EVOTER: Evolution of Transparent Explainable Rule-sets

Hormoz Shahrzad[1,2], Babak Hodjat[1], and Risto Miikkulainen[1,2]
[1]Cognizant AI Labs; [2]The University of Texas at Austin

ABSTRACT

Most AI systems are black boxes generating reasonable outputs for given inputs. Some domains, however, have explainability and trustworthiness requirements that cannot be directly met by these approaches. Various methods have therefore been developed to interpret black-box models after training. This paper advocates an alternative approach where the models are transparent and explainable to begin with. This approach, EVOTER, evolves rule-sets based on simple logical expressions. The approach is evaluated in several prediction/classification and prescription/policy search domains with and without a surrogate. It is shown to discover meaningful rule sets that perform similarly to black-box models. The rules can provide insight to the domain, and make biases hidden in the data explicit. It may also be possible to edit them directly to remove biases and add constraints. EVOTER thus forms a promising foundation for building trustworthy AI systems for real-world applications in the future.

1 INTRODUCTION

Most of today's popular and powerful Artificial intelligence and machine learning approaches rely on inherently black-box methods like neural networks, deep learning, and random forests. Therefore, they are difficult to explain, and their functionality in critical/sensitive use-cases cannot be thoroughly understood. It is also difficult to modify them to remove biases or add known constraints. In some domains, such lack of transparency can lead to costly failures (e.g., in self-driving cars) and in others, mandatory regulations (e.g. insurance) or ethical concerns (e.g. gender or racial biases) cannot be met [35].

Methods have been developed to try to explain behavior of black-box models (e.g., deep learning networks) by interrogating them after training. In this manner, it is possible to e.g. identify the areas of the input to which the model is paying attention when making a prediction. However, post-training interrogation-based explanations are not always accurate or complete, and they can be open to interpretation.

Interpretability and explainability, therefore, are different concepts [17, 19]. Interpretability can be defined as "the ability to explain or to present in understandable terms to a human" [8]. Another popular definition is "the degree to which a human can understand the cause of a decision" [23]. These definitions point to behavior-based interpretation of systems where the actual cause of the behavior is not transparent. In contrast, the internal processes of explainable models are transparent and can be understood in mechanical terms. As a result, they can be audited for compliance, which is useful for various stakeholders from data scientists and business owners to risk analysts and regulators [1, 4].

This paper introduces such an approach, evolution of transparent explainable rule-sets, or EVOTER. These rule sets are transparently explainable, and lead to domain insights that would be difficult to achieve through other machine learning methods. EVOTER is general and effective, as demonstrated through in several benchmark problems and real-world domains.

2 BACKGROUND

In building explainable AI, there are two main directions: (1) explain behavior of existing black-box models, or (2) use a modeling technique that is inherently explainable in the first place.

The former applies primarily to deep learning models, which are difficult to explain because behavior is defined by nonlinear interactions of a very large number of parameters (i.e. connection weights). The most common approaches analyze opaque models to interpret their behavior locally. For instance, they may extract feature importance for every individual prediction, or perform localized distillation based on sub-spaces [1, 12, 18, 20, 27, 36]. However, they do not result in a comprehensive characterization of the solution.

Another approach is to divide the responsibilities of the system into three models: the predictor model, which predicts the outcomes of a certain decision, a prescriptor model, which recommends a decision for a given context, and an uncertainty model [10, 21, 26]. In such a setup, upon getting a decision recommendation, it is possible to ask the 'Why' question, for which the predictor will provide the answer "because I predict these outcomes as a result of this decision". This, of course, is an indirect explanation of the behavior of the prescriptor model, and does not explain the prediction. However, in this setup it is also possible to provide a user with a scratchpad, allowing the user to explore alternative decisions and their outcomes. In this manner, the user can verify that the best decisions are indeed made.

In the second direction, the substrate of the model is replaced with an inherently explainable structure. Instead of running gradient descent in a deep neural network, for instance, a rule-based substrate is evolved. Thus, constructing the model results in a human-readable set of rules or equations. By reviewing this set, it is possible to understand what input features are used, and how they are brought together in rules to make a decision. It is thus possible to explain, on a case-by-case basis, the behavior of the model, in an exact human-readable form.

A number of early such systems were based on a fixed- or variable-length-chromosome genetic algorithm (GA), or a tree- or list-based genetic programming (GP) [16, 24, 25, 32]. The rules were propositional logic expressions where each term is built from a feature compared against a constant, for example: IF (Quality = Medium OR High) AND (Advertisement = Yes OR Telemarketing = Yes) AND (Gifts = Yes) AND (Sales Profile = Good OR Medium) THEN (Profit = Good) ELSE (Profit = Low)[32]. These techniques were applied to several datasets in the UCI machine learning repository [9]. In contrast in EVOTER, these rules are expanded to more powerful syntax that allows comparing features, performing linear and nonlinear operations, and observing time lags. This approach has made more complex applications possible, such as trading in the stock market and predicting sepsis from blood pressure time series [13, 30].

How does one determine which explainability technique to apply to a given ML model? Post-facto explainability methods are useful for pre-existing deep learning based models that are hard to replace given the cost of training. In many cases, deep learning is the preferred model type, especially in very large data problems such as unstructured text, sound, images, and video. In problems where feature engineering has produced a structured data set with familiar input and output features (e.g., tabular data), rule-set evolution can be competitive, and in some cases even more accurate, making it a preferable choice. However, this method can also be applied to unstructured data sets with continuous input or output features, and deep learning methods can sometimes do well in structured and tabular data problems. So the second, equally important consideration is whether interpretability is sufficient, or whether transparent explainability is required. If explainability is for example mandated by regulations (as it is e.g. in some insurance use-cases), then it is important to consider transparently explainable modeling techniques such as EVOTER in this paper.

$$\begin{aligned}
< \mathit{rules} > &::= < \mathit{rule} > \mid < \mathit{rule} > \textbf{OR} < \mathit{rules} > \\
< \mathit{rule} > &::= < \mathit{conditions} > \longrightarrow [< \mathit{certainty} >] < \mathit{action} > \\
< \mathit{conditions} > &::= < \mathit{condition} > \mid < \mathit{condition} > \textbf{AND} < \mathit{conditions} > \\
< \mathit{condition} > &::= < \mathit{leading} >< \mathit{operator} >< \mathit{trailing} > \\
< \mathit{leading} > &::= < \mathit{coefficient} > * < \mathit{feature} >^{[<\mathit{power}>]} \ [(\mathit{lag})] \\
< \mathit{trailing} > &::= < \mathit{leading} > \mid < \mathit{value} > \\
< \mathit{operator} > &::= \text{"} < \text{"} \ \mid \ \text{"} \leq \text{"} \ \mid \ \text{"} > \text{"} \ \mid \ \text{"} \geq \text{"} \\
< \mathit{coefficient} > &::= 0.[0-9]+ \\
< \mathit{value} > &::= \text{(a number between min and max feature value)} \\
< \mathit{power} > &::= [1-3]\text{(keep it to more meaningful values.)} \\
< \mathit{lag} > &::= [0-9]\text{(or the desired lag range to be explored.)} \\
< \mathit{feature} > &::= \text{(an arbitrary input feature)} \\
< \mathit{action} > &::= \text{(a predicted class or prescribed action)} \\
< \mathit{certainty} > &::= < \mathit{coefficient} > \ \text{(confidence of a prediction or action)}
\end{aligned}$$

Figure 1: BNF grammar for rule set representation in EVOTER.

3 METHOD

Rule sets in EVOTER are based on propositional logic expressions. They are collections of statements of the form “IF antecedent A is met THEN consequence B occurs”; the antecedents are conjunctions of conditions and logical OR applies between the rules (Figure 1). Conditions compare single features with constant values, or compare features with each other; they support linear scaling of features with coefficients and nonlinear scaling through power expressions, as well as time lags in feature values. The consequences can have probabilities associated with them [15].

There are several reasons why such rules are a good representation. First, they closely resemble verbal reasoning we use in our daily lives. Thus, they are explicit and interpretable, which is vital in applications where predictions need to be auditable so that experts can understand how and why a recommendation or forecast was made. Second, such rules have a linear list structure, which helps avoid the usual problems of tree evolution (e.g., bloat, mutation/crossover), while still remaining logically complete. Third, they have the ability to uncover nonlinear relationships and interactions among the domain features—even as complex as those between time lags in time series data [13]. Fourth, the conditions can be easily augmented to represent probabilities, as well as a variety of further mathematical operations and functions.

When parsing an individual’s rule-set, conditions are evaluated in order and all actions for conditions that are met are returned. It is up to the domain to decide which subset of these actions to apply. For example, in some domains (e.g. those in Sections 4.1- 4.3), only the first action is executed; in others (e.g. Sections 4.4–4.5), a hard-max filter is used to select one of the actions based on action coefficients; and in yet others, all actions may be executed in parallel or in sequence.

At reproduction, parent individuals are chosen through tournament selection. A variety of crossover functions are possible. A straightforward method is to pick a random crossover index less than the number of rules in one individual, and in the offspring rule-set, replace the remainder rules in that individual with rules past the crossover index from the other parent individual. This method thus produces offspring in which the number of rules can potentially grow or shrink relative to their parents. Another form of crossover does a logical multiplication of one parent individual’s rules into the second parent, producing offspring with longer rules than the parents. In the experiments for this paper the crossover style is selected randomly for each reproduction.

Mutation is defined as a single random change to an element of the rule-set. Mutation may apply at the condition level, changing an element of the condition, or at the rule level, replacing, removing, or adding a condition to the rule, or changing the rule’s action. Mutation can also apply at the rule level, removing an entire rule from the individual, or changing the default rule, or changing the rule order. Note that mutations can thus make the offspring smaller or larger than the original individual.

In order to reduce bloat, all conditions recognized as tautologies or falsehoods are removed from offspring individuals. A useful tool is a counter called times_applied, associated with each rule to keep track of how many times the rule evaluated to true, thus contributing to individual’s behavior and fitness. This counter can be used to filter out inactive individual rules from participating in crossover. The counter can also be useful to get a sense of each rule’s coverage and generality: for instance, a very low count could be an indication of overfitting or even a corner case bias).

Potential extensions include encouraging shorter rules through an explicit secondary objective, incorporated e.g. through NSGA-II [7]. Similarly, exploration and creativity can be encouraged through a secondary novelty objective [29, 30]. In cases where the dataset may be excessively large, partial and incremental evaluation through age-layering can be used to make the process more efficient [13, 14, 31].

1. $(\mathit{Mean}[4] < 72.75\text{mmHg})$ & $(\mathit{Kurtosis}[3] < 4.09)$	$\longrightarrow$ *Low*
2. $(\mathit{Skew}[10] > 2.01)$ & $(\mathit{Mean}[8] < 88.92\text{mmHg})$ & $(\mathit{Skew}[4] < 0.15)$	$\longrightarrow$ Normal
3. $(\mathit{Mean}[0] < 72.75\text{mmHg})$	$\longrightarrow$ *Low*
4. $(\mathit{Mean}[10] < 73.10\text{mmHg})$	$\longrightarrow$ *Low*
5. $(\mathit{Mean}[1] < 121.96\text{mmHg})$ & $(\mathit{Mean}[4] > 88.92\text{mmHg})$ & $(\mathit{Mean}[1] > 73.10\text{mmHg})$	$\longrightarrow$ *High*
6. $(\mathit{Mean}[0] < 97.53\text{mmHg})$	$\longrightarrow$ *Normal*
7. $(\mathit{Mean}[0] < 97.53\text{mmHg})$ & $(\mathit{Kurtosis}[0] > 12.71)$	$\longrightarrow$ *Normal*
8. $(\mathit{Mean}[4] < 72.75\text{mmHg})$ & $(\mathit{Kurtosis}[7] > 4.03)$	$\longrightarrow$ *Low*
9. $(\mathit{Mean}[4] > 121.96\text{mmHg})$ & $(\mathit{Kurtosis}[5] > 12.71)$ & $(\mathit{Kurtosis}[3] > 1.00)$	$\longrightarrow$ *Normal*
10. $(\mathit{Std}[0] < 10.76)$	$\longrightarrow$ *High*
11. $(\mathit{Kurtosis}[0] > 1.00)$	$\longrightarrow$ *High*
12. $(\mathit{Mean}[0] < 72.75\text{mmHg})$ & $(\mathit{Std}[4] > 0.01)$	$\longrightarrow$ *Low*
13. $(\mathit{Kurtosis}[0] < 4.09)$ & $(\mathit{Skew}[3] > 2.01)$	$\longrightarrow$ *Normal*
14. $(\mathit{Skew}[9] > 0.06)$	$\longrightarrow$ *High*
15. $(\mathit{Skew}[0] < 1.95)$	$\longrightarrow$ *High*
16. $(\mathit{Mean}[0] < 72.75\text{mmHg})$ & $(\mathit{Mean}[5] < 52.12\text{mmHg})$	$\longrightarrow$ *Low*
17. Default	$\longrightarrow$ *Normal*

Figure 2: *A sample rule set evolved to predict blood pressure.* All the features are extracted from aggregations of MAP [13]. EVOTER discovered sets of features at specific time points to provide a useful signal for prediction. For instance, Std[4] specifies the standard deviation of the aggregated mean arterial pressure (MAP) over four minutes earlier. The evolved rules predict sepsis accurately and are interpretable and meaningful to experts.

4 EXPERIMENTS

This section presents examples of the application of rule-set evolution to prediction / classification problems as well as prescription / action determination problems. Similar prescriptions are shown to emerge when evaluation is done directly and through a surrogate model. Prescription performance of evolved rule sets is shown similar to the performance of evolved neural networks. The evolved rules are meaningful to domain experts and in some cases provide useful insight to them.

4.1 Rule-sets for Prediction/Classification

In the first experiment, EVOTER was evaluated in predicting whether a patient in ICU will go into a septic shock in the next 30 minutes, based on a time series of blood pressure measurements.

Experiment. Consider the mean arterial pressure (MAP) signal at time t as $x(t) \in \mathbb{R}$. The goal is to predict if the value of the statistic $\bar{m} = \text{avg}([x(t+\alpha), ..., x(t+\alpha+\beta)])$ falls into one of the three intervals:

- If $\bar{m} \leq$ 55mmHg $\longrightarrow$ Low;
- If 55mmHg $< \bar{m} \leq$ 85mmHg $\longrightarrow$ Normal;
- If $\bar{m} >$ 85mmHg $\longrightarrow$ High.

Results. The first of these intervals indicates acute hypotension indicative of septic shock. EVOTER was trained with approximately 4000 patients' arterial blood pressure wave-forms from the MIMIC II v3 dataset [13]. Rules were evolved to observe various features of these wave-forms at various times, and predict whether acute hypotension was likely to develop within the prediction window of 30 minutes (Figure 2). In a massively parallel implementation on 1000 clients running for several days, EVOTER achieved an accuracy of 0.895 risk-weighted error on the withheld set, with true positive rate of 0.96 and false positive rate of 0.394.

Note that the approach is general and does not need transformation of the data. Even though many machine learning methods could be used for this prediction task, EVOTER provides a solution that is transparent and interpretable. Indeed, the rules were evaluated by emergency-room physicians who found them meaningful. Without such an understanding and verification, it would be difficult to trust the system enough to deploy it. Thus, working in tandem with human experts, EVOTER can be useful in interpreting and understanding the progression of the patient's health and sensitivity, and serve as an early indicator of problems.

4.2 Rule-sets for Prescription/Action Determination

The second set of experiments focuses on problems where the goal is not to predict a set of known labels, but instead to prescribe, i.e. generate actions for an agent in a reinforcement learning environment. Results from two standard such domains are included: cart-pole and flappy bird. In both domains, evaluation is done observing the effects of the actions directly in a simulation of the domain.

Cart-Pole Experiment. Cart-pole is one of the standard *Reinforcement Learning* benchmarks. In the popular CartPole-v0 implementation in the OpenAI Gym platform [5], there is a single pole on a cart that moves left and right depending on the force applied to it. The controller inputs are position of cart, velocity of cart, angle of pole, and rotation rate of pole. A reward is given for each time step that the pole stays near vertical and the cart stays near the center of the track; otherwise the episode ends [10].

Cart-Pole Results. EVOTER finds solutions reliably within five generations. These rule sets keep the pole from falling indefinitely when started with situations in the standard validation set. Figure 3 shows a sample such rule-set. It is remarkably simple, based on a relationship between pole angle and cart velocity. In this manner, EVOTER can discover and take advantage of physical relationships, which is in general a powerful ability of evolutionary optimization [28].

1. $(0.11 * \mathit{velocity.of.cart}^3 < 0.87 * \mathit{angle.of.pole})$	$\longrightarrow$ *LEFT*
2. Default Action	$\longrightarrow$ *RIGHT*

Figure 3: *A sample solution rule-set for the Cart-Pole problem discovered by direct evolution.* The first rule expresses a simple mathematical relationship between two input features; such discoveries can be useful in uncovering insights into the nature of the physical system.

Flappy-Bird Experiment. Flappy Bird is a side-scroller game where the player attempts to fly an agent between columns of pipes without hitting them by performing flapping actions at carefully chosen times. The experiment was based on a PyGame [34] implementation running at a speed of 30 frames per second. The goal of the game is to finish ten episodes of two minutes, or 3,600 frames each, through random courses of pipes. A reward is given for each frame where the bird does not collide with the boundaries or the pipes; otherwise the episode ends. The score of each candidate is the average reward over the ten episodes[10].

Flappy-Bird Results. EVOTER finds perfect solutions to this problem typically within 400 generations. Figure 4 shows a sample solution rule-set. As is typical in this domain, the rules identify cases where the agent should flap. The conditions appear complex, however, several of the clauses are redundant and can be removed to form a final solution. Such redundancy is common in evolved solutions likely because it makes the solutions robust to mutations and crossover. Innovation is unlikely to break them completely, and they can be refined to be useful. While this principle applies to many evolutionary settings (e.g. neuroevolution [11]), rule-sets make it explicit.

$$\begin{array}{ll}
1. (0.99 * \mathit{next.pipe.dist.to.player} < 0.93 * \mathit{next.next.pipe.bottom.y}) \;\; \& & \\
\quad (0.99 * \mathit{next.pipe.dist.to.player} < 0.83 * \mathit{next.next.pipe.bottom.y}) \;\; \& & \\
\quad (0.98 * \mathit{player.y} \leq 0.78 * \mathit{next.pipe.bottom.y}) \;\; \& & \\
\quad (0.95 * \mathit{player.y} \leq 0.65 * \mathit{next.pipe.bottom.y}) \;\; \& & \\
\quad (0.76 * \mathit{player.vel} > -0.98[-8.0..10.0]) \;\; \& & \\
\quad (0.47 * \mathit{next.next.pipe.bottom.y} > 0.82 * \mathit{player.vel}) \;\; \& & \\
\quad (0.41 * \mathit{player.y} \leq 0.78 * \mathit{next.pipe.bottom.y}) \;\; \& & \\
\quad (0.26 * \mathit{next.pipe.top.y} < 0.76 * \mathit{player.y}) \;\; \& & \\
\quad (0.17 * \mathit{next.pipe.top.y} \leq 84.48[0..192.0]) & \longrightarrow \mathit{FLAP} \\
2. (0.95 * \mathit{player.y} \leq 0.65 * \mathit{next.pipe.bottom.y}) \;\; \& & \\
\quad (0.76 * \mathit{player.vel} > -0.98[-8.0..10.0]) \;\; \& & \\
\quad (0.47 * \mathit{next.next.pipe.bottom.y} > 0.82 * \mathit{player.vel}) \;\; \& & \\
\quad (0.41 * \mathit{player.y} \leq 0.78 * \mathit{next.pipe.bottom.y}) \;\; \& & \\
\quad (0.19 * \mathit{next.pipe.dist.to.player} < 0.64 * \mathit{next.pipe.bottom.y}) \;\; \& & \\
\quad (0.17 * \mathit{next.pipe.top.y} \leq 84.48[0..192.0]) & \longrightarrow \mathit{FLAP} \\
3. (0.92 * \mathit{next.pipe.dist.to.player} < 0.95 * \mathit{next.pipe.top.y}) \;\; \& & \\
\quad (0.78 * \mathit{next.pipe.bottom.y} \geq 175.2[0..292.0]) \;\; \& & \\
\quad (0.71 * \mathit{next.next.pipe.bottom.y} > 0.71 * \mathit{next.pipe.dist.to.player}) \;\; \& & \\
\quad (0.49 * \mathit{next.next.pipe.top.y} \geq 0.12 * \mathit{next.pipe.dist.to.player}) \;\; \& & \\
\quad (0.53 * \mathit{next.pipe.top.y} < 0.63 * \mathit{next.pipe.dist.to.player}) & \longrightarrow \mathit{NO\ FLAP} \\
4. \text{ Default Action} & \longrightarrow \mathit{NO\ FLAP}
\end{array}$$

Figure 4: *A sample solution rule-set discovered by direct evolution for the Flappy Bird domain. The rules primarily identify situations where a flap is warranted. Rule 3 is redundant, and the conditions in all rules have redundancy, which is common in evolved solutions, making the search robust.*

4.3 Evolving rule sets with a surrogate (ESP)

Many real-world domains are too costly for direct evolution, and evaluation of candidates needs to be done more economically against a surrogate model [10]. The next experiment demonstrates that solutions evolved with a surrogate can be remarkably similar to those evolved directly in the domain.

Experiment. Evolutionary Surrogate-Assisted Prescription, or ESP [10] is a general approach to decision making where the decision policy is discovered through evolution and the surrogate is constructed through machine learning. The surrogate, or the predictor, can be an opaque model such as a random forest or neural network trained with gradient descent. Similarly the decision policy, or the prescriptor, is often a neural network; however by evolving a rule-set strategy instead results in an explainable solution.

In this experiment, ESP was set up to solve the same cart-pole domain as the previous section. The prescriptor population was initially random, and therefore generated a diverse set of actions. The predictor was trained with the outcome of these actions. The trained predictor was then used to evaluate prescriptor candidates through evolution.

Results. The predictor trained reliably with 100 action examples from the random initial population. Evolution then took only three generations to find agents that solved the examples in the validation set. Interestingly, the resulting rules (Figure 5) are logically almost identical to those resulting from direct evolution (Figure 3). While this validity of the surrogate modeling approach has been demonstrated before in terms of performance [10], rule-set evolution makes this conclusion explicit.

$$\begin{array}{ll}
1. (0.16 * \mathit{velocity.of.cart}^3 > 0.89 * \mathit{angle.of.pole}) & \longrightarrow \mathit{RIGHT} \\
2. \text{ Default Action} & \longrightarrow \mathit{LEFT}
\end{array}$$

Figure 5: *A sample rule-set discovered through ESP in the Cart-Pole domain.* These rules are logically almost identical to those discovered through direct evolution (i.e., Figure 3). This result verifies that surrogate-based optimization is a viable approach, i.e. it discovers the same solutions as direct evolution.

4.4 Performance of evolved rule sets versus neural networks

The standard approach in ESP is to evolve neural networks as the surrogate. Neural networks are powerful and flexible, but they are opaque. In contrast, rule sets are transparent; an important question is whether they perform approximately as well as neural networks in the same task. This question is evaluated in this section in the task of recommending treatment for heart failures through the ESP approach.

Experiment. A random-forest predictor was trained on the Heart Failure data-set [6] to predict the probability of death for a patient given two possible interventions, i.e., ejection fraction and serum creatinine. The predictor achieved an out-of-sample Mathews correlation coefficient of 0.33.

The prescriptor receives the patient condition as its input and prescribes one of the two interventions. The predictor is queried to evaluate its fitness in minimizing deaths. The neural network prescriptor's input nodes correspond to the patient condition and outputs to the two possible treatments, and it has a hidden layer of 16 nodes. Its weights were evolved over 100 generations using a population size of 100. Similarly, the rule-set prescriptor includes left-hand side clauses referring to the patient's condition, and right-hand side actions specifying interventions. Evolution was similarly set to run for 100 generations with a population of 100.

Results. Effectiveness of the two prescriptor models in preventing death is shown over evolution time in Figure 6, averaged over ten runs. It took more generations for rule set evolution to reach the same level as the neural network evolution, and there is more diversity and variance in the rule set populations. The likely reason is that even small changes in rules (e.g., adding or removing a clause) can have large effect on performance. However, on average the final results are very similar: The best rule set on average performs at 10.8% and the best neural network at 10.7%. Thus, explainability is achieved with practically no cost in performance. With such variability, an interesting question is: How can a one candidate in the end be chosen such that it most likely generalizes to unseen data, given the multiple hypothesis problem [2, 3]? It turns out that under a smoothness assumption that similar candidates perform similarly, picking a candidate based on the average performance of its neighbors provides a viable strategy [22].

A sample solution rule set is shown in Figure 7. It uses serum creatinine as a default, and the rules primarily identify situations where ejection fraction is better. As usual, many of the rules are redundant (indicating a robust search process) and could be cleaned up if needed.

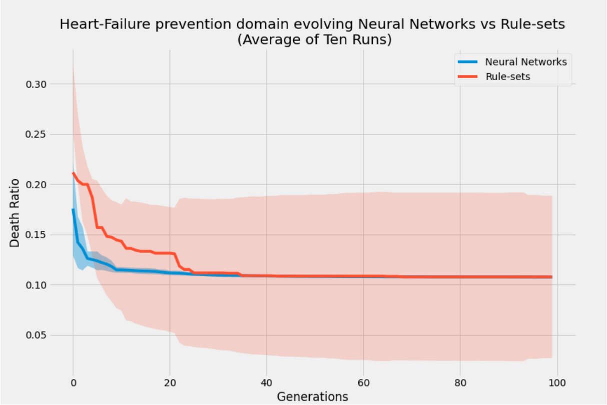

Figure 6: *Evolution of neural network and rule-set prescriptors for ESP in the heart-failure prevention domain.* The rule-set population varies more throughout evolution, but the final best solutions perform almost the same on average with a death rate of slightly less than 0.11. Thus, explainability is achieved with more diversity and practically no cost in performance.

4.5 Obtaining insights and avoiding biases

In the last experiment, rule-based ESP was applied to the task of recommending treatment for diabetes that would minimize hospital readmissions. The results were evaluated together with physicians, giving them a plain-text interpretation, and indicating insights that are possible to obtain with this approach.

Experiment. The diabetes dataset [33] represents 10 years (1999-2008) of clinical care at 130 US hospitals and integrated delivery networks. Information was extracted from the database for encounters that satisfied the following criteria:

- It is an inpatient encounter (a hospital admission).

$$\begin{array}{rll}
1. & (0.08 * \mathit{anaemia} < 0.44 * \mathit{platelets}) & \longrightarrow 0.21 * \mathit{ejection.fraction} \\
2. & (0.08 * \mathit{platelets} \leq 0.45[0.0..1.0]) & \longrightarrow 0.59 * \mathit{ejection.fraction} \\
3. & (0.84 * \mathit{anaemia}^3 \leq 0.40 * \mathit{age}) & \longrightarrow 0.04 * \mathit{serum.creatinine} \\
4. & (0.89 * \mathit{serum.sodium} > 0.97 * \mathit{smoking}) \ \ \& & \\
 & (0.47 * \mathit{serum.sodium} > 0.09 * \mathit{creatinine.phosphokinase}) & \longrightarrow 0.32 * \mathit{ejection.fraction} \\
5. & (1.00 * \mathit{age} > 0.97 * \mathit{platelets}) & \longrightarrow 0.49 * \mathit{ejection.fraction} \\
6. & (0.39 * \mathit{creatinine.phosphokinase} < 0.92 * \mathit{anaemia}) & \longrightarrow 0.03 * \mathit{serum.creatinine} \\
7. & (0.94 * \mathit{platelets} \geq 0.02 * \mathit{anaemia}) & \longrightarrow 0.65 * \mathit{ejection.fraction} \\
8. & (0.46 * \mathit{smoking} \leq 0.50 * \mathit{diabetes}) & \longrightarrow 0.51 * \mathit{ejection.fraction} \\
9. & (0.49 * \mathit{anaemia} \leq 0.92[0.0..1.0]) \ \ \& & \\
 & (0.46 * \mathit{smoking} \leq 0.50 * \mathit{diabetes}) \ \ \& & \\
 & (0.09 * \mathit{platelets} \leq 0.78[0.0..1.0]) & \longrightarrow 0.36 * \mathit{ejection.fraction} \\
10. & (0.08 * \mathit{high.blood.pressure}^3 \leq 0.85 * \mathit{diabetes}) & \longrightarrow 0.21 * \mathit{ejection.fraction} \\
11. & (0.91 * \mathit{platelets} < 0.62 * \mathit{high.blood.pressure}) \ \ \& & \\
 & (0.61 * \mathit{age} < 0.88 * \mathit{high.blood.pressure}) \ \ \& & \\
 & (0.10 * \mathit{creatinine.phosphokinase} < 0.43 * \mathit{sex}) & \longrightarrow 0.46 * \mathit{ejection.fraction} \\
12. & (0.47 * \mathit{serum.sodium} > 0.09 * \mathit{creatinine.phosphokinase}) & \longrightarrow 0.32 * \mathit{ejection.fraction} \\
13. & (0.46 * \mathit{smoking} \leq 0.50 * \mathit{diabetes}) \ \ \& & \\
 & (0.06 * \mathit{anaemia} > 0.82 * \mathit{creatinine.phosphokinase}) \ \ \& & \\
 & (0.03 * \mathit{high.blood.pressure} \geq 0.21 * \mathit{sex}) & \longrightarrow 0.51 * \mathit{ejection.fraction} \\
14. & (0.31 * \mathit{creatinine.phosphokinase} >= 0.02 * \mathit{high.blood.pressure}) & \longrightarrow 0.48 * \mathit{ejection.fraction} \\
15. & (0.47 * \mathit{serum.sodium} > 0.09 * \mathit{creatinine.phosphokinase}) \ \ \& & \\
 & (0.40 * \mathit{diabetes} \geq 0.33 * \mathit{serum.sodium}) & \longrightarrow 0.32 * \mathit{ejection.fraction} \\
16. & (0.17 * \mathit{smoking} < 0.30 * \mathit{high.blood.pressure}) \ \ \& & \\
 & (0.08 * \mathit{high.blood.pressure} \leq 0.85 * \mathit{diabetes}) & \longrightarrow 0.21 * \mathit{ejection.fraction} \\
17. & (0.94 * \mathit{platelets} \geq 0.02 * \mathit{anaemia}) \ \ \& & \\
 & (0.46 * \mathit{smoking} \leq 0.50 * \mathit{diabetes}) & \longrightarrow 0.65 * \mathit{ejection.fraction} \\
18. & (1.00 * \mathit{age} > 0.97 * \mathit{platelets}) \ \ \& & \\
 & (0.97 * \mathit{platelets} \leq 0.36 * \mathit{diabetes}) \ \ \& & \\
 & (0.46 * \mathit{smoking} \leq 0.50 * \mathit{diabetes}) & \longrightarrow 0.49 * \mathit{ejection.fraction} \\
19. & (0.46 * \mathit{smoking} \leq 0.50 * \mathit{diabetes}) \ \ \& & \\
 & (0.08 * \mathit{age} > 0.85 * \mathit{diabetes}) \ \ \& & \\
 & (0.03 * \mathit{high.blood.pressure} \geq 0.21 * \mathit{sex}) & \longrightarrow 0.51 * \mathit{ejection.fraction} \\
20. & (0.91 * \mathit{platelets} < 0.62 * \mathit{high.blood.pressure}) \ \ \& & \\
 & (0.10 * \mathit{creatinine.phosphokinase} < 0.43 * \mathit{sex}) & \longrightarrow 0.46 * \mathit{ejection.fraction} \\
21. & (0.46 * \mathit{smoking} \leq 0.50 * \mathit{diabetes}) \ \ \& & \\
 & (0.06 * \mathit{anaemia} > 0.82 * \mathit{creatinine.phosphokinase}) & \longrightarrow 0.51 * \mathit{ejection.fraction} \\
22. & (0.88 * \mathit{anaemia} > 0.76[0.0..1.0]) \ \ \& & \\
 & (0.84 * \mathit{creatinine.phosphokinase} < 0.40 * \mathit{age}) \ \ \& & \\
 & (0.45 * \mathit{sex} < 0.85[0.0..1.0]) & \longrightarrow 0.04 * \mathit{serum.creatinine} \\
23. & (0.46 * \mathit{smoking} \leq 0.50 * \mathit{diabetes}) \ \ \& & \\
 & (0.08 * \mathit{anaemia} < 0.44 * \mathit{platelets}) & \longrightarrow 0.21 * \mathit{ejection.fraction} \\
24. & (0.63 * \mathit{high.blood.pressure} \leq 0.79 * \mathit{serum.sodium}) \ \ \& & \\
 & (0.46 * \mathit{smoking} \leq 0.50 * \mathit{diabetes}) \ \ \& & \\
 & (0.31 * \mathit{creatinine.phosphokinase} \geq 0.02 * \mathit{high.blood.pressure}) & \longrightarrow 0.48 * \mathit{ejection.fraction} \\
25. & (0.08 * \mathit{age} > 0.85 * \mathit{diabetes}) & \longrightarrow 0.21 * \mathit{ejection.fraction} \\
26. & (0.09 * \mathit{platelets} \leq 0.78[0.0..1.0]) \ \ \& & \\
 & (0.08 * \mathit{age} > 0.85 * \mathit{diabetes}) & \longrightarrow 0.36 * \mathit{ejection.fraction} \\
27. & (0.91 * \mathit{platelets} < 0.62 * \mathit{high.blood.pressure}) \ \ \& & \\
 & (0.24 * \mathit{serum.sodium} > 0.29 * \mathit{diabetes}) \ \ \& & \\
 & (0.10 * \mathit{creatinine.phosphokinase} < 0.43 * \mathit{sex}) \ \ \& & \\
 & (0.08 * \mathit{age} > 0.85 * \mathit{diabetes}) & \longrightarrow 0.46 * \mathit{ejection.fraction} \\
28. & (0.91 * \mathit{platelets} < 0.62 * \mathit{high.blood.pressure}) & \longrightarrow 0.46 * \mathit{ejection.fraction} \\
29. & (0.40 * \mathit{diabetes} \geq 0.33 * \mathit{serum.sodium}) & \longrightarrow 0.32 * \mathit{ejection.fraction} \\
30. & (0.31 * \mathit{creatinine.phosphokinase} \geq 0.02 * \mathit{high.blood.pressure}) & \longrightarrow 0.48 * \mathit{ejection.fraction} \\
31. & (0.84 * \mathit{creatinine.phosphokinase} \geq 0.40 * \mathit{age}) & \longrightarrow 0.04 * \mathit{serum.creatinine} \\
32. & (0.94 * \mathit{platelets} \geq 0.02 * \mathit{anaemia}) \ \ \& & \\
 & (0.08 * \mathit{platelets} \leq 0.45[0.0..1.0]) & \longrightarrow 0.59 * \mathit{ejection.fraction} \\
33. & (0.08 * \mathit{anaemia} < 0.44 * \mathit{platelets}) \ \ \& & \\
 & (0.08 * \mathit{age} > 0.85 * \mathit{diabetes}) & \longrightarrow 0.21 * \mathit{ejection.fraction} \\
34. & (0.47 * \mathit{serum.sodium} > 0.09 * \mathit{creatinine.phosphokinase}) \ \ \& & \\
 & (0.46 * \mathit{smoking} \leq 0.50 * \mathit{diabetes}) \ \ \& & \\
 & (0.40 * \mathit{diabetes} \geq 0.33 * \mathit{serum.sodium}) & \longrightarrow 0.32 * \mathit{ejection.fraction} \\
35. & (0.39 * \mathit{creatinine.phosphokinase} < 0.92 * \mathit{anaemia}) \ \ \& & \\
 & (0.10 * \mathit{anaemia} \geq 0.06 * \mathit{serum.sodium}) & \longrightarrow 0.03 * \mathit{serum.creatinine} \\
36. & (0.91 * \mathit{platelets} < 0.62 * \mathit{high.blood.pressure}) \ \ \& & \\
 & (0.46 * \mathit{smoking} \leq 0.50 * \mathit{diabetes}) \ \ \& & \\
 & (0.10 * \mathit{creatinine.phosphokinase} < 0.43 * \mathit{sex}) & \longrightarrow 0.46 * \mathit{ejection.fraction} \\
37. & (0.46 * \mathit{smoking} \leq 0.50 * \mathit{diabetes}) \ \ \& & \\
 & (0.08 * \mathit{age} \leq 0.85 * \mathit{diabetes}) & \longrightarrow 0.21 * \mathit{ejection.fraction} \\
38. & (0.46 * \mathit{smoking} \leq 0.50 * \mathit{diabetes}) \ \ \& & \\
 & (0.08 * \mathit{platelets} \leq 0.45[0.0..1.0]) & \longrightarrow 0.59 * \mathit{ejection.fraction} \\
39. & (0.46 * \mathit{smoking} \leq 0.50 * \mathit{diabetes}) \ \ \& & \\
 & (0.03 * \mathit{high.blood.pressure} \geq 0.21 * \mathit{sex}) & \longrightarrow 0.51 * \mathit{ejection.fraction} \\
40. & (0.41 * \mathit{high.blood.pressure} > 0.48 * \mathit{sex}) \ \ \& & \\
 & (0.09 * \mathit{platelets} \leq 0.78[0.0..1.0]) \ \ \& & \\
 & (0.08 * \mathit{age} > 0.85 * \mathit{diabetes}) & \longrightarrow 0.36 * \mathit{ejection.fraction} \\
41. & (0.46 * \mathit{smoking} \leq 0.50 * \mathit{diabetes}) \ \ \& & \\
 & (0.09 * \mathit{platelets} \leq 0.78[0.0..1.0]) & \longrightarrow 0.36 * \mathit{ejection.fraction} \\
42. & (0.21 * \mathit{high.blood.pressure} \leq 0.85 * \mathit{diabetes}) \ \ \& & \\
 & (0.17 * \mathit{smoking} < 0.30 * \mathit{high.blood.pressure}) & \longrightarrow 0.21 * \mathit{ejection.fraction} \\
43. & (1.00 * \mathit{age} > 0.97 * \mathit{platelets}) \ \ \& & \\
 & (0.97 * \mathit{platelets} \leq 0.36 * \mathit{diabetes}) & \longrightarrow 0.49 * \mathit{ejection.fraction} \\
44. & (0.97 * \mathit{platelets} \leq 0.36 * \mathit{diabetes}) & \longrightarrow 0.49 * \mathit{ejection.fraction} \\
45. & (0.23 * \mathit{diabetes} > 0.89 * \mathit{platelets}) \ \ \& & \\
 & (0.08 * \mathit{high.blood.pressure}^3 \leq 0.85 * \mathit{diabetes}) & \longrightarrow 0.21 * \mathit{ejection.fraction} \\
46. & (0.14 * \mathit{creatinine.phosphokinase} > 0.75 * \mathit{smoking}) \ \ \& & \\
 & (0.08 * \mathit{age} \leq 0.85 * \mathit{diabetes}) & \longrightarrow 0.21 * \mathit{ejection.fraction} \\
47. & \text{Default} & \longrightarrow 0.13 * \mathit{serum.creatinine}
\end{array}$$

Figure 7: *A sample solution rule set for the heart-failure domain.* The default rule indicates serum, and the others mostly identify situations where ejection fraction is preferable. Many of the rules are redundant as is common in the search process, and could be cleaned up if needed.

- It is a diabetic encounter, that is, one during which any kind of diabetes was entered to the system as a diagnosis.
- The length of stay was at least 1 day and at most 14 days.
- Laboratory tests were performed during the encounter.
- Medications were administered during the encounter.

The dataset includes over 50 features characterizing the patient and the hospital context, including patient number, race, gender, age, admission type, time in hospital, medical specialty of admitting physician, number of lab test performed, HbA1c test result, and diagnosis, as well as the number of outpatient, inpatient, and emergency visits by the patient in the year before the hospitalization. As actions, 21 different treatments can be prescribed, i.e. different diabetic and other medications. Two objectives are optimized: readmission rate and discharge disposition. The three possible readmission categories are: "no readmission" (+1 point), "readmission in less than 30 days" (0), and "readmission in more than 30 days" (-1). The discharge-disposition categories were: "sent back home" (+2 points), "remained in the hospital" (-1), "left with advised medical attention" (-1), "sent to another hospital" (-1), and "died" (-4). The goal is to maximize both the readmission score and the discharge-disposition score simultaneously.

A neural-network predictor was trained with ∼66k samples in the historical data, with ∼33k reserved for validation and ∼33k for testing. A population of 100 rule sets was then evolved for 40 generations to generate a Pareto front: Given the patient and hospital contexts, they represent different tradeoffs between reducing readmissions and improving discharge dispositions. Note that these objectives are somewhat at odds: e.g. sending the patient home too early may result in an early readmission.

Results. The predictor training achieved a test accuracy of ∼80% (of correct classifications into the three categories). Rule-set evolution started with a no-readmission rate of ∼29%, which improved to ∼49% over the 40 generations; a similar improvement was observed along the discharge-disposition objective. Remarkable, tradeoffs were discovered that performed much better than the actual prescriptions in the dataset. Whereas 78% of patients were sent home and the no-readmission rate was 60% in the dataset, the most dominant solution recommended treatments that were predicted to result in sending home as well as no readmission 99% of the time.

Figure 8 shows this highly effective rule-set discovered by EVOTER. The rules are relatively transparent, but again contain redundancies that make them somewhat difficult to read. To clarify and to evaluate them, these rules were given to a domain expert who removed the redundancies and translated them into plain text (Figure 9). For instance, given that their features take on binary values, the last three clauses of Rule 6 can be simplified to "not Asian", "not court/law", and emergency. The expert verified that indeed this set constitutes a meaningful treatment policy.

This example also illustrates the potential of rule-set evolution to make biases in machine learning systems explicit. For instance, Rule 6 is conditioned upon the race of the patients being Asian. In this case, race is indeed a medically valid consideration, but it is easy to see that in other domains with incomplete data, such constraints could be simply due to biases in the data. With rule sets, it is possible to identify such biases. Furthermore, it is possible to edit them out when they are not warranted, which would be very difficult to do with other machine-learning approaches.

$$\begin{array}{ll}
1.(\mathit{diag1.injury} \leq 2.2 * \mathit{diag1.respiratory}) \ \ \& & \\
\quad (\mathit{diag2.neoplasms} > 33) & \longrightarrow 18 \\
2.(\mathit{age}[60-70) \ \geq 1.2 * \mathit{age}[10-20)) & \longrightarrow 7 \\
3.(\mathit{diag2.diabetes} \geq 0.82 * \mathit{diag1.digestive}) & \longrightarrow 17 \\
4.(\mathit{admission.Court/Law} \geq 1.045 * \mathit{diag3.circulatory}) & \longrightarrow 9 \\
5.(\mathit{age}[80-90) \ \geq 2.29 * \mathit{admission.type.id.Newborn}) & \longrightarrow 14 \\
6.(\mathit{age}[30-40) \ \leq 0.47 * \mathit{age}[50-60)) \ \ \& & \\
\quad (\mathit{admission.type.id.Newborn} \leq 0.47 * \mathit{age}[50-60)) \ \ \& & \\
\quad (\mathit{diag2.respiratory} \geq 1.86 * \mathit{diag3.musculoskeletal}) \ \ \& & \\
\quad (\mathit{diag1.diabetes} \geq 1.42 * \mathit{age}[30-40)) \ \ \& & \\
\quad (0.18 * \mathit{gender} \geq 0.38 * \mathit{race.Asian}) \ \ \& & \\
\quad (0.12 * \mathit{race.AfricanAmerican} \geq 0.15 * \mathit{admission.Court/Law}) \ \ \& & \\
\quad (0.03 * \mathit{race.Hispanic} \leq 0.76 * \mathit{admission.Emergency.Room}) & \longrightarrow 15 \\
7.\ \text{Default} & \longrightarrow 8
\end{array}$$

Figure 8: ***A sample rule-set discovered through ESP to reduce hospital readmissions for diabetes.*** **The rules are transparent but contain redundancies that make them sometimes hard to read. A domain expert can verify that the set is meaningful, and translate them into plain English where the insights and biases are easier to see (Figure 9).**

1. If the patient has a respiratory problem but not a neoplasms problem	*prescribe Metformin-Pioglitazone*
2. If the patient is any age except 60 to 70	*prescribe Glyburide*
3. If the patient has Diabetic Diagnostic	*prescribe Glipizide- Metformin*
4. If the patient has circulatory diagnosis	*prescribe Pioglitazone*
5. If the admission type is Newborn	*prescribe Tolazamide*
6. If the patient's age is between 30 to 60 & the patient has musculoskeletal diagnosis & is not Asian & was not sent by court/law & was admitted in emergency	*prescribe Insulin*
7. If none of the above	*prescribe Glipizide*

Figure 9: ***Expert interpretation of the diabetes treatment rule set. In particular, Rule 6 is now easier to read. It also makes it possible to identify a potential racial bias; in this case the bias is medically indicated, but in other applications it may be due to inaccurate training data. With rule sets, such biases can be identified and edited out, which would be difficult to do with other machine-learning approaches.***

5 DISCUSSION AND FUTURE WORK

The experiments in this paper show that rule-set evolution is a viable approach to a range of different machine-learning problems ranging from prediction and classification to prescription and action policies. This approach can be used to discover explainable models directly, or through a surrogate model of the domain. Compared to black-box models such as neural networks, there may be a slight performance cost for this transparent explainability, but the performance of rule sets is in the general ballpark with more opaque models. The resulting rule sets make relationships between concepts explicit, and can uncover insights into the domain, as well as biases.

An important direction for future work is to include a facility for the expert to edit the discovered rule sets. First, it is useful to identify redundancies and remove them, thereby making the rules easier to read. Much of such simplification can be done automatically. Second, the approach can provide a foundation for building models that are fair. Unlike in opaque models, the learned biases are explicit, and they can be evaluated and edited out if desired. Similarly, it may be possible to edit the rules to add further knowledge that may be well understood but difficult to learn, such as safety limits or other real-world constraints.

An interesting extension would be to gradually expand the expression vocabulary during evolution. For example, the process could be started with a restricted set of operators which could then be expanded as evolution progresses, thus implementing a form of curricular learning.

Another interesting direction is to build rule-set twins for existing SoTA black-box models. In this manner, it may be possible to convert an opaque but well-performing model into a transparent rule-set equivalent. This model can then be used to explain the learned behavior and identify biases and potential weaknesses in it. Such rule-set models are executable, and if they replicate the performance of the black-box model accurately enough, they could be deployed instead of the black box in explanation-critical applications. In this manner, rule-set distillation can serve as a crucial step in deploying machine learning results in the real world.

6 CONCLUSION

Rule-set evolution, as implemented in EVOTER, is a promising direction for developing inherently transparent explainable models, in contrast with the dominant black-box modeling approaches in machine learning. They can be used in a range of prediction, classification, prescription, and policy search domains, with and without surrogates, and with only a small cost on performance. Given how important it is to understand how and why machine-learning models perform the way they do, rule-set evolution may provide a crucial step in deploying AI in the real world in the future.

REFERENCES

[1] Amina Adadi and Mohammed Berrada. 2018. Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI). *IEEE Access* 6 (2018), 52138–52160.

[2] RS Austin, Isaac Dialsingh, and NS Altman. 2014. Multiple Hypothesis Testing: A Review. *Journal of the Indian Society of Agricultural Statistics* 68 (01 2014), 303–314.

[3] Ligiia Igorevna Avdeeva. 1966. *Simultaneous statistical inference.* Springer.

[4] Vaishak Belle and Ioannis Papantonis. 2021. Principles and Practice of Explainable Machine Learning. *Frontiers in Big Data* 4 (2021).

[5] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. 2016. OpenAI Gym. *arXiv:1606.01540* (2016).

[6] Davide Chicco and Giuseppe Jurman. 2020. Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone. *BMC medical informatics and decision making* 20, 1 (2020), 1–16.

[7] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6, 2 (2002), 182–197. https://doi.org/10.1109/4235.996017

[8] Finale Doshi-Velez and Been Kim. 2017. Towards A Rigorous Science of Interpretable Machine Learning. *arXiv:1702:08608* (2017).

[9] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. (2017). http://archive.ics.uci.edu/ml

[10] Olivier Francon, Santiago Gonzalez, Babak Hodjat, Elliot Meyerson, Risto Miikkulainen, Xin Qiu, and Hormoz Shahrzad. 2020. Effective reinforcement learning through evolutionary surrogate-assisted prescription. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference.* 814–822.

[11] Faustino Gomez and Risto Miikkulainen. 1997. Incremental evolution of complex general behavior. *Adaptive Behavior* 5, 3-4 (1997), 317–342.

[12] David Gunning and David Aha. 2019. DARPA's Explainable Artificial Intelligence (XAI) Program. *AI Magazine* 40, 2 (Jun. 2019), 44–58.

[13] Erik Hemberg, Kalyan Veeramachaneni, Babak Hodjat, Prashan Wanigasekara, Hormoz Shahrzad, and Una-May O'Reilly. 2014. Learning Decision Lists with Lags for Physiological Time Series. In *Third Workshop on Data Mining for Medicine and Healthcare, at the 14th SIAM International Conference on Data Mining.* 82–87.

[14] Babak Hodjat and Hormoz Shahrzad. 2013. *Introducing an Age-Varying Fitness Estimation Function.* 59–71. https://doi.org/10.1007/978-1-4614-6846-2_5

[15] Babak Hodjat, Hormoz Shahrzad, Risto Miikkulainen, Lawrence Murray, and Chris Holmes. 2018. PRETSL: Distributed Probabilistic Rule Evolution for Time-Series Classification. In *Genetic Programming Theory and Practice XIV.* Springer, 139–148.

[16] Julie Jacques, Julien Taillard, David Delerue, Clarisse Dhaenens, and Laetitia Jourdan. 2015. Conception of a dominance-based multi-objective local search in the context of classification rule mining in large and imbalanced data sets. *Applied Soft Computing* 34 (2015), 705–720.

[17] Isaac Lage, Emily Chen, Jeffrey He, Menaka Narayanan, Been Kim, Sam Gershman, and Finale Doshi-Velez. 2019. An evaluation of the human-interpretability of explanation. *arXiv:1902.00006* (2019).

[18] Himabindu Lakkaraju, Ece Kamar, Rich Caruana, and Jure Leskovec. 2019. Faithful and Customizable Explanations of Black Box Models. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society (AIES '19).* Association for Computing Machinery, New York, NY, USA, 131–138.

[19] Pantelis Linardatos, Vasilis Papastefanopoulos, and Sotiris Kotsiantis. 2020. Explainable ai: A review of machine learning interpretability methods. *Entropy* 23, 1 (2020), 18.

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

[21] Risto Miikkulainen, Olivier Francon, Elliot Meyerson, Xin Qiu, Darren Sargent, Elisa Canzani, and Babak Hodjat. 2021. From Prediction to Prescription: Evolutionary Optimization of Non-Pharmaceutical Interventions in the COVID-19 Pandemic. *IEEE Transactions on Evolutionary Computation* 25 (2021), 386–401.

[22] Risto Miikkulainen, Hormoz Shahrzad, Nigel Duffy, and Phil Long. 2017. How to Select a Winner in Evolutionary Optimization?. In *Proceedings of the IEEE Symposium Series in Computational Intelligence.* IEEE. http://www.cs.utexas.edu/users/ai-lab?miikkulainen:ssci17

[23] Tim Miller. 2019. Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence* 267 (2019), 1–38.

[24] Jaume Bacardit Peñarroya. 2004. Pittsburgh genetic-based machine learning in the data mining era: representations, generalization, and run-time.

[25] Arnak Poghosyan, Ashot Harutyunyan, Naira Grigoryan, and Nicholas Kushmerick. 2021. Incident Management for Explainable and Automated Root Cause Analysis in Cloud Data Centers. *Journal of Universal Computer Science* 27, 11 (2021), 1152–1173.

[26] Xin Qiu, Elliot Meyerson, and Risto Miikkulainen. 2020. Quantifying Point-Prediction Uncertainty in Neural Networks via Residual Estimation with an I/O Kernel. In *Proceedings of the International Conference on Learning Representations.*

[27] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16).* Association for Computing Machinery, New York, NY, USA, 1135–1144.

[28] Michael Schmidt and Hod Lipson. 2009. Distilling Free-Form Natural Laws from Experimental Data. *Science* 324, 5923 (2009), 81–85.

[29] Hormoz Shahrzad, Daniel Fink, and Risto Miikkulainen. 2018. Enhanced Optimization with Composite Objectives and Novelty Selection. In *Proceedings of the 2018 Conference on Artificial Life.* Tokyo, Japan. http://www.cs.utexas.edu/users/ai-lab?shahrzad:alife18

[30] Hormoz Shahrzad, Babak Hodjat, Camille Dolle, Andrej Denissov, Simon Lau, Donn Goodhew, Justin Dyer, and Risto Miikkulainen. 2020. Enhanced Optimization with Composite Objectives and Novelty Pulsation. In *Genetic Programming Theory and Practice XVII*, Wolfgang Banzhaf, Erik Goodman, Leigh Sheneman, Leonardo Trujillo, and Bill Worzel (Eds.). Springer, New York, 275–293.

[31] Hormoz Shahrzad, Babak Hodjat, and Risto Miikkulainen. 2016. Estimating the Advantage of Age-Layering in Evolutionary Algorithms. 693–699. https://doi.org/10.1145/2908812.2908911

[32] Sujatha Srinivasan and Sivakumar Ramakrishnan. 2011. Evolutionary multi objective optimization for rule mining: A review. *Artificial Intelligence Review* 36 (10 2011), 205–248. https://doi.org/10.1007/s10462-011-9212-3

[33] Beata Strack, Jonathan Deshazo, Chris Gennings, Juan Luis Olmo Ortiz, Sebastian Ventura, Krzysztof Cios, and John Clore. 2014. Impact of HbA1c Measurement on Hospital Readmission Rates: Analysis of 70,000 Clinical Database Patient Records. *BioMed Research International* 2014 (2014), 781670.

[34] Norman Tasfi. 2016. PyGame Learning Environment. https://github.com/ntasfi/PyGame-Learning-Environment. (2016).

[35] Erico Tjoa and Cuntai Guan. 2021. A Survey on Explainable Artificial Intelligence (XAI): Toward Medical XAI. *IEEE Transactions on Neural Networks and Learning Systems* 32 (2021), 4793–4813.

[36] Joseph Townsend, Thomas Chaton, and Joao M Monteiro. 2019. Extracting relational explanations from deep neural networks: A survey from a neural-symbolic perspective. *IEEE transactions on neural networks and learning systems* 31, 9 (2019), 3456–3470.