Rule-Based Classification

Johannes Fürnkranz
Knowledge Engineering Group
TU Darmstadt

juffi@ke.informatik.tu-darmstadt.de
Local vs. Global Rule learning

Local Rule Discovery
- Find a rule that allows to make predictions for some examples
- Techniques:
  - Association Rule Discovery
  - Subgroup Discovery
  - ...

Global Rule Learning
- Find a rule set with which we can make a prediction for all examples
- Techniques:
  - Decision Tree Learning / Divide-And-Conquer
  - Covering / Separate-And-Conquer
  - Weighted Covering
  - Classification by Association Rule Discovery
  - Statistical Rule Learning
  - ...
Local Patterns and Covering

- Covering is a simple, proto-typical strategy for constructing a global theory out of local patterns

```
function COVERING(Examples)

# initialize the classifier
GlobalClassifier ← Ø

# loop until all examples are covered
while Examples ≠ Ø

    # find the best local pattern
    LocalPattern ← FINDBESTLOCALPATTERN(Examples)

    # add the local pattern to the classifier
    GlobalClassifier ← GlobalClassifier ∪ LocalPattern

    # remove the covered examples
    Examples ← Examples \ COVERED(LocalPattern,Examples)

return GlobalClassifier
```

**Key Problem:**
- What is the best local pattern?
What is the Best Local Pattern?

- We have a global requirement...
  - We want a rule set that is as accurate as possible
  - ... that needs to be translated into local constraints.

→ What local properties are good for achieving the global requirement?
  - class probability close to 1?
  - class probability different from prior probability?
  - coverage of the pattern?
  - size of the pattern?
  - ...

- Typically decided by a single rule learning heuristic / rule evaluation metric
What is measured by a Rule Learning Heuristic?

- Rule learning heuristics focus on good discrimination between positive and negative examples
  - **Consistency:**
    - cover few negative examples
  - **Coverage:**
    - cover many positive examples

- Commonly used heuristics
  - information gain, m-Estimate, weighted relative accuracy / Klösgen measures, correlation, ...
  - Study of trade-off between consistency and coverage in many popular rule learning heuristics (Janssen & Fürnkranz, submitted to MLJ-08)
What should be measured by a Rule Learning Heuristics?

- **Discrimination**
  - How good are the positive examples separated from the negative examples?

- **Completeness**
  - How many positive examples are covered?

- **Gain**
  - How good is the rule in comparison to other rules (e.g., default rule, predecessor rules)?

- **Novelty**
  - How different is the rule from known or previously found rules?

- **Utility**
  - How good / useful will be the local pattern in a team with other patterns?

- **Bias**
  - How will the quality estimate change on new examples?

- **Potential**
  - How close is the rule to a good rule?
Discrimination

- How good are the positive examples separated from the negative examples?

- Typically ensured by some sort of purity measure
  - e.g., precision \( h_{\text{prec}} = \frac{p}{p+n} \)

- Most other measures try to achieve different goals at the same time!
  - e.g., Laplace / m-Estimate
    → bias correction and coverage
Completeness

- How many positive examples are covered?

- Can be maximized in different ways
  - directly
    - include an explicit term that captures coverage
      - weighted relative accuracy
        \[ h_{WRA} = \frac{p + n}{P + N} \left( \frac{p}{p + n} - \frac{P}{P + N} \right) \]
      - information gain
        \[ h_{foil} = -p \left( \log_2 c - \log_2 \frac{p}{p + n} \right) \]
  - indirectly
    - implicit biases towards coverage
    - e.g., Laplace or m-Estimate
      \[ h_{Lap} = \frac{p + 1}{p + n + 2} \]
  - algorithmically
    - the covering loop makes sure that successive rules cover at least one new examples
    - can also be found, e.g., in many classification by association algorithms
Gain

- How good is the rule in comparison to other rules?

  Can be found in various heuristics
  - information gain compares to predecessor rule
    \[ h_{foil} = -p \left( \log_2 \frac{p'}{p'+n}, - \log_2 \frac{p}{p+n} \right) \]
  - weighted relative accuracy compares to default rule
    \[ h_{WRA} = \frac{p+n}{P+N} \left( \frac{p}{p+n} - \frac{P}{P+N} \right) \]

- Lift / Leverage compare to a rule with empty body
  \[ h_{lift} = \frac{\text{confidence}(A \rightarrow B)}{\text{confidence}(\rightarrow B)} \quad h_{leverage} = \text{confidence}(\rightarrow B) - \text{confidence}(A \rightarrow B) \]

- Various concepts in association rule discovery
  - e.g., prune a condition if it doing so does not change the support
  - e.g., closed itemsets / rules
Novelty

- How different is the rule from known or previously found rules?

- Novelty is an important criterion for local pattern discovery by itself
  - part of the classical definition of Knowledge Discovery by Fayyad et al.
  - however, difficult to formalize what is known

- In the context of global pattern discovery, the covering loop can be used to ensure that new patterns are found
  - the knowledge of the past is implicitly handled by removing the examples that are covered by known rules

- trade-off between novelty and other criteria can be realized by weighted covering
  - instead of entirely removing covered examples, only reduce their weight
  - has also been used for local pattern discovery (e.g., Lavrac et al.)
(Global) Utility

- How good / useful will be the local pattern in a team with other patterns?

- The covering loop only takes care of the past (novelty)
  - We also should consider how well the remaining examples will be covered by future rules

- The future is tried to be captured by some heuristics, in particular in decision trees
  - rule learning heuristics typically only consider the examples covered by the current rule
  - decision tree heuristics try to optimize all branches / rules simultaneously
  - Foil's information gain heuristic vs. C4.5's information gain

- Ripper's optimization loop
  - repeatedly try to re-learn a rule in the context of all other rules

- Pattern team selection heuristics
  - (Knobbe et al., Bringmann & Zimmermann, Rückert)
Bias

- How will the quality estimate change on new examples?

- Various works on estimating the out-of-sample precision/confidence/etc. of a local pattern
  - statistical
    - modeling the distribution of local patterns (Scheffer, IDAJ 05)
    - correct optimistic evaluations (Mozina et al. ECML-06)
  - meta-learning
    - trying to predict the performance of a rule on an independent test set (Janssen & Fürnkranz, ICDM-07)
  - pruning / evaluation on a separate pruning set
    - I-REP (Fürnkranz & Widmer 1994), Ripper (Cohen 1995) for classification rules
    - recently also proposed for local pattern evaluation (Webb, MLJ 2008)
Potential

- How close is the rule to a good rule?

- If exhaustive search is not feasible, **heuristic search** might be an option
  - Typically, heuristic search algorithms evaluate candidate patterns by their quality according to some rule learning heuristic

- We need a **clear formulation as a search problem**
  - do not evaluate the quality of the rule
  - but how close it gets us to the goal (a high-quality rule)

- Approaches
  - use bounds to bound the quality function
    - optimistic pruning (Webb, Zimmermann et al.)
      - assume that the best refinement of the rule will cover all positives and no negatives
      - if not better → prune
  - reinforcement learning to learn a function for the search problem
    - preliminary (bad) results
Conclusion

- Inducing good Rule-Based Classifiers is still a *not very well understood* problem
  - despite decades of research

- Various algorithms are known to **perform well**
  - but their solutions are *ad hoc* and not very principled

- Typical **rule learning heuristics** address (too) many problems at once
  - maybe trying to understand each of them separately is a first step for understanding their interplay

- **Rule-Based Classification is not an old hat!**