

I view my role as an educator to have 3 essential parts: classroom teaching, research mentorship, and outreach beyond the university. As the head TA for *Intro to Machine Learning* at Brown University (fall 2013), I led weekly recitations and office hours for over 50 students. I have mentored over 10 undergraduate, masters, and Ph.D. students on research projects at Brown and Harvard, including 4 honors theses, 2 masters theses, and 5 co-authored publications at major machine learning conferences. Finally, my outreach experience includes 2 years as an AP Computer Science instructor at Boston Latin Academy via the TEALS “Computer Science in Every High School” volunteer program<sup>1</sup>, as well as work on open-source remote sensing tools to respond to village burnings with the Harvard Humanitarian Initiative<sup>2</sup> (with help from two CS students at Brown). I view all these efforts – teaching, mentorship, and outreach – as interrelated ways to train more people to turn data into actionable knowledge and bring the benefits of data-driven insights to the broader public. Looking forward, I have three guiding principles as an educator: students should (1) learn by doing meaningful work, and (2) seek and reflect on feedback, and (3) feel welcome.

**Goal 1: Students should Learn by *Doing* and Do Meaningful Work.** My ideal course favors active learning methods throughout. In the classroom, I give brief lectures to motivate new concepts and model different approaches, but students spend most time in small groups collaborating to solve sample problems. For assigned homework, I prefer case studies requiring both math and code motivated by real datasets and applications. This approach helps students practice situated learning. I close each course with an open-ended final project that lets students demonstrate mastery of core concepts. Throughout each course, I articulate learning outcomes up front and accept regular student feedback on pace and content via surveys or in-class minute papers.

My emphasis on learning-by-doing comes from my undergraduate days at Olin College, whose core mission is hands-on engineering education. In my Olin courses, I built a hovercraft, synthesized quantum dots, and fabricated heat sinks. From this experience, I view students as capable individuals who should take ownership and learn by doing. I am particularly excited about course projects that cover the entire data science pipeline: cleaning messy data, choosing methods wisely, evaluating rigorously, and communicating insights effectively (including to non-technical audiences). When building an object detector for satellite images in my humanitarian outreach work, I solicited help from two Brown CS students in an advanced computer vision course taught by Prof. James Hays. Their final projects became exciting experiments in neural network feature extraction and human-in-the-loop detection. When student projects are self-directed and tackle meaningful problems, they help students build *lifelong* passions as well as employable skills.

**Goal 2: Students should Interact, Receive Feedback, Reflect, and Repeat.** As a student and postdoc, I have found that the best way to make progress on difficult problems is to seek and iterate upon feedback from others. At Brown, my adviser Erik and I led weekly meetings where the 4-6 students working on projects related to our Bayesian Nonparametrics in Python (BNPy) package would meet together rather than one-on-one. Each student gives a short update (via whiteboard or slides) followed by feedback from the group. As a postdoc, I led similar meetings for thesis

---

<sup>1</sup>[www.tealsk12.org](http://www.tealsk12.org)

<sup>2</sup>My work with HHI was featured in a TedX talk: <http://youtu.be/u7l9rBwOnwU>

students. In these discussions, I model *critical thinking* and *debugging strategies*. For example, when told that an inference method is not working as expected, students and I collaborate to form testable hypotheses, collect appropriate evidence (e.g. trace plots of the objective function), and suggest concrete next steps (e.g. freeze one variable in the model). These interactions build analytical skills as well as presentation skills, because all participants need to be brought up to speed quickly.

Another exportable rapid feedback success is our Harvard lab's series of 10-minute chalk talks at "tea time" each Monday, Wednesday, and Friday. In a rotation, each member discusses either an in-progress research question or an interesting paper. These chalk talks create an lively atmosphere of rapid exploration and awareness of others' work. Given my focus on open-source software, I'm excited to bring public code reviews into this rotation as well.

**Goal 3: Students should Feel Welcome and Willing to Take Risks.** I hope all my students know that I see them as worthwhile, multi-dimensional people and that poor academic performance does not make me think any less of them. Prof. Francis Su at Harvey Mudd calls this "Teaching with Grace"<sup>3</sup>. Through informal conversations (e.g. over snacks at office hours or coffee with new group members), I listen to student thoughts and concerns, so I can understand their goals and help them succeed. Students who feel welcome are more willing to try new ideas and more receptive to feedback. Rather than hide my own failures from students, I try to model how to respond to criticism or feelings of imposter syndrome. I will gladly recount the time when I blanked on an easy question during a guest lecture by Prof. Michael Jordan (of U.C.-Berkeley), but then went on to coauthor a statistics journal article with him. I hope to foster a risk-taking growth mindset necessary to thrive in graduate school and beyond.

I especially hope that students from all backgrounds feel welcome in my classroom and my lab. To encourage participation from underrepresented groups, I will promote and fund attending conferences such as Women in Machine Learning (WiML)<sup>4</sup>, the Grace Hopper Celebration, and the ACM Richard Tapia Conference, especially early in a student's career. I will also mentor outreach efforts like TEALS that help younger students of diverse backgrounds get excited about computing.

**Courses I'd Love to Teach.** I am prepared to teach many undergraduate courses across computer science, applied math, and statistics. I would especially like to teach an introductory course on Machine Learning, covering core mathematical concepts as well as practical implementations using open-source toolkits. I would happily also teach courses on Artificial Intelligence, Data Science, Optimization, or Biomedical Informatics. Finally, I am excited about co-teaching with other instructors (especially those from outside CS) on topics at the intersection of data science and X, where X could be medicine, biology, or other sciences. One concrete idea would be an Intro to Computing for Pre-Med Students course, to help future clinicians practice computational thinking via projects focused on clinical and public health datasets.

For upper-level graduate courses, I prefer a seminar-and-project approach where students try reading, criticizing, and remixing recent research papers. I could cover different modeling approaches (Graphical Models, Deep Probabilistic Models, Bayesian Nonparametrics), different inference methods (Markov Chain Monte Carlo, Variational Methods, Probabilistic Programming), or different application needs (Semi-supervised Methods, Time-series Analysis, Fairness & Transparency in Machine Learning).

---

<sup>3</sup><http://mathyawp.blogspot.com/2013/01/the-lesson-of-grace-in-teaching.html>

<sup>4</sup><http://wimlworkshop.org/>