CEGARE: A Coevolutionary Approach for Interpretable Rule Ensembles

Simon De Lange*†, Matthias Bogaert*†, Koen W. De Bock‡ and Dirk Van den Poel*†
*Department of Marketing, Innovation and Organisation
Ghent University, Belgium
†FlandersMake@UGent—Corelab CVAMO, Belgium
‡Audencia Business School, France
Email: simon.delange@ugent.be, matthias.bogaert@ugent.be, kdebock@audencia.com, dirk.vandenpoel@ugent.be

***Abstract*—Interpretability is an essential aspect of machine learning for high-stakes decision-making. Rule ensembles, which combine simple decision rules in a linear model, offer a balance between the interpretability of an additive model and the flexibility to capture non-linear relationships and interactions of tree-based ensembles. However, current rule ensemble methods rely on a two-step approach that first generates a large set of rules from decision trees and then selects a subset of these rules using a regularized linear model. This decoupling can lead to suboptimal rule ensembles, as the rules are not optimized for their combined performance in the linear model, which can result in redundant rules in the final model. To address this limitation, we propose CEGARE, a CoEvolutionary Genetic Algorithm for Rule Ensembles, which simultaneously optimizes the rules and their coefficients in a bilevel optimization framework. By coevolving both components, CEGARE aims to discover more compact and relevant rule ensembles that achieve competitive predictive performance while providing a sparser model. We benchmark CEGARE against RuleFit and other models on multiple real-world datasets, demonstrating its ability to produce sparser rule ensembles without sacrificing predictive performance.**

***Index Terms*—coevolutionary algorithms, interpretable machine learning, rule ensembles, classification**

I. Introduction

Model interpretability has become an increasingly important aspect of machine learning, particularly in high-stakes domains such as healthcare, finance, and criminal justice, where understanding the decision-making process is crucial for trust, accountability, and regulatory compliance [1]. In these domains, inherently interpretable white-box models, such as linear regression and decision trees, are often the preferred choices. However, black-box models, such as random forest (RF) and gradient boosted trees (GBT), have been shown to achieve higher performance due to their ability to capture complex non-linear relationships and interactions between features. These models can be explained using post-hoc explanation methods such as LIME [2] and SHAP [3], but these explanations may not always be faithful to the model's true decision-making process [1].

To bridge the gap between the predictive performance of black-box models and the interpretability of white-box models, various methods have been proposed. One of those methods is rule ensembles [4], which combine simple decision rules extracted from decision trees with the original features in a linear fashion to obtain an interpretable model. The key challenge in constructing rule ensembles lies in identifying a compact and relevant set of rules from a potentially vast search space that, when linearly combined, yields strong predictive performance. The predominant approach to building rule ensembles is to sequentially grow decision trees to determine a large set of rule candidates. In the original RuleFit [4] algorithm, rules are extracted from an ensemble of decision trees trained using gradient boosting; however, other tree ensemble methods, such as RF, can also be used. Each rule then corresponds to a conjunction of feature conditions derived from a decision tree split. Subsequently, an L1 regularized linear model is fitted to the extracted rules and original features to make predictions. This results in a sparse, additive, and interpretable model that can achieve competitive predictive performance by allowing for simple non-linearities and interactions between features.

However, this two-step approach is greedy in nature and does not guarantee an optimal rule ensemble. First, tree ensembles are typically designed to maximize predictive performance by combining multiple weak learners rather than to extract a compact and relevant set of rules for a linear model. Consequently, the decision trees yield a large collection of rules, many of which may remain redundant or irrelevant in the context of the final linear model, even after regularization during the second phase. Second, the decision trees are trained without considering the final linear model. On the one hand, this leads to the tree ensemble attempting to approximate linear relationships using multiple rules, which could be more efficiently captured by a single linear term. On the other hand, decision trees that are meaningful in their ensemble context may not be optimal when their rules are combined linearly.

To address these limitations of traditional rule ensemble methods, we propose CEGARE, a CoEvolutionary Genetic Algorithm Rule Ensemble, which simultaneously optimizes the rules and the coefficient vector of the rule ensemble. By coevolving both components, CEGARE aims to discover a more compact and relevant set of rules that work synergistically with the linear terms, leading to a sparser model and improved predictive performance. With a benchmark against RuleFit and other models on 7 real-world datasets from different domains, we demonstrate that CEGARE can achieve competitive predictive performance while producing sparser

and more interpretable rule ensembles. The rest of this paper is organized as follows. Section II introduces RuleFit and its limitations. Section III covers related work on coevolutionary algorithms for machine learning. Section IV describes the CEGARE algorithm in detail. Section V presents the experimental setup, including datasets, baselines, evaluation metrics, and hyperparameter tuning. Section VI discusses the experimental results, comparing CEGARE to RuleFit and other baselines in terms of predictive performance and model sparsity. Finally, Section VII concludes the paper and outlines future research directions.

II. Rule Ensembles

Rule-based models have long been valued for their interpretability, as they provide clear and human-readable reasoning behind their predictions [5]. Different approaches for rule induction exist, ranging from single decision trees to sequential covering algorithms (IF-THEN-ELSE rules) and disjunctive normal form learning (IF-THEN rules). Rule ensembles, on the other hand, for which RuleFit [4] is the most prominent algorithm, operate in two main phases. In the first phase, an ensemble of decision trees is trained using methods like GBT or RF. From these trees, rules are extracted as conjunctions of feature conditions corresponding to root-to-leaf paths. In the second phase, a Lasso-penalized linear model is fitted to these extracted rules and the original features. This results in a sparse model that balances interpretability and predictive accuracy through its native ability to capture both linear and non-linear relationships, as well as feature interactions. Mathematically, the rule ensemble is expressed as:

$$\hat{y} = \hat{\beta}_0 + \sum_{j=1}^{M} \hat{\beta}_j R_j(x) + \sum_{k=1}^{P} \hat{\gamma}_k x_k \tag{1}$$

where $\hat{y}$ is the predicted outcome, $\hat{\beta}_0$ is the intercept, M is the number of extracted rules, $R_j(x)$ are the extracted rules, $\hat{\beta}_j$ are their corresponding coefficients, P is the number of original features, x_k are the original features, and $\hat{\gamma}_k$ are their coefficients.

Although efficient, RuleFit's two-stage approach decouples rule generation from coefficient optimization, which can lead to suboptimal ensembles. Decision trees are trained to maximize ensemble predictive performance rather than to generate rules suitable for linear combination. Consequently, many extracted rules may be redundant or irrelevant in the final linear model context, and the tree ensemble may attempt to approximate linear relationships through multiple rules that could be more efficiently captured by single linear terms. Although L1 regularization effectively reduces model complexity, it can also shrink the coefficients of potentially meaningful covariates. This issue is particularly pronounced in high-dimensional settings with correlated features, where Lasso may arbitrarily select one feature while discarding others that are equally relevant [6]. Other works [7]–[11] have attempted to address this issue by generating a larger and more diverse set of rules and by applying more sophisticated regularization techniques, leading to more compact rule ensembles. However, this approach still relies on the initial rule generation step and does not address the limitations of decoupling rule generation from coefficient optimization. This motivates the need for methods that jointly optimize both rules and coefficients.

III. Coevolutionary Machine Learning

Coevolutionary algorithms decompose complex optimization problems into multiple interacting subcomponents, each evolved in separate species [12]. Unlike traditional genetic algorithms that evolve complete solutions, coevolution evolves partial solutions that must collaborate to form complete solutions. Each individual's fitness is evaluated by combining it with collaborators selected from other species, enabling the algorithm to explore modular problem structures more efficiently [13].

A particularly successful application of coevolution in machine learning is neuroevolution, where network architectures and weights can be coevolved. For instance, SANE [14] coevolves neurons and network blueprints, which are then combined to form complete neural networks. The fitness of neurons depends on the performance of the networks they help create, promoting the evolution of useful building blocks. This approach maintains diversity and encourages the discovery of specialized components that contribute to overall performance. CoDeepNEAT [15] extends this idea by using a bilevel optimization approach: network modules and blueprints are coevolved while the weights of the networks are optimized using gradient descent. The fitness of modules and blueprints is determined by the performance of the networks after weight optimization, allowing the algorithm to discover architectures that work well with learned weights. This bilevel framework is closely related to Baldwinian learning [16], where the fitness of individuals is influenced by their ability to learn during their lifetime.

For rule induction, coevolution offers a middle ground between Michigan-style algorithms, where each individual represents a single rule, and Pittsburgh-style algorithms, where each individual encodes an entire rule set [17]. While Michigan-style algorithms have a low computational cost per fitness evaluation, they struggle to capture rule interactions without explicit diversity maintenance mechanisms, such as niching [5]. Pittsburgh-style algorithms naturally model rule interactions but face scalability challenges due to the exponentially larger search space [18]. Coevolution addresses both limitations by evolving individual rules in separate species while evaluating them in the context of complete rule sets through collaboration. Similar to SANE [14], CORE [18] coevolves a population of rules and a population of rule sets. Each rule's fitness is determined by the performance of the rule sets it helps form, promoting the evolution of complementary rules that work well together. The success of coevolutionary algorithms in rule induction and neuroevolution suggests that similar principles can be applied to rule ensemble learning to decompose the problem into manageable subcomponents

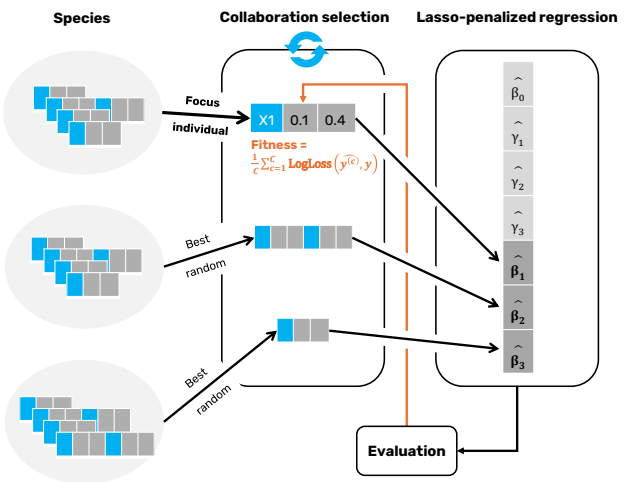

Fig. 1. CEGARE algorithm overview. Multiple species evolve rules at fixed ensemble positions. Each generation, collaborations form complete ensembles, coefficients are optimized via coordinate descent, and rules are evaluated based on ensemble performance.

while maintaining global coordination through collaborative evaluation.

IV. CEGARE Implementation

To evolve compact rule ensembles, we propose CEGARE, a CoEvolutionary Genetic Algorithm for Rule Ensembles. This is a bilevel optimization framework where multiple subpopulations explore the discrete space of rule structures, while efficiently optimizing the continuous coefficient space through local search.

A. Rule Encoding

As shown in Fig. 1, the rule population in CEGARE consists of multiple species, each representing a fixed coefficient position in the rule ensemble. Each rule is encoded as a chromosome with an integer feature pointer, a minimum quantile, and a maximum quantile for each condition. For example, a rule condition such as "Feature 3 $\leq$ 3" can be represented by the chromosome (3, 0, 0.5), where 3 is the feature pointer, 0 is the minimum quantile, and 0.5 is the maximum quantile, assuming that the value 3 corresponds to the 50th percentile of Feature 3. For numerical features, one of the quantiles is set to 0 or 1 to create one-sided rules. Categorical features can be handled by setting both quantiles to either 0 or 1, effectively creating equality conditions. Rules with multiple conditions can be formed by concatenating multiple such triples.

B. Evaluation

To obtain a compact and performant rule ensemble, rules need to cooperate to pursue predictive performance. Therefore, specialization among the species should be effectively captured in the fitness evaluation of the rules. At each generation, for each individual in the rule population, two collaborations are formed by selecting (1) the representatives, which are the fittest members from each of the other species, and (2) a random member from each of the other species, resulting in two complete rule ensembles that are separately evaluated. Each set of selected rules is combined with the original features, creating matrix $\mathbf{X}_{aug} = [\mathbf{X}_{scaled}, \mathbf{R}]$, where $\mathbf{X}_{scaled}$ represents the standardized original features scaled by 0.4 to equalize a priori importance with the rules [4], and $\mathbf{R}$ denotes the binary matrix of rule activations. In contrast to RuleFit, the number of candidate rules is limited to the number of species, a parameter that can be set manually to control the complexity of the rule ensemble. Although feature selection is less important than in RuleFit, L1 regularization remains important to penalize redundant rules. Without the lasso penalty, the logistic regression (LR) optimization may assign non-zero coefficients to multiple similar rules, leading to overfitting and reduced interpretability. The coefficients of the linear model are then optimized by minimizing the logistic loss with L1 regularization:

$$\begin{aligned} \min_{\beta_0, \boldsymbol{\beta}, \boldsymbol{\gamma}} \quad & \sum_{i=1}^{N} [\log(1 + e^{z_i}) - y_i z_i] + \lambda \left(\|\boldsymbol{\beta}\|_1 + \|\boldsymbol{\gamma}\|_1 \right) \\ \text{where} \quad & z_i = \beta_0 + \sum_{j=1}^{M} \beta_j R_j(x_i) + \mathbf{x}_i^\top \boldsymbol{\gamma} \end{aligned} \tag{2}$$

and N is the number of samples, $y_i \in \{0, 1\}$ are the true labels, and λ is the regularization parameter. The fitness of each rule is then evaluated based on the predictive performance of the complete rule ensembles in which it participates. Mathematically, the fitness of a rule R_j can be expressed as:

$$\text{Fitness}(R_j) = -\frac{1}{C} \sum_{c=1}^{C} \text{LogLoss}(\hat{y}^{(c)}, y) \tag{3}$$

where C is the number of collaborations involving rule R_j, 2 in this case, $\hat{y}^{(c)}$ are the predicted probabilities from collaboration c, and y are the true labels.

To assess the predictive performance of the rule ensembles, log loss is used as it provides a differentiable measure for the optimization of the LR coefficients and encourages good calibration of predicted probabilities. As the optimization of the coefficients for a fixed set of rules is a convex problem, the optimal coefficients for a collaboration of rules can be found efficiently using local search methods, more specifically, coordinate descent for LR with L1 regularization. The optimization of the coefficients serves as a Baldwinian learning mechanism, as the fitness of individual rules is determined by their value in the context of the complete rule ensemble after coefficient optimization. The performance of the rule ensemble after coefficient optimization then provides a meaningful fitness signal for the rules, guiding the search towards more relevant and complementary rules as rules that work well together will lead to better rule ensemble performance.

C. Evolutionary Dynamics

To maintain diversity in the population and avoid premature convergence, we implement several mechanisms. Tournament

selection identifies parents for breeding, with the top 25% of each species preserved through elitism, while the remaining 75% undergo uniform crossover, with a 50% probability for each gene to be exchanged, and mutation. This balance between high selection pressure and high genetic operator rates promotes both the exploitation of successful rules and the exploration of the search space. High mutation rates on the quantile bounds encourage exploration of different rule boundaries, while lower mutation rates on feature pointers maintain some stability in feature selection. The bounds are mutated with Gaussian noise, with a decreasing standard deviation and probability of occurrence over generations to allow for finer adjustments as the population converges. Feature mutations involve selecting a random other feature and occur with a decreasing probability over generations due to their disruptive nature. Additionally, a prune operation is applied with a small probability to simplify rules by randomly removing a condition, promoting the evolution of shorter rules.

D. Adaptive Population Management

To speed up the search process, complexification [19] is employed, where the algorithm starts with a single rule species and gradually increases the number of species over time. A new species is added every 5 generations if the previous addition led to an improvement in the rule ensemble's performance on a separate validation set. At each generation, a solution is formed using the representative rule from each species. After coefficient optimization, this rule ensemble is evaluated on the separate validation set, which is also used for early stopping if no improvement is observed for a set number of generations. If a newly added species did not lead to improvement, it is simply replaced by the next candidate species. The newly generated species is initialized randomly, but with a reduced probability for features that are already present in the existing species representatives. This approach allows the algorithm to start with simpler rule ensembles and gradually increase complexity as needed. To further improve efficiency, computational resources are dynamically allocated when a species has converged, as measured by low semantic dispersion. Specifically, when the mean pairwise Jaccard distance calculated on the binary rule activation vectors of all individuals in a species falls below a threshold, the population size of that species is halved until a minimum is reached. This allows for larger initial population sizes that promote early exploration, while later focusing computational resources on more diverse species.

V. Experimental Setup

To evaluate the performance of CEGARE, we conduct experiments on 7 real-world binary classification datasets from domains where interpretability is crucial. These datasets include the Australian Credit dataset (Credit) [20], Breast Cancer (Cancer) [21], Heart Disease (Heart) [22], Taiwanese Bankruptcy Prediction (Bankrupt) [23], Telco Customer Churn (Telco) from Kaggle, a B2B sales dataset from a Belgian company (B2B), and a customer churn dataset from a karting software company (Karting). The characteristics of these datasets are summarized in Table I. Preprocessing includes: removal of columns with $> 30\%$ missing data, removal of rows with any missing values, one-hot encoding of categorical features, and random forest feature selection (importance based on mean decrease in impurity > 0.05) for high-dimensional data ($P > 20$).

TABLE I
SUMMARY OF DATASETS USED IN THE EXPERIMENTS.

Dataset	Samples	Features after selection (P)	Positive class
Credit	690	14	44.5%
Heart	303	18	45.9%
B2B	22 439	12	6.1%
Bankrupt	6819	45	3.2%
Cancer	286	15	23.8%
Telco	7043	20	26.5%
Karting	4553	4	20.2%

TABLE II
CANDIDATE HYPERPARAMETERS FOR EACH MODEL. GBT: GRADIENT BOOSTED TREES, RF: RANDOM FOREST, P: NUMBER OF FEATURES.

Model	Hyperparameters
CEGARE	Max. rule conditions: 3 Species size: 80 L1 regularization (λ): [0.01, 0.05, 0.1] Early stopping generations: 30
RuleFit	Rule generator: [GBT, RF] Number of trees: [100, 150] Max. tree depth: 3 L1 regularization (λ): [0.01, 0.05, 0.1, 0.5, 1]
Logistic Regression	L1 regularization (λ): [0.01, 0.1, 1, 10]
Random Forest	Number of trees: [100, 200] Max. tree depth: [2, 5, 10] Max. features: ($\sqrt{P}$)

CEGARE is benchmarked against RuleFit using GBT and RF for rule generation. LR with L1 regularization and RF are included as baselines to assess the trade-off between interpretability and predictive performance. For predictive performance evaluation, area under the ROC curve (AUC) is used, as it is a threshold-independent measure of the discriminative ability of the models. Brier score is also reported to assess the calibration of the predicted probabilities. Lastly, the F1-score is reported particularly to evaluate performance on imbalanced datasets. To quantitatively evaluate model interpretability, we report the average number of rules and conditions per rule in the final rule ensembles.

All experiments are conducted using 5 x 2-fold cross-validation to ensure the robustness and generalizability of the results. Here, the full dataset is split into 2 halves, of which one is used for training and hyperparameter tuning using a fixed 25% validation subset, while the other is used for testing. This process is repeated 5 times with different random splits to obtain reliable estimates of model performance. Within these splits, grid search is conducted to find the optimal hyperparameters for each model based on validation performance. The candidate hyperparameters are listed in Table II.

VI. Results

A. Model Complexity

To quantitatively assess the interpretability of the rule ensembles produced by CEGARE and RuleFit, we analyze the complexity of the final models in terms of the number of rules and the average number of conditions per rule. Table III summarizes these metrics for both methods across the evaluated datasets. CEGARE consistently produces sparser rule ensembles with fewer rules compared to RuleFit. Additionally, the average number of conditions per rule is lower for CEGARE than for RuleFit, indicating that CEGARE favors simpler rules. On average, the number of rules decreased by 68% and the number of conditions per rule decreased by 38% when using CEGARE. This reduction in model complexity enhances interpretability, making it easier for practitioners to analyze the model's predictions.

B. Predictive Performance

Table IV summarizes the predictive performance of CEGARE compared to RuleFit, LR, and RF across the evaluated datasets. For each dataset, we report the average AUC, Brier score, and F1 score over the nested cross-validation folds. In general, RF attains the best performance across most datasets and metrics, which is expected given its ability to capture complex non-linear relationships and interactions. This, however, is at the cost of interpretability. Compared to RuleFit and other baselines, CEGARE achieves high performance, particularly on the B2B, Telco, Karting, and Cancer datasets.

To assess the statistical significance of the differences between these models, we conduct Bayesian hypothesis testing using the hierarchical correlated t-test [24]. This method accounts for the dependencies introduced by cross-validation and provides posterior probabilities for the performance differences between models, which are more straightforward to interpret than traditional p-values [25]. Table V presents the posterior probabilities that the difference in the performance metric between CEGARE and each baseline method (Δ) is negative ($P(\Delta < -\delta)$), practically equivalent ($P(-\delta \leq \Delta \leq \delta)$), or positive ($P(\Delta > \delta)$), with δ being the region of practical equivalence (ROPE) threshold. Note that for the Brier score, lower values indicate better performance, so the interpretations of the probabilities are inverted. A ROPE threshold of 0.01 is used for AUC and F1 score, while a threshold of 0.0025 is used for the Brier score, reflecting their different scales. The results indicate that CEGARE achieves practically equivalent performance to RuleFit and LR on AUC (ROPE probabilities > 0.87), while performing slightly worse than RF on AUC. For the Brier score and F1 score, CEGARE significantly outperforms RuleFit and shows mixed results against the other baselines.

TABLE III
Model complexity comparison: number of terms in the final rule ensemble and average number of conditions per rule. Lowest values are highlighted in bold.

Dataset	# terms		# conditions per rule	
	CEGARE	RuleFit	CEGARE	RuleFit
Credit	**18**	48	**1.42**	2.33
Heart	**23**	31	**1.42**	2.43
B2B	**15**	85	**1.68**	2.57
Bankrupt	**42**	94	**1.59**	2.29
Cancer	**19**	31	**1.68**	2.53
Telco	**9**	71	**1.06**	2.09
Karting	**9**	66	**1.23**	2.42

VII. Conclusion and Future Work

In this paper, we introduce CEGARE, a novel coevolutionary genetic algorithm for constructing interpretable rule ensembles. By simultaneously optimizing the rules and coefficients of the ensemble, CEGARE addresses the limitations of traditional rule ensemble methods that decouple rule generation from coefficient optimization. This greedy approach often leads to suboptimal ensembles with redundant or irrelevant rules. CEGARE employs a bilevel optimization framework, where multiple species evolve individual rules while the coefficients are optimized using local search methods. The fitness of each rule is evaluated based on the performance of the complete rule ensembles in which it participates, promoting the evolution of complementary and relevant rules. The number of rules in the ensemble, as encoded by the number of species, is automatically adapted during the evolutionary process based on the observed improvements in performance. Experimental results on 7 datasets, validated through Bayesian hypothesis testing, demonstrate that CEGARE achieves competitive predictive performance compared to LR and RF while likely outperforming RuleFit. Importantly, CEGARE produces sparser rule ensembles, enhancing interpretability without sacrificing predictive performance, as demonstrated by a reduction in the number of rules and conditions per rule.

Concerning future work, several avenues can be explored to further enhance the capabilities of CEGARE. Firstly, we intend to explore evolving the coefficients of the rule ensemble alongside the rules themselves using a genetic algorithm. This approach could enable direct optimization of different evaluation metrics, potentially leading to improved predictive performance in specific contexts. Another promising direction is the development of intelligent mechanisms for initializing the rule populations based on data characteristics to speed up convergence. Lastly, multiobjective optimization techniques could be integrated into CEGARE to simultaneously optimize for predictive performance and interpretability proxy metrics, such as the L1 norm of the coefficient vector or the total number of conditions in the rule ensemble.

References

[1] C. Rudin, "Stop explaining black box models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[2] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should i trust you?": Explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 1135–1144.

TABLE IV
PERFORMANCE COMPARISON ACROSS DATASETS AND METRICS. BEST VALUES PER METRIC ARE IN BOLD. LR: LOGISTIC REGRESSION, RF: RANDOM FOREST.

Model	Credit	Heart	B2B	Bankrupt	Cancer	Telco	Karting
AUC							
CEGARE	0.929	0.889	0.944	0.908	0.679	0.813	0.693
LR	0.928	0.895	0.943	0.927	0.658	0.807	0.685
RF	**0.935**	**0.900**	**0.948**	**0.931**	**0.698**	**0.819**	**0.703**
RuleFit	0.928	0.883	0.946	0.918	0.643	0.814	0.679
Brier							
CEGARE	0.101	0.140	0.039	0.026	0.197	0.146	0.154
LR	0.102	**0.134**	0.040	0.026	0.206	0.148	0.155
RF	**0.100**	0.135	**0.039**	**0.023**	**0.184**	**0.144**	**0.153**
RuleFit	0.122	0.164	0.048	0.029	0.250	0.170	0.173
F1							
CEGARE	0.847	0.779	0.300	**0.311**	0.405	0.522	0.881
LR	**0.849**	**0.798**	0.293	0.252	0.361	**0.522**	**0.882**
RF	0.848	0.782	**0.308**	0.271	**0.410**	0.510	0.880
RuleFit	0.821	0.776	0.247	0.293	0.390	0.418	0.865

TABLE V
BAYESIAN COMPARISON OF CEGARE AGAINST BASELINE METHODS. PROBABILITIES INDICATE THE POSTERIOR BELIEF THAT THE DIFFERENCE BETWEEN THE PERFORMANCE METRIC FOR CEGARE AND THE BASELINE METHOD (Δ) IS NEGATIVE $P(\Delta < -\delta)$, PRACTICALLY EQUIVALENT $P(-\delta \leq \Delta \leq \delta)$, OR POSITIVE $P(\Delta > \delta)$, WITH δ BEING THE REGION OF PRACTICAL EQUIVALENCE (ROPE) THRESHOLD. $\delta = 0.01$ FOR AUC AND F1 SCORE, AND $\delta = 0.0025$ FOR BRIER SCORE. LR: LOGISTIC REGRESSION, RF: RANDOM FOREST.

Comparison	$P(\Delta < -\delta)$	$P(\lvert\Delta\rvert \leq \delta)$	$P(\Delta > \delta)$
AUC			
CEGARE - RuleFit	0.13	**0.87**	0.00
CEGARE - LR	0.004	**0.96**	0.03
CEGARE - RF	0.00	0.37	**0.63**
Brier			
CEGARE - RuleFit	0.00	0.00	**1.00**
CEGARE - LR	0.01	**0.90**	0.09
CEGARE - RF	0.34	**0.66**	0.00
F1			
CEGARE - RuleFit	**0.97**	0.11	0.00
CEGARE - LR	0.42	**0.58**	0.00
CEGARE - RF	0.12	**0.88**	0.00

[3] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, p. 4768–4777.

[4] J. H. Friedman and B. E. Popescu, "Predictive learning via rule ensembles," *The Annals of Applied Statistics*, vol. 2, no. 3, pp. 916–954, 2008.

[5] A. A. Freitas, *A Survey of Evolutionary Algorithms for Data Mining and Knowledge Discovery*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 819–845.

[6] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 67, no. 2, pp. 301–320, 2005.

[7] M. Nalenz and M. Villani, "Tree ensembles with rule structured horseshoe regularization," 2018. [Online]. Available: https://arxiv.org/abs/1702.05008

[8] M. Nalenz and T. Augustin, "Compressed rule ensemble learning," in *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, G. Camps-Valls, F. J. R. Ruiz, and I. Valera, Eds., vol. 151. PMLR, 28–30 Mar 2022, pp. 9998–10 014.

[9] P. Jawanpuria, J. S. Nath, and G. Ramakrishnan, "Efficient rule ensemble learning using hierarchical kernels," in *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ser. ICML'11. Madison, WI, USA: Omnipress, 2011, p. 161–168.

[10] K. W. De Bock and A. De Caigny, "Spline-rule ensemble classifiers with structured sparsity regularization for interpretable customer churn modeling," *Decision Support Systems*, vol. 150, p. 113523, 2021, interpretable Data Science For Decision Making.

[11] F. Yang, P. Le Bodic, M. Kamp, and M. Boley, "Orthogonal gradient boosting for simpler additive rule ensembles," in *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, S. Dasgupta, S. Mandt, and Y. Li, Eds., vol. 238. PMLR, 02–04 May 2024, pp. 1117–1125. [Online]. Available: https://proceedings.mlr.press/v238/yang24b.html

[12] M. A. Potter and K. A. De Jong, "A cooperative coevolutionary approach to function optimization," in *Parallel Problem Solving from Nature — PPSN III*, Y. Davidor, H.-P. Schwefel, and R. Männer, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1994, pp. 249–257.

[13] R. P. Wiegand, "An analysis of cooperative coevolutionary algorithms," Ph.D. dissertation, George Mason University, 2004.

[14] D. E. Moriarty and R. Miikkulainen, "Forming Neural Networks Through Efficient and Adaptive Coevolution," *Evolutionary Computation*, vol. 5, no. 4, pp. 373–399, Dec. 1997.

[15] R. Miikkulainen, J. Liang, E. Meyerson, A. Rawal, D. Fink, O. Francon, B. Raju, H. Shahrzad, A. Navruzyan, N. Duffy, and B. Hodjat, "Evolving deep neural networks," in *Artificial Intelligence in the Age of Neural Networks and Brain Computing*. Elsevier, 2024, pp. 269–287.

[16] G. E. Hinton, S. J. Nowlan *et al.*, "How learning can guide evolution," *Complex systems*, vol. 1, no. 3, pp. 495–502, 1987.

[17] A. Fernández, S. García, J. Luengo, E. Bernadó-Mansilla, and F. Herrera, "Genetics-based machine learning for rule induction: State of the art, taxonomy, and comparative study," *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 6, pp. 913–941, 2010.

[18] K. C. Tan, Q. Yu, and J. H. Ang, "A coevolutionary algorithm for rules discovery in data mining," *International Journal of Systems Science*, vol. 37, no. 12, pp. 835–864, 2006.

[19] K. O. Stanley and R. Miikkulainen, "Competitive coevolution through evolutionary complexification," *J. Artif. Int. Res.*, vol. 21, no. 1, p. 63–100, Feb. 2004.

[20] R. Quinlan, "Statlog (Australian Credit Approval)," UCI Machine Learning Repository, 1987.

[21] M. Zwitter and M. Soklic, "Breast cancer," 1988.

[22] Andras Janosi, William Steinbrunn, Matthias Pfisterer, Robert Detrano, "Heart disease," 1989.

[23] Unknown, "Taiwanese bankruptcy prediction," 2020.

[24] G. Corani, A. Benavoli, J. Demšar, F. Mangili, and M. Zaffalon, "Statistical comparison of classifiers through bayesian hierarchical modelling," *Machine Learning*, vol. 106, no. 11, pp. 1817–1837, Nov 2017. [Online]. Available: 10.1007/s10994-017-5641-9

[25] A. Benavoli, G. Corani, J. Demšar, and M. Zaffalon, "Time for a change: a tutorial for comparing multiple classifiers through bayesian analysis," *Journal of Machine Learning Research*, vol. 18, no. 77, pp. 1–36, 2017. [Online]. Available: http://jmlr.org/papers/v18/16-305.html