Teaching Statement

Brian Plancher
brian.plancher@gmail.com

Inspired by my professional experiences, and my interdisciplinary research interests, I approach designing and teaching courses with two main goals. First, I aim to create courses that not only teach the core course material but also the knowledge and skills that students can use in their future courses, research, and careers. Second, I aim to create courses that promote diversity, inclusion, and belonging, understanding that all students and colleagues enter new courses and jobs with different backgrounds, expectations, preparations, and life contexts. To achieve these goals, I strive to design and teach courses that are foundational, project-based, accessible, and inclusive.

Figure 1: Starting the final race for the project-based robotics summer program for high school students.

During my PhD I have gained valuable experience designing and teaching courses across a range of topics and settings to a variety of students. From running Harvard's introduction to artificial intelligence course with over 150 student and 10 teaching assistants, to designing advanced seminar courses on robotics hardware-software co-design and tiny machine learning (and a corresponding introductory level MOOC), to helping launch a tuition-free, project-based robotics summer program for high school students, I have found that teaching has brought some of my most enjoyable and meaningful moments in graduate school and has inspired me to pursue a career in academia.

At the core of my teaching philosophy is my commitment to diversity, inclusion, and belonging (DIB). I include a DIB statement on course syllabi that explicitly calls out the fact that the course staff may have blind spots and that science has historically been dominated by a small subset of privileged voices. I work with the course staff to actively try to address these issues and encourage students to support the course staff in this effort. This focus underlies my principles of foundational, project-based, accessible and inclusive learning.

Foundational

Whether teaching an introductory undergraduate course or an advanced seminar course, my goal is that students will gain a foundational understanding of the topics covered in the course. I hope that students, regardless of their background, will leave my courses with core knowledge that they can build upon in future courses, in future research, and in their careers.

In introductory courses this manifests in a focus on teaching the concepts upon which the discipline is based. In the introduction to artificial intelligence (AI) class, my goal was that students came away understanding core algorithmic concepts that appear throughout the computer science curriculum. Therefore, I structured my lectures and sections around teaching recursion, local approximation, complexity analysis, etc., using AI algorithms like the value iteration equation as examples.

In advanced seminar courses the level of detail is increased but the concepts stay the same. By carefully choosing papers to be read and analyzed and by asking probing questions during group discussion, the foundational themes of the specific sub-field can be reinforced in every class. In a course on optimization algorithms for robotics, I gave an example paper presentation using one of my published papers, and focused on the differences and trade-offs between gradient descent, Newton's method, and the method used in our paper. My goal was that students would leave the class understanding where, when, and why the various approaches should be applied in research and industry.

Project-Based

I have come to realize that while it is important for a student to learn all of the facts presented in a given course, some of the most valuable skills I learned in the classroom were hands-on experiences, soft skills, and collaborative problem solving. As a teacher, I aim to center learning on projects and assignments that help students build and refine these skills alongside their technical knowledge.

We developed the Tiny Machine Learning course around 3 project-based assignments and a collaborative final project. To connect the assignments to real-world challenges, we had guest lectures from industry experts at leading companies such as Google, ARM, Microsoft, and Qualcom. In the first assignment, students trained a custom Keyword Spotting model (think "OK Google") on their own data and deployed the model onto an Arduino microcontroller. By closing the loop with hardware tests, students not only got to see the direct impact of their TinyML design choices, but also gained valuable experience with the entire TinyML flow. This assignment was graded not on the final model accuracy, but on the explanation and design of the data collection and testing scheme. This ensured that students focused not only on technical knowledge but also on problem solving, critical writing, and communication skills.

The project-based robotics summer program for high school students was structured with weekly team-based challenges to build core competencies in motion planning and computer vision, culminating in a final autonomous car race. To build excitement and community, we not only gave out awards to teams that won the challenges, but also to the teams that came up with the most creative solutions, and were the most effective at collaboration. We shuffled the teams between weeks, and in the second year we added mid-week challenges to both give students early feedback on how well their new team was working together and to provide clear, achievable checkpoints to build confidence and confirm understanding. Collaborative exercises with actionable feedback like these were instrumental in helping students from diverse backgrounds collectively solve problems, and gain confidence in the material. At the conclusion of the course, students self reported, and the staff observed, significant increases in collaborative problem solving techniques, soft skills, and technical skills.

Accessible and Inclusive

By teaching with a focus on foundational and project-based learning, my hope is that students are able to build a toolkit that they can use in their future courses of study, research projects, and careers. In order to ensure that all students are able to build that toolkit, it is important to structure courses and teach in a way that is accessible and inclusive for all of the students in the class, across their diverse backgrounds.

I have tried to address this issue by ensuring that students see the same topic through different lenses of learning. For example, in the introduction to AI class I used multiple teaching tactics including:

1. Contextualizing the algorithm through videos of it in action in the real-world and/or in active research
2. Deriving the algorithm, inviting students to ask questions at each step
3. Providing a graphical explanation of the algorithm to aid visual learners
4. Connecting, comparing, and contrasting that algorithm to other algorithms learned in the class (and from previous required courses if applicable)
5. Assigning theoretical questions about the algorithm's fundamental properties in problem sets
6. Assigning coding assignments to code up the algorithm for experiential learning

By using these multiple approaches, students who are visual, mathematical, and/or experiential learners get exposure to the topic in the way they learn best. And, by approaching the topic from multiple angles, students have more chances to evaluate their understanding and signal confusion which often raises helpful collective problem solving discussions for the entire class.

At the same time, I strive to create an inclusive environment in the course and make my office hours a safe space to ask any question. I set the tone that office hours are meant to be a space for shared learning and try to ensure that the teaching staff's office hours are available to all students by spreading them out across different days and times, and then offering additional times by appointment. I am also committed to 360° feedback and ask for anonymous student feedback after every lecture to improve the course in realtime.

Finally, this year I am co-designing a free MOOC course series adapted from a new seminar course I co-designed and am currently teaching: HarvardX's Tiny Machine Learning (TinyML). By offering this course as a free MOOC with hands-on labs leveraging Google Colaboratory's free compute for all model training and low cost Arduino microcontrollers, and assuming no prerequisites beyond some basic programming, my collaborators and I hope to democratize access to this new impactful technology.

References

[1] Sandy Baum and Patricia Steele. Who goes to graduate school and who succeeds?, 2017.

[2] National Science Foundation. Table 12-1 - NCSES Survey of Doctorate Recipient: Fall 2017