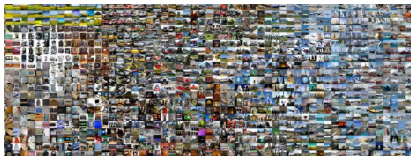
COMPSCI 514: Algorithms for Data Science (Fall 2023)





Time: Tuesday/Thursday 1:00pm-2:15pm

Location: Goessmann Laboratory. Room 20. Lectures will be recorded and posted along with annotated slides under the Schedule tab (schedule.html).

Professor: Cameron Musco (http://people.cs.umass.edu/~cmusco/)

- Email: cmusco at cs dot umass dot edu.
- Office: CS 234
- Office Hours: Tuesday 2:30pm-3:30pm (directly after class) in CS 234.
- How to Contact: If you need to chat or schedule an individual meeting, you can reach out over email, via a Piazza message, or in person, after class or during office hours.

Teaching Assistants:

- Weronika Nguyen
  - Email: thuytrangngu at cs dot umass dot edu
  - Office Hours: Tuesday 11:00am-12:00pm, in CS 207. Wednesday 3:00pm-4:00pm, over Zoom (https://umass-amherst.zoom.us/j/92596997751).
- Ed Almusalamy
  - Email: malmusalamy at umass dot edu
  - Office Hours: Monday 9:00am-10:00am, over Zoom (https://umass-amherst.zoom.us/j/98109929802?pwd=cHFPVTZXSndTTFVtejFEWkMzc0RKdz09). Thursday 4:00pm-5:00pm, in CS 207.

Course Description: With the advent of social networks, ubiquitous sensors, and large-scale computational science, data scientists must deal with data that is massive in size, arrives at blinding speeds, and often must be processed within interactive or quasi-interactive time frames. This course studies the mathematical foundations of big data processing, developing algorithms and learning how to analyze them. We explore methods for sampling, sketching, and distributed processing of large scale databases, graphs, and data streams for purposes of scalable statistical description, querying, pattern mining, and learning. 3 credits.

Prerequisites: The undergraduate prerequisites are COMPSCI 240 or STAT 515 (Probability) and COMPSCI 311 (Algorithms). This is a theoretical course with an emphasis on algorithm design, correctness proofs, and analysis. Aside from a general background in algorithms, a strong mathematical background, particularly in linear algebra and probability is required. If you are a masters student with a limited background in either of these subjects, please email me at the start of the semester to discuss your preparation.

Textbooks: This is no official textbook for this class. We will use some material from:

- Foundations of Data Science (https://www.cs.cornell.edu/jeh/book.pdf), Avrim Blum, John Hopcroft and Ravi Kannan.

- Mining of Massive Datasets (http://www.mmds.org/), Jure Leskovec, Anand Rajaraman and Jeff Ullman.
- Probability and Computing (https://www.cs.purdue.edu/homes/spa/courses/pg17/mu-book.pdf), Michael Mitzenmacher and Eli Upfal.

Readings from these books and other sources will be posted before class under the Schedule tab (schedule.html).

Related Classes: You may also find some helpful reference material in these similar classes taught at other universities:

- The Modern Algorithmic Toolbox (http://web.stanford.edu/class/cs168/index.html), Gregory Valiant at Stanford.
- Sketching Algorithms for Big Data (https://www.sketchingbigdata.org/), Piotr Indyk and Jelani Nelson at MIT/Harvard.
- Algorithmic Techniques for Big Data (https://web.stanford.edu/class/cs369g/), Moses Charikar at Stanford.
- COMPSCI 514 last year (Fall 2021) (https://people.cs.umass.edu/~cmusco/CS514F21/index.html).

Piazza: We will use Piazza for class discussion, questions, and annoucements. Sign up here. (https://piazza.com/umass/fall2023/cs514) We hope for Piazza to be a key interactive component of the class. Thus, we encourage posting and good answering of other students' questions as part of up to 5% extra credit for class participation (see below).

Grade Components:

- Problem Sets (5 total): 40%, split equally between core competency problems and challenge problems, see details below.
- Weekly Quizzes: 10%, weighted equally, lowest score dropped.
- Midterm: 25%.
- Final: 25%.

Grade Scale: The course is graded on a standard scale. I will typically shift this scale down to account for any difficult exams/problems sets. I will never shift it up. I.e., if you obtain a 90% in the course, you will definitely achieve an A-, and potentially an A. If applicable, I will publish the shifted scale at the conclusion of the course. The standard grade scale is: A (100-93), A- (92-90), B+ (89-87), B (86-83), B- (82-80), C+ (79-77), C (76-73), C- (72-70), D+ (69-67), D (66-63), D- (62-60), F (below 60).

Problem Sets: The problem sets will be split into two components: **core competency questions** and **challenge questions**.

- Core competency questions are designed to help you master the key algorithmic and mathematical tools introduced in the course. They will be similar in difficulty to exam questions, and if you are able to solve them, you should be well prepared for the in-class exams.
- You are expected to complete all core competency questions. They will be graded numerically, and count for 20% of the final grade (equally weighted accross problem sets).
- Challenge questions are designed to strengthen your ability to think creatively about algorithmic problems and push beyond known approaches, to develop solutions of your own. They will require significantly more time to digest and solve than core competency questions.
- Each problem set will contain roughly three challenge questions. You can choose which ones you wish to complete., and may attempt as many as you like. In total, the challenge questions will count for 20% of your final grade.
- Each challenge question will be graded on a scale of X, ✓-,✓,✓+. These marks will count towards your grade as follows: each ✓- is worth 1 point, and each ✓ is worth 2 points, each ✓+ is worth 3 points. An

X is worth 0 points. **Full credit is obtained by scoring 15 points total throughout the semester.** Partial credit is assigned accordingly (e.g., if you score 12 points total throughout the semeseter, you will receive an 80% on this component of the course.)

- The rubric for challenge question grading is as follows:
    - ✓+: Submitted work is fully correct and clearly presented. It could be used as a reference solution for the problem. Any errors are minor and easily correctible.
    - ✓: Submitted work demonstrates a full understanding of the problem. There may be some errors, omissions, or unclear steps, but overall, a reader would be able to understand how to solve the problem by looking at the submitted work.
    - ✓-: Submitted work demonstrates partial understanding of the concepts, but contains significant omissions or errors.
    - X: Submitted work doesn't not provide enough information to determine whether there is understanding of the problem.

Problem Set Submissions: Problem sets can be completed in groups of up to three students. If you work in a group, you submit a single problem set together. You may talk to people not in your group about the problem sets at a high level, but may not work through the detailed solutions together, write them up together, etc. We very strongly encourage you to work in a three person group, as it will give an advantage in doing the problem sets. At the beginning of the semester we will make a Piazza post where you can look for teammates.

- While we encourage working in groups, we expect all members of a group to collaborate on and understand their submitted solutions. Some exam problems may closely resemble previous homework problems, and so understanding their solutions will be critical to your success in the course.
- Problem set submissions will be via Gradescope (https://www.gradescope.com/). If working in a group, only one member of each group should submit the problem set, marking the other members in the group as part of the submission in Gradescope.
- The entry code for Gradescope is `WB38GP`.
- **Core competency problems and challenge problems will be submitted separately in Gradescope -- you will see a separate Gradescope assignment for each component.** You do not neccesarily need to submit with the same group for the different questions types or different problem sets.
- No late homework submissions will be accepted unless there are extenuating circumstances, approved by the instructor before the deadline.
- I strongly encourage students to type up problem sets using Latex (https://astrobites.org/2018/01/20/getting-started-with-latex/). A Latex template for problem sets can be downloaded here (./template.tex). While it may seem cumbersome at first, getting proficient in Latex will save you a lot of time in the long run!

Weekly Quizzes: A quiz will be posted in Moodle (https://umass.moonami.com/course/view.php?id=37402) each Thursday after class, due the following Monday at 8pm. These are short quizzes (designed to take ~15 minutes) to check that you are following the material and help me make adjustments if needed. Quizzes will include check-in questions asking for feedback on class pacing and on topics that need clarification, or that you would like to see discussed more. While we will not allow any excused misses of quizzes, **the lowest quiz grade will be dropped,** so that each student can miss one quiz during the course of the semeseter without it affecting their grade.

Exams: The midterm will be held in class on Tuesday October 24th, and the final will held on Thursday December 14th from 10:30am-12:30pm (also in Goessmann Lab, Room 20). Both will be closed notes. We will post extensive review material, past exams, and other practice questions to help you prepare. There is

no option to take the exams remotely. Any makeup exams needed due to illness or other excused absences will be held in person.

Class Participation: Up to 5% extra credit may be awarded for class participation. This may come in many forms, e.g.:

- Asking good clarfiying questions and answering questions during lecture.
- Actively participating in office hours.
- Asking good clarfiying questions and answering other students' or instructor questions on Piazza.
- Posting helpful links on Piazza, e.g., resources that cover class material, research articles related to the topics covered in class, etc.

Course Academic Honestly Policy: If caught violating the problem set or quiz rules, students will receive a 0% on the assignment for the first violation, and fail the class for a second violation. Any cheating on the midterm or final will lead to failing the class. For fairness, we apply these rules universally, without exceptions.

UMass Academic Honesty Statement: Since the integrity of the academic enterprise of any institution of higher education requires honesty in scholarship and research, academic honesty is required of all students at the University of Massachusetts Amherst. Academic dishonesty is prohibited in all programs of the University. Academic dishonesty includes but is not limited to: cheating, fabrication, plagiarism, and facilitating dishonesty. Appropriate sanctions may be imposed on any student who has committed an act of academic dishonesty. Instructors should take reasonable steps to address academic misconduct. Any person who has reason to believe that a student has committed academic dishonesty should bring such information to the attention of the appropriate course instructor as soon as possible. Instances of academic dishonesty not related to a specific course should be brought to the attention of the appropriate department Head or Chair. Since students are expected to be familiar with this policy and the commonly accepted standards of academic integrity, ignorance of such standards is not normally sufficient evidence of lack of intent.

Disability Accommodations: The University of Massachusetts Amherst is committed to providing an equal educational opportunity for all students. If you have a documented physical, psychological, or learning disability on file with Disability Services (DS) (http://www.umass.edu/disability), you may be eligible for reasonable academic accommodations to help you succeed in this course. If you have a documented disability that requires an accommodation, please notify me within the first two weeks of the semester so that we may make appropriate arrangements.
I understand that people have different learning needs, home situations, etc. If something isn't working for you in the class, please reach out and let's try to work it out.

Helpful UMass Resources:

- Academic Honesty Policy (https://www.umass.edu/honesty/) (https://www.umass.edu/honesty/)
- (https://www.umass.edu/honesty/)English as a Second Language (ESL) Program (https://www.umass.edu/esl/)
- Center for Counseling and Psychological Health (https://www.umass.edu/counseling/)

Learning Objectives:

- Students will learn about modern tools for data processing, including random sampling and hashing, low-memory streaming algorithms, linear and non-linear dimensionality reduction, spectral graph theory, and continuous optimization. A major goal is to be familiar at a high level with a breadth of algorithmic tools beyond combinatorial algorithms, which are the main focus of most undergraduate algorithms courses.

- Through problem sets, students will develop the ability to apply and modify these algorithmic tools to tackle new problems, beyond those discussed in class. They will strengthen their ability to think creatively about algorithmic problems and push beyond known approaches, to develop solutions of their own.
- Through assessments that emphasize formal proofs, students will strengthen their ability to formulate problems mathematically and analyze them rigorously.
- Through algorithmic problems, students will practice applying fundamental tools in probability theory and linear algebra, which are broadly applicable in data science and machine learning. These include concentration bounds and methods for decomposing complex random variables, eigendecomposition, orthogonal projection, important matrix identities, and fundamentals of high-dimensional geometry and random matrix theory.