

Daniel Fried

Teaching Statement

My interest in teaching started in high school when I led a weekly recitation section for a history class and has continued since then. Some of the most rewarding experiences of my undergrad were teaching opportunities: substitute lecturing a discrete math class for a traveling instructor as a TA and running tutorial talks on topics like functional programming in a student club. I've continued to gain teaching experience during my time at UC Berkeley:

- **Running a course:** I was one of two primary instructors for Berkeley's Intro to Artificial Intelligence course (cs188) during the 2018 summer session. This was an upper-division undergraduate course of 160 students which runs at twice the normal speed during the summer. I lectured up to 4 times per week, managed a course staff and handled logistics, and designed curriculum: reworking a unit on neural networks, helping lead design of a new collaborative project, and writing exams. Student evaluations rated my teaching effectiveness as 6.2 / 7.0, which is in the top 25% of instructors for this course in the last 10 years. I was recognized for this teaching with UC Berkeley's Outstanding Graduate Student Instructor award.
- **Curriculum design:** As an instructor, a TA, and a volunteer, I've contributed a range of content to the undergraduate intro-to-AI course (cs188) and a graduate-level natural language processing course (cs288). I've created or co-created projects for these courses that have ranged from heavily-scaffolded to more open-ended and from individual to collaborative. These projects have been used by thousands of students in total. I've also written course notes, exams, and section problems for the undergrad AI course and guest lectured for the graduate NLP course.
- **Research mentoring:** I've mentored several students on research: an undergraduate, a master's student, and a younger PhD student in their first projects with our group. Depending on students' needs and stage in gaining research independence, this has involved feedback on project selection, help with model and experiment design, implementation advice, and paper editing and presentation feedback.

A common theme in these teaching experiences is to *personalize learning*: using multiple pedagogical strategies to engage students, interacting with students as individuals, and tailoring content to students' past experiences and future needs.

Engaging students with multiple strategies

Some students get most engaged with a course, and learn the most, when they have freedom to explore. Other students get most engaged when collaborating with a partner, teaching and being taught by them. Since students' intrinsic motivations vary, I've taken multiple approaches in my teaching. Making a large lecture class interactive and engaging to students can be challenging, but showing demos and examples and giving students chances to respond both to the instructor and each other can go a long way. In student evaluations for the Intro to AI course I co-taught, one student wrote,

From lecture, (Daniel) encouraged participation by asking questions and having us discuss with people around us. I think he did an amazing job.

While some parts of a course, like exams, need to be highly-structured individual evaluations, assignments afford more flexibility. When teaching the AI course, I helped introduce a new project where students created game agents that could be partnered with other students' in a collaborative version of the game PacMan. Feedback on the project showed that the collaborative aspect was a big motivator, and the project was used in future semesters of the course. In other projects, I've built in a progression from a high degree of scaffolding to more open-ended exploration. For example, when I was a TA for the AI course I helped design a project where students first implemented parts of a simple machine learning library for automatic differentiation, then used this library to explore neural network approaches for problems including classifying images. Over 4,000 students have completed our project in the semesters since then. In a class where students have a large range of past experiences and degrees of comfort with programming, a combination of supportive scaffolding and room for creativity can be crucial to their engagement and success.

Interacting with students as individuals

Office hours, focused use of interactive online tools, and impromptu after-lecture QA sessions can outweigh the impact of a lecture for some students, both in solidifying the material and in opening up new avenues of investigation. While it's impossible to interact with every student in a large course, engaging with as many as possible gives a barometer of what's working well and what's not, and allows adapting approaches and focus as needed. I enjoy putting time into smaller-group interactions: in evaluations, one student wrote,

(Daniel's) office hours were incredible... he is really really great at explaining concepts. He is willing to work with students through problems/examples on pen and paper to really solidify their understanding.

Another student wrote,

He researched questions that I had that were beyond the scope of the class and got back to me during a break during lecture which was absolutely incredible! Possibly the best professor-student interaction I have ever had!

It's been rewarding to see students develop their interest within the class and afterward, and it's been incredibly gratifying to write recommendations for students and see them go on to do research or become teaching assistants, both in AI labs at Berkeley and graduate programs elsewhere. Personal interaction with students is enabled, particularly in a large course, by recruiting and training a strong course staff, and I also enjoyed seeing our staff develop. One staff member we hired for our summer class was a first-time TA who frequently sought feedback from us on teaching. He returned to TA for the course four more times after our summer, eventually as head TA.

Tailoring content to what students have and need

It's typically easiest to get concepts to stick when they are built on a scaffold of past skills and concrete examples. During the summer, the AI course has students from a mix of different universities and departments, so there was significant variation in how people have learned the prerequisites. We used lectures structured around motivating examples which we then generalized later. Projects and assignments were designed to give students practice developing skills that they reused later on. Students seemed to appreciate this incremental approach, with one writing,

Daniel's knowledge and ability to communicate said knowledge in a digestible format has been refreshing in a large CS lecture. Really really impressive grasp of the subject matter combined with a very perceptive ability to convey the complex concepts in a way that is both complete yet not overwhelming.

Beyond the classroom, tailoring instruction to individual students is particularly important in research mentoring. The main objectives in research mentoring should be to teach students skills and enable them to meet their career goals. In working with younger students, I've tried to strike a balance between encouraging them to do short-term tasks to push a project forward and stay motivated, skill development to enable other research directions, and longer-term exploration to develop their interests. As in the classroom, this often leads to an incremental approach, where we first work on a concrete subproblem that we can iterate on quickly, but one that could be extended in one of several directions that all seem interesting.

Teaching plans

I'd be glad for the chance to introduce students to AI, machine learning, and natural language processing, as well as to teach them more advanced tools in these areas. My experience as an instructor for an upper-level undergraduate intro to AI course has both prepared me and motivated me to teach a similar course in the future. I'd also be excited to teach a graduate course or upper-level undergraduate course on natural language processing, or graduate seminars on my research areas within it. One of the main reasons that I'm interested to remain in academia is the chance to repay to others the instruction and mentoring that I've received, and I look forward to continuing to teach in the future.