ORF 363/COS 323, Fall 2025

Computing and Optimization for the Physical and Social Sciences

http://aaa.princeton.edu/orf363

Course description

An introduction to several fundamental and practically-relevant areas of numerical computing with an emphasis on the role of modern optimization. Topics include convex optimization, computational linear algebra, linear and semidefinite programming, optimization for statistical regression and classification, computational complexity theory, and techniques for dealing with uncertainty and intractability in optimization problems. Extensive hands-on experience with high-level optimization software. Applications drawn from operations research, statistics and machine learning, finance, economics, and engineering.

Course website

- The course will be on Canvas and we will use Ed Discussion.
- Lecture notes and problem sets will also be posted here: http://aaa.princeton.edu/orf363

Lecture time and location

Tue, Thu 1:20 pm-2:40 pm EST, Friend 006

Distribution area

Quantitative reasoning

Certificate programs/minors

The course counts towards the certificate programs/minors in

- Optimization and Quantitative Decision Science
- Applications of Computing
- Statistics and Machine Learning
- Robotics and intelligent systems

Instructor

Amir Ali Ahmadi, Professor at ORFE
Web: http://aaa.princeton.edu/
Email: aaa@p...

Graduate student TAs/UCAs

- Ben Budway (TA)
- bb2584@p...

- Yixuan Hua (TA)
- yh7422@p...

- Albert Shi (UCA)
- as6369@p...

- Yukai Tang (TA)
- yt3846@p...

Office hours

Mon	Tue	Wed	Thu	Fri
4:30-6:30pm Sherrerd 122 (Ben)	9-11am Sherrerd 122 (Albert)	4:30-6:30pm Sherrerd 125 (AAA)	9-11am Sherrerd 122 (Albert)	9-11am Sherrerd 122 (Yixuan)
7-9pm Sherrerd 122 (Yixuan)	7-9pm Sherrerd 122 (Yukai)	7-9pm Sherrerd 122 (Ben)	7-9pm Sherrerd 122 (Yukai)	

Ed Discussion

Please sign up via Canvas. Please take advantage of Ed Discussion to initiate mathematical discussions with your teaching staff and classmates.

Prerequisites

- Multivariable Calculus: MAT201 or MAT203
- Linear Algebra: MAT202 or MAT204
- Familiarity with MATLAB
  (You are free to use other software; e.g., Python)

Textbooks

Recommended as reference (instructor lecture notes are the only required reference):

- *An Introduction to Optimization,* K. P. Chong & S. H. Zak, *4th ed.*
- *Algorithms,* S. Dasgupta, C. Papadimitriou, and U. Vazirani.
- *Introduction to Applied Linear Algebra – Vectors, Matrices, and Least Squares*
  https://web.stanford.edu/~boyd/vmls/
- *Linear Programming: Foundations and Extensions,* R.J. Vanderbei.
  Available for free download to Princeton students:
  http://link.springer.com/book/10.1007/978-1-4614-7630-6
- *Convex Optimization,* S.Boyd, and L. Vandenberghe.
  Available for free download:
  http://web.stanford.edu/~boyd/cvxbook/

Software

Students will use the MATLAB-based optimization software CVX (http://cvxr.com/cvx/) to solve a variety of real-world (but simplified) problems having to do with optimization and computing in operations research, machine learning, finance, economics, and engineering. Students are free to use other programming languages (e.g., CVXPY in Python) if they feel more comfortable, but our coding questions have been tested more extensively in MATLAB.

Course grade

- 50% homework (around 8 problem sets; will drop the lowest score); almost all problem sets will involve a computational component.
- 20% midterm; in class, during regular lecture time, closed notes (except for one page of notes)
- 30% final; 48-hour take-home, open notes from class but no internet, involves a computational component

Homework

Homework will be due before lecture, at 1:30 pm EST. Homework must always be submitted on Gradescope as a *single PDF file* which includes your code. Both typed and hand-written solutions are accepted. Unless there is an *extremely valid* reason, requests for extension on homework will not be accepted. To help stick with this policy, we drop your lowest homework score.

Midterm

Midterm will be in-class, for the entire length of a regular lecture. Midterm (date to be announced) is closed-book and closed-notes. However, you can take a single sheet of standard-sized paper with you (double-sided) with anything you want written on it. No electronic devices are allowed; no calculators, no phones, etc. There will be no computer-based problems on the midterm.

Final

There will be a take-home final exam. This will be a cumulative exam but with more focus on the second half of the course. You cannot discuss the exam with anyone other than the professor and the TAs. The date of the exam is to be announced later in the semester. The final (just like the homework) will have both a mathematical component and a computational component.

Collaboration policy

Homework: You are allowed and in fact encouraged to collaborate on the homework. You must turn in your individual assignment.

Midterm and final exam: No collaboration allowed.

Policy on use of artificial intelligence-based tools

You cannot use the internet during the final exam (except for debugging code), and hence the use of AI-based tools is also not allowed. The midterm exam is in-class, with no internet access allowed, and therefore the use of AI tools is similarly forbidden.

For the problem sets, the use of AI-based tools such as ChatGPT is allowed—**subject to the following caveats and rules—but strongly discouraged**. If you use an AI tool, you must specify this on your solution sheet—specifically next to each part or subpart of the question where the tool was used. Failure to do so is a violation of University policy. If you use an AI tool and your argument is not completely correct, you will receive **no** partial credit. (By contrast, your own solutions will receive partial credit if they are partially correct.)

We have tried entering examples of homework questions into such tools and have repeatedly received incorrect answers. These include proofs that seem correct but are flawed, and counterexamples that are invalid. Even with back-and-forth instructions, tools like ChatGPT can continue to make mathematical and logical mistakes. We are concerned that this can confuse students, as the mistakes are sometimes subtle. Again, **you are responsible for any mistakes made by such tools** and will receive no partial credit. You may not simply enter your question into an AI tool and then ask the TAs or the instructor to verify its response.

You are allowed to use AI tools to help with coding, as long as you (i) declare it, (ii) check its correctness, and (iii) aim to learn from it. Instead of asking the tool to write code for you, you should write your own code and use the tool for debugging—otherwise, you will not learn effectively. You may also use an AI tool to get examples of how certain syntax works in MATLAB, Python, etc., and then apply what you learn to the problem set.

We reserve the right to change this policy as we learn more about how these tools work. Most importantly, when deciding how to approach a homework problem, make sure your **primary objective is to learn**.

Tentative list of lectures and problem sets:

- Lec1: Let's play two games! (Optimization, P and NP.)
- Lec2: What you should remember from linear algebra and multivariate calculus.
- HW1: Brush up on linear algebra, multivariate calculus, and MATLAB.
- Lec3: Unconstrained optimization, least squares, optimality conditions.
- Lec4: Convex optimization I.
- HW2: Image compression and SVD, optimality conditions, convex sets.
- Lec5: Convex optimization II.
- Lec6: Applications in statistics and machine learning: LASSO+SVMs.
- HW3: Convex analysis and convex optimization.
- Lec7: Root finding and line search: bisection, Newton, and the secant method.
- Lec8: Gradient descent methods, analysis of steepest descent, rates of convergence.
- HW4: Support vector machines.
- Lec9: Quadratic convergence of Newton's method, nonlinear least squares and Gauss-Newton.
- Lec10: Conjuagate direction methods, solving linear systems, Leontief economy.
- HW5: New gym and movie theater for Princeton + Newton fractals.
- Lec11: Linear programming: applications, geometry, and the simplex algorithm.
- Lec12: Duality and robust linear programming.
- HW6: Leontief economy + conjuagate gradients + radiation treatment planning.
- Lec13: Semidefinite programming, SDP relaxations for nonconvex optimization.
- Lec14: A working knowledge of computational complexity for an optimizer.
- HW7: Optimal control + linear programming.

- Lec15: William Pierson Field Lecture by Sanjeeb Dash: Optimization at IBM Research.
- Lec16: Limits of computation, course recap.
- HW8: End-of-semester party at AAA's + Doodle and scheduling + SDP + NP-completeness.

Honor code

We strictly adhere to Princeton University's Undergraduate Honor System.