New York University Tandon School of Engineering
Computer Science and Engineering

CS-GY 6923
Machine Learning, Fall 2025

Professor Pavel Izmailov
Friday 2:00-4:30pm, 2 Metrotech, Room 315
Virtual lectures TBD.

PROFESSOR CONTACT

Email: pi390@nyu.edu.

Office: TBD.

Office Hours: TBD.

COURSE ASSISTANTS

Majid Daliri: daliri.majid@nyu.edu

Harsha Mupparaju: sm12754@nyu.edu

Shubham Rastogi: sr7421@nyu.edu

Office Hours: See course webpage.

ESSENTIAL LINKS

Course Web page: https://izmailovpavel.github.io/ny_fall25_ml_grad.

Brightspace (for Zoom links): https://brightspace.nyu.edu/d2l/home/484236

Ed Discussion: TBD

COURSE DESCRIPTION

This course provides a graduate-level introduction to machine learning through a mixture of hands-on exercises and theoretical foundations. We will cover fundamentals of machine learning: regression, classification, linear models, neural networks, numerical optimization methods (gradient descent, backpropagation), unsupervised learning, and a number of other topics. We will also cover basics of language modeling and understand how systems such as ChatGPT operate. The course includes hands-on exercises with machine learning methods and covers a broad range of applications.

COURSE PREREQUISITES

For this class you will need a solid understanding of linear algebra, calculus and basic probability theory. You will also need to be able to code in Python. Here is a sample of specific topics you should know:

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

2. **Linear Algebra**: Matrices and vectors, 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 independence, matrix rank, column/row span, null space, orthogonal matrices, basics of eigenvectors, eigenvalues, eigendecomposition, singular vectors.

3. **Calculus**: Derivatives and partial derivatives, chain rule, gradients and directional derivatives, critical points, local minima and maxima, convex and concave functions, Taylor series expansion, basic optimization (unconstrained optimization, Lagrange multipliers), multivariable functions, limits and continuity.

4. **Programming**: Basic Python syntax and data types, control structures (loops, conditionals), functions and modules, data structures (lists, dictionaries, tuples, sets), NumPy for numerical computing and array operations, understanding of vectorized operations.

If you are not familiar with these topics, you will need to study them on your own, as they will not be taught in the class.

COURSE STRUCTURE AND GRADING

There is one class meetings per week, involving lecture and demonstrations. You will also work on assignments at home and take two in-class tests. 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. You can ask good questions, or answer those of your peers, on the class Ed forum. You can attend and actively contribute to office hours.

Weekly programming labs (20% of grade) and written problem sets (20% of grade). These assignments are completed at home and reinforce the material discussed in class. 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 exams. Assignments and their due dates will be posted on the course webpage. Over the course of the semester, you may obtain a one day extension for any three assignments you want, no questions asked. Addition extensions are only possible under extenuating circumstances and if you have obtained prior permission from the instructor.

In-class midterm (25% of grade) and final exam (25% of grade). For both exams you will be allowed a cheat-sheet (a two-sided piece of paper with whatever information you want on it).

COURSE POLICIES

Programming labs: Turned in via Gradescope as completed and evaluated Jupyter notebooks.

Written problem sets: Turned in via Gradescope. 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 machine learning research and implementation are rarely done alone. However, any machine learning practitioner must be able to communicate and work through the details of a solution individually.
- *Do not write 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.
- We have a zero tolerance policy for copied code or solutions: any students with duplicate or very similar material will receive a zero on the offending assignment. This means both the student who copied material and the student they copied from will receive a zero. My advice is to never share code or solutions with other students.

ChatGPT, Claude Code, and Other AI Tools: You can feel free to AI tools as you wish for homeworks. However, make sure the tools do not impede your own learning. Usually it is better to struggle through a difficultly homework problem on your own or with a friend, as doing so will help ensure you actually learn the concept being focused on (and, better prepare you for the exam).

Midterm: The midterm exam will be held during class on **October 24th** and the final on **December 19th**. Make sure you are available on those days. If there is any issue, let me know ASAP.

Grade changes: Fair grading is very important to me. If you feel that you were incorrectly docked points on an assignment or exam, please contact the TAs on Ed, who will escalate to me as necessary. Do not wait until the end of the semester! 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 and the TAs *early*.

READINGS

There is no textbook to purchase. Primary course material will consist of lecture slides and my handwritten notes from class (I use a tablet as a digital whiteboard and make those notes available.) To supplement that information, optional readings will be posted from each lecture. Many will come from:

- *Probabilistic Machine Learning: An Introduction*, by Kevin Murphy. This comprehensive introduction to probabilistic approaches in machine learning is available free for download here.
- *Pattern Recognition and Machine Learning*, by Christopher Bishop. A foundational text covering Bayesian methods and pattern recognition techniques. A free PDF version is available here.
- *Deep Learning*, by Ian Goodfellow, Yoshua Bengio, and Aaron Courville. The definitive textbook on deep learning methods and theory. The complete book is freely available online.
- *An Introduction to Statistical Learning*, by James, Witten, Hastie, and Tibshirani. This book is available free for download from the NYU library.
- *Elements of Statistical Learning*, by Hastie, Tibshirani, and Friedman for some more advanced topics. A free version of the book is available here.

COURSE SCHEDULE

An updated schedule will be maintained at `https://izmailovpavel.github.io/ny_fall25_ml_grad`.

COURSE OBJECTIVES

1. Students will learn how to view and formulate real world problems in the language of machine learning. Categories of problems include those involving prediction, classification, pattern recognition, and decision making.
2. Through in-class demonstrations and at-home programming labs, students will gain experience applying the most popular and most successful machine learning algorithms to example problems. The goal is to prepare students to use these tools in industrial or academic positions.
3. In addition to experimental exploration, students will learn how theoretical analysis can help explain the performance of machine learning algorithms, and ultimately guide how they are used in practice, or lead to the design of entirely new methods.
4. Students will build experience with the most important mathematical tools used in machine learning, including probability, statistics, and linear algebra. This experience will prepare them for more advanced coursework or research in the subject.
5. A major goal is to prepare students to read and understand contemporary research in machine learning, including papers from NeurIPS, ICML, ICLR, AAAI, JMLR, and other major machine learning venues. Since machine learning is a rapidly evolving field, many of its most powerful tools today may no longer be relevant in 15 years. The goal is to provide students with a theoretical foundation that will allow them to keep up with changes in the field.

At the end of this course, every student should feel confident holding a conversation about machine learning with anyone, no matter their experience in the field. This would include a potential employer, a professor or graduate student working in the field, or a stranger on the subway.

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.

As an NYU community member, it is your responsibility to know your rights and responsibilities around academic misconduct. Please click here to review the NYU Tandon Student Code of Conduct. Any questions about how this policy relates to this class, please ask your professor.

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 School's rules on academic integrity will be sanctioned under this Policy. Students are responsible for familiarizing themselves with the School's 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 person's 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 one's 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, advocacy.tandonstudentlife@nyu.edu, is the Coordinator of Student Advocacy, Compliance and Student Affairs and handles excused absences. The Office of Student Advocacy is located in 5 MTC, LC240C and they can assist you should it become necessary or if you have any questions.

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.

Acknowledgements

This class and this syllabus document are based on the previous iterations of CS-GY 6923 taught by professor Christopher Musco.