Machine Learning: Course Overview

Yingyu Liang
Computer Sciences 760
Fall 2017

http://pages.cs.wisc.edu/~yliang/cs760/

Some of the slides in these lectures have been adapted/borrowed from materials developed by Mark Craven, David Page, Jude Shavlik, Tom Mitchell, Nina Balcan, Elad Hazan, Tom Dietterich, and Pedro Domingos.

Class enrollment

- typically the class was limited to 30
- we've allowed ~100 to register
- the waiting list full

- unfortunately, many on the waiting list will not be able to enroll
- but CS760 will be offered in the Spring semester!

Instructor

- Yingyu Liang

  email: yliang@cs.wisc.edu

  office hours: 3-5pm, Monday, or by appointment

  office: 6393 Computer Sciences

TA

- Heemanshu Suri
  - email: hsuri@wisc.edu
  - office hours: 3-5pm, Thursday
  - office: TBA

Monday, Wednesday *and* Friday?

- we'll have ~30 lectures in all, just like a standard TR class
- will push the lectures forward (finish early, leave time for projects and review)

- ***see the schedule on the course website***

Course emphases

- a variety of learning settings: supervised learning, unsupervised learning, reinforcement learning, active learning, etc.
- a broad toolbox of machine-learning methods: decision trees, nearest neighbor, neural nets, Bayesian networks, SVMs, etc.
- some underlying theory: bias-variance tradeoff, PAC learning, mistake-bound theory, etc.
- experimental methodology for evaluating learning systems: cross validation, ROC and PR curves, hypothesis testing, etc.

Two major goals

1. Understand what a learning system should do

2. Understand how (and how well) existing systems work

Course requirements

- 5-6 homework assignments: ~75%
  - programming
  - computational experiments (e.g. measure the effect of varying parameter x in algorithm y)
  - some written exercises
- final exam or project (choose one): ~25%
  - project group: 3-5 people

Expected background

- CS 540 (Intro to Artificial Intelligence) or equivalent
- good programming skills
- probability
- linear algebra
- calculus, including partial derivatives

Programming languages

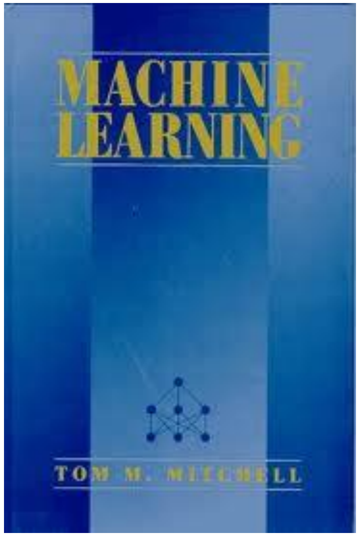
- for the programming assignments, you can use
  - C
  - C++
  - Java
  - Perl
  - Python
  - R
  - Matlab

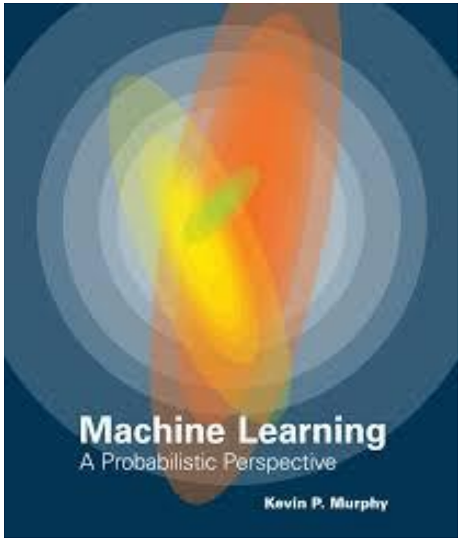
- programs must be callable from the command line and *must run on the CS lab machines (this is where they will be tested during grading!)*

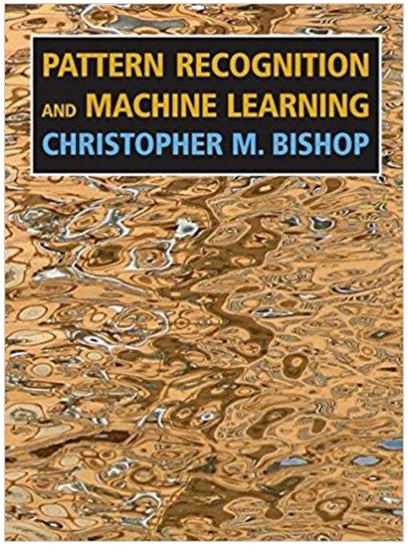
Course readings

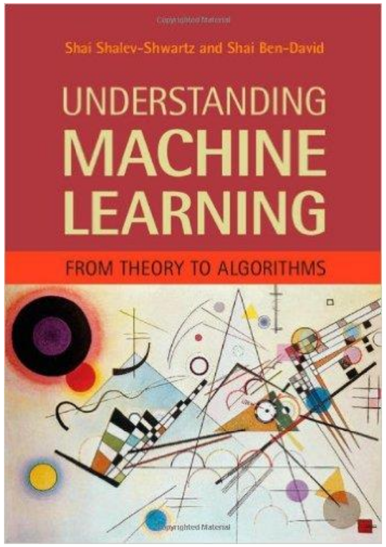
Recommend to get <u>one</u> of the following books

- *Machine Learning*. T. Mitchell. McGraw Hill, 1997.
- *Pattern Recognition and Machine Learning.* C. Bishop. Springer, 2011.
- *Machine Learning: A Probabilistic Perspective*. K. Murphy. MIT Press, 2012.
- *Understanding Machine Learning: From Theory to Algorithms.* S. Shalev-Shwartz, S. Ben-David. Cambridge University press, 2014.









Course readings

- the books can be found online or at Wendt Commons Library

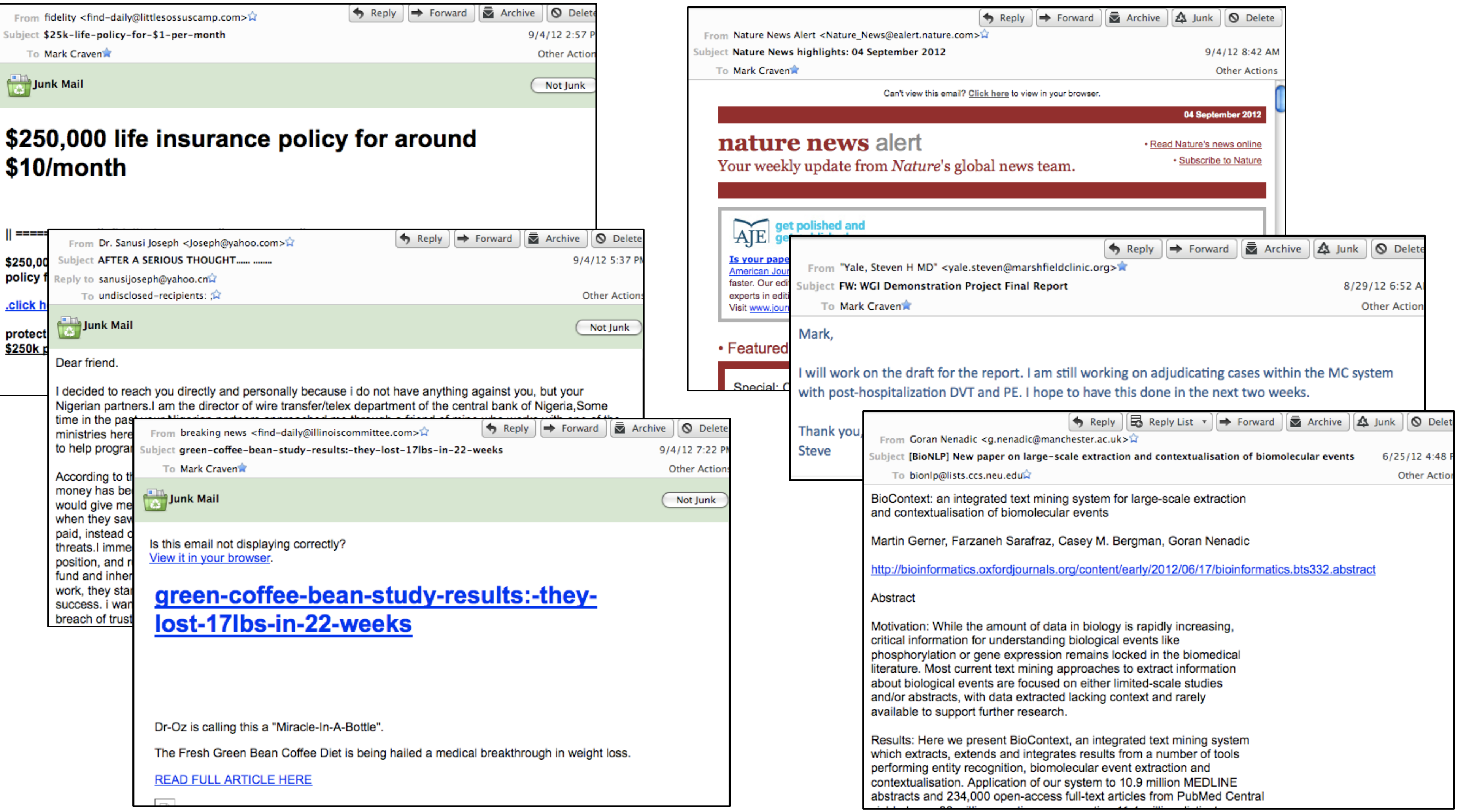
- additional readings will come from online articles, surveys, and chapters
- will be posted on course website

What is machine learning?

- the study of algorithms that
  improve their performance P
  at some task T
  with experience E

- to have a well defined learning task,
  we must specify: $< P, T, E >$

ML example: spam filtering



ML example: spam filtering

- T : given new mail message, classify as spam vs. other
- P : minimize misclassification costs
- E : previously classified (filed) messages

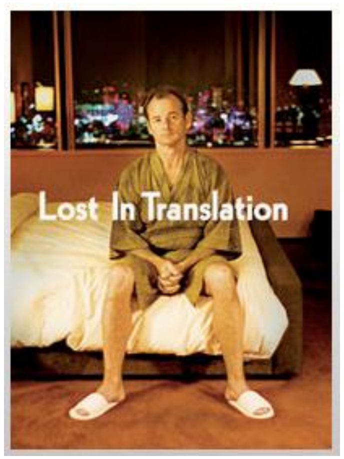
ML example: predictive text input

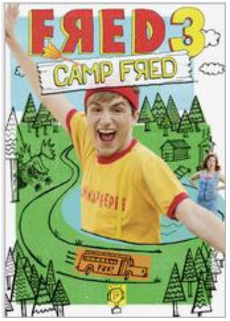
ML example: predictive text input

- T : given (partially) typed word, predict the word the user intended to type
- P : minimize misclassifications
- E : words previously typed by the user
  (+ lexicon of common words + knowledge of keyboard layout)

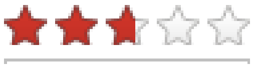
domain knowledge

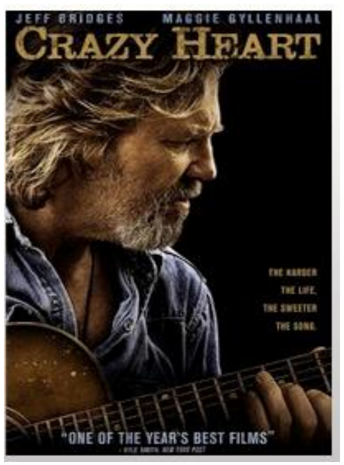
ML example: Netflix Prize

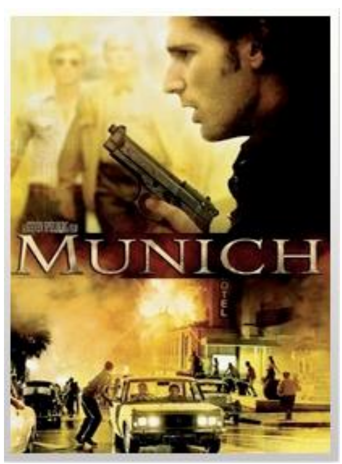


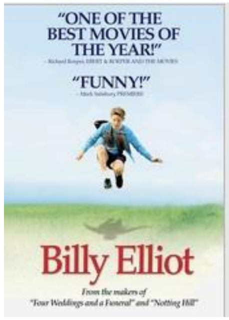


Our best guess for Mark:

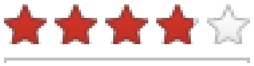








Our best guess for Mark:



ML example: Netflix

- T : given a user/movie pair, predict the user's rating (1-5 stars) of the movie
- P : minimize difference between predicted and actual rating
- E : histories of previously rated movies (user/movie/rating triples)

ML example: reinforcement learning to control an autonomous helicopter

video of Stanford University autonomous helicopter from http://heli.stanford.edu/

ML example: autonomous helicopter

- T : given a measurement of the helicopter's current state (orientation sensor, GPS, cameras), select an adjustment of the controls
- P : maximize reward (intended trajectory + penalty function)
- E : state, action and reward triples from previous demonstration flights

Reading assignment

- for Friday, read
  - Chapter 1 of Mitchell **or** Chapter 1 of Murphy
  - article by Dietterich on course website
  - article by Jordan and Mitchell on course website

- course website:
  http://pages.cs.wisc.edu/~yliang/cs760/

HW1: Background test

- posted on course website; due in two weeks (Sep 20)
- will set up how to submit the solutions soon

- contains: minimum and medium tests

- if pass medium: in good shape
- if pass minimum but not medium: can still take but expect to fill in background
- if fail both: suggest to fill in background before taking the course

Minimum background test

- 80 pts in total; pass: 48pts
- linear algebra: 20 pts
- probability: 20 pts
- calculus: 20 pts
- big-O notations: 20 pts

Minimum test example

$$X = \begin{pmatrix} 9 & 8 \\ 7 & 6 \end{pmatrix} \qquad \mathbf{y} = \begin{pmatrix} 9 \\ 8 \end{pmatrix} \qquad \mathbf{z} = \begin{pmatrix} 7 \\ 6 \end{pmatrix}$$

1. What is the inner product of the vectors $\mathbf{y}$ and $\mathbf{z}$? (this is also sometimes called the *dot product*, and is sometimes written as $\mathbf{y}^T\mathbf{z}$)

2. What is the product $X\mathbf{y}$?

3. Is X invertible? If so, give the inverse, and if no, explain why not.

4. What is the rank of X?

Minimum test example

1. If $y = 4x^3 - x^2 + 7$ then what is the derivative of y with respect to x?

2. If $y = \tan(z)x^{6z} - \ln(\frac{7x+z}{x^4})$, what is the partial derivative of y with respect to x?

Medium background test

- 20 pts in total; pass: 12 pts
- algorithm: 5 pts
- probability: 5 pts
- linear algebra: 5 pts
- programming: 5 pts

Medium test example

Match the distribution name to its probability density / mass function. Below, $|\boldsymbol{x}| = k$.

(a) Laplace

(b) Multinomial

(c) Poisson

(d) Dirichlet

(e) Gamma

(f) $f(\boldsymbol{x}; \boldsymbol{\Sigma}, \boldsymbol{\mu}) = \frac{1}{\sqrt{(2\pi)^k \boldsymbol{\Sigma}}} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{\mu})\right)$

(g) $f(x; n, \alpha) = \binom{n}{x} \alpha^x (1-\alpha)^{n-x}$ for $x \in \{0, \ldots, n\}$; 0 otherwise

(h) $f(x; b, \mu) = \frac{1}{2b} \exp\left(-\frac{|x-\mu|}{b}\right)$

(i) $f(\boldsymbol{x}; n, \boldsymbol{\alpha}) = \frac{n!}{\Pi_{i=1}^k x_i!} \Pi_{i=1}^k \alpha_i^{x_i}$ for $x_i \in \{0, \ldots, n\}$ and $\sum_{i=1}^k x_i = n$; 0 otherwise

(j) $f(x; \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}$ for $x \in (0, +\infty)$; 0 otherwise

(k) $f(\boldsymbol{x}; \boldsymbol{\alpha}) = \frac{\Gamma(\sum_{i=1}^k \alpha_i)}{\prod_{i=1}^k \Gamma(\alpha_i)} \prod_{i=1}^k x_i^{\alpha_i - 1}$ for $x_i \in (0, 1)$ and $\sum_{i=1}^k x_i = 1$; 0 otherwise

(l) $f(x; \lambda) = \lambda^x \frac{e^{-\lambda}}{x!}$ for all $x \in Z^+$; 0 otherwise

Medium test example

Draw the regions corresponding to vectors $\mathbf{x} \in \mathbb{R}^2$ with the following norms:

1. $||\mathbf{x}||_1 \leq 1$ (Recall that $||\mathbf{x}||_1 = \sum_i |x_i|$)
2. $||\mathbf{x}||_2 \leq 1$ (Recall that $||\mathbf{x}||_2 = \sqrt{\sum_i x_i^2}$)
3. $||\mathbf{x}||_\infty \leq 1$ (Recall that $||\mathbf{x}||_\infty = \max_i |x_i|$)