Machine Learning

Thomas G. Dietterich

notes by Urszula Chajewska
February 20, 1996

Philosophical Foundations

Defining learning can be reduced to defining knowledge. Learning is simply an increase in knowledge.
Possible definitions of knowledge:

- An agent knows a fact F if the agent has been told F or can logically infer F from its other knowledge (*deductive closure knowledge*). According to this definition, learning occurs when an agent is told a fact F (that it did not know) and when an agent makes an "inductive leap" and chooses to believe some fact F that is not entailed by its existing knowledge.
- Distinction between explicit (facts told) and implicit (facts that can be inferred) knowledge. Learning - finding new explicit beliefs.

Learning systems:

- *speed-up learning* - no input, increased efficiency
- *learning by being told* - new knowledge acquired only via inputs
- *inductive learning* - new knowledge acquired by inductive leaps

Inductive learning can provide no guarantee of correctness. Solution: probabilistic framework (Leslie Valiant) - the fact is learned if there is high probability that it is approximately correct (and the learning system is then called *PAC - probably approximately correct*).

- a "learned" fact F is "approximately correct" (with respect to actual fact G) if the symmetric difference F G is smaller than the *accuracy parameter* (small constant ε)
- probability that the learning system will produce an approximately correct fact must be bigger than
  1 - *confidence parameter* δ assuming the training sample is representative of the universe.

Theoretical Results on Learning From Examples

<u>Bias</u> - assumptions about the fact F that enable inductive leaps

- <u>restricted-hypothesis-space bias</u> - assumption that the correct concept F is a member of some hypothesis set H, where H contains only some of the 2|U| possible concepts over U.
  - ♣ Finite hypothesis spaces
    Bounds on the number of examples required for learning in a given hypothesis space:
    $m \geq 1/\varepsilon (\ln 1/\delta + \ln |H|)$ theorem by Blumer et al. 1987
    Hypothesis space sizes for various concept representations - table 1 page 262
  - ♣ Real-valued parameters
    Vapnik-Chervonenkis dimension - measure of bias
    Idea: although an hypothesis space may contain uncountably many hypotheses, those hypotheses may still have restricted expressive power

A set of hypotheses can *completely fit* a collection of examples $E \subseteq U$ if, for every possible way of labeling the elements of E positive or negative, there exists an hypothesis in H that will produce that labeling. VC-dimension is the size |E| of the largest set of examples that H can completely fit.
VC dimension for linear threshold units over n-dimensional Euclidean space is n+1.

- ♣ A space of hypotheses H is PAC learnable iff it has finite VC-dimension - theorem by Blumer et al. 1989
- ♣ Any PAC learning algorithm for H must examine $\Omega(1/\varepsilon\ (\ln 1/\delta + \text{VC-dim}(H)))$ training examples - theorem by Ehrenfeucht et al. 1988.

〈 <u>preference bias</u> - a preference ordering is imposed over the set of hypotheses; goal - find the best hypothesis according to this ordering

- ♣ I(F) - numerical position of hypothesis F in this ordering
  Number of examples required: $1/\varepsilon\ (\ln 1/\delta + \ln I(F)))$ - from the first Blumer's theorem

All of the above assumes that the training examples are complete and correct.
Adjustment for <u>noisy data</u>: find a concept that is consistent with fraction $1 - \varepsilon/2$ of the training examples.

<u>Computational Complexity</u>

An hypothesis space H is polynomially learnable if
(a) only a polynomial number of training examples are required (as a function of input features n, $1/\varepsilon$ and $1/\delta$);
(b) a consistent hypothesis from H can be found in time polynomial in n, $1/\varepsilon$ and $1/\delta$
See table 2 on page 268 for complexities for several hypothesis spaces.

Recent Developments in Practical Learning Algorithms

〈 Improvements to Decision-Tree Methods

- ♣ ID3 - top-down recursive algorithm for constructing a decision tree (by Quinlan)
  Shortcomings: 1. the number of training examples for deep recursive calls is very small;
  2. "replication problem" (not compact for DNF representation); 3. a batch algorithm
- ♣ Converting decision trees to production rules (Quinlan) solves problems 1 and 2.
- ♣ incremental version of ID3 called ID5 (by Utgoff) solves problem 3. Needs to rearrange the tree to avoid problems 1 and 2.

〈 The Back-Propagation Algorithm for Training Multi-Layer Neural Networks (by Rumelhart et al. 1986)

- ♣ Goal: minimizing the squared error between the output of the network and the correct outputs provided in the training examples
- ♣ Process: gradient descent search in weight space
- ♣ Problem: the functions computed by individual units must be differentiable; linear threshold units are discontinuous at θ
- ♣ Solution: approximation of the linear threshold unit by the function $y = 1/(1 + e(wx + \theta))$
- ♣ Shortcomings of the algorithm: 1. the training set must be processed many times before the weight values converge; 2. sometimes the weight values converge to a local optimum that is not a global optimum.
- ♣ Shortcomings of neural networks: 1. the need to choose the number of hidden units; 2. high cost of the learning process.

- ♣ Advantages: 1. the ability to learn a wide variety of concepts; 2. the ability to learn concepts involving real-valued functions.

〈 Hybrid Algorithms
- ♣ Stagger system (Schlimmer and Granger 1986) - combines Bayesian weight-learning algorithm with a method for constructing Boolean expressions.
- ♣ Fringe algorithm (Pagallo 1989) - combines decision trees and Boolean feature combinations.
  Avoids problems 1 and 2 of ID3.
- ♣ Perceptron tree algorithm (Utgoff 1988) - a decision tree in which the leaf nodes are perceptrons and the internal nodes are standard decision nodes. The algorithm is incremental.

Explanation-Based Learning

〈 The basic EBL procedure - a method for speed-up learning; a technique for improving the computational efficiency of reasoning programs
- ♣ analogous to caching with pattern matching
- ♣ a process of converting problem-solving search into pattern-matching search
- ♣ the solution to a solved problem is generalized and stored together with applicability constraints
- ♣ intermediate representation: an AND tree
- ♣ the results of EBL can be represented as new rules and added to the rule base
- ♣ mEBL - exploits multiple (similar) training examples; finds the largest subtree shared by all the proofs of individual examples; the rule extracted is more general

〈 Integrating EBL into problem-solving architectures
- ♣ problem-solving architectures that perform EBL automatically: SOAR (Laird et al. 1986) and Prodigy (Minton 1988)
- ♣ work by acquiring *control rules* (EBL at meta-level)
- ♣ Advantages: 1. usually there are only a few meta-level rules for many base-level rules; 2. meta-level rules can be significantly more effective

〈 Lessons and problems
- ♣ the vocabulary of the domain theory must be chosen carefully
- ♣ the quality of the rule learned by the system can be greatly affected by the quality of the explanation given to the EBL procedure
- ♣ some form of post-optimization of the learned rules is critical
- ♣ it is important to be selective in EBL

〈 Generalization-to-n - problems with learning iterative procedures
- ♣ solution: analyze the explanation to find iterative structure and represent it as a recursive rule

〈 Imperfect domain theories require introduction of some form of inductive learning
- ♣ *abduction theory completion* (e.g. ODYSSEUS by Wilkins 1988) - when the domain theory is unable to produce a complete proof that the example is an instance of the target concept, construct a maximal partial proof and then hypothesize new rules to fill the remaining "holes". It is implemented by a parser; expensive (requires both top-down and bottom-up parsing).
- ♣ Induction Over Explanations (IOE, Dietterich and Flann 1988) - approach to refining promiscuous domain theories (theories that produce

incorrect explanation because they try to explain everything) by finding a maximally specific explanation.