

# Teaching Statement

Steven Holtzen

Teaching is an integral part of my life, and I am deeply proud in the positive impact my teaching has already had on my students and mentees during my PhD. I see teaching as the foundation of all higher learning: when I teach students, I always aim to form a deeper understanding of the topic myself, and convey that insight as I am forming it. I have been fortunate to help teach two courses while at UCLA, as well as mentor a variety of students in research. For my teaching I was recognized by the UCLA computer science department with an *Outstanding Teaching Assistant Honorable Mention* in 2019.

I have extensive teaching experience from my time as a PhD. student. With my advisor Guy Van den Broeck I helped develop a new advanced graduate class, *CS267A: Probabilistic Programming and Relational Learning*. This past quarter I was a teaching assistant for *CS30: Principles and Practices of Computing*, an introductory computer science course targeted at brand new programmers taught by my advisor Todd Millstein. I have also helped lead reading groups on special topics like probabilistic programming, combinatorics, and theoretical computer science.

**CS267A: An Advanced Graduate Course** I was extremely fortunate to have the opportunity to develop an CS267A, an entirely new course. I developed curriculum including homeworks, exams, and a final project. When designing assignments, I aim for a balance of accessibility and challenge: I want the students to be comfortable pushing themselves. I also like to give students agency during the course: in CS267A we gave the students a choice between an open-ended final project, or a guided one that had more structure and a well-defined objective. Throughout the course I also held regular office hours to guide the students through the assignments, and I lectured on probabilistic programming. In the end I received anonymous teaching evaluations from 41 students, who gave me an overall average evaluation score of 8.09/9.0. For context, the average teaching evaluation for professors at UCLA is around 7.5/9.0. Some anonymous students gave me written feedback, which I quote here in full:

- “Steven is a great TA and insanely knowledgeable about the course material. I appreciate that he is excited about the material, and that he makes himself available to questions.”
- “The TA was really helpful whenever approached. Overall, a good experience interacting with him.”
- “Steven is a good TA. He is always available on online forums and during class.”

**CS30: An Introductory Computer Science Course** I am especially attracted to teaching new students how to program and inviting new people into the computer science community. This past quarter I was fortunate to be a teaching assistant for CS30, an introductory computer science course targeted at brand-new programmers, with a particular focus on increasing diversity and accessibility in computer science. The majority of my students were first-time programmers from diverse backgrounds: most are not computer science majors, and roughly half of the course consists of women. My responsibilities including giving a weekly 2-hour discussion, creating assignments, and holding regular office hours. I received an overall T.A. rating of 8.45/9.0 for this course. Moreover, this course was taught entirely online.

My philosophy in this introductory class was to be inviting and open to *any* questions, letting the student’s curiosity drive and motivate course material. As one student put it, “As someone who had basically 0 exposure to coding, I went to A LOT of office hours and even though it was hard to articulate my questions I always felt like [Steven] listened to every part of them and was able to help me through it.” I try to be approachable to all students: in the words of a different anonymous student, “[Steven] is friendly and very understanding. Steven brings in a warm human element into a difficult academic setting”.

Every CS30 discussion is centered around a worksheet that the students work through together in groups. These worksheets are critical for students: according to an anonymous student, “The practice

problems we were given and the opportunities we had to work with each other allowed me to be comfortable with the course material. It was a very comforting environment in which I was not nervous in asking questions.” This engagement is critically important during the current era of remote learning: I firmly believe that individually reaching students is essential for retention in computer science courses.

This worksheet informs the future design of lectures: we identify key weaknesses that we can cover in more detail, and identify strengths that let us accelerate and challenge the students. The students noticed this dynamic effect the worksheets had on the class in anonymous evaluations, stating “Steven goes out of his way to expand upon the class material and try to adjust teaching in such a way that students who are having difficulty grasping the material are catered to and helped.”

**Reading Groups & Unstructured Learning** In addition to teaching in a structured environment, I also enjoy leading reading groups to explore advanced topics. I am currently leading a reading group on theoretical computer science with 5 other graduate students and an undergraduate student. I helped to run a reading group with 12 other students on mathematics in computer science based on *Concrete Mathematics: A Foundation for Computer Science*.

**Mentorship** While at UCLA I have been fortunate to mentor a variety of students with different backgrounds. I am currently working with an undergraduate, Ellie Cheng, on research projects with the goal of helping her prepare for graduate school in computer science. I am working with a Master’s student, Meet Teraviya, on a research project that will become his Master’s thesis. Both Ellie and Meet intend to publish their work at top-tier conferences under my guidance. In the past, I mentored a PhD. student Honghua Zhang on a research project that ultimately yielded a publication at the Conference on Uncertainty in Artificial Intelligence (UAI 2020).

When mentoring students my first task is project selection. My goal in helping a student choose a project is guiding them towards problems that are interesting, accessible, and educational: I want them to be excited and learn something while working on it. I consider it a good project if my mentee has ideas or extensions that I did not initially consider: this means that it had depth and that the mentee was truly engaged.

**Teaching Interests & Course Design** I am well-qualified to teach general courses at any level on machine learning, artificial intelligence, probabilistic graphical models, and programming languages. I am interested in teaching more specialized advanced topics courses in probabilistic programming and statistical relational learning. I enjoy computer science theory (complexity and computability, analysis of algorithms, randomized algorithms), mathematics of computer science (discrete mathematics, combinatorics, proofs by induction, probability theory), and software engineering and systems design, and I can teach introductory courses on these topics.

I am excited about the opportunity to design entirely new courses. A subsequent offering of CS267A taught the students how to use `Dice`, a probabilistic programming language that I developed during my PhD. I was amazed when about 150 students successfully downloaded and ran my tool, and used it to solve homework and projects during the course. This deeply affected me: in the future, I will use `Dice` as an interactive tool for teaching students probabilistic inference, and aim to develop courses around it. I see this potentially as a course on its own – teaching students the basics of probabilistic modeling – or as a replacement for a standard statistics and probability offering.