UIC ENGINEERING

CS-594 Final report

Prof. Glavic Boris
Leonardo Borgioli

9th December 2022

Introduction

The paper introduces Falling Rule Lists (FRLs), an approach to building interpretable classification models. FRLs are designed for applications where transparency and trust are essential, such as healthcare. For example, prioritizing patients and actions based on risk is crucial, with the most at-risk patients needing immediate attention. Traditional predictive models often lack both this prioritization logic and interpretability, creating a gap between the desired outcomes and what these models can achieve.

The model balances interpretability with competitive accuracy, addressing the challenge of “black-box” predictions in traditional machine learning models.

	Conditions		**Probability**	**Support**
IF	IrregularShape AND Age ≥ 60	THEN risk is	85.22%	230
ELSE IF	SpiculatedMargin AND Age ≥ 45	THEN risk is	78.13%	64
ELSE IF	IlDefinedMargin AND Age ≥ 60	THEN risk is	69.23%	39
ELSE IF	IrregularShape	THEN risk is	63.40%	153
ELSE IF	LobularShape AND Density ≥ 2	THEN risk is	39.68%	63
ELSE IF	RoundShape AND Age ≥ 60	THEN risk is	26.09%	46
ELSE	-	THEN risk is	10.38%	366

Table 1: Falling Rule List example

The prediction as shown in Tab.1 consists of an ordered list of if-then rules, sorted by an importance criterion, where the estimated probability of success decreases monotonically as one moves down the list.

The paper aims to develop an algorithm for constructing a falling rule list for patient diagnosis, focusing on creating a highly interpretable model that allows physicians to easily understand the decision criteria by examining the ordered rules. The proposed method involves a binary classification model to estimate $p(Y|x)$, where Y represents the presence of a disease and x denotes the patient's features. The model utilizes an ordered list of if-then rules with the probability $p(Y=1)$ decreasing monotonically down the list. A Bayesian parametrization is employed to characterize the posterior distribution of the falling rule list, ensuring both accuracy and interpretability.

1 Methodology

The paper utilizes several common discrete distributions to model its approach. The Bernoulli distribution is employed to capture binary cases, such as coin tosses, with $x \in [0,1]$, and it is parameterized by $p = P(X=1) \in [0,1]$. The Poisson distribution is used to describe the rare event limit, where an increasing number of Bernoulli random variables ($n \to \infty$) have decreasing chances of success ($p = \frac{\lambda}{n} \to 0$). It is parameterized by $\lambda > 0$. Additionally, the Gamma distribution is utilized to model the waiting time until the occurrence of k events in a process, parameterized by the shape parameter α (number of events) and the rate parameter β.

In terms of Bayesian inference, the paper applies a statistical method that updates the probability of a hypothesis as new evidence becomes available. For the discrete case, the posterior distribution $p_{\Theta|X}(\theta|x)$ is computed as:

$$p_{\Theta|X}(\theta|x) = \frac{p_{\Theta}(\theta)p_{X|\Theta}(x|\theta)}{\sum_t p_{\Theta}(\theta)p_{X|\Theta}(x|t)}$$

Here, $p_{\Theta}(\theta)$ represents the prior distribution, which reflects our belief about the unknown truth Θ. $p_{X|\Theta}(x|\theta)$ is the likelihood, describing the relationship between the observed data X and Θ. $p_{\Theta|X}(\theta|x)$ is the posterior distribution, representing updated beliefs after observing Θ.

A point estimator $\hat{\theta}$ is then used to perform a single estimation $\hat{\theta}$ maps an observation x to a realistic θ, used for single observations. The Maximum A Posteriori (MAP) estimator is defined as:

$$\hat{\theta}_{MAP} = \arg\max_{\theta} p_{\Theta|X}(\theta|x)$$

The proposed plan will consist of training a falling rule list by parameterizing the model with priors and likelihoods, mining the optimal algorithm, and summarizing results. The primary objective is to find the optimal rule list while enforcing monotonicity in the risk scores (r_l) associated with each IF condition. This involves building a specific prior and determining an optimal point estimator to generate the ideal rule list for the model.

2 Training FRLs

2.1 Parematrization

	Conditions		Probability	Supp.
c_0 : IF	IrregularShape AND Age ≥ 60	THEN **r0** is	85.22%	230
c_1 : ELSE IF	SpiculatedMargin AND Age ≥ 45	THEN **r1** is	78.13%	64

Table 2: Falling Rule list parameters

A show in Tab.2 In this model, $L \in Z^+$ represents the size of the rule list, which is 2 in this example. The IF clauses, denoted as $c_l(.) \in B_x(.)$ for $l = 0, \ldots, L-1$, define the conditions of the rules. Each rule has an associated risk score $r_l \in R$ for $l = 0, \ldots, L$, with the constraint $r_{l+1} \leq r_l$ to enforce monotonicity in the list. These risk scores are fed into a logistic function to calculate the corresponding risk probabilities. The model includes $L+1$ nodes, accounting for an additional default case (ELSE) for patients who do not match any of the L rules.

The plan to compute the prior involves several key steps:

1. A reparametrization is applied to enforce monotonicity on the risk scores r_l.

2. Build a specific prior is built to reflect the constraints of the model.

3. The prior specific is exposed only to the outputs of a Mining algorithm to help with computations.

2.2 Reparametrization

The reparametrization is done as follows:

$$r_l = \log(v_l) \quad \text{for} \quad l = 0, \ldots, L$$

$$v_l = K \prod_{l'=l}^{L-1} y_{l'} \quad \text{for} \quad l = 0, \ldots, L-1$$

With the following constraints:

$$v_L = K$$

$$y_l \geq 1$$

$$K \geq 0$$

Therefore r_L (risk of default rule) is equal to $\log(K)$. After parametrization, we obtain the following:

$$\theta = \{L, \{c_l(.)\}_{l=0}^{L-1}, \{\gamma_l\}_{l=0}^{L-1}, K\}$$

The positive prior probability of $\{c_l\}_{l=0}^{L-1}$ is assigned exclusively over a list of booleans B, which is generated by a mining algorithm, specifically FPGrowth in this case. The input to the process is a binary dataset where $\mathbf{x}$ represents a boolean vector, and the output is a set of subsets of the dataset's features. Therefore the input will be a binary dataset , where x is a boolean vector and the output will be the set of subsets of the features of the dataset.

2.3 Prior Specific

It is now therefore possible to build the prior specific, here below are described the step used to do so:

1. Initialize hyperparameter
$$H = \{B, \lambda, \{\alpha_l\}_{l=0}^{|B|-1}, \alpha_K, \beta_K, w_{l}{}_{l=0}^{|B|-1}\}$$
2. Initialize $\Theta \leftarrow \{\}$
3. To ensure that the list will not have too many rows:
$$L \sim Poisson(\lambda)$$
4. Iterate: $For\ l = 0, .., L - 1$

$$\begin{aligned} &c_l(\cdot) \sim p_{c(\cdot)}\left(\cdot \mid \Theta; B, \{w_l\}_{l=0}^{|B|-1}\right) \\ &p_{c(\cdot)}\left(c(\cdot) = c_j(\cdot) \mid \Theta; B, \{w_l\}_{l=0}^{|B|-1}\right) \qquad \propto w_j \text{ if } c_j(\cdot) \notin \Theta \text{ and } 0 \text{ otherwise.} \\ &\text{ Update } \Theta \leftarrow \Theta \cup \{c_l(\cdot)\cdot\} \end{aligned}$$

Now the objective is to find the decision list with the maximum posterior probability, expressed as

$$p_{post}(L, c_{0,...,L-1}(.), K, \gamma_{0,...,L-1} \mid y_{1,...,N}; c_{1,...,N}).$$

However, the posterior does not have a simple solution and can be computationally expensive to calculate. To address this, Monte Carlo sampling is employed from the posterior distribution over the decision parameter. Monte Carlo sampling works by estimating the expected value of a function $f(x)$ over a probability distribution $p(x)$ by generating random samples from $p(x)$. The key idea is to approximate the integral

$$E[f(x)] = \int f(x)p(x)dx$$

using the empirical average of $f(x)$ evaluated at N random samples $x_1, x_2, \ldots, x_N$ drawn from $p(x)$. The approximation is given by:

$$E[f(x)] \approx \frac{1}{N}\sum_{i=1}^{N} f(x_i).$$

This method relies on the Law of Large Numbers, which ensures that the average converges to the true expected value as N increases. In our case employing this sampling method from the posterior distribution over the decision parameter:

$$\theta = \{L, \{c_l(.)\}_{l=0}^{L-1}, \{\gamma_l\}_{l=0}^{L-1}, K\}.$$

$\theta^* = \{L^*, c_{0,\ldots,L^*-1}(\cdot)^*, K^*, \gamma_{0,\ldots,L^*-1}\}$, where

$$L^*, c^*_{0,\ldots,L^*-1}(\cdot), K^*, \gamma^*_{0,\ldots,L^*-1} \in \text{argmax}_{L,c_0\ldots\ldots L-1}(\cdot), K, \gamma_{0,\ldots,L-1}\mathcal{L}$$

where $\mathcal{L} = log(p_{post})$.
This optimization problem is equivalent to finding:

$$L^*, c_{0,\ldots,L^*-1}(\cdot)^* \quad \in \text{argmax}_{L,\{c_l(\cdot)\}_{l=0}^{L-1}} \mathcal{L}\left(L, \{c_l(\cdot)\}_{l=0}^{L-1}, K^*, \gamma^*_{0,\ldots,L-1}\right)$$

where

$$K^*, \gamma^*_{0,\ldots,L-1} \quad \in \text{argmax}_{K,\gamma_{0,\ldots,L-1}} \mathcal{L}\left(L, \{c_l(\cdot)\}_{l=0}^{L-1}, K, \gamma_{0,\ldots,L-1}\right)$$

Note that K^* and $\gamma^*_{0,\ldots,L-1}$ depend on $L, \{c_l(\cdot)\}_{l=0}^{L-1}$.

To summarize: FRL takes the mined rules and constructs a sequential list of rules, referred to as a decision list. Each rule is evaluated based on its ability to explain the positive and negative samples, denoted as X_{pos} and X_{neg}. Rules that most effectively separate positive samples from negative ones are given priority. A Bayesian parameterization is employed to characterize the posterior distribution of the falling rule list.

3 Results and experiments

	FRL	NF FRL	NF GRD	RF	SVM	Logreg	Cart
Mean AUROC	.80(.02)	.75(.02)	.75(.02)	.79(.03)	.62(.06)	.82(.02)	.52(.01)

Table 3: Mean AUROC applied on preliminary readmission data

	Conditions		Probability	Support
IF	BedSores AND Noshow	THEN read. risk is:	33.25%	770
ELSE IF	PoorPrognosis AND MaxCare	THEN read. risk is:	28.42%	278
ELSE IF	PoorCondition AND Noshow	THEN read. risk is:	24.63%	337
ELSE IF	BedSores	THEN read. risk is:	19.81%	308
ELSE IF	NegativeIdeation AND Noshow	THEN read. risk is:	18.21%	291
ELSE IF	MaxCare	THEN read. risk is:	13.84%	477
ELSE IF	Noshow	THEN read. risk is:	6.00%	1127
ELSE IF	MoodProblems	THEN read. risk is:	4.45%	1325
ELSE		Read. risk is:	0.88%	3031

Table 4: Readmission risk parameters

Falling Rule Lists were applied to preliminary readmission data to predict whether a patient would be readmitted to the hospital within 30 days. The data set included preoperative and postoperative data for 8000 patients, along with 30 additional features. No parameters were tuned and the prior condition on L ensured that each rule had an equal chance of being included in the rule list. The value of λ was set to 8, and a simulated annealing search was conducted for 5000 steps. Out-of-sample performance was measured using the AUROC from 5-fold cross-validation, where the MAP decision list was used to predict each fold's test set. We can see in Tab.3 that the FRL method maintains a correct mean AUROC while having the advantage of being much more interpretable as shown in the resulting FRL in Tab.4, resulting therefore on a satisfactory result. It is important to note, however, that the models of comparisons are no longer state-of-the-art models and that the results of SVM are surprisingly low; this has also been commented on by the authors. Lastly, the relatively small number of features (30) in the hospitalization dataset might limit the model's potential for capturing more complex patterns in the data.

Performance was evaluated on several datasets from the UCI Machine Learning Repository, which is a widely used resource for benchmarking machine learning models. One such dataset is the Mammographic Mass dataset, which includes features such as BI-RADS assessment, Age, Shape, Margin, Density, and Severity. These attributes are used to classify mammographic masses as benign or malignant, providing a valuable testbed for decision-making algorithms.

Here, the FRL method performs a bit more poorly than other methods like Random Forest (RF). But still keep the advantage of being very interpretable. However, the comparison of the model with only a few approaches is limited; including boosting models such as XGBoost or LightGBM could strengthen the analysis.

Dataset	FRL	NF_FRL	NF_GRD	SVM	Logreg	CART	RF
Spam	.91(.01)	.90(.03)	.91(.03)	.97(.03)	.97(.03)	.88(.05)	.97(.03)
Mamm	.82(.02)	.67(.03)	.72(.04)	.83(.01)	.85(.02)	.82(.02)	.83(.01)
Breast	.95(.04)	.70(.11)	.82(.12)	.99(.01)	.99(.01)	.93(.04)	.98(.01)
Cars	.89(.08)	.60(.21)	.62(.20)	.94(.08)	.92(.09)	.72(.17)	.92(.05)

Table 5: Transposed comparison of methods across datasets

4 Conclusion and comments

This report presents an in-depth exploration of the Falling Rule Lists (FRLs) framework, emphasizing its capability to combine interpretability with competitive predictive performance.

The methodology includes the parameterization of risk scores to enforce monotonicity, the use of frequent itemset mining for rule extraction, and the application of Monte Carlo sampling to approximate the posterior distribution. These techniques collectively ensure that the generated rule list is both statistically robust and interpretable; the right choice of the distribution and reparametrization create a not-too-long list with a decreasing risk probability (therefore enforcing its monotonicity).

In summary, the FRL approach has potential for enhancing trust and usability in predictive modeling, particularly in domains where interpretability is as crucial as accuracy. However multiple negative aspects can be pointed out:

- The method relies on binary datasets and pre-mined feature sets, which might limit its applicability to more complex or noisy datasets.
- The complexity of constructing and optimizing the FRLs could pose challenges for real-time applications.

The paper itself was well written and presented the method clearly. However, it is to be mentioned that the models used for comparison are outdated.