Teaching Statement

Gregory Rosenthal

November 15, 2024

1 Teaching philosophy

Teaching is important. Teaching is obviously one of the core purposes of a university, alongside research. Few people are complete autodidacts, so without teaching a field will die out. There are also self-interested reasons for a researcher to care about teaching: In *Surely You're Joking, Mr. Feynman!*, Richard Feynamn observes that a) teaching boosts his morale even when he's stuck on research, b) re-thinking the basics of a field can lead to new research, and c) questions asked by students can lead to new research. To which I'd also add, d) my classes may motivate some students to help me with my research.

Teach induction and proof-writing in the context of discrete math... Where I did my PhD, the introductory CS theory course for undergrads essentially *only* covered induction and proof-writing.[1] In contrast, the analogous course that I took as an undergrad myself covered these topics in the context of basic discrete math (specifically logic, set theory, number theory, combinatorics, probability, graph theory, and automata theory). I firmly prefer the latter approach, not only because discrete math is important, but because it's hard to teach proof-writing without an actually substantive statement to prove.

...and explain why discrete math is interesting. When I was an undergrad, I thought DFAs were just a toy model used to build up to Turing machines, and only years later learned that DFAs have applications in areas such as software verification. I can easily imagine why some students might feel similarly about the rest of the content in a discrete math course. To make bored students interested and make interested students excited, I think it's worth briefly mentioning Gödel incompleteness when teaching propositional logic, RSA encryption when teaching modular arithmetic, and so on.

Lean breadth over depth, at least after introductory courses. Obviously there are some core topics that every student needs to understand. But beyond that, my inclination is to introduce students to the breadth of topics within a general area, and point students to references where they can read more about any of these topics if interested (as researchers do). In particular, in upper-level courses where students are expected to already know what constitutes a formal proof, I think it's fine to present proof sketches rather than complete proofs sometimes (again, as researchers do when talking to each other).

[1] As well as analysis of simple algorithms, but similarly this can be taught in a proper algorithms course.

Make sure undergrads know to consider doing research. Many people don't know that professors do research in addition to teaching. For example, I have a family member who for years never understood that the professor in my family had responsibilities over the summer. Even in core undergrad courses, I'd like to (briefly, and in a manner integrated into the course content) talk about modern research topics that I'm excited about and the human experience of doing research, so that students know to consider this.

Post online lecture notes. Even students who want to learn miss classes sometimes. For example maybe they feel sick, slept in, are overwhelmed with other work, or just learn more efficiently by reading than by watching lectures. I consider it important to serve these students with written material that covers the content of the lectures, either by writing lecture notes myself, requiring each student to scribe a lecture (and then proofreading their scribe notes), or indicating a textbook section if the lecture closely follows one.

Give partial credit for saying "I don't know". In CS theory courses I TA'd as a grad student, it was customary to give 20% credit for just writing "I don't know", 10% credit for leaving a question blank, and 0% credit for complete nonsense. This obviously saved time for the graders by incentivizing students to write less—presumably this point is even more relevant now that LLMs exist—but I also see two ways that it benefits the students. First, it teaches the importance of intellectual humility: confidently doing something wrong can cause much more damage in life than asking a colleague for help. Second, it teaches the importance of clear and concise writing.

Customize my PhD supervising style to the student. My PhD advisors adopted a relatively hands-off style with me and that worked well. Instead of weekly meetings, we'd just meet when there was a specific reason to do so (which would occasionally be more than weekly) and also exchanged lots of messages in place of meetings. However, I recognize that many students need more structure than this, especially early on.

2 Teaching experience

TAing I've TA'd several courses, including core undergrad courses, topics-level PhD courses, and everything in between. I have experience with office hours, small-group lectures, grading, manging the scribing of lecture notes, and proofreading assignments.

Mentoring As a postdoc I've a) helped teach a PhD student coauthor on one of my papers how to write well, b) given another PhD student in my group significant advice on which research problems to pursue, and c) recently started mentoring another PhD student as part of a semi-formal mentorship program within my college at Cambridge.