What this course is about

From climate change and COVID to biological evolution and exoplanets, computation is an essential element of modern science. It allows us to find insights in a sea of data, ask principled questions about the future, and perform experiments without a laboratory. In this class, weâ€™ll learn the practice of computer programming, and quantitative questioning, through both data-driven and model-oriented case studies focused on the earth, the universe, and living systems. To explore these topics weâ€™ll use the Python programming language to create dynamic computational notebooks that interweave code, images, comments, questions, and analysis. This introductory class will illuminate how computation is changing the nature of science, and provide undergraduate students with the experience and tools to engage in scientific research.

Getting started

- The first class meeting is on Tuesday January 23rd at 12:00-1:15 in room G115 of the Maxwell Dworkin Laboratory on Oxford St.

How this course works

- This is an introductory programming course.
  - All are welcome!
  - We'll learn programming through the applications to modern scientific problems.
  - No prior programming experience is required.
  - No prior calculus experience is required.
  - This course only uses the widely used Python programming language.
  - All assignments will be hosted in the cloud through Google Colabs so you don't have to install anything on your own device.
  - We will learn how to write our own programs and integrate powerful tools written by others.
- Course meetings
  - Lectures will be on Tuesdays and Thursdays from 12:00-1:15 in room G115 of the Maxwell Dworkin meeting on Oxford St. Live class meetings are an essential and required part of this course and we hope to make them opportunities for questions, discussions, and interacting with your fellow students and the teaching staff.
  - TF/TA led sections will be hosted weekly at dates and times listed below. The purpose of the sections is to review material from lectures and provide additional assistance with assignments.
- Course work
  - 1 writing assignment
  - 7 programming assignments
  - 1 final project
- This course is approved to satisfy Harvard's Quantitative Reasoning with Data (QRD) requirement.

Teaching staff

- Professor: **Brendan Meade** (meade@fas.harvard.edu)
  - Office hours: Mondays 9:00am-10:00am (via Zoom appointment)
- Professor: **Jeremy Bloxham** (jeremy_bloxham@harvard.edu)
  - Office hours: TBD
- Teaching Fellow: **Sarah Hancock** (sarahhancock@g.harvard.edu)
  - Office Hours: Tuesdays 8:00PM-9:00PM on Zoom
- Teaching Fellow: **Sammy Piascik** (spiascik@fas.harvard.edu)
  - Office Hours: Wednesdays 9:00am-10:00am (https://harvard.zoom.us/j/99913903057?pwd=SCtlRFAvVzByd0tVbUVNaGVYVVFNZz09)
  - Link to Lab Section Colabs

Labs and locations

There will be weekly laboratory sessions throughout the semester. Attendance at laboratory sessions is required.

Lab times and locations:

Tuesday, 6:45PM-8:45PM, Room: MD123, TF: Sammy

Wednesday, 9AM-11AM, Room: MD123, TF: Sarah

Assignments and Grading

- Class attendance (10%). Come to class, meet your classmates and be a part of the discussion!
- Weekly coding exercises and writing opportunities (65%). There are weekly assignments in the form of computational exercises and writing opportunities. Most of the assignments are computational exercises where students can describe exercise goals and computational thinking, write code, and plot results. These notebooks should be heavily described rather than a simple listing of code. A goal is that every student-created notebook should be explicit enough and well enough documented so that another student could read through it, understand what was being done, and describe it accurately. These will all be carried out using Google Colabs, which are available online to all students through their g.harvard.edu accounts. Weekly TF-led sessions will focus on helping students develop their assignments and review lecture material. Late homework policy: Homeworks handed in after the submission deadline and up to 1 day late will see a score reduction of 33%, between 1 and 2 days late will result in a 67% reduction, and 3 or more days late will result in a 100% reduction (no credit).
- Final group projects (25%). The final assessment in the class is a group project. The topic for the group project will be based on a 1/2-page proposal that must be approved by the teaching staff, and we will work with the students to help define a project that is within scope and relevant. There are two parts to the final project. The first is a 8-10 minute presentation summarizing the problem, the data and methods used to study the problem, and what was learned from studying the problem. The second part of the final project is supporting computational notebooks and code.

Submitting homework

- Homework will be assigned in class and submitted through the course website (right here). For most assignments, you will be submitting a Colab notebook individually. Specific instructions that detail how to submit homeworks can be found in these Google Slides.

On the use of machine learning tools for class materials

- The use of AI tools (including but not limited to ChatGPT, Llama, Claude, CoPilot) for writing, coding, and presentations is not prohibited but is discouraged.

  We do not prohibit their use for the following reasons: 1) The use of these tools may make you any better at programming, and 2) We respect the decisions that students make about how much they want to learn as individuals, and 3) There is the possibility that in some way the use of the tools does make you smarter. However, we discourage their use because programming is a skill that requires practice and that skill cannot be honed simply by using AI tools.

  We also note that while these AI tools may be good at contributing to the completion of problem sets, in the real world their use for science and programming applications has so far proved limited. Instead, it is the human programmers of today who are creating the AI and other computational tools of tomorrow. Further, the hype about AI displacing people from technical jobs is now decaying quickly.

If you choose to use AI assistance to do any of your homework, you must note this (in writing) on the homework at the time of submission.

Class meetings

Class 1 - Foundation: Introduction [Slides]

- Programming as science, craft, art, and the 3rd way of science.
- Asking â€œWhat if?â€
- Semester outline.

Class 2 - Foundation: Python and computational notebooks [Slides] [Colab]

- Why python and computational notebooks?
- Introduction to Colab in the browser.
- Elements of a program.

Class 3 - Foundation: Variables and state [Colab]

- Why do we want to store things?
- Imperative programming.
- Fahrenheit to Celsius conversion exercise in class

Class 4 - Foundation: Flow control in python [Colab]

- Iteration with a â€œforâ€ loop.
- Logic and â€œifâ€ statements.
- Event detection example.

Class 5 - Foundation: Functions [Colab]

- Functions as a recipe.
- Syntax and where to define them.
- The difference between arguments and returned values.
- Writing good docstrings.

Class 6 - Foundation: Pandas [Colab]

- Introduction to pandas and dataframes.
- Read and sort climate data.
- Selecting subsets of data.

Class 7 - Foundation: Debugging and reading documentation [Colab]

- Reading python error messages
- Working backwards
- How to read and use documentation

Class 8 - Foundation: Learning from online sources

- StackOverflow
- Github Copilot

Class 9 - Foundation: Making plots with Matplotlib [Colab]

- Why matplotlib?
- Building the temperature anomaly time series plot.

Class 10 - Foundation: Mathematics with Numpy [Colab]

- Probability and rolling dice.

Class 11 - Climate change: observations I [Colab]

- Explaining where ice core data come from.
- Reading and plotting ice core data.

Class 12 - Climate change: Observations II [Colab]

- Comparing ice core data and contemporary data

Class 13 - Climate change: Temperature trends [Colab]

- Finding and fitting linear trends

Class 14 - Climate change: Extrapolating CO2 trends [Colab]

- Autoregression

Class 15 - COVID: Human population growth [Colab]

- Global population growth over the last 2000 years observations and model.
- Integrate numerically with Eulerâ€™s method.
- Fit loop to solve for the constant that best fits the real-world data.
- Solving with a scipy ODE solver.

Class 16 - COVID: The mathematics of pandemics [Slides]

- Covid overview (stats, growth rate, current state, challenges).
- Introduce the concept of a conceptual model where we have rules about how people transition from Susceptible, Infected and Recovered.

Class 17 - COVID: Modeling pandemic interventions [Colab]

- How to use a scipy integration library.
- What happens if we try different lockdown scenarios?
- What happens if we change vaccination rates.

Class 18 - COVID: Interacting pandemic populations [Colab]

- How do two populations interact and spread COVID?

- The SIR model with six parameters.

Class 19 - Machine learning: Handwriting recoginition

- The problem that brought ML to prominence
- Support vector machines

Class 20 - Machine learning: Neural network fundamentals

- Fully connected encoders
- Libraries

Class 21 - Machine learning: Neural network fundamentals [slides]

Class 22 - Machine learning: Image classification [Colab]

Class 23 - Simple approaches to next word prediction [Colab]

Class 24 - Student project presentations

Class 25 - Outlook

Bok Center resources for learning

Please refer to this document for any assistance you may need related to learning. The document includes support for students for:

- Student safety and health issues
- Mental and emotional well-being
- Inclusion and belonging
- Financial aid
- Academic support
- Academic planning
- Co-curricular activities and programs

Online connections

Course website: https://canvas.harvard.edu/courses/115258

Academic Integrity

Please read Harvardâ€™s policy on academic integrity, in the Undergraduate Handbook. Also, please read Harvardâ€™s guide to using sources, which includes a section on â€œAvoiding Plagiarismâ€. It is your responsibility as a student to read and understand the provisions of the Harvard College Honor Code. Cases of suspected Honor Code violation will be referred to the Honor Council. Note: Course materials are the property of the instructional staff, Harvard University, or other copyright holders, and are provided for your personal use. You may not distribute them or post them on websites. We encourage collaboration in lab and section activities. However, it is expected that all assignments and lab reports submitted for academic credit will be the studentâ€™s own work. If you use the work of others in preparing your responses, this must be properly acknowledged as outlined in the Harvard College Student Handbook. Absolutely no collaboration or use of auxiliary materials is allowed during exams. All exams must be completed independently with no aids.

Accommodations

Students needing academic adjustments or accommodations because of a documented disability must present their Faculty Letter from the Accessible Education Office (AEO) and speak with the professor by the end of the second week of the term. Failure to do so may result in the Course Head's inability to respond in a timely manner. All discussions will remain confidential, although Faculty are invited to contact AEO to discuss appropriate implementation.