New York University Tandon School of Engineering
Computer Science and Engineering

CS-GY 9223D (CS-UY 3943B)
Algorithmic Machine Learning and Data Science, Fall 2020

Professor Christopher Musco
Wednesdays 11:00am-1:30pm, 370 Jay St + Zoom

CONTACT INFORMATION

Email: cmusco@nyu.edu.

Office: 370 Jay St., #1105 (Floor 11).

Virtual Zoom Office:

Piazza Forum (for questions):

Virtual Office Hours: Weekly time TBA, or by appointment.

Course Webspage:

COURSE DESCRIPTION

This course gives a behind-the-scenes look into the algorithms and computational methods that make machine learning and data science work at large scale. How does a service like Shazam match a sound clip to a library of 10 million songs in under a second? How do scientists find patterns in terabytes of genetic data? How can we efficiently train neural networks with millions of parameters on millions of labeled images? We will address these questions and others by studying advanced algorithmic techniques like randomization, approximation, sketching, continuous optimization, spectral methods, and Fourier methods.

COURSE PREREQUISITES

We require a previous course in machine learning (for example, CS-UY 4563, CS-GY 6923, or ECE-GY 6143), a previous course in algorithm design and analysis (for example, CS-UY 2413, CS-GY 6033, or CS-GY 6043), and a previous course in linear algebra (for example, MA-UY 2034, 3044, or 3054). Experience with probability and random variables is also necessary. In general, *modern algorithm design uses a lot of math!* This is partially what makes it such an interesting subject to study. In this course, we rely most heavily on probability and linear algebra, but we will also learn some approximation theory, Fourier analysis, and high dimensional geometry (a student favorite in the past). Be prepared for theoretically rigorous problem solving, which may require you to refresh topics you covered in prior courses. In particular, you should know:

1. **Probability**: Random variables, discrete and continuous probability distributions, expectation, variance, covariance, independence, correlation, conditional and joint probability, Gaussian random variables, law-of-large-numbers.

2. **Linear Algebra**: Vector inner and outer products, matrix-vector and matrix-matrix multiplication, vector norms (e.g., Euclidean), matrix norms (e.g., Frobenius, operator), triangle inequality, Cauchy-Schwarz, solving systems of linear equations, linear subspaces, linear independence, projection, matrix rank, column/row span, null space, orthogonal matrices, basics of eigenvectors and eigenvalues.

Finally, you do a small amount of programming in this course. You can use whatever language you like (most students use MATLAB or Python) and will need to use basic linear algebra packages in these languages.

COURSE STRUCTURE AND GRADING

You will complete short weekly quizzes, work on problem sets at home, take one in-class midterm, and complete a final project. Details follow:

Class participation (10% of grade). This grade captures how much you contribute to your own learning and that of your peers. Since different students have different styles, there are *many ways* to earn full credit for this part of the course. You can actively participate in class (if you are able to attend synchronously). You can ask good questions, or answer those of your peers, on the class Piazza forum. You can contribute to the reading group. You can attend and actively contribute to office hours.

Weekly quizzes (10% of grade). Every week a short Google Forms quiz will be posted after lecture as a way for both you and I to assess how well you understood the class material. While they are designed to require no more than $\sim$ 15 minutes, you can take as long as you want for the quiz, as long as its completed before the following week's lecture. These quizzes will be graded leniently, and any effort will be rewarded, even if you don't get the answers right.

Bi-weekly written problem sets (40% of grade). These assignments are completed at home and involve the analysis and application of methods learned in class, with occasional programming exercises to further explore lecture content. I expect a lot of your learning to occur while working on these exercises, and investing time on them is the best way to prepare for the midterm exam and final project. Assignments and their due dates will be posted on the course webpage. Late assignments will only be accepted if there are extenuating circumstances and you have obtained prior permission from me.

Take-home midterm exam (15% of grade). Exact structure TBA. You will be able to take the exam asynchronously, or during the first hour of class. You will be allowed a cheat-sheet (a two-sided piece of paper with whatever information you want on it).

Final project (25% of grade). To be completed in teams of 2. This is a completely open ended project that must involve advanced algorithmic methods like those learned in the class. You can either complete an applied project or theoretical project. Details on expectations for both will be released early in the semester.

Optional Reading Group (ungraded). It is an exciting time for research at the intersection of algorithm design and the data sciences. Many of the ideas covered in this course are still the subject of active research. We will hold a virtual reading group to discuss recent papers every week for 2 hours, time TBA. The reading group is a great way to explore topics for the final project, or dip your feet into research.

LECTURES

There is one 2.5 hour class meeting per week which is divided into a *flipped* and *lecture component*. Both components can be attended either in person or on Zoom, and will be recorded. *You are not required to attend either component live, but must review the recordings if you do not.* There will also be an hour of recorded lecture content posted on Thursdays, to be watched asynchronously.

1. **Flipped**: During the **first hour (11am - 12pm)** of lecture, we will discuss solutions to the previous week's quiz, work on simple exercises in small groups, and discuss questions about the problem sets. I will also field general questions and review content as in a typical office hour. *Why are we doing this?* Some people will get more value out of this part of class than others. I am including it for two reasons:
   - Due to the pandemic, I am not able to hold in-person office hours. This more direct and collaborative way of working with students is how I prefer to teach, and has been an important part of my previous courses. Since some students learn better face-to-face, I think we get more value out of having some office-hours-like time in person. Lecture content on the other hand is more easily made virtual.
   - 2.5 hours of lecture is tough to sit through in person, even with a break in the middle. It is even tougher if you are participating online. Since we have many remote students this semester, I think it will be better overall if we avoid such a long block.
2. **Lecture**: After a 15 minute break, the **remaining 1.25 hours (12:15pm-1:30pm)** will be a more typical lecture. You will be able to ask questions in-person or via Zoom. It is fine to use the Zoom chat to discuss with other students, but I request that you use your microphone if you have a question for me.
3. **Asynchronous Lecture**: To make up for my shorter lecture time, I will record an additional ≤ 1 hour of additional lecture content every Thursday, which you should watch on your own time and will be covered on your weekly quiz.

COVID-19 CONSIDERATIONS

In accordance with NYU guidelines, you must wear a mask covering your face and nose whenever you are attending lecture in person. A challenge with our class is that the timeslot spans most student's natural lunch time, and due to the mask guideline, you *are not allowed to eat in class.* I asked my department for clarification on this, and that's the policy. This is the main reason I have included a longer break than last year, although I know 15 minutes is not really enough time for lunch. But hopefully it lets you run down to Metrotech and eat something quick.

COURSE POLICIES

Written problem sets: Turned in via NYU Classes. While not required, I encourage students to prepare written problem sets in LaTeX or Markdown (with math support.) Students will receive 10% extra credit on the Problem Set 1 for preparing it in LaTeX or Markdown.

Homework collaboration policy:

- *Discussion of high-level ideas for solving problems sets or labs is allowed, but all solutions and any code must be written up independently.* This reflects the reality that algorithms and machine learning research are rarely done alone. However, any researcher or practitioner must be able to communicate and work through the details of a solution individually.
- *Students must name any collaborators they discussed problems with at the top of each assignment (list at the top of each problem separately).*
- *Do not write solutions or code in parallel with other students.* If problem ideas are discussed, solutions should be written or implemented at a later time, individually. I take this policy very seriously. Do not paraphrase, change variables, or in any way copy another students solution.

Grade changes: Fair grading is very important to me. If you feel that you were incorrectly docked points on an assignment or exam, please let me know – it is very possible I misunderstood your work. Do not wait until the end of the semester to bring up grading issues! If you notice something off, better to ask ASAP.

Questions about performance: If you are struggling in the class, contact me as soon as possible so that we can discuss what's not working, and how we can work together to ensure you are mastering the material. It is difficult to address questions about performance at the end of the semester, or after final grades are submitted: by Tandon policy no extra-credit or makeup work can be used to improve a students grade once the semester closes. So if you are worried about your grade, seek help from me *early.*

READINGS

There is no textbook to purchase. Many of the topics covered are new, and not sufficiently addressed in existing textbooks. Course material will consist of my written lecture notes, as well as assorted online resources, including papers, notes from other courses, and publicly available surveys. This year I will also be trying to post type-written lecture ntoes. Please refer to the course webpage (https://www.chrismusco.com/9223_2020/) before and after lecture to obtain links.

COURSE SCHEDULE

The following schedule is tentative and subject to change.

THE POWER OF RANDOMNESS

1. 9/2, Concentration of random variables, applications to hashing and load balancing
   9/9, NO CLASS – MONDAY SCHEDULE BECAUSE OF LABOR DAY.
2. 9/16, Sketching and streaming algorithms, models of computation for data processing
3. 9/23, The Johnson-Lindenstrauss lemma, applications to high dimensional data
4. 9/30, Nearest neighbor search, locality sensitive hashing

OPTIMIZATION

5. 10/7, Convexity in machine learning, vanilla, stochastic, and online gradient
6. 10/14, Conditioning, acceleration, coordinate descent, quasi-Newton methods
7. 10/21, Learning from experts, multiplicative weights
8. 10/28, Constrained Optimization, linear programming, relaxation

SPECTRAL METHODS AND LINEAR ALGEBRA

9. 11/4, Singular value decomposition, Krylov methods.
10. 11/11, Spectral graph theory, clustering, stochastic block models.
11. 11/18, Randomized linear algebra, sketching for linear regression, -nets arguments.
12. 11/25, High Dimensional Geometry

FOURIER METHODS

13. 12/2, Compressed sensing, the restricted isometry property, basis pursuit
14. 12/9, Kernel methods in machine learning, random Fourier features

COURSE OBJECTIVES

1. Students will build experience with the most common algorithmic tools used in machine learning, including optimization, relaxation, spectral methods, Fourier methods, and Monte Carlo algorithms. They will also strengthen their understanding of the mathematical concepts behind these tools.

2. Through problem sets and in-class assignments, students will practice applying, combining, and modifying the methods learned in lectures to specific problems in machine learning and data analysis. The goal is to prepare students to use these tools in industrial or academic positions where they are developing computationally efficient machine learning methods.

3. Students will learn how theoretical questions (involving asymptotic complexity, communication complexity, convergence rate, approximation quality, etc.) can guide the design of new algorithmic methods in machine learning. We will practice framing and rigorously answering such questions.

4. The course will prepare students to contribute to research projects on computational methods for machine learning and data science.

5. Through assigned papers and discussions, students will improve their ability to read, summarize, and extract value from contemporary research papers in machine learning and data mining conferences like NeurIPS, ICML, KDD, ICLR and AAAI.

ADDITIONAL INFORMATION

MOSES CENTER STATEMENT OF DISABILITY.

If you are student with a disability who is requesting accommodations, please contact New York University's Moses Center for Students with Disabilities (CSD) at 212-998-4980 or mosescsd@nyu.edu. You must be registered with CSD to receive accommodations. Information about the Moses Center can be found here.

NYU SCHOOL OF ENGINEERING POLICIES AND PROCEDURES ON ACADEMIC MISCONDUCT.

The complete Student Code of Conduct can be found here.

A. Introduction: The School of Engineering encourages academic excellence in an environment that promotes honesty, integrity, and fairness, and students at the School of Engineering are expected to exhibit those qualities in their academic work. It is through the process of submitting their own work and receiving honest feedback on that work that students may progress academically. Any act of academic dishonesty is seen as an attack upon the School and will not be tolerated. Furthermore, those who breach the Schools rules on academic integrity will be sanctioned under this Policy. Students are responsible for familiarizing themselves with the Schools Policy on Academic Misconduct.

B. Definition: Academic dishonesty may include misrepresentation, deception, dishonesty, or any act of falsification committed by a student to influence a grade or other academic evaluation. Academic dishonesty also includes intentionally damaging the academic work of others or assisting other students in acts of dishonesty. Common examples of academically dishonest behavior include, but are not limited to, the following:

    1. Cheating: intentionally using or attempting to use unauthorized notes, books, electronic media, or electronic communications in an exam; talking with fellow students or looking at another persons work during an exam; submitting work prepared in advance for an in-class examination; having someone take an exam for you or taking an exam for someone else; violating other rules governing the administration of examinations.
    2. Fabrication: including but not limited to, falsifying experimental data and/or citations.
    3. Plagiarism: intentionally or knowingly representing the words or ideas of another as ones own in any academic exercise; failure to attribute direct quotations, paraphrases, or borrowed facts or information.
    4. Unauthorized collaboration: working together on work meant to be done individually.
    5. Duplicating work: presenting for grading the same work for more than one project or in more than one class, unless express and prior permission has been received from the course instructor(s) or research adviser involved.
    6. Forgery: altering any academic document, including, but not limited to, academic records, admissions materials, or medical excuses.

NYU SCHOOL OF ENGINEERING POLICIES AND PROCEDURES ON EXCUSED ABSENCES.

The complete policy can be found here.

A. Introduction: An absence can be excused if you have missed no more than 10 days of school. If an illness or special circumstance has caused you to miss more than two weeks of school, please refer to the section labeled Medical Leave of Absence.

B. Students may request special accommodations for an absence to be excused in the following cases:

    1. Medical reasons
    2. Death in immediate family
    3. Personal qualified emergencies (documentation must be provided)
    4. Religious Expression or Practice

Deanna Rayment, deanna.rayment@nyu.edu, is the Coordinator of Student Advocacy, Compliance and Student Affairs and handles excused absences. She is located in 5 MTC, LC240C and can assist you should it become necessary.

NYU SCHOOL OF ENGINEERING ACADEMIC CALENDAR

The full calendar can be found here. Please pay attention to notable dates such as Add/Drop, Withdrawal, etc. For confirmation of dates or further information, please contact Susana Garcia: sgarcia@nyu.edu.