE-SC0023194, NSF IIS-2130250, NIDA R01 DA054994, NSF IIS-2147061, and a grant from Fujitsu. We also acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC). Nous remercions le Conseil de recherches en sciences naturelles et en génie du Canada (CRSNG) de son soutien.

Impact Statement

There are significant societal impacts discussed in this work, with the most important points summarized as: (1) Rashomon sets often admit many good models, giving rise to the existence of high-performing models that obey constraints such as interpretability and fairness; such constraints are crucial in high-stakes settings. (2) Machine learning algorithms now exist within the new Rashomon set paradigm. These algorithms can find whole Rashomon sets for a given dataset, mitigating the *interaction bottleneck*, and allowing users to easily create usable machine learning models for a huge variety of applications. (3) We can determine, before seeing any data, and by knowing only that noise is present in the data generation process, whether a large Rashomon set will exist, and (thus) whether simpler and/or fairer well-performing models will exist. Policy-makers can use this information as evidence for mandating that interpretable models be used for many high-stakes decisions by default. In this way, knowledge of the Rashomon set and its origins can help make the practical uses of machine learning safer and fairer across society. (4) *Knowledge* of the Rashomon Effect changes the way we view just about everything in machine learning, including uncertainty, variable importance measurements, interpretability, fairness, interactivity, and even the classical paradigm of machine learning.

References

Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., and Kim, B. Sanity checks for saliency maps. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.

Ahanor, I., Medal, H., and Trapp, A. C. Diversitree: Computing diverse sets of near-optimal solutions to mixed-integer optimization problems. *arXiv preprint arXiv:2204.03822*, 2022.

Angelino, E., Larus-Stone, N., Alabi, D., Seltzer, M., and Rudin, C. Learning certifiably optimal rule lists for categorical data. *Journal of Machine Learning Research*, 18 (234):1–78, 2018.

Barron, A. R. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Transactions on Information Theory*, 39(3):930–945, 1993.

Black, E., Raghavan, M., and Barocas, S. Model multiplicity: Opportunities, concerns, and solutions. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, pp. 850–863, 2022.

Black, E., Koepke, J. L., Kim, P., Barocas, S., and Hsu, M. Less discriminatory algorithms. *Georgetown Law Journal*, 113(1), 2024. Washington University in St Louis Legal Studies Research Paper (forthcoming).

Breiman, L. Random Forests. *Machine Learning*, 45(1): 5–32, 2001a.

Breiman, L. Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical Science*, 16(3):199–231, 2001b.

Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. *Classification and Regression Trees*. CRC press, 1984.

Chen, C., Lin, K., Rudin, C., Shaposhnik, Y., Wang, S., and Wang, T. A holistic approach to interpretability in financial lending: Models, visualizations, and summary-explanations. *Decision Support Systems*, 152:113647, 2022.

Coker, B., Rudin, C., and King, G. A theory of statistical inference for ensuring the robustness of scientific results. *Management Science*, 67(10):6174–6197, 2021.

Cooper, A. F., Lee, K., Choksi, M., Barocas, S., Sa, C. D., Grimmelmann, J., Kleinberg, J., Sen, S., and Zhang, B. Arbitrariness and prediction: The confounding role of variance in fair classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 22004–22012, Mar. 2024.

Coston, A., Rambachan, A., and Chouldechova, A. Characterizing fairness over the set of good models under selective labels. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 139, pp. 2144–2155, 18–24 Jul 2021.

Dong, J. and Rudin, C. Exploring the cloud of variable importance for the set of all good models. *Nature Machine Intelligence*, 2(12):810–824, 2020.

Donnelly, J., Katta, S., Rudin, C., and Browne, E. P. The rashomon importance distribution: Getting RID of unstable, single model-based variable importance. In *Neural Information Processing Systems (NeurIPS)*, 2023.

D'Amour, A., Heller, K., Moldovan, D., Adlam, B., Alipanahi, B., Beutel, A., Chen, C., Deaton, J., Eisenstein, J., Hoffman, M. D., et al. Underspecification presents challenges for credibility in modern machine learning. *Journal of Machine Learning Research*, 2020.

FICO, Google, Imperial College London, MIT, University of Oxford, UC Irvine, and UC Berkeley. Explainable Machine Learning Challenge. `https://community.fico.com/s/explainable-machine-learning-challenge`, 2018.

Fisher, A., Rudin, C., and Dominici, F. All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. *Journal of Machine Learning Research*, 20(177):1–81, 2019.

Freund, Y. and Schapire, R. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, August 1997.

Han, T., Srinivas, S., and Lakkaraju, H. Which explanation should I choose? a function approximation perspective to characterizing post hoc explanations. In *Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 5256–5268, 2022.

Holte, R. C. Very simple classification rules perform well on most commonly used datasets. *Machine Learning*, 11: 63–90, 1993.

Hsu, H. and Calmon, F. Rashomon capacity: A metric for predictive multiplicity in classification. In *Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 28988–29000, 2022.

Kleinberg, J. Inherent trade-offs in algorithmic fairness. *SIGMETRICS Perform. Eval. Rev.*, 46(1):40, June 2018.

Kleinberg, J. and Mullainathan, S. Simplicity creates inequity: Implications for fairness, stereotypes, and interpretability. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, EC '19, pp. 807–808, 2019.

Kurosawa, A. Rashomon. RKO Radio Pictures, 1950.

Larson, J., Mattu, S., Kirchner, L., and Angwin, J. How we analyzed the COMPAS recidivism algorithm. *ProPublica*, 2016.

Lin, J., Zhong, C., Hu, D., Rudin, C., and Seltzer, M. Generalized and scalable optimal sparse decision trees. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 6150–6160, 2020.

Liu, J., Zhong, C., Li, B., Seltzer, M., and Rudin, C. Fasterrisk: Fast and accurate interpretable risk scores. In *Neural Information Processing Systems (NeurIPS)*, 2022a.

Liu, J., Zhong, C., Seltzer, M., and Rudin, C. Fast sparse classification for generalized linear and additive models. In *Proceedings of Artificial Intelligence and Statistics (AISTATS)*, 2022b.

Liu, J., Rosen, S., Zhong, C., and Rudin, C. OKRidge: Scalable optimal k-sparse ridge regression. In *Neural Information Processing Systems (NeurIPS)*, 2023.

Lou, Y., Caruana, R., Gehrke, J., and Hooker, G. Accurate intelligible models with pairwise interactions. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 623–631, 2013.

Marx, C., Calmon, F., and Ustun, B. Predictive multiplicity in classification. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 6765–6774, 2020.

Mata, K., Kanamori, K., and Arimura, H. Computing the collection of good models for rule lists. *arXiv preprint arXiv:2204.11285*, 2022.

McElfresh, D., Khandagale, S., Valverde, J., C, V. P., Feuer, B., Hegde, C., Ramakrishnan, G., Goldblum, M., and White, C. When do neural nets outperform boosted trees on tabular data? In *Neural Information Processing Systems (NeurIPS)*, 2023.

McTavish, H., Zhong, C., Achermann, R., Karimalis, I., Chen, J., Rudin, C., and Seltzer, M. Fast sparse decision tree optimization via reference ensembles. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 9604–9613, 2022.

Novakovsky, G., Dexter, N., Libbrecht, M. W., Wasserman, W. W., and Mostafavi, S. Obtaining genetics insights from deep learning via explainable artificial intelligence. *Nature Reviews Genetics*, pp. 1–13, 2022.

Quinlan, J. R. *C4.5: programs for machine learning*, volume 1. Morgan Kaufmann, 1993.

Rodolfa, K. T., Lamba, H., and Ghani, R. Empirical observation of negligible fairness–accuracy trade-offs in machine learning for public policy. *Nature Machine Intelligence*, 3:896—904, October 2021.

Rudin, C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1:206–215, May 2019.

Rudin, C. *Intuition for the Algorithms of Machine Learning*. self-published at https://users.cs.duke.edu/~cynthia/teaching.html, 2020.

Rudin, C. and Wagstaff, K. L. Machine learning for science and society. *Machine Learning*, 95(1), 2014.

Rudin, C., Wang, C., and Coker, B. The *Age* of secrecy and unfairness in recidivism prediction. *Harvard Data Science Review*, 2(1), 1 2020.

Rudin, C., Chen, C., Chen, Z., Huang, H., Semenova, L., and Zhong, C. Interpretable machine learning: Fundamental principles and 10 grand challenges. *Statistics Surveys*, 16:1–85, 2022.

Semenova, L., Rudin, C., and Parr, R. On the existence of simpler machine learning models. In *ACM Conference on Fairness, Accountability, and Transparency (ACM FAccT)*, 2022.

Semenova, L., Chen, H., Parr, R., and Rudin, C. A path to simpler models starts with noise. In *Neural Information Processing Systems (NeurIPS)*, 2023.

Smith, G., Mansilla, R., and Goulding, J. Model class reliance for random forests. In *Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 22305–22315, 2020.

Sun, Y., Chen, Z., Orlandi, V., Wang, T., and Rudin, C. Sparse and faithful explanations without sparse models. In *Proc. Artificial Intelligence and Statistics (AISTATS)*, 2024.

Tollenaar, N. and van der Heijden, P. Which method predicts recidivism best?: A comparison of statistical, machine learning and data mining predictive models. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 176(2):565–584, 2013.

Wagstaff, K. L. Machine learning that matters. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 1851–1856, 2012.

Wang, C., Han, B., Patel, B., and Rudin, C. In Pursuit of Interpretable, Fair and Accurate Machine Learning for Criminal Recidivism Prediction. *Journal of Quantitative Criminology*, pp. 1–63, 2022a.

Wang, F., Huang, S., Gao, R., Zhou, Y., Lai, C., Li, Z., Xian, W., Qian, X., Li, Z., Huang, Y., et al. Initial whole-genome sequencing and analysis of the host genetic contribution to COVID-19 severity and susceptibility. *Cell Discovery*, 6(1):83, 2020.

Wang, T., Rudin, C., Doshi-Velez, F., Liu, Y., Klampfl, E., and MacNeille, P. A Bayesian framework for learning rule sets for interpretable classification. *Journal of Machine Learning Research*, 18(70):1–37, 2017.

Wang, Z. J., Kale, A., Nori, H., Stella, P., Nunnally, M., Chau, D. H., Vorvoreanu, M., Vaughan, J. W., and Caruana, R. Gam changer: Editing generalized additive models with interactive visualization. *Advances in Neural Information Processing Systems, Bridging the Gap: From Machine Learning Research to Clinical Practice (Research2Clinics) Workshop*, 2021.

Wang, Z. J., Zhong, C., Xin, R., Takagi, T., Chen, Z., Chau, D. H., Rudin, C., and Seltzer, M. Timbertrek: Exploring and curating sparse decision trees with interactive visualization. In *2022 IEEE Visualization and Visual Analytics (VIS)*, pp. 60–64. IEEE, 2022b.

Watson-Daniels, J., Parkes, D. C., and Ustun, B. Predictive multiplicity in probabilistic classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 10306–10314, 2023.

Wikipedia. Right to explanation. https://en.wikipedia.org/wiki/Right_to_explanation, 2024.

Xin, R., Zhong, C., Chen, Z., Takagi, T., Seltzer, M., and Rudin, C. Exploring the whole Rashomon set of sparse decision trees. In *Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 14071–14084, 2022.

Yanagawa, M. and Sato, J. Seeing is not always believing: Discrepancies in saliency maps. *Radiology: Artificial Intelligence*, 6(1):e230488, 2024.

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

Zhong, C., Chen, Z., Liu, J., Seltzer, M., and Rudin, C. Exploring and interacting with the set of good sparse generalized additive models. In *Neural Information Processing Systems (NeurIPS)*, 2023.

Zhu, C. Q., Tian, M., Semenova, L., Liu, J., Xu, J., Scarpa, J., and Rudin, C. Fast and interpretable mortality risk scores for critical care patients. *arXiv preprint arXiv:2311.13015*, 2023.

A. Metrics to Gauge the Rashomon Effect

There are many different ways to assess the Rashomon Effect.

- For measuring the **size** of Rashomon set: TreeFARMS (Xin et al., 2022) computes the number of sparse decision trees. The GAM Rashomon set algorithm (Zhong et al., 2023) computes the number of unique support sets (for GAMs, including linear and additive models); the Rashomon set includes a convex set of models for each support set. CorelsEnum of Mata et al. (2022) enumerates the Rashomon set of rule lists, while DiversiTree of Ahanor et al. (2022) gives a set of diverse, close-to-optimal mixed integer programming solutions. For ridge regression, the size of the Rashomon set can be computed in closed-form (Semenova et al., 2022).
- For measuring **diversity of predictions** (for classifiers), we can use the pattern diversity metric of Semenova et al. (2022) or the pairwise disagreement of Black et al. (2022). The ambiguity and discrepancy metrics of Marx et al. (2020) can further help to understand the conflicting predictions from the Rashomon set's models. For example, ambiguity tells how many people's bail decision could be changed by using a different model from the Rashomon set, while discrepancy tells us the model in the Rashomon set with the most bail decisions changed relative to a baseline (deployed) model. The Hacking Interval framework of Coker et al. (2021) contains calculations that show maximum and minimum predictions within the Rashomon set for several different types of algorithms.
- For measuring **variable importance diversity**, we can use Model Class Reliance of Fisher et al. (2019) or Smith et al. (2020) to get a range of variable importance values in the Rashomon set. We can visualize the "cloud" of variable importance using the approach of Dong & Rudin (2020), which plots each model in variable importance space.
- For **probabilistic classification**, the Rashomon capacity metric of Hsu & Calmon (2022) can be used, or probabilistic ambiguity/discrepancy of Watson-Daniels et al. (2023).
- The Rashomon ratio or pattern Rashomon ratio of Semenova et al. (2022), as well as the fraction of good models in the hypothesis space, can help to understand the **simplicity of the learning problem**.