CSCI 1550/2540 Probabilistic Methods in Computing Science

Instructor: Prof. Upfal (CIT 319) – Eli_Upfal@Brown.edu

Description:

CSCI 1550/2540 is a course on the mathematics that motivates, formulates, and explains many of the great successes of computing, including statistical machine learning, Monte Carlo methods, and modern cryptography. Probability, randomness, and statistics play a key role in these and almost any other modern computer science application. This course introduces the novel mathematical and computation methods that were developed at the interplay of probability and computing. *The course focuses on mathematical models, theorems, and proofs, and leaves implementation and experiments to other courses.*

Syllabus:

- Randomized algorithms and probabilistic analysis:
  - Identity testing, min-cut algorithm, …
- Large deviation bounds:
  - Chernoff and Hoeffding's bounds
  - sub-Gaussian distributions
- The occupancy problem:
  - hashing and random graphs
  - the Poisson approximation
- Martingales
  - Definition and applications
  - Azuma-Hoeffding's concentration bound
- Uniform convergence - the fundamental theorem of machine learning:
  - VC-dimension,
  - covering number and chaining
  - Rademacher complexity.
- Dimension reduction and the Johnson-Lindenstrauss theorem,
- Monte Carlo methods
  - Monte Carlo Markov Chains.
  - Rapidly mixing chains and the coupling method

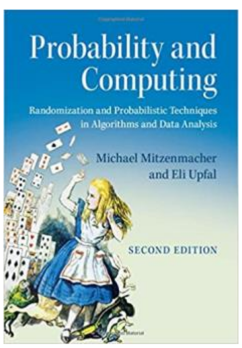
Website: http://cs.brown.edu/courses/cs155/home.html

Check the website and Canvas for instructor's and TA's office hours, locations, and other practical information.

Course staff email: cs155tas@cs.brown.edu

Prerequisites: No formal Prerequisites. Basic understanding of probability theory, mathematical maturity, and interest in rigorous computer science are needed to benefit from this course.

Textbook:
The textbook for the course is Probability and Computing: Randomized Algorithms and Probabilistic Analysis (second edition by Michael Mitzenmacher and Eli Upfal:



.

Course Work and Grading:

Assignments:

Weekly assignments (problem sets) are a major part of the class work. Submitting all assignments is required to take the final exam. Assignments will be graded for correctness and mathematical writing. Assignments grades do not count toward the course grade, but at least half the midterm and final exams questions will be from the weekly assignments.

Assignments will be posted on the course website/Canvas and submitted online. Assignments must be typeset in Latex or written in a VERY clear handwriting. Answers must be concise and mathematically correct. Some of the assignments will be solved in class.

Assignments submitted by their deadline will be graded and returned with corrections. Assignments submitted after their deadline will be counted as submitted but may not be graded.

Final grade:

For 1550 credit: 40% midterm, 60% final**.**

For 1550 credit as a capstone course: 30% midterm, 40% final, 30% project.

For 2540 credit: 30% midterm, 40% final, 30% project.

Assignments collaboration policy:

In preparing the assignments you are allowed to use the textbook, the course slides, the discussions on the course's Ed Discussion, and help from the TA's. Any other source must be disclosed in the submission.

Project (for capstone and/or 2000 level credit):

Submit a 6-8 page research project (with possible appendices) on any application of probabilistic method in CS, and give a 15-minute presentation. A project can be done by 1 or 2 students. Two students who submit a joint project will receive the same grade on their project.

Students who choose to prepare projects will interact with the instructor and the TA's on preparing their project throughout the course.

Time Requirements: Total time spent in and out of class for this course is estimated at ~180 hours. Students will spend 3 hours in class each week (a total of 39 hours). Although specific out-of-class time investments may vary for individual students, a reasonable estimate to support this course's learning outcomes is 140 -150 total out-of class hours, or on average, 10 hours weekly over a 13-week term, in reviewing class material and answering the weekly problem sets, and 10-20 hours working on the take home final.

Accommodations: If you feel you have physical, psychological, or learning disabilities that could affect your performance in the course, we urge you to contact SAS (https://www.brown.edu/campus-life/support/accessibility-services/). We will do whatever we can to support accommodations recommended by SAS.

Please review the Brown University Academic Code:

http://www.brown.edu/academics/college/degree/sites/brown.edu.academics.college.degree/file s/uploads/Academic-Code.pdf

Violations of the Academic Code will lead to strict disciplinary action as outlined in the Code. Misunderstanding of the Code will not be accepted as an excuse for dishonest work.

AI Use Policy for This Course:

Many students are looking for ways to enhance, supplement, or speed up their learning with GenAI tools. However, many uses of GenAI can undermine students' learning and instructors' ability to assess students' learning and give feedback for growth. In this course, the following guidelines are meant to clarify what uses are detrimental to student learning and which may be supportive.

PROHIBITED USES OF GENERATIVE AI
Based on the learning outcomes, assessments and learning activities of this course, the following uses of generative AI are not permitted unless otherwise specified on an assignment:
- Using GenAI to draft explanations, definitions, or comparisons of core theoretical concepts.
- Using GenAI to synthesize theories or establish connections between disparate concepts.

PERMITTED USES OF GENERATIVE AI (WITH GUARDRAILS)
The following uses of generative AI are permitted in this course. Given that GenAI tools can often hallucinate or produce incorrect information, students are responsible for double-checking all AI output.
- Socratic Tutor: Explains complex, abstract concepts; provides alternative analogies and breakdowns. AI must not write explanations or definitions for graded submissions.
- Literature & Discovery Aggregator: Summarizes technical papers or scans global developments to help find fields of interest. AI must not compile the final literature review or evaluate source validity.

- Study Aid Architect: Generates custom study flashcards, vocabulary drills, mnemonic devices, or mock quizzes. AI must not be used to look up or generate answers during actual testing/grading.
- Linguistic Syntax Coach: Identifies grammar errors in written target languages and explains underlying structural rules. AI must not write or translate whole sentences/paragraphs for graded submission.
- Late-Stage Copy-Editor: Suggests changes to syntax, citation formats, and prose to ensure readability of a student's fully completed draft. AI must not generate initial drafts, write paragraph transitions, or introduce new ideas.

Students should reach out to the course instructor(s) if they have questions about this policy. If a GenAI use case not listed here may be supportive for your learning, check with the instructor first.