

CSE 512 LABORATORY – Week 8, Winter 2017

Prof. Kerstin Voigt

Implement the GSCA algorithm as described in Nilsson's textbook and in the lecture notes. Implement the "greedy variant" which strives to add the attribute the results in the greatest coverage of positive training instances in each step.

Apply Divide-and-Conquer to the task at hand:

- Start by choosing a representation of the set (table) of training instances. Use the training data from the lecture handout (Nilsson's book example).

Then ask yourself (in this or any other order):

- How do you keep track of the attributes (which ones are they, which ones have already been used up for the generation of prior rules etc.)?
- How do you determine that a training instance is positive (negative)?
- How do you want to represent an individual rule? The emerging set of rules? (Hint:: since all rules are of the sort "antecedent \rightarrow v+", storing the antecedent for each rule will be sufficient).
- How will you determine that a current set of learned rules covers only positive instances? Covers all positive instances?

The answers to the above questions should all lead to one or more relatively small and straightforward functions. The functions should then be put to use in a top-level function, e.g., `learn_rules(data)`, which will iteratively learn decision rules from the set of training data until the body of learned rules accounts for all positive instances (e.g., all instances with `LOAN-OK = 1`).

What to hand in? Nothing for today's lab. However, expect a next homework assignment to consist of the completion of this lab, and the application of our learning program to other sets of training data.