CSCI 5582
Artificial Intelligence

Lecture 19

Jim Martin

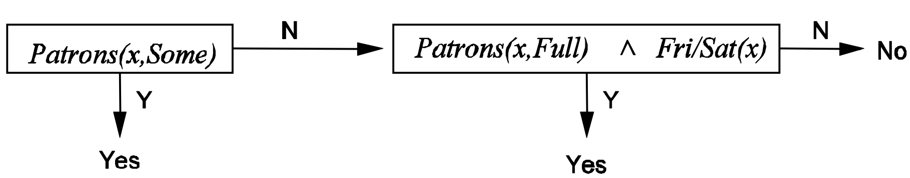
Today 11/13

- Decision Lists
- Break
  - Quiz review
  - New HWs
- Boosting

Decision Lists

- Each element in the list is a test that an object can pass or fail.
- If it passes, emit the label associated with the test.
- If it fails, move to the next test.
- If an object fails all the tests emit a default answer.
- The tests are propositional logic statements where the feature/value combinations are atomic propositions.

Decision Lists



Decision Lists

- Key parameters:
  - Maximum allowable length of the list
  - Maximum number of elements in a test
  - Logical connectives allowed in the test
- The longer the lists, and the more complex the tests, the larger the hypothesis space.

Decision List Learning

function DECISION-LIST-LEARNING(*examples*) **returns** a decision list, *No* or failure

if *examples* is empty **then return** the value *No*
t ← a test that matches a nonempty subset $examples_t$ of *examples*
such that the members of $examples_t$ are all positive or all negative
if there is no such *t* **then return** failure
if the examples in $examples_t$ are positive **then** *o* ← *Yes*
else *o* ← *No*
return a decision list with initial test *t* and outcome *o*
and remaining elements given by DECISION-LIST-LEARNING($examples - examples_t$)

Training Data

#	F1 (In/Out)	F2 (Meat/Veg)	F3 (Red/Green/Blue)	Label
1	In	Veg	Red	Yes
2	Out	Meat	Green	Yes
3	In	Veg	Red	Yes
4	In	Meat	Red	Yes
5	In	Veg	Red	Yes
6	Out	Meat	Green	Yes
7	Out	Meat	Red	No
8	Out	Veg	Green	No

Decision Lists

- Let's try
  [F1 = In] → Yes

Training Data

#	F1 (In/Out)	F2 (Meat/Veg)	F3 (Red/Green/Blue)	Label
1	In	Veg	Red	Yes
2	Out	Meat	Green	Yes
3	In	Veg	Red	Yes
4	In	Meat	Red	Yes
5	In	Veg	Red	Yes
6	Out	Meat	Green	Yes
7	Out	Meat	Red	No
8	Out	Veg	Green	No

Decision Lists

- [F1 = In] → Yes
- [F2 = Veg] → No

Training Data

#	F1 (In/Out)	F2 (Meat/Veg)	F3 (Red/Green/Blue)	Label
1	In	Veg	Red	Yes
2	Out	Meat	Green	Yes
3	In	Veg	Red	Yes
4	In	Meat	Red	Yes
5	In	Veg	Red	Yes
6	Out	Meat	Green	Yes
7	Out	Meat	Red	No
8	Out	Veg	Green	No

Decision Lists

- [F1 = In] → Yes
- [F2 = Veg] → No
- [F3=Green] → Yes

Training Data

#	F1 (In/Out)	F2 (Meat/Veg)	F3 (Red/Green/Blue)	Label
1	In	Veg	Red	Yes
2	Out	Meat	Green	Yes
3	In	Veg	Red	Yes
4	In	Meat	Red	Yes
5	In	Veg	Red	Yes
6	Out	Meat	Green	Yes
7	Out	Meat	Red	No
8	Out	Veg	Green	No

Decision Lists

- [F1 = In] → Yes
- [F2 = Veg] → No
- [F3=Green] → Yes
- No

Covering and Splitting

- The decision tree learning algorithm is a splitting approach.
  - The training set is split apart according to the results of a test
  - Until all the splits are uniform
- Decision list learning is a covering algorithm
  - Tests are generated that uniformly cover a subset of the training set
  - Until all the data are covered

Choosing a Test

- What tests should be put at the front of the list?
  - Tests that are simple?
  - Tests that uniformly cover large numbers of examples?
  - Both?

Choosing a Test

- What about choosing tests that only cover small numbers of examples?
  - Would that ever be a good idea?
    - Sure, suppose that you have a large heterogeneous group with one label.
    - And a small homogeneous group with a different label.
    - You don't need to characterize the big group, just the small one.

Decision Lists

- The flexibility in defining the tests and the length of the lists is a big advantage to decision lists.
  - (Decision trees can end up being a bit unwieldy)

What Does Matter?

- I said that in practical applications the choice of ML technique doesn't really matter.
- They will all result in the same error rate (give or take)
- So what does matter?

What Matters

- Having the right set of features in the training set
- Having enough training data

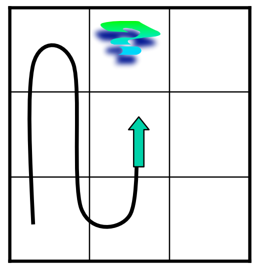
Break

- The next quiz will be on 11/28.
- It will cover the ML material and the probabilistic sequence material.
- The readings for this quiz are:
  - Chapter 18
  - Chapter 19
  - Chapter 20: 712-718
  - HMM chapter posted on the web

Quiz

- 1. True
- 2. Soundness: All inferred sentences are entailed.
- 3. Stench and Wumpus
- 4. Probabilities and Wumpus
- 5. Belief nets.

Wumpus



Wumpus

- What do you know about the presence or absence of a wumpus in [2,3] before the game even begins?
- What do you know about it after you first detect a stench in [1,3]?

Wumpus (Q 3)

a) ~S22 => (~w23 ^ ~w12 ^ ~w32 ^ ~w21)

b) By MP

    a) ~S22 and the above rule:

        ~w23 ^ ~w12 ^ ~w32 ^ ~w21

        By And elim

        ~w23

c)We know ~Wx,y in all but W33. We know that there has to be one wumpus

(w11 or w12 or w13…)

Successive resolutions will result in W33.

Wumpus (Q4)

P(W|S) = P(S|W) P(W)/P(S)

= P(W)/P(S)

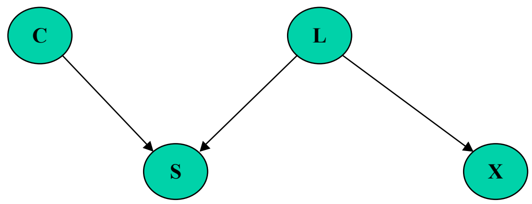
= .25 / P(S)

= .25 / P(S,W)+P(S,~W)

=.25 /P(S|W)P(W)+P(S|~W)P(~W)

=.25 /(.25 + P(S|~W)*.75)

Q 5



P(C)P(L)P(S|C,L)P(X|L)

CSCI 5582 Fall 2006

Q5

- b) P(L|S) = P(L,S)/P(S)

$$\frac{P(L)\sum_c P(c)P(S \mid L,c)}{\sum_x \sum_c \sum_l P(c)P(l)P(S \mid l,c)P(x \mid l)}$$

- c) P(C|S,~X) = P(C,S, ~X)/P(S,~X)

$$\frac{P(C)\sum_l P(S \mid l,C)P(\sim X \mid l)}{\sum_c \sum_l P(c)P(l)P(S \mid l,c)P(\sim X \mid l)}$$

CSCI 5582 Fall 2006

HWs

- We'll have two remaining HWs.
- The next one will be due 12/5; the second is due on 12/14.
- Basic idea for assignment 1:
  - I give you two bodies of texts by two authors (labeled). You train a system to recognize the work of each author.
  - To test I give you new texts by each author.

Colloquium

- I'm giving the colloquium on Thursday on natural language processing research here at CU.
- Your chance to heckle me in public.
  - Ask me where the HWs are.

Computational Learning Theory

- The big elements of this are:
  - |H| the size of the hypothesis space
    - For lists, the number of possible lists
  - The number of training instances
  - The acceptable error rate of a hypothesis
  - The probability that a given hypothesis is a good one (has an error rate in the acceptable range).

CLT

- First an exercise with a coin...
  - A bunch of folks get identical copies of a coin. Their job is to say its either a normal coin or a two-headed coin. By getting the results of flips (without examining both sides)
  - Let's say that you go about this by assuming one hypothesis and try to disprove that hypothesis via flips

Coin Flipping

- Ok, given this framework what's a good hypothesis (fair vs. fake).
  - Fake
    - Fake can be disproved by one flip (tails)
    - Fair can't be logically disproved by any number of flips.

Coin Flipping

- You let these people flip five times.
- The lucky folks will encounter a tails and report the coin is fair.
- The unlucky folks will get 5 heads in a row and… report that they think its fake.
  - How many? 1/32

Coin Flipping

- Say there are 320 flippers… How many unlucky folks will there be?
  - 10
- Ok… now you decide you're going to ask a random person what they think about this coin and that's the answer you're going to go with.
- What's the probability that you'll stumble over an unlucky flipper
  - 1/32

CLT

- Back to learning…
  - Learning is viewed as candidate hypothesis elimination.
  - Each training example can be seen as a filter on the space of possible hypotheses
  - Hypotheses inconsistent with a training example are filtered out leaving the ones that are consistent (give the right answer)
  - What do we know about those?

CLT

- Ok what about the ones that are consistent… two kinds
  - Hypotheses that are flat out wrong, but just coincidentally give the right answer
  - Hypotheses that are basically right (and got the right answer because they're normally right)

CLT

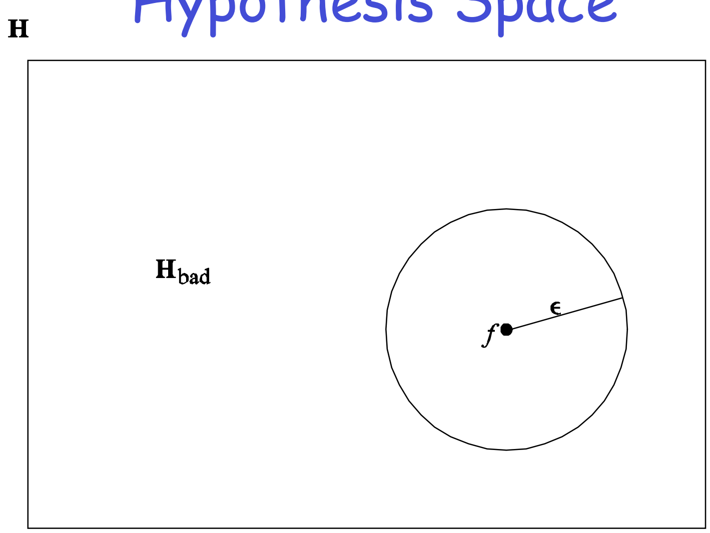
- So run the training data as a filter on the hypotheses
- When the data runs out pick a random hypothesis from among the ones still left standing (remember we don't know what the right answer is).
- Can we say something about the probability of being unlucky and picking a hypothesis that is really wrong?

CLT

- Yup… its clearly based on the size of the hypothesis space and just how long a bad hypothesis can keep giving the right answers (ie. The size of the training set).

CSCI 5582 Fall 2006

Hypothesis Space



CSCI 5582 Fall 2006

Bad Hypotheses

- Say we're happy with any hypothesis that has an error rate no more than 5%.
- So any hypothesis with an error rate greater than 5% is in H_bad.

Bad Hypotheses

- Look at one with error rate of 20%. The probability of it being correct on any given example is?
  .8
- Probability correct on n examples?
  .8^n
- How many of those will there be?
  ? < .8^n * |H_bad| < .8^n*|H|

So…

- The name of the game is to say that if I want to be X% sure that I'm going to have a solution with an error rate no worse than Y% then either I have to
  - Reduce the number of surviving bad hypotheses
    - More training examples
  - Or reduce |H|
    - Restrict the hypothesis space by restricting the expressiveness of the possible answers
  - Or provide a bias for how to select from among surviving hypotheses (Occam).

Next

- Thursday
  - Ensembles (Sec 18.4)
  - SVMs and NNs (20.5 and 20.6)
- Next week
  - Chapter 19: learning with knowledge