Incremental Approach to Interpretable Classification Rule Learning*

Bishwamittra Ghosh and **Kuldeep S. Meel**
School of Computing
National University of Singapore

The recent advances in the machine learning techniques have led autonomous decision making systems be adopted in wide range of domains to perform data-driven decision making. As such the domains range from movie recommendations, ad predictions to legal, medical, and judicial. The diversity of domains mandate different criteria for the machine learning techniques. For domains such as movie recommendations and ad predictions, accuracy is usually the primary objective but for safety critical domains (Otte 2013) such as medical and legal, interpretability, privacy, and fairness (Barocas, Hardt, and Narayanan 2017) are of paramount importance.

It has been long observed that the interpretable techniques are typically trusted and adopted by decision makers as interpretability provides them understanding of reasoning behind a tool's decision making (Ribeiro, Singh, and Guestrin 2016). At this point, it is important to acknowledge that formalizing interpretability is a major challenge (Doshi-Velez and Kim 2017) and we do not claim to have final word on this. In this context, it is worth noting that for several domains such as medical domain, which was the motivation for our investigation, decision rules with small number of rules tend to be most interpretable (Letham et al. 2015).

Since the problem of rule learning is known to be in NP-hard, the earliest efforts focused on heuristic approaches that sought to combine heuristically chosen optimization functions with greedy algorithmic techniques. Recently, there has been surge of effort to achieve balance between accuracy and rule size via principled objective functions and usage of combinatorial optimization techniques such as linear programming (LP) relaxations, sub-modular optimization, or Bayesian methods (Bertsimas, Chang, and Rudin 2012; Marchand and Shawe-Taylor 2002; Malioutov and Varshney 2013; Wang et al. 2015). Motivated by the success of MaxSAT solving over the past decade, Malioutov and Meel proposed a MaxSAT-based approach, called MLIC (Maliotov and Meel 2018), that provides a precise control of accuracy vs. interpretability. The said approach was shown to provide interpretable Boolean formulas without significant loss of accuracy compared to the state of the art classifiers. MLIC, however, has poor scalability in terms of training time and times out for most instances beyond hundreds of samples. In this context, we ask: *Can we design a MaxSAT-based framework to efficiently construct interpretable rules without loss of accuracy and scaling to large real-world instances?*

The primary contribution of this paper is an affirmative answer to the above question. We first investigate the reason for poor scalability of MLIC and attribute it to large size (i.e., number of clauses) of MaxSAT queries constructed by MLIC. In particular, for training data of n samples over m boolean features, MLIC constructs a formula of size $\mathcal{O}(n \cdot m \cdot k)$ to construct a $k-$clause Boolean formula. We empirically observe that the performance of MaxSAT solvers has worse than quadratic degradation in runtime with increase in the size of query. This leads us to propose a novel incremental framework, called IMLI, for learning interpretable rules using MaxSAT. In contrast to MLIC, IMLI makes p queries to MaxSAT solvers with each query of the size $\mathcal{O}(\frac{n}{p} \cdot m \cdot k)$. IMLI relies on first splitting the data into p batches and then incrementally learning rules on the p batches in a linear order such that rule learned for the i-th batch not only uses the current batch but regularizes itself with respect to the rules learned from the first $i-1$ batches.

We now discuss briefly about formal logic theory and MaxSAT. A CNF (Conjunctive Normal Form) formula on a set of Boolean variables is a conjunction of clauses where each clause is a disjunction of literals. Here a literal is either a variable or its complement. Given a CNF formula, the SAT (satisfiability) problem finds an assignment to the variables that satisfies all the clauses in the formula, wherein a clause is satisfied when at least one literal in that clause is satisfied. MaxSAT is an optimization analogue to SAT, where the goal is to find an assignment that satisfies most of the clauses in the formula. In this problem, we consider a weighted variant of a CNF formula where each clause is given a positive weight. Based on the weight, there are two types of clauses in a formula: a hard clause, where the weight is ∞ and a soft clause, where the weight $\mathbb{R}^+$. To learn rules incrementally over batches of the dataset, we consider a partial weighted MaxSAT formula, where the goal is to find an optimal as-

[*] This is an extended abstract of our paper: IMLI: An Incremental Framework for MaxSAT-Based Learning of Interpretable Classification Rule, AIES-19.

signment that satisfies all the hard clauses and most of the soft clauses such that the total weight of the satisfied soft clauses is maximized.

In this paper, we reduce the learning problem as an optimization problem, where we optimize both the interpretability and the prediction accuracy of a rule. We consider a standard binary classification problem on a dataset with binary features. Features with categorical and real-valued features can be converted to binary features by applying standard discretization techniques as in (Maliotov and Meel 2018). Let $\mathbb{1}\{true\} = 1$ and $\mathbb{1}\{false\} = 0$. To learn a k-clause CNF rule from the dataset, we consider two types of boolean decision variables: feature variable $b_i^j = \mathbb{1}\{j\text{-th feature is selected in } i\text{-th clause}\}$ and noise variable $\eta_q = \mathbb{1}\{\text{sample } q \text{ is misclassified}\}$. In our proposed incremental approach, we first split the original dataset into fixed number p of batches. Given a training set $(\mathbf{X} \in \{0,1\}^{n \times m}, \mathbf{y} \in \{0,1\}^n)$ in the τ-th batch, we consider the following optimization function.

$$\min \sum_{i,j} b_i^j \cdot I(b_i^j) + \lambda \sum_q \eta_q$$

where indicator function $I(\cdot)$ is defined as follows.

$$I(b_i^j) = \begin{cases} -1 & \text{if } b_i^j = 1 \text{ in the } (\tau-1)\text{-th batch } (\tau \neq 1) \\ 1 & \text{otherwise} \end{cases}$$

The first term in the objective function tries to keep the assignment of all feature variables in the previous batch except in the first batch, where the preference is given on the sparsity (i.e., interpretability) of the rule. The second term in the objective function emphases on minimizing the prediction error. λ is the data fidelity parameter balancing the trade-off between the sparsity and the prediction accuracy of the learned rule. Higher value of λ guarantees less prediction error while sacrificing the sparsity of $\mathcal{R}$ by adding more literals in $\mathcal{R}$, and vice versa.

We now discuss how to construct the MaxSAT formula for learning rule in a batch. We construct soft clauses to encode the objective function and hard clauses to encode the constraints for all samples, that is, a positive labeled sample must satisfy the learned rule and a negative labeled sample must dissatisfy the rule, otherwise the sample is detected as a classification noise. The weight of the soft clause is derived from the coefficients in the objective function. The clauses in the MaxSAT formula are defined as follows:

$$S_j^i := \begin{cases} b_i^j & \text{if } b_i^j = 1 \text{ in the } (\tau-1)\text{-th batch}(\tau \neq 1) \\ \neg b_i^j & \quad \text{otherwise} \end{cases},$$

$$\mathsf{wt}(S_j^i) = 1;$$

$$E_q := \neg \eta_q, \qquad \mathsf{wt}(S_j^i) = \lambda;$$

$$H_q := \neg \eta_q \rightarrow \Big(y_q \leftrightarrow \bigwedge_{i=1}^{k} \mathbf{X}_q \circ \mathbf{b}_i\Big), \qquad \mathsf{wt}(H_q) = \infty.$$

In the hard clause H_q, $\mathbf{X}_q$ is the q-th row of input matrix $\mathbf{X}$, y_q is the q-th element of $\mathbf{y}$, and $\mathbf{b}_i = \{b_i^j \mid j \in \{1, \dots, m\}\}$. Between two vectors $\mathbf{u}$ and $\mathbf{v}$ over boolean variables or constants (i.e., $0, 1$), we refer $\mathbf{u} \circ \mathbf{v}$ to represent the inner product of $\mathbf{u}$ and $\mathbf{v}$, i.e., $\mathbf{u} \circ \mathbf{v} = \bigvee_i u_i \wedge v_i$, where u_i and v_i denote a variable/constant at the i-th index of $\mathbf{u}$ and $\mathbf{v}$ respectively.

Once we construct all soft and hard clauses, the MaxSAT query Q is the conjunction of all clauses.

$$Q := \bigwedge_{q=1}^{n} E_q \wedge \bigwedge_{i=1,j=1}^{i=k,j=m} S_j^i \wedge \bigwedge_{q=1}^{n} H_q$$

Our learned rule consists of features that are assigned 1 in the optimal solution of Q by an off-the-shelf MaxSAT solver.

We conduct a comprehensive experimental study over a large set of benchmarks and show that IMLI significantly improves upon the runtime performance of MLIC by achieving a speedup of up to three orders of magnitude. Furthermore, the rules learned by IMLI are significantly small and easy to interpret compared to that of the state-of-the-art classifiers such as RIPPER and MLIC. We think IMLI highlights the promise of MaxSAT-based approach and opens up several interesting research directions at the intersection of AI and SAT/SMT community. In particular, it would be an interesting direction of future research if the MaxSAT solvers can be designed to take advantage of incrementality of IMLI.

References

Barocas, S.; Hardt, M.; and Narayanan, A. 2017. Fairness and machine learning. *NIPS Tutorial*.

Bertsimas, D.; Chang, A.; and Rudin, C. 2012. An integer optimization approach to associative classification. In *Proc. of NIPS*.

Doshi-Velez, F., and Kim, B. 2017. Towards a rigorous science of interpretable machine learning.

Letham, B.; Rudin, C.; McCormick, T. H.; Madigan, D.; et al. 2015. Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model. *The Annals of Applied Statistics*.

Maliotov, D., and Meel, K. S. 2018. Mlic: A maxsat-based framework for learning interpretable classification rules. In *Proc. of CP*.

Malioutov, D. M., and Varshney, K. R. 2013. Exact rule learning via boolean compressed sensing. In *Proc. of ICML*.

Marchand, M., and Shawe-Taylor, J. 2002. The set covering machine. *Journal of Machine Learning Research* (Dec).

Otte, C. 2013. Safe and interpretable machine learning: a methodological review. In *Computational intelligence in intelligent data analysis*.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. Why should i trust you?: Explaining the predictions of any classifier. In *Proc. of KDD*.

Wang, T.; Rudin, C.; Doshi-Velez, F.; Liu, Y.; Klampfl, E.; and MacNeille, P. 2015. Or's of and's for interpretable classification, with application to context-aware recommender systems.