Self-improvement for dummies
(Machine Learning)

COS 116, 4/21/2011

Sanjeev Arora

Artificial Intelligence

- Definition of AI (Merriam-Webster):

Next time:

1. The capability of a machine to imitate intelligent human behavior

Today:

2. Branch of computer science dealing with the simulation of intelligent behavior in computers

- Definition of Learning:
  - To gain knowledge or understanding of or skill in by study, instruction, or experience

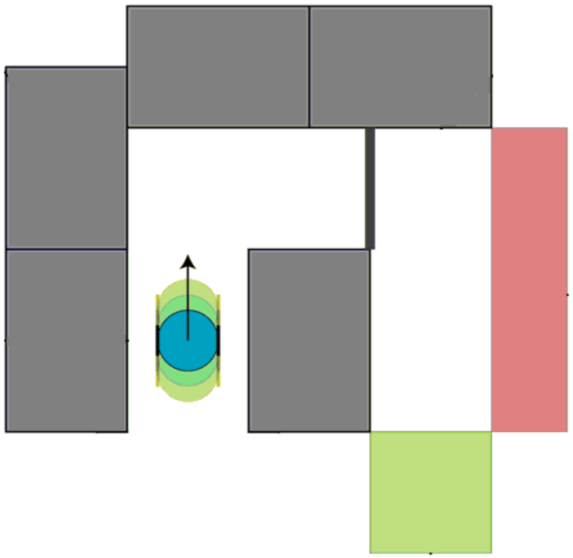
Today's lecture: Machine Learning

- Machine learning =
  "Programming by example"
- Show the computer <u>what</u> to do,
  without explaining <u>how</u> to do it.
- The computer programs itself!

In fact, continuous improvement via more data/experience.

Recall your final Scribbler lab

- Task: Program Scribbler to navigate a maze.
  - Avoid walls, avoid “lava”, head towards the goal.



- As maze becomes more complex, programming becomes much harder. (Why?)

~~Program~~ *Teach* Scribbler to navigate a maze

Start with a simple program:

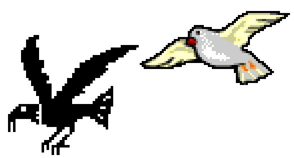
1. Run the maze.
2. Label this trial GOOD or BAD, depending on whether goal was reached.
3. Submit data from the trial to a “learning algorithm”, which uses it to devise a better program.
4. Repeat as needed.

- Is this how you learned to drive a car?

Note: imitating nature may not be best

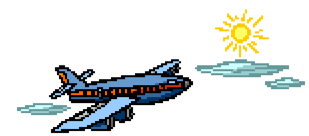
- Examples:

Birds



vs

Airplanes



Cheetahs

vs

Race cars

Machine's "experience" of the world

- *n* sensors, each produces a number:
  "experience" = an array of *n* numbers
- Example: video camera: 480 x 640 pixels
  $n = 480 \times 640 = 307200$
- In practice, reduce *n* via some processing

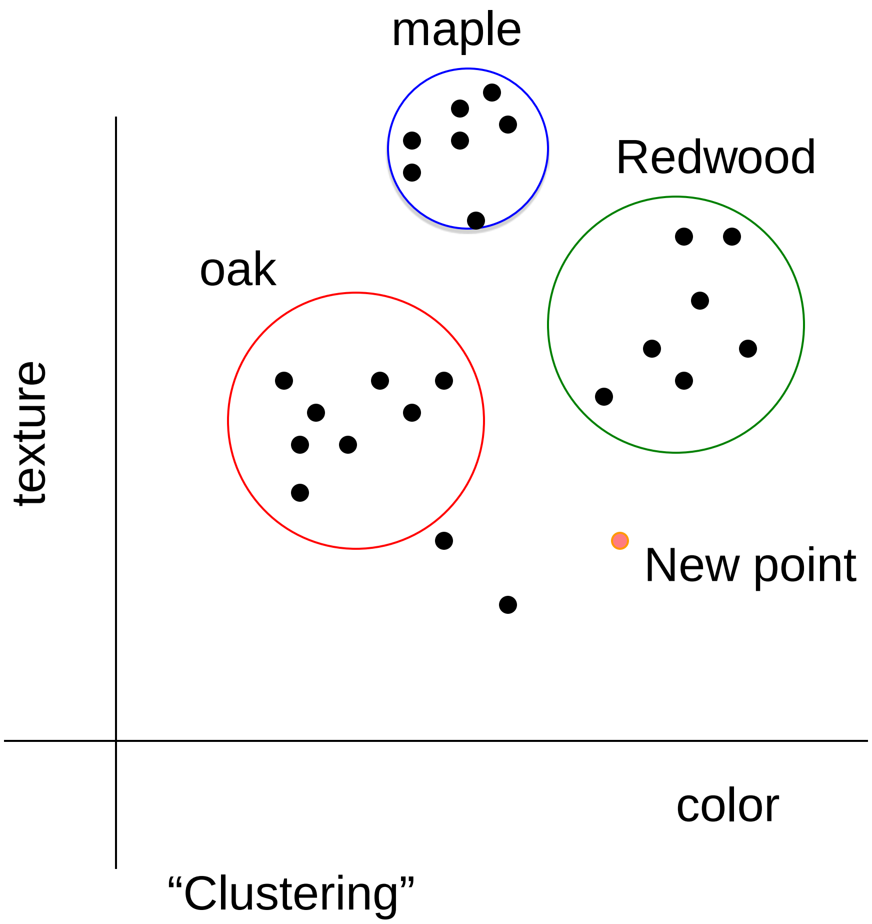
Example: Representing wood samples

Brownness scale	1 light	…	10 dark
Texture scale	1 smooth	…	10 rough

(3, 7) = wood that is fairly light brown but kind of on the rough side

A learning task and its mathematical formulation

- Given: 100 samples of oak, maple
- Figure out labeling (“clustering”)
- Given a new sample, classify it as oak, maple…



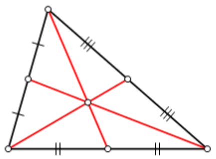
Fact: producing “best” clustering is NP-complete

3-means algorithm (produces 3 clusters)

- Some notions:
  - Mean of k points $(x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k)$ is

$$\left(\frac{x_1 + x_2 + \ldots + x_k}{k}, \frac{y_1 + y_2 + \ldots + y_k}{k} \right)$$

("center of gravity")



  - Distance between points $(x_1, y_1), (x_2, y_2)$ is $((x_1 - x_2)^2 + (y_1 - y_2)^2)^{1/2}$

3-means Algorithm (cont.)

Start by randomly picking 3 data points as your “means”

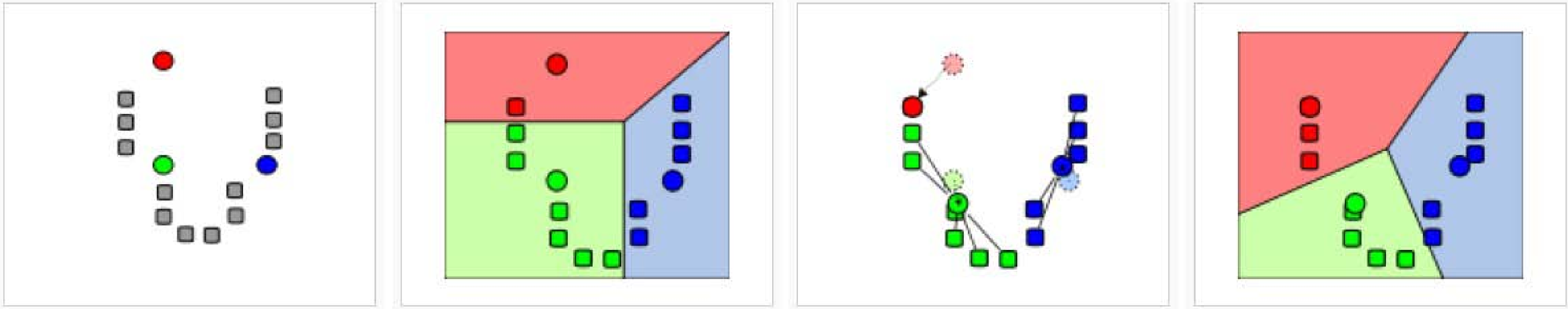
Repeat many times:

{

- Assign each point to the cluster whose mean is closest to it
- Compute means of the clusters

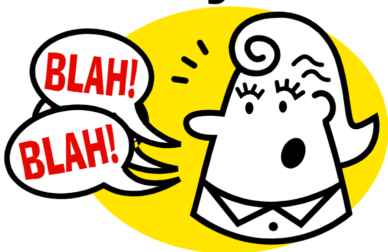
}

http://en.wikipedia.org/wiki/K-means_clustering



What about learning a more complicated object?

- Speech?

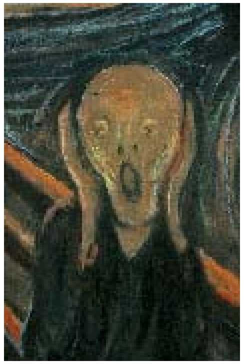


- Motion?

- Handwriting?

Similar data representations, but more "dimensions"

One major idea: modeling uncertainty using probabilities



- Example: Did I just hear
  "Ice cream" or "I scream"?
- Assign probability ½ to each
- Listen for subsequent phoneme
  - _?_ "is": use knowledge of usage patterns…
  - Increase probability of "Ice cream" to 0.9


SPAM®
Ingredients: Pork with Ham, Salt, Water, Sugar, Sodium Nitrite.
GET
SPAM
STUFF
SEE BACK FOR DETAILS
NET WT. 12 OZ. (340g)
U.S. INSPECTED AND PASSED BY DEPARTMENT OF AGRICULTURE
Hormel Foods
Serving
Sugges

Spam filtering

- How would you define Spam to a computer?
- Descriptive approach:
  - "Any email in ALL CAPS, unless it's from my kid brother, or that contains the word 'mortgage', unless it's from my real estate agent, …"
  - Difficult to come up with an good description!
- Learning approach:
  - "Train" the computer with labeled examples of spam and non-spam (a.k.a. ham) email.
  - Easy to find examples of spam – you probably get hundreds a day!

Spam Filtering

- Given: A spam corpus and ham corpus.
- Goal: Determine whether a new email is spam or ham.

- Step 1: Assign a "spam score" to each *word:*
  - $F_{spam}(word)$ = Fraction of emails in spam corpus that contain *word*.
  - $F_{ham}(word)$ = Fraction of emails in ham corpus that contain *word*.

$$SpamScore(word) = \frac{F_{spam}(word)}{F_{ham}(word)}$$

  - Observe:
    - SpamScore(*word*) > 1 if *word* is more prevalent in spam.
    - SpamScore(*word*) < 1 if *word* is more prevalent in ham.

Spam Filtering

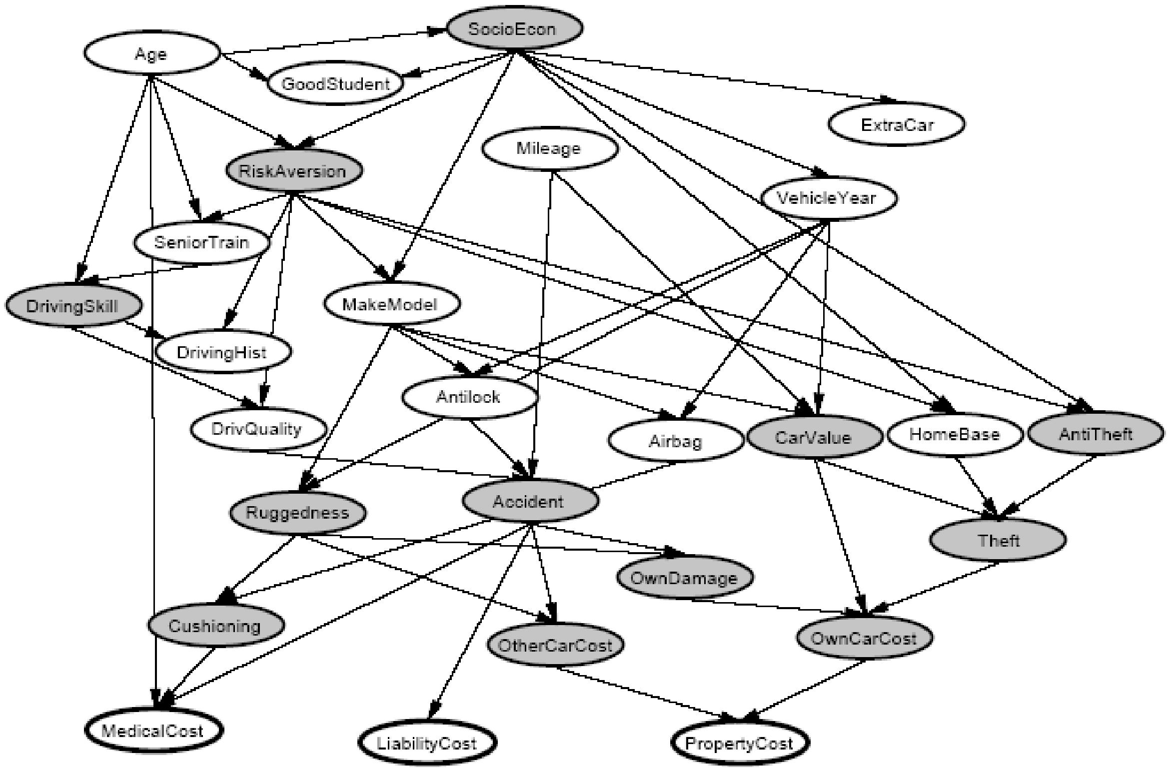
- Step 2: Assign a “spam score” to the *email*:
  - SpamScore(*email*) = SpamScore($word_1$) x … x SpamScore($word_n$),
    where $word_i$ is the ith word in *email*.
  - Observe:
    - SpamScore(*email*) >> 1 if *email* contains many spammy words.
    - SpamScore(*email*) << 1 if *email* contains many hammy words.
- Step 3: Declare *email* to be spam if SpamScore(*email*) is high

Spam Filtering

- Advantages of this type of spam filter:
  - Though simple, catches 90+% of spam!
  - No explicit definition of spam required.
  - Customized for your email.
  - Adaptive – as spam changes, so does filter

Dominant ML paradigm today

Learn a probabilistic "model" (i.e. succinct description of data)



Bayes net model for identifying low-risk customers for automobile insurance (arrows denote "dependent random variables")

Preview of lab: Text synthesis (simplistic version)

- Idea: Use example text to generate similar text.
  - Input: 2007 State of the Union Address.
  - Output: “This war is more competitive by strengthening math and science skills. The lives of our nation was attacked, I ask you to make the same standards, and a prompt up-or-down vote on the work we've done and reduce gasoline usage in the NBA.”

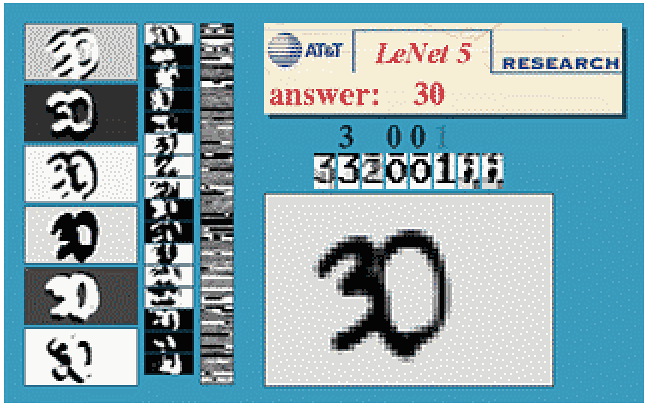
Text synthesis

- How it works: Output one word at a time.
  1. Let (v, *w)* be the last two words outputted.
  2. Find all occurrences of (v, *w)* in the input text.
  3. Of the words following the occurrences of (v, w), output one at random.
  4. Repeat.
- Variants: Last *k* words instead of last 2 words.

Handwriting recognition

[LeCun et al, AT&T, 1998]

- The LeNet-5 system
  - Trained on a database: 60,000 handwritten digits
  - Reads ~10% of all checks cashed in the US; can tolerate weird handwriting, errors etc.



Aside: How to get large amounts of data? (major problem in ML)

- Answer 1: Use existing corpuses (lexis-nexis, WWW for text)

- Answer 2: Create new corpuses by enlisting people in fun activities. (Recall Image-Labeling Game in Lab 1)

Example: SAT Analogies

- Bird : Feathers :: Fish : ____
- Idea: Search web to learn relationships between words. [Turney 2004]
  - Example: Is the answer above "water" or "scales"?
    - Most common phrases on the web:
      "bird *has* feathers", "bird *in* air", "fish *has* scales", "fish *in* water".
    - Conclusion:
      Right answer is "scales".

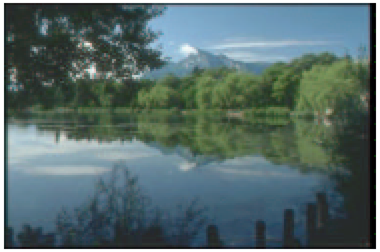
SAT Analogies [Turney 2004]

- On a set of 374 multiple-choice SAT analogies, this approach got 56% correct.
- High-school seniors on the same set:
  - 57% (!)
- Mark of "Scholastic Aptitude"?

Image labeling [Blei et al, 2003]

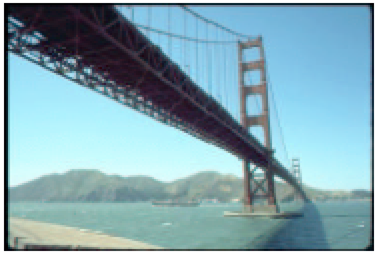
Princeton prof!

- Another solution:
  Learn captions from examples.
  - System trained on a Corel database
    - 6,000 images with captions.
  - Applied to images without captions.



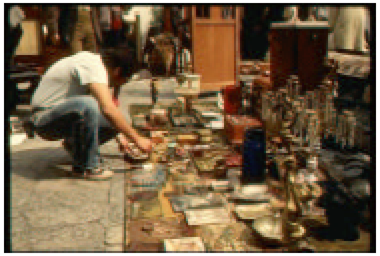
SKY WATER TREE MOUNTAIN PEOPLE

SCOTLAND WATER FLOWER HILLS TREE



SKY WATER BUILDING PEOPLE WATER

FISH WATER OCEAN TREE CORAL



PEOPLE MARKET PATTERN TEXTILE DISPLAY

BIRDS NEST TREE BRANCH LEAVES

Helicopter flight [Abbeel et al 2005]

- Idea: Algorithm learns to pilot a helicopter by observing a human pilot.
- Results: Even <u>better</u> than the human pilot.

Next time: Artificial Intelligence

What do we mean by it? Turing's test for intelligence.

To do before then:

(a)Complete assigned readings.

(b) Participate in a Turing test on Turinghub.com. Cut and paste the entire conversation into a document and bring to class to hand in (and pass around).

(c)Write a para on how convincing you find Searle's objection.