CS 468 / CS 568 / BME 568:
Computational Modeling for Bioengineering
Spring 2024

Syllabus

Matthew R. Lakin

January 4, 2024

Course Information

Semester: Spring 2024.
Course title: Computational Modeling for Bioengineering.
Course numbers: CS468 / CS568 / BME568.
Course credits: 3.
Class meeting time: Tuesdays and Thursdays, 3:30pm–4:45pm.
Class meeting location: Mechanical Engineering room 210.

Instructor

Name: Matthew Lakin.
Email: `mlakin@cs.unm.edu`
Office hours time: Mondays 2–4pm.
Office hours location: Farris Engineering Center room 3240.

Course description

This course will cover the use of computational modeling as a tool for analyzing scientific data, making predictions based on that analysis, and rational design of engineered systems. The course and course examples will use the Python programming language. As such, the early part of the course will offer a brief overview of Python programming in general, and of the libraries used for the examples in the course.

The course will cover case studies of modeling applied to a range of scientific problems from the biological and chemical sciences and bioengineering. There will be a particular focus on modeling the kinetics

of biomolecular interactions in various situations and at various levels of abstraction, including analysis of gene expression levels in genetic regulatory networks or other biomolecular systems.

The course is intended for advanced undergraduate and graduate students in Computer Science and Biomedical Engineering. Prior experience with Python is helpful but not required. No knowledge of the specific scientific concepts used as examples will be assumed, but basic knowledge of scientific and mathematical fundamentals will be helpful.

The course will consist of lectures, occasional written, modeling, and programming assignments, and examinations.

Student learning outcomes

- Students can define the basic principles and concepts in scientific modeling.
- Students can select and apply appropriate tools to carry out scientific modeling tasks.
- Students can write and use Python scripts for scientific data analysis.
- Students can describe and apply basic principles of the case studies covered.

Textbook and course materials

The Python programming component of the course will broadly follow the textbook:

- "A Primer on Scientific Programming with Python (5th edition)" by Hans Petter Langtagen (ISBN: 978-3-662-49886-6).

This textbook can be downloaded for free in PDF format via the UNM network, from:

- https://link.springer.com/book/10.1007%2F978-3-662-49887-3

All lecture materials will be made available via UNM Canvas.

The UNM email system will be used for administrative announcements.

Course requirements

You are expected to attend class regularly, read any assigned reading before class, and participate in class discussions.

Short written and/or programming homework assignments will be given to consolidate lecture material. In the early part of the course, these tasks will be drawn from the general domains of mathematics, science, and engineering, to practice programming skills; in the later part of the course the tasks will be drawn from the case studies covered in class.

There will be two in-class, closed book midterm exams.

Instead of a final exam there will be a final project, which will draw on the tools and techniques covered in the course to model and analyze the behavior of biological and chemical systems.

Grading

You are expected to attend class regularly, read any assigned reading before class, and participate in class discussion. The grade will be determined as follows:

- Homeworks: 30% total
- Exams: 40% total (20% for each of two midterm exams)
- Final project: 20%
- Quizzes: 10%

The purpose of exams is to test one's knowledge of the material at a specific moment in time. As such, unofficial requests for accommodations or delays will not be granted except in circumstances that make in-person attendance on the day of the exam unavoidable (e.g., prearranged conference travel). In such cases, students will be required to sit a catchup exam as soon as possible after the main exam date. All exams must be taken in person and not remotely.

No requests regarding grading, such as grade mode changes, will be considered after the final class period. There will also be no extra credit assignments or "do-overs" for homeworks, exams, or quizzes. I cannot predict or discuss possible grades until all assessments have been graded.

Course schedule

Broadly, the course will cover basic Python programming, theoretical preliminaries on chemical kinetics and chemical reaction networks, and case studies of computational modeling from the primary literature. Examples of case studies from the primary literature include:

- A. A. K. Nielsen, B. S. Der, J. Shin, P. Vaidyanathan, V. Paralanov, E. A. Strychalski, D. Ross, D. Densmore, and C. A. Voigt, "Genetic circuit design automation," Science, vol. 352, no. 6281, aac7341, 2016. DOI: 10.1126/science.aac7341
- C. Y. Hu, M. K. Takahashi, Y. Zhang, and J. B. Lucks, "Engineering a functional small RNA negative autoregulation network with model-guided design," ACS Synthetic Biology, vol. 7, no. 6, pp. 1507–1518, 2018. DOI: 10.1021/acssynbio.7b00440
- L. Qian and E. Winfree, "Scaling up digital circuit computation with DNA strand displacement cascades," Science, vol. 332, pp. 1196–1201, 2011. DOI: 10.1126/science.1200520
- Y.-J. Chen, N. Dalchau, N. Srinivas, A. Phillips, L. Cardelli, D. Soloveichik, and G. Seelig, "Programmable chemical controllers made from DNA," Nature Nanotechnology, vol. 8, pp. 755–762, 2013. DOI: 10.1038/nnano.2013.189

Credit-hour statement

This is a three credit-hour course. Class meets for two 75-minute sessions of direct instruction for fifteen weeks during the Fall 2021 semester. Students are expected to complete a minimum of six hours of out-of-class work (or homework, study, assignment completion, and class preparation) each week.

Academic integrity statement

Each student is expected to maintain the highest standards of honesty and integrity in academic and professional matters. The University reserves the right to take disciplinary action, up to and including dismissal, against any student who is found guilty of academic dishonesty or otherwise fails to meet the standards. Any student judged to have engaged in academic dishonesty in course work may receive a reduced or failing grade for the work in question and/or for the course.

Academic dishonesty includes, but is not limited to, dishonesty in quizzes, tests, or assignments; claiming credit for work not done or done by others; hindering the academic work of other students; misrepresenting academic or professional qualifications within or without the University; and nondisclosure or misrepresentation in filling out applications or other University records.

All students will be required to sign and submit a warning regarding issues of academic integrity and possible sanctions prior to any submissions being graded.

Accommodation statement

UNM is committed to providing equitable access to learning opportunities for students with documented disabilities. As your instructor, it is my objective to facilitate an inclusive classroom setting, in which students have full access and opportunity to participate. To engage in a confidential conversation about the process for requesting reasonable accommodations for this class and/or program, please contact Accessibility Resource Center at arcsrvs@unm.edu or by phone at 505-277-3506.

Title IX statement

To meet obligations under Title IX, UNM faculty, Teaching Assistants, and Graduate Assistants are considered "responsible employees". This designation requires that any report of gender discrimination, which includes sexual harassment, sexual misconduct and sexual violence made to a faculty member, TA, or GA, must be reported to the Title IX Coordinator at the Office of Compliance, Ethics, and Equal Opportunity. For more information on the campus policy regarding sexual misconduct, see: https://policy.unm.edu/university-policies/2000/2740.html