Incremental Approach to Interpretable Classification Rule Learning

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

CP 2019
Practical applications of machine learning

- Hiring employees
- Giving a loan to a person
- Predicting recidivism: likelihood of a person convicted of a crime to offend again
- ...

Should we believe the prediction of machine learning models?
Practical applications of machine learning

▶ Hiring employees
▶ Giving a loan to a person
▶ Predicting recidivism: likelihood of a person convicted of a crime to offend again
▶ ...

Should we believe the prediction of machine learning models?
Practical applications of machine learning

- Hiring employees
- Giving a loan to a person
- Predicting recidivism: likelihood of a person convicted of a crime to offend again
- ...

Should we believe the prediction of machine learning models?

**Interpretable classification model**
## Example Dataset

### Samples (instances, observations)

<table>
<thead>
<tr>
<th></th>
<th>Sepal length</th>
<th>Sepal width</th>
<th>Petal length</th>
<th>Petal width</th>
<th>Class label</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.1</td>
<td>3.5</td>
<td>1.4</td>
<td>0.2</td>
<td>Setosa</td>
</tr>
<tr>
<td>2</td>
<td>4.9</td>
<td>3.0</td>
<td>1.4</td>
<td>0.2</td>
<td>Setosa</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>...</td>
</tr>
<tr>
<td>50</td>
<td>6.4</td>
<td>3.5</td>
<td>4.5</td>
<td>1.2</td>
<td>Versicolor</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>...</td>
</tr>
<tr>
<td>150</td>
<td>5.9</td>
<td>3.0</td>
<td>5.0</td>
<td>1.8</td>
<td>Virginica</td>
</tr>
</tbody>
</table>

### Features

- **Petal**
- **Sepal**

### Class labels (targets)
A sample is predicted as Iris Versicolor if
(sepal length > 6.3 \textbf{OR} sepal width > 3
\textbf{OR} petal width \leq 1.5 )
\textbf{AND}
(sepal width \leq 2.7 \textbf{OR} petal length > 4
\textbf{OR} petal width > 1.2)
\textbf{AND}
(petal length \leq 5)

Interpretable Model

Black Box Model
A CNF (Conjunctive Normal Form) formula is a conjunction of clauses where each clause is a disjunction of literals

\[(a \lor \neg b \lor c) \land (d \lor e)\]

A DNF (Disjunctive Normal Form) formula is a disjunction of clauses where each clause is a conjunction of literals

\[(a \land b \land \neg c) \lor (d \land e)\]
A CNF (Conjunctive Normal Form) formula is a conjunction of clauses where each clause is a disjunction of literals

\[(a \lor \neg b \lor c) \land (d \lor e)\]

A DNF (Disjunctive Normal Form) formula is a disjunction of clauses where each clause is a conjunction of literals

\[(a \land b \land \neg c) \lor (d \land e)\]

Decision rules in CNF and DNF are highly interpretable

[Malioutov’18; Lakkaraju’19]
Definition of interpretability in rule-based classifiers

- There exists different notions of interpretability of rules
Definition of interpretability in rule-based classifiers

- There exists different notions of interpretability of rules

\[ \mathcal{R} = (a \lor b \lor \neg c \lor d \lor e) \land (f \lor g \lor h \lor \neg i) \land (j \lor k \lor \neg l) \land (\neg m \lor n \lor o \lor p \lor q) \land (a \lor b \lor \neg c) \land (f \lor g) \]

- Rules with fewer terms are considered interpretable in medical domains [Letham’15]
Definition of interpretability in rule-based classifiers

- There exists different notions of interpretability of rules

\[ \mathcal{R} = (a \lor b \lor \neg c \lor d \lor e) \land (f \lor g \lor h \lor \neg i) \land (j \lor k \lor \neg l) \land (\neg m \lor n \lor o \lor p \lor q) \land \mathcal{R} = (a \lor b \lor \neg c) \land (f \lor g) \]

- Rules with fewer terms are considered interpretable in medical domains [Letham’15]
- We refer rule size as a proxy of interpretability in rule-based classifiers
- For rules expressed as CNF/DNF, rule size = number of literals
Outline

1. Introduction
2. Preliminaries
3. Design of an interpretable rule-based classifier
4. Incremental learning
5. Experimental Evaluation
6. Conclusion
We design objective function to

- minimize prediction error
- minimize rule size (i.e., maximize interpretability)
We design objective function to
  - minimize prediction error
  - minimize rule size (i.e., maximize interpretability)

Consider decision variables:
  - feature variables $b^j_i = 1 \{j\text{-th feature is selected in } i\text{-th clause}\}$
  - noise variables $\eta_q = 1 \{\text{sample } q \text{ is misclassified}\}$

$$\min \sum_{i,j} b^j_i + \lambda \sum_q \eta_q$$

Constraints:
  - a positive labeled sample satisfies the rule
  - a negative labeled sample does not satisfy the rule
  - otherwise the sample is considered as noise
In MaxSAT

- **Hard Clause:** always satisfied, weight $= \infty$
- **Soft Clause:** can be falsified, weight $= \mathbb{R}^+$

MaxSAT finds an assignment that satisfies all hard clauses and most soft clauses such that the weight of satisfied soft clauses is maximized.
MaxSAT-based approach for interpretable rule-based classification

- the objective function is encoded as soft clauses
- the constraints are encoded as hard clauses

Analysis

- To generate a $k$-clause CNF rule for a dataset of $n$ samples over $m$ boolean features, the number of clauses of the MaxSAT instance is $\mathcal{O}(n \cdot m \cdot k)$
An Incremental Rule-learning Approach [Ghosh’19]

- We attribute large formula size of the MaxSAT instance for the poor scalability
- We propose mini-batch incremental learning
We propose a mini-batch incremental learning framework with the following objective function on batch $t$:

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

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

$$I(b^i_j) = \begin{cases} 
-1 & \text{if } b^i_j = 1 \text{ in the } (t-1)\text{-th batch } (t \neq 1) \\
1 & \text{otherwise}
\end{cases}$$
Continued...

\((t - 1)\)-th batch

we learn assignment

- \(b_1 = 0\)
- \(b_2 = 1\)
- \(b_3 = 0\)
- \(b_4 = 1\)

\(t\)-th batch

we construct soft unit clause

- \(\neg b_1\)
- \(b_2\)
- \(\neg b_3\)
- \(b_4\)
Experimental Results
# Accuracy and training time of different classifiers

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Size $n$</th>
<th>Features $m$</th>
<th>LR</th>
<th>SVC</th>
<th>RIPPER</th>
<th>IMLI</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIMA</td>
<td>768</td>
<td>134</td>
<td>75.32 (0.3s)</td>
<td>75.32 (0.37s)</td>
<td>75.32 (2.58s)</td>
<td>73.38 (0.74s)</td>
</tr>
<tr>
<td>Credit-default</td>
<td>30000</td>
<td>334</td>
<td>80.81 (6.87s)</td>
<td>80.69 (847.93s)</td>
<td>80.97 (20.37s)</td>
<td>79.41 (32.58s)</td>
</tr>
<tr>
<td>Twitter</td>
<td>49999</td>
<td>1050</td>
<td>95.67 (3.99s)</td>
<td>Timeout</td>
<td>95.56 (98.21s)</td>
<td>94.69 (59.67s)</td>
</tr>
</tbody>
</table>

**Table:** Each cell in the last 5 columns refers to test accuracy (%) and training time (s).

**IMLI** exhibits better training time by costing a little bit of accuracy.
Size of rules of different rule-based classifiers

<table>
<thead>
<tr>
<th>Dataset</th>
<th>RIPPER</th>
<th>IMLI</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIMA</td>
<td>8.25</td>
<td>3.5</td>
</tr>
<tr>
<td>Twitter</td>
<td>21.6</td>
<td>6</td>
</tr>
<tr>
<td>Credit</td>
<td>14.25</td>
<td>3</td>
</tr>
</tbody>
</table>

Table: Average size of the rules of different rule-based models.

IMLI generates shorter rules compared to other rule-based models.
Conclusion

- Interpretable ML model ensures reliability of prediction models in practice
- We propose an incremental learning approach of classification rules
- IMLI\textsuperscript{1} achieves up to three orders of magnitude improvement in training time by sacrificing a bit of accuracy
- The generated rules appear to be more interpretable

Python library:

$ pip install rulelearning

Thank You !!

\textsuperscript{1}Source code: https://github.com/meelgroup/MLIC