Final Review

Coverage–comprehensive, slight post-midterm emphasis

B&T ch 1-3,5,9, continuous, limits, hypothesis testing, mle, em.
Intro algorithms (especially DPV chap 6)

everything in slides, hw, daily problems.

Mechanics

closed book, aside from one page of notes (8.5 x 11, both sides, handwritten)

Format–similar to midterm:

T/F, short problems, one or two slightly longer.

Hell's library

what to expect on the final in more detail

...

counting principle (product rule)

permutations

combinations

binomial coefficients

binomial theorem

inclusion/exclusion

pigeon hole principle

sample spaces & events

axioms

complements, Venn diagrams, deMorgan, mutually exclusive events, etc.

equally likely outcomes

chapter 1: conditional probability and independence

conditional probability

chain rule, aka multiplication rule

total probability theorem

bayes rule

independence

discrete random variables

probability mass function (pmf)

expectation, variance of X

expectation of g(X) (i.e., a function of an r.v.)

linearity: expectation of X+Y and aX+b

conditional expectation, law of total expectation

cumulative distribution function (cdf)

- cdf as sum of pmf from -∞

joint and marginal distributions

important examples:

- bernoulli, binomial, poisson, geometric, uniform

know pmf, mean, variance of these

some important (discrete) distributions

Name	PMF	$E[k]$	$E[k^2]$	σ^2
Bernoulli(p)	$f(k) = \begin{cases} 1-p & \text{if } k=0 \\ p & \text{if } k=1 \end{cases}$	p	p	$p(1-p)$
Binomial(p, n)	$f(k) = \binom{n}{k} p^k (1-p)^{n-k}, k = 0, 1, \ldots, N$	np		$np(1-p)$
Poisson(λ)	$f(k) = e^{-\lambda} \frac{\lambda^k}{k!}, k = 0, 1, \ldots$	λ	$\lambda(\lambda+1)$	λ
Geometric(p)	$f(k) = p(1-p)^{k-1}, k = 0, 1, \ldots$	$1/p$	$(2-p)/p^2$	$(1-p)/p^2$
Hypergeometric(n, N, m)	$f(k) = \frac{\binom{m}{k}\binom{N-m}{n-k}}{\binom{N}{n}}, k = 0, 1, \ldots, N$	nm/N	$\frac{nm}{N}\left(\frac{(n-1)(m-1)}{N-1} + 1\right)$	$\frac{nm}{N}\left(\frac{(n-1)(m-1)}{N-1} + 1 - \frac{nm}{N}\right)$

chapter 3: continuous random variables

probability density function (pdf)

cdf as integral of pdf from $-\infty$

expectation and variance

expectation of g(X)

conditional expectation

law of total expectation

important examples

uniform, normal (incl Φ, “standardization”), exponential

know pdf and/or cdf, mean, variance of these

tail bounds

- Markov
- Chebyshev
- Chernoff (lightly)

limit theorems

- weak/strong laws of large numbers
- central limit theorem

likelihood, parameter estimation, MLE (b&t 9.1)

likelihood

- "likelihood" of observed data given a model
- usually just a product of probabilities (independence assumption)
- a function of (unknown?) parameters of the model

parameter estimation

- if you know/assume the *form* of the model (e.g. normal, poisson,...), can you estimate the *parameters* based on observed data
- many ways

maximum likelihood estimators

- one way to do it–choose values of the parameters that maximize likelihood of observed data
- method (usually) – solve
  - "derivative (wrt parameter/s) of (log) likelihood = 0"

EM

- iterative algorithm trying to find MLE in situations that are analytically intractable
- usual framework: there are 0/1 *hidden variables* (e.g. from which component was this datum sampled) & problem much easier if they were known
- E-step: given rough parameter estimates, find expected values of hidden variables
- M-step: given expected values of hidden variables, find (updated) parameter estimates to maximize likelihood
- Algorithm: iterate above alternately until convergence

hypothesis testing (b&t 9.3)

I have data, and 2 (or more) hypotheses about the process generating it. Which hypothesis is (more likely to be) correct?

Again, a very rich literature on this.

One of the many approaches: the "Likelihood Ratio Test"

calculate: $\dfrac{\text{likelihood of data under } \textit{alternate} \text{ hypothesis}}{\text{likelihood of data under } \textit{null} \text{ hypothesis}}$

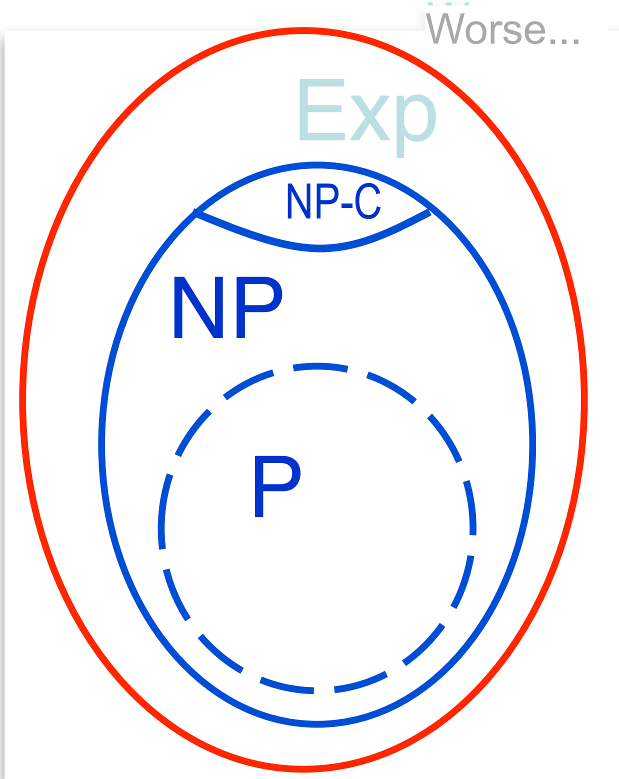
ratio > 1 favors alternate, < 1 favors null, etc.

(false rejection prob, false acceptance prob) noise, uncertainty & variability are pervasive

learning to model it, derive knowledge and compute despite it are critical

Some of the important applications:

- cryptography
- simulation
- statistics via sampling
- machine learning
- systems and queueing theory
- data compression
- error-correcting codes
- randomized algorithms



Big-O – good

P – good

Exp – bad

Verification easy? NP

NP-hard, NP-complete – bad (I bet)

To show NP-complete – reductions

NP-complete = hopeless? – no, but you need to lower your expectations: heuristics, approximations and/or small instances.

Decades of work on many computational problems has led to remarkable improvements in speed/memory/solution quality/etc.

Algorithmic progress dwarfs Moore's law in many cases

Some broadly applicable generic approaches like "greedy", "divide and conquer" and "dynamic programming" have emerged

"Polynomial time": good 1^{st} approx to "feasibly computable"; scales nicely

Unfortunately, for many problems no polynomial time algorithm is known, we are not able to routinely solve large, arbitrary instances, and progress on doing so has been slow and erratic.

Most such problems are NP-hard; many are NP-complete (a subset)

Key characteristics of NP problems:

- searching for a "solution" among exponentially many possibilities
- each solution can be described by polynomially many bits
- a potential solution can be verified in polynomial time

Key technique: reduction

want more?

Stat 390/1	probability & statistics
CSE 421	**algorithms**
CSE 431	computability and complexity
CSE 427/8	computational biology
CSE 440/1	human/computer interaction
CSE 446	**machine learning**
CSE 473	artificial intelligence

and others!

(If you're really motivated and up for a significant challenge, next spring I'm teaching a graduate course on randomized algorithms and probabilistic analysis :)

Thanks and Good Luck!