Decision Stumps

Rob Holte, *Machine Learning*, 11, 63-91 (1993)

Classification rules judged by two criteria:

- accuracy on an independent test set
- complexity

Relationship between the two is "of keen interest".

Indications in a number of studies that very simple rules may achieve surprisingly high accuracy.

Slide CS478–1

Goals of the Paper

Examines 1-rules:

- 1-level decision trees
- "decision stumps"
- classify an object on the basis of a single attribute

Presents a method for learning 1-rules.

Evaluates on 16 commonly used data sets: *UCI data sets*

Slide CS478–2

UCI Data Sets

123 data sets at last count.

- wide variety of characteristics
- size: 47 (soybean) to 8124 (mushroom)
- missing values: 10 data sets
- attributes: 4 (iris) to 36 (chess end-game)
- continuous features: 10 data sets
- "mixed" features: 7 data sets
- number of attribute values: 1 (20 or so attrs), 2 (150+ attrs), ..., > 6 (10 attrs)

Slide CS478–3

1R Algorithm

Create a 1-rule for each attribute

- numeric attributes treated as interval-based discrete values
  - avoid overfitting by requiring all intervals to contain > N instances
  - N = 6 for larger data sets
  - N = 3 for smaller ones
- manually deleted attributes that uniquely identify each instance (2 of these)
- "missing" values treated as legitimate values
- classification at each leaf is the majority class

Choose the 1-rule that maximizes accuracy on the training set.

Slide CS478–4

1R vs. C4.5 (1990 version)

1. randomly split the data set into two parts
   - training set (2/3)
   - test set (1/3)
2. generate a 1-rule using the training set
3. measure accuracy of the rule on the test set
4. repeat steps 1–3 25 times and average the results

Slide CS478–5

Results

		CH						
1R	68.7	67.6	53.8	72.9	73.4	76.3	81.0	97.2
C4.5	72.0	99.2	63.2	74.3	73.6	81.2	83.6	99.1
						SO		
1R	93.5	71.5	70.7	98.4	95.0	81.0	95.2	86.8
C4.5	93.8	77.2	77.5	100.0	97.7	97.5	95.6	89.4

- 1R is 5.7% < C4.5 on average
- Without CH and SO, only 3.1% less
- Half of the data sets within 2.6% of C4.5

Slide CS478–6

What's wrong with C4.5?

- Is C4.5 missing opportunities to exploit additional complexity? Pruned trees have same accuracy as the unpruned ones.
- C4.5 is probably not overfitting: forcing C4.5 to build 1-rules didn't do better.
- In fact, C4.5's performance on these data sets is better than most learning systems (based on a survey of the literature).
- Maybe it's just the data sets. We'll revisit this.

Anything special about CH and SO?

Only one attribute with more values than there are classes.

CH	2 classes	1 attr w/3 values	35 attrs w/2 values
SO	4 classes	1 attr w/7 values	4 w/4, 30 w/ < 4 values

Attributes with fewer values than classes can't predict all classes.

This characteristic doesn't always cause 1-rules to perform poorly. Depends on the distribution of class values across the leaves.

Practical significance

Most of the examples in most of the data sets studied can be classified correctly by very simple rules.

The practical significance of this results hinges on whether or not the procedures and data sets faithfully reflect the conditions that arise in practice.

Intuition is that "real" classification problems wouldn't be solved by such simple rules. We know that some real-world data sets are "hard" in that simple rules are inadequate: protein structure prediction.

Data set is representative of real problems if...

1. It is drawn from a real-life domain.
2. Examples and attributes must be typical of those that naturally arise in the domain.

Are the data sets representative of those that arise in practice?

1. Drawn from a real-life domain
   - all of the data sets fulfill this
2. Examples and attributes must be typical of those that naturally arise in the domain.
   - Six data sets: ok (well documented)
   - Two voting data sets: probably ok. Nine possible positions on a bill (in original data set) were reduced to three. But groupings are natural; not contrived to improve results of learning.

- Three data sets: probably ok; involved "cleaning" of the data which is not described in detail.
- SO: created for ML experiments; likely to have been engineered to ensure classification with relatively simple rules.
- CH: doesn't qualify: features engineered by a chess expert working with ID3.
- Six data sets: no published information on their creation.

"Simplicity first" Methodology

Many learning algorithms search in a very *large hypothesis space* containing very *complex hypotheses*. Much work focuses on developing better heuristics for navigating in these spaces towards simple, accurate hypotheses.

Holte et al. paper suggests an alternative. Search in a relatively *small space* containing only *simple hypotheses*. Develop methods for expanding the search space to include slightly more complex hypotheses that fix specific deficiencies.

Advantages of "simplicity first" methodology

1. Guaranteed to produce rules that are near-optimal w.r.t. simplicity. If accuracy isn't high enough, then consider more complex hypotheses.
2. Formal analysis of simple hypothesis spaces and the associated simple learning algorithms is easier.
3. Simple hypothesis spaces are smaller, allowing the application of algorithms that would be impractical in a larger space.
4. Difficult issues (noise, continuous attributes, overfitting) are more easily studied in the smaller, simpler context.