Decision Tree Learning

CS4780/5780 – Machine Learning
Fall 2014

Thorsten Joachims
Cornell University

Reading: Mitchell Sections 2.1-2.3, 2.5-2.5.2, 2.7, Chapter 3

Supervised Learning

- Task:
  - Learn (to imitate) a function f: X → Y
- Training Examples:
  - Learning algorithm is given the correct value of the function for particular inputs → training examples
  - An example is a pair (x, f(x)), where x is the input and f(x) is the output of the function applied to x.
- Goal:
  - Find a function h: X → Y that approximates f: X → Y as well as possible.

Hypothesis Space

	correct (complete, partial, guessing)	color (yes, no)	original (yes, no)	presentation (clear, unclear)	binder (yes, no)	A+
1	complete	yes	yes	clear	no	yes
2	complete	no	yes	clear	no	yes
3	partial	yes	no	unclear	no	no
4	complete	yes	yes	clear	yes	yes

Instance Space X: Set of all possible objects described by attributes.

Target Function f: Maps each instance x ∈ X to target label y ∈ Y (hidden).

Hypothesis h: Function that approximates f.

Hypothesis Space H: Set of functions we consider for approximating f.

Training Data S: Set of instances labeled with target function f.

Inductive Learning Strategy

- Strategy and hope (for now, later theory):

  Any hypothesis h found to approximate the target function f well over a sufficiently large set of training examples S will also approximate the target function well over other unobserved examples.
- Can compute:
  - A hypothesis h ∈ H such that h(x)=f(x) for all x ∈ S.
- Ultimate Goal:
  - A hypothesis h ∈ H such that h(x)=f(x) for all x ∈ X.

Consistency

Definition: *A hypothesis h is* **consistent** *with a set of training examples S if and only if $h(x) = y$ for each training example $(x, y) \in S$.*

$$Consistent(h, S) \equiv [\forall (x, y) \in S : h(x) = y]$$

	correct (complete, partial, guessing)	color (yes, no)	original (yes, no)	presentation (clear, unclear)	binder (yes, no)	A+
1	complete	yes	yes	clear	no	yes
2	complete	no	yes	clear	no	yes
3	partial	yes	no	unclear	no	no
4	complete	yes	yes	clear	yes	yes

Version Space

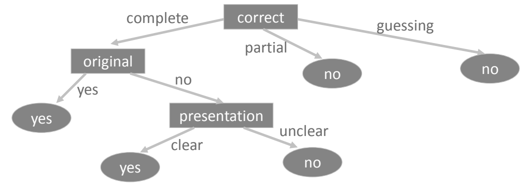
Definition: *The* **version space**, $VS_{H,S}$, *with respect to hypothesis space H and training examples S, is the subset of hypotheses from H consistent with all training examples in S.*

$$VS_{H,S} \equiv \{h \in H | Consistent(h, S)\}$$

List-Then-Eliminate Algorithm

- Init VS ← H
- For each training example (x, y) ∈ S
  - remove from VS any hypothesis h for which h(x) ≠ y
- Output VS

Decision Tree Example: A$^+$ Homework



	correct (complete, partial, guessing)	color (yes, no)	original (yes, no)	presentation (clear, unclear)	binder (yes, no)	A+
1	complete	yes	yes	clear	no	yes
2	complete	no	yes	clear	no	yes
3	partial	yes	no	unclear	no	no
4	complete	yes	yes	clear	yes	yes

Top-Down Induction of DT (simplified)

Training Data: $S = ((\vec{x}_1, y_1), \dots, (\vec{x}_n, y_n))$

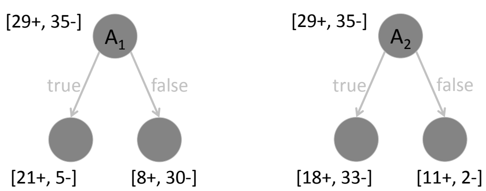
TDIDT(S,y_{def})
- IF(all examples in S have same class y)
  - Return leaf with class y (or class y_{def}, if S is empty)
- ELSE
  - Pick A as the "best" decision attribute for next node
  - FOR each value v_i of A create a new descendent of node
    - $S_i = \{ (\vec{x}, y) \in S : \text{attribute } A \text{ of } \vec{x} \text{ has value } v_i \}$
    - Subtree t_i for v_i is TDIDT(S_i,y_{def})
  - RETURN tree with A as root and t_i as subtrees

Example: TDIDT

TDIDT(S,y_{def})
- IF(all ex in S have same y)
  - Return leaf with class y (or class y_{def}, if S is empty)
- ELSE
  - Pick A as the "best" decision attribute for next node
  - FOR each value v_i of A create a new descendent of node
    - $S_i = \{ (\vec{x}, y) \in S : \text{attr } A \text{ of } \vec{x} \text{ has val } v_i \}$
    - Subtree t_i for v_i is TDIDT(S_i,y_{def})
  - RETURN tree with A as root and t_i as subtrees

C O P	A^+
$\vec{x}_1 = (\text{c, y, c})$	$y_1 = +1$
$\vec{x}_2 = (\text{c, n, u})$	$y_2 = -1$
$\vec{x}_3 = (\text{c, y, u})$	$y_3 = +1$
$\vec{x}_4 = (\text{c, n, c})$	$y_4 = +1$
$\vec{x}_5 = (\text{p, y, c})$	$y_5 = -1$
$\vec{x}_6 = (\text{g, y, c})$	$y_6 = -1$
$\vec{x}_7 = (\text{c, y, c})$	$y_7 = +1$
$\vec{x}_8 = (\text{c, y, u})$	$y_8 = +1$
$\vec{x}_9 = (\text{p, y, c})$	$y_9 = -1$
$\vec{x}_{10} = (\text{c, y, c})$	$y_{10} = +1$

Which Attribute is "Best"?



Example: Text Classification

- Task: Learn rule that classifies Reuters Business News
  - Class +: "Corporate Acquisitions"
  - Class -: Other articles
  - 2000 training instances
- Representation:
  - Boolean attributes, indicating presence of a keyword in article
  - 9947 such keywords (more accurately, word "stems")

+ LAROCHE STARTS BID FOR NECO SHARES

Investor David F. La Roche of North Kingstown, R.I., said he is offering to purchase 170,000 common shares of NECO Enterprises Inc at 26 dlrs each. He said the successful completion of the offer, plus shares he already owns, would give him 50.5 pct of NECO's 962,016 common shares. La Roche said he may buy more, and possible all NECO shares. He said the offer and withdrawal rights will expire at 1630 EST/2130 gmt, March 30, 1987.

- SALANT CORP 1ST QTR FEB 28 NET

Oper shr profit seven cts vs loss 12 cts. Oper net profit 216,000 vs loss 401,000. Sales 21.4 mln vs 24.9 mln. NOTE: Current year net excludes 142,000 dlr tax credit. Company operating in Chapter 11 bankruptcy.

Decision Tree for “Corporate Acq.”

```
• vs = 1: -
• vs = 0:
• |   export = 1:
…
• |   export = 0:
• |   |   rate = 1:
• |   |   |   stake = 1: +
• |   |   |   stake = 0:
• |   |   |   |   debenture = 1: +
• |   |   |   |   debenture = 0:
• |   |   |   |   |   takeover = 1: +
• |   |   |   |   |   takeover = 0:
• |   |   |   |   |   |   file = 0: -
• |   |   |   |   |   |   file = 1:
• |   |   |   |   |   |   |   share = 1: +
• |   |   |   |   |   |   |   share = 0: -
… and many more
```

Learned tree:

- has 437 nodes
- is consistent

Accuracy of learned tree:

- 11% error rate

Note: word stems expanded for improved readability.