"Learning is not the product of teaching.
Learning is the product of the activity of learners."
–John Holt

As a student, I often had trouble paying attention in class. My issue was that, despite being interested in the subject matter, I would get bored quite quickly, and I often found myself mentally wandering off when forced to work on formulaic or repetitive tasks without being given any context. "Solve this set of 100 multiplication problems." Sure, I enjoy math, but what's the point? "Memorize the rulers of Mesopotamia in chronological order." … Probably not going to happen; you lost me. I assumed the issue was on my end, that I didn't fit the mold of an "ideal" student, and it wasn't until I neared the end of my undergraduate studies that I finally heard otherwise.When I was enrolled in UC San Diego's *Molecular Sequence Analysis* course, I was ecstatic to learn from Pavel Pevzner, world-renowned Bioinformatician. To my surprise, however, Dr. Pevzner began the first meeting of the course with "I will not be giving any lectures in this course," and thus began my first exposure to a "flipped classroom."

Flipping the Classroom

Motivated by my experiences in Dr. Pevzner's course as well as by Benjamin Bloom's landmark 1984 paper highlighting the significantly lower student performance in conventional lecture instruction compared to more interactive approaches, I decided that I would "flip" my courses as well: instead of covering the entirety of a given topic during the allotted lecture time, my students would perform the initial steps of the learning process at our own pace at home beforehand, and during the in-person meetings, I could focus on helping them overcome the unique "learning breakdowns" they encountered at home. I found numerous studies demonstrating the efficacy of flipped classrooms, and a brief history and review can be found in Cynthia Brame's 2013 teaching guide. I largely follow the method proposed by Jacob Bishop and Matthew Verleger in 2013 emphasizing that students should self-study using instructor-provided learning materials outside of class, and that these learning materials should utilize computer technology to automate the learning process. I have observed that, based on student scores for online challenges in the courses I have taught, *almost all* students are able to comprehend the majority of the concepts of a given topic at home without *any* direct involvement from me. Thus, I enter my in-person sessions with the assumption that all students have performed the assigned learning at home (which I incentivize by including quizzes/challenges with grade point values), and instead of lecturing…

Active Learning

… I spend the entirety of the in-person session having students work in small groups to solve problems I have designed to trigger what I have observed to be common misconceptions students encounter. When designing the activities, I employ the principles suggested by Douglas Barnes (1989): the activity must be purposive, reflective, negotiated, critical, complex, situation-driven, and engaged. I typically wander among the groups to help stimulate discussion by asking guiding questions: I aim to play the role of a facilitator, and I let the students lead the flow of instruction. After giving the students ample time to formulate and collect their thoughts, they discuss/debate their proposed solutions to the problem, and I help facilitate discussion regarding the pros and cons of each approach.

Inquiry-Based Learning

As mentioned, as a student, I would often lose interest in learning tasks that seemed arbitrary. The courses I teach are largely focused on solving problems (e.g. data structures, algorithms), and a key philosophy of mine is to motivate topics with real-world problems. Aligned with Barnes' suggestion of situation-driven and engaged content, prior to any technical discussion, in order to captivate the students, I introduce the real-world problem in the form of a story. For example, before introducing Dijkstra's Algorithm, I begin with a story about finding the cheapest flight route possible for my next vacation; before introducing Eulerian Cycles and their applications to Genome Assembly, I begin with the story of the enigmatic pathogenic strain of *E. coli* from the 2011 outbreak in Germany. By turning the problem into a story, I have found that students stay attentive because they wish to find out "what happens next." Further, by posing the topic using real-world problems, students will have the required context to directly apply the concept to new problems they encounter in the future.

Discovery Learning

As psychologist Jerome Bruner described in 1961, when a student finds the solution to an open-ended problem on their own, the student benefits two-fold: the student typically has a stronger fundamental understanding of the solution, and the student has an improved perception of his or her own abilities to solve problems of this nature. When teaching specific algorithms or data structures, I try to "trick" students into designing the algorithm or data structure with minimal guidance. For example, when teaching students about the Sequence Alignment Problem, instead of explicitly teaching them the Needleman-Wunsch algorithm, I attempt to guide them towards "inventing" it themselves. Any relevant background information provided by the teacher to facilitate Discovery Learning in the students requires effective instructor-created content, and the chosen mechanism of knowledge delivery is of utmost importance to retention of the foundational information. For the content I create for my own classes, …

Massive Adaptive Interactive Texts (MAITs)

… I am a strong proponent of Massive Adaptive Interactive Texts (MAITs, Compeau & Pevzner 2015), which are online texts embedded with activities, quizzes, and challenges throughout the instructional content. The motivation is to enhance the student's learning experience by having the student immediately engage with a given concept as it is being introduced, as opposed to leaving all engagement at the end of the learning process in the form of a series of quizzes. The MAITs I have written (e.g. *Data Structures*) allow students to interact with multiple-choice quizzes, short-answer questions, interactive puzzles, and even auto-graded coding challenges that can be solved directly in a web browser or via the Stepik iOS or Android app, and these activities are embedded *within* the content delivery. They are adaptive in that, by using a collection of common mistakes I have observed students make, I have created challenges that not only provide the student binary "Correct" vs. "Incorrect" feedback, but that actually attempt to predict what specific *unique* misconception the student's response suggests (e.g. "You may have an off-by-one error"), thus providing each student an immediate *individualized* learning experience. They are massive in that, because all activities are graded automatically, the MAITs can support an unbounded number of students: I have over 100,000 students in many of the MAITs I have developed, yet they scale perfectly. The *Data Structures* MAIT I wrote has been utilized by multiple professors at UC San Diego to flip the *Advanced Data Structures* course, and student reception has been largely positive. The “massive” aspect of MAITs is important to me because, in addition to utilizing the text in my own classes, I can make the content accessible to the public, where it can hopefully have far-reaching impact. For example, outside of UC San Diego, my *Data Structures* text has been integrated into courses at the University of San Diego and the University of Puerto Rico, into a MOOC (*Data Structures: An Active Learning Approach*), and into a full-fledged textbook, *Design and Analysis of Data Structures* (Moshiri & Izhikevich 2018).

Making Learning Fun and Accessible!

Motivated by my own learning experiences, I make a conscious effort to make my courses and learning materials genuinely enjoyable. In his 2009 book *Why Don't Students Like School?*, Daniel Willingham proposes that learning objectives are rarely very interesting on their own, which generally leads to student drop-off, but when posed as compelling problems, these same objectives can become quite interesting. My MAITs and courses are filled with stories, jokes, and puns, and I attempt to avoid the use of unnecessarily complex jargon when describing concepts to ensure that students of a wide range of backgrounds are able to follow successfully. As James Holt suggested, the success to learning is in the hands of the learners, and it is the responsibility of the teacher as the expert to design the educational journey to be genuinely captivating. Intuitively, it is much easier to teach when students *want* to learn. Based on widely positive student reviews from my UCSD and online courses, I feel confident that my strategy is successfully making it through to students. For example, in *Data Structures: An Active Learning Approach*, an online student from India shared “I am in love with the course! It is delivered perfectly in bite-sized chunks that makes it so engaging. It is not like reading a textbook, but at the same time, it is not passive like videos that quickly get boring. You have created a masterpiece that every computer science enthusiast must absolutely enroll in. Thank you very much for putting in the effort and providing this for free! I look forward to completing the course under your guidance.”

Goals as a Lecturer with Potential Security of Employment at the University of California, San Diego

In the short term, I wish to teach courses relevant to my academic interests (e.g. CSE 8A-B, 11, 12, 20, 21, 100, 101, 180, 181, 182, 184, 185, 202, 280A, 282, 283) using the existing curricula but with interactive classroom meetings as described above, and I would want to integrate auto-graded coding and math challenges into these courses. In the longer term, I wish to create new flipped courses for Computer Science and Data Science students, ideally with a focus on the intersection of algorithms and the analysis of large biomedical datasets. To do so, I wish to write new MAITs to integrate into the flipped classes, to make accessible for free to the general public, and to transform into physical textbooks. The two next topics especially of interest to me are introductory programming in Python, which I feel would be of great use to students interested in Data Science, and stochastic models in phylogenetics. I also wish to increase collaboration between the School of Medicine and the Computer Science & Engineering Department, as I firmly believe that computational analysis is at the heart of the future of personalized medicine. I sincerely believe that being a Lecturer with Potential Security of Employment (LPSOE) at UC San Diego will allow me to pursue my dream of making an impact on the world through education.

References

Barnes, D. R. (1989). *Active learning*. Leeds: Leeds University TVEI Support Project.

Bishop, J., & Verleger, W. (2013). *The Flipped Classroom: A Survey of the Research*. Atlanta, GA: 120th ASEE Annual Conference & Exposition.

Bloom, B. S. (1984). The 2 Sigma Problem: The Search for Methods of Group Instruction as Effective as One-to-One Tutoring. *Educational Researcher*, 13(6), 4. doi:10.2307/1175554

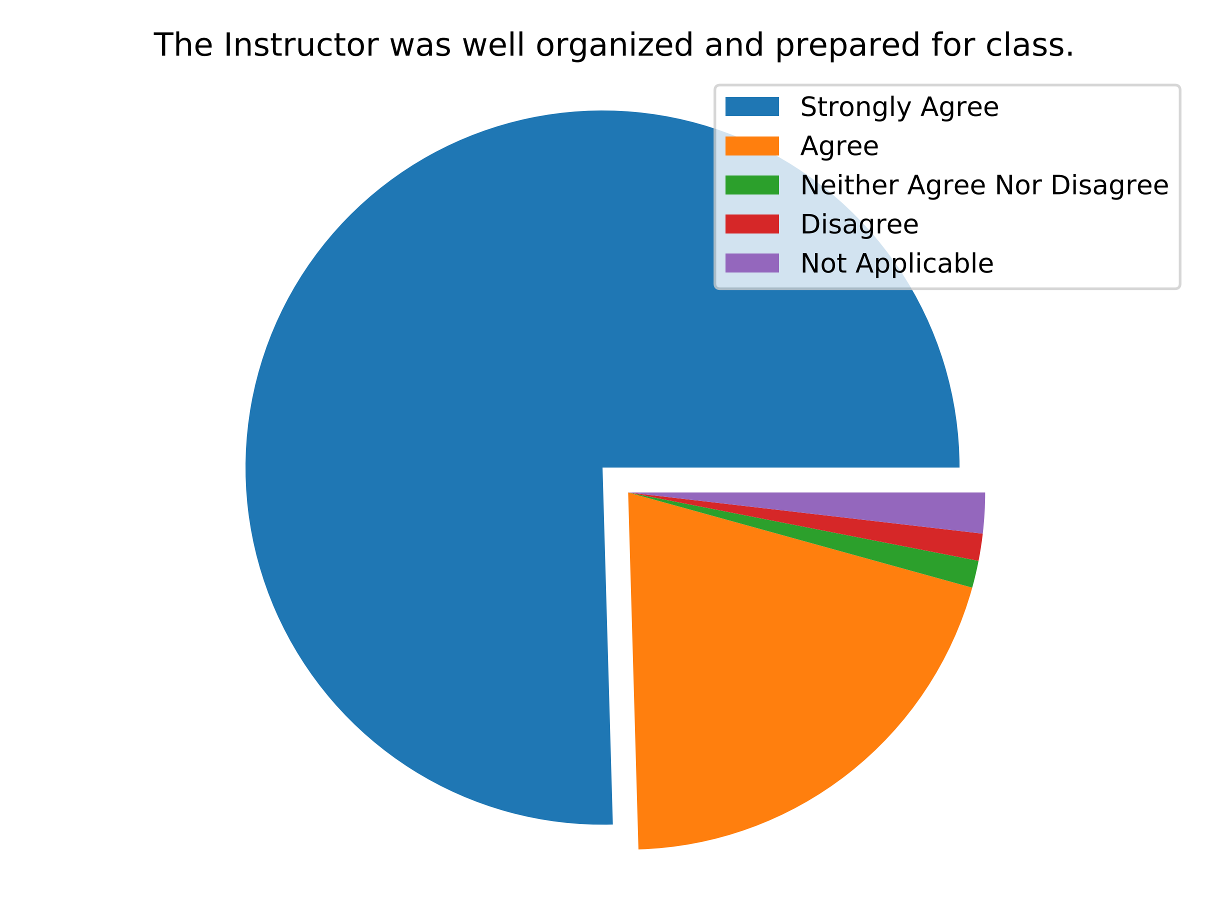
Brame, C. (2013). Flipping the Classroom. *Vanderbilt University Center for Teaching*. https://cft.vanderbilt.edu/guides-sub-pages/flipping-the-classroom

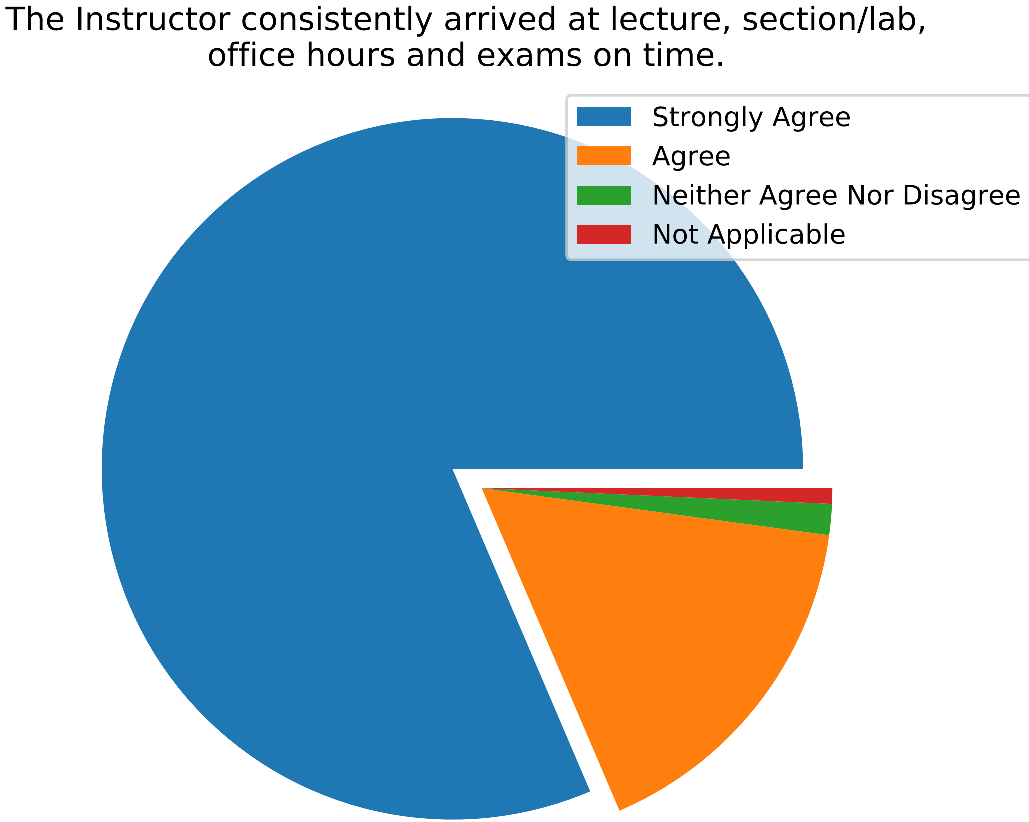
Bruner, J. (1961). The Art of Discovery. *Harvard Educational Review*, 31, 21-32.

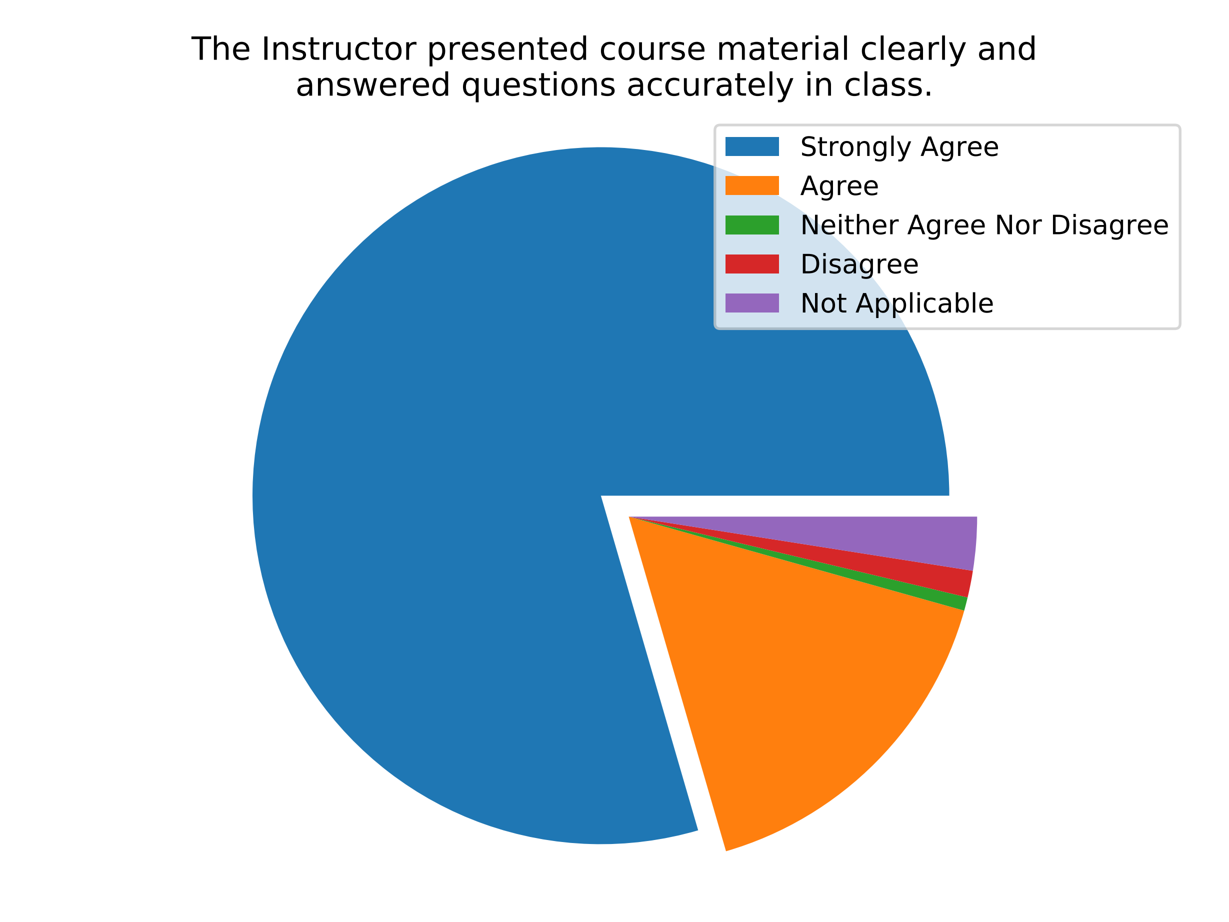
Compeau, P., & Pevzner, P. A. (2015). Life after MOOCs. *Communications of the ACM*, 58(10), 41-44. doi:10.1145/2686871

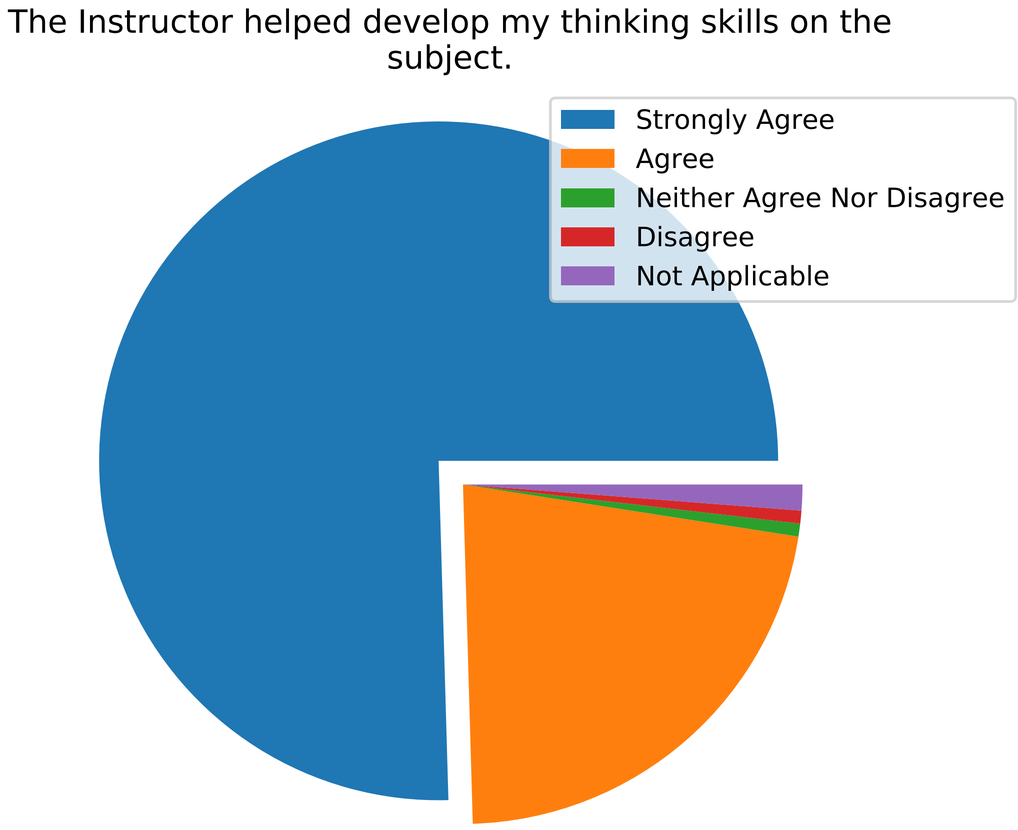
Moshiri, N., & Izhikevich, L. (2018). *Design and Analysis of Data Structures*. Amazon Kindle Direct Publishing.

Willingham, D. T. (2010). *Why don't students like school?: A cognitive scientist answers questions about how the mind works and what it means for the classroom*. San Francisco, CA: Jossey-Bass.

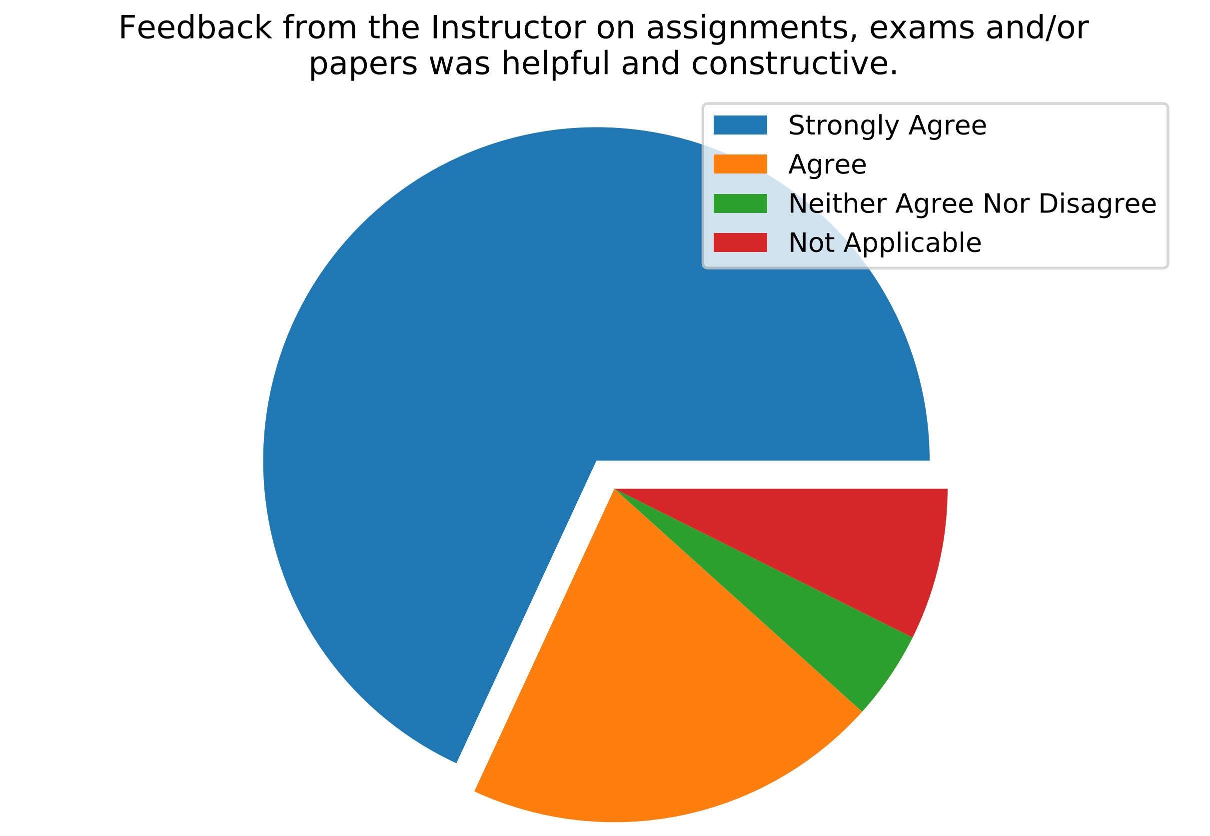

The Instructor was well organized and prepared for class.
Strongly Agree
Agree
Neither Agree Nor Disagree
Disagree
Not Applicable

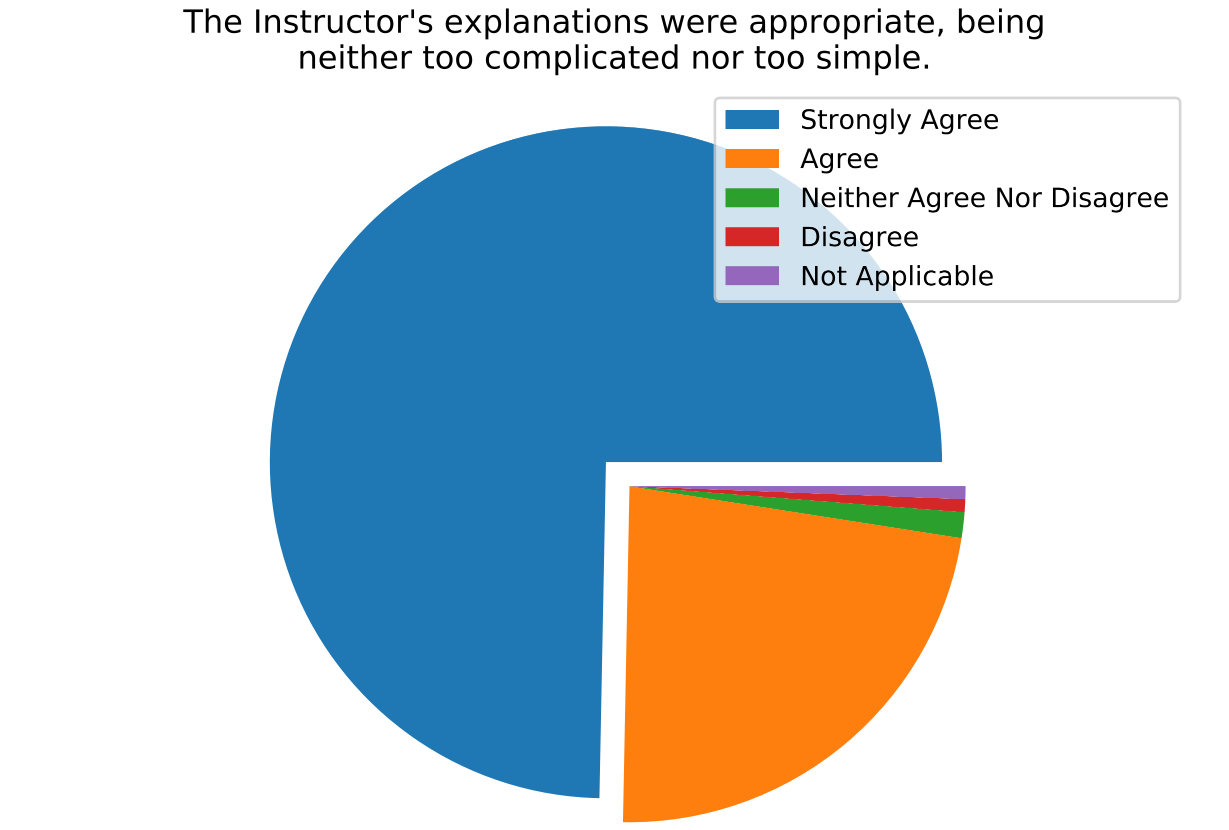

The Instructor consistently arrived at lecture, section/lab,
office hours and exams on time.
Strongly Agree
Agree
Neither Agree Nor Disagree
Not Applicable


The Instructor presented course material clearly and answered questions accurately in class.
Strongly Agree
Agree
Neither Agree Nor Disagree
Disagree
Not Applicable

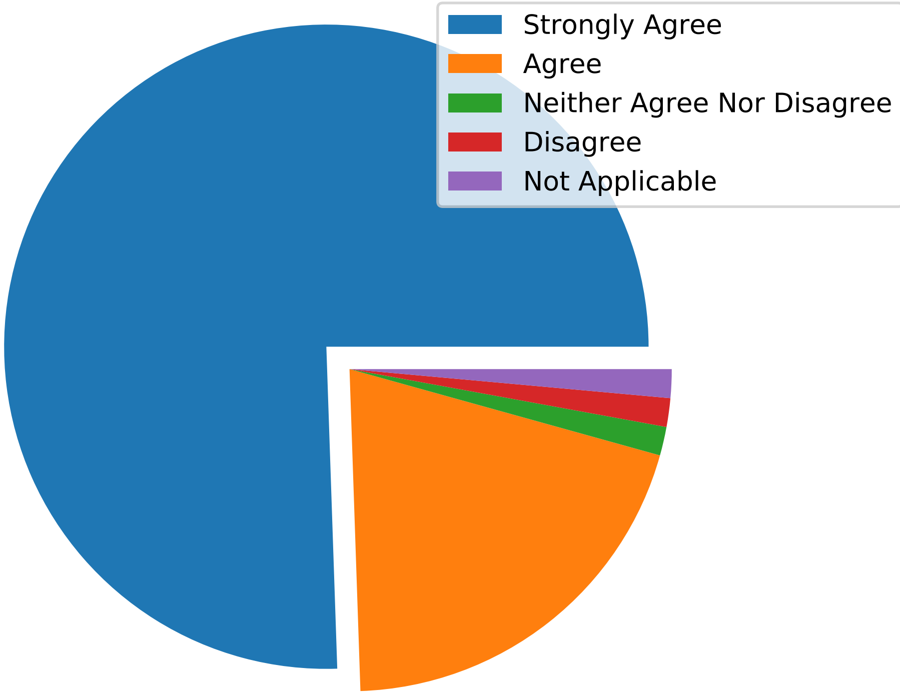


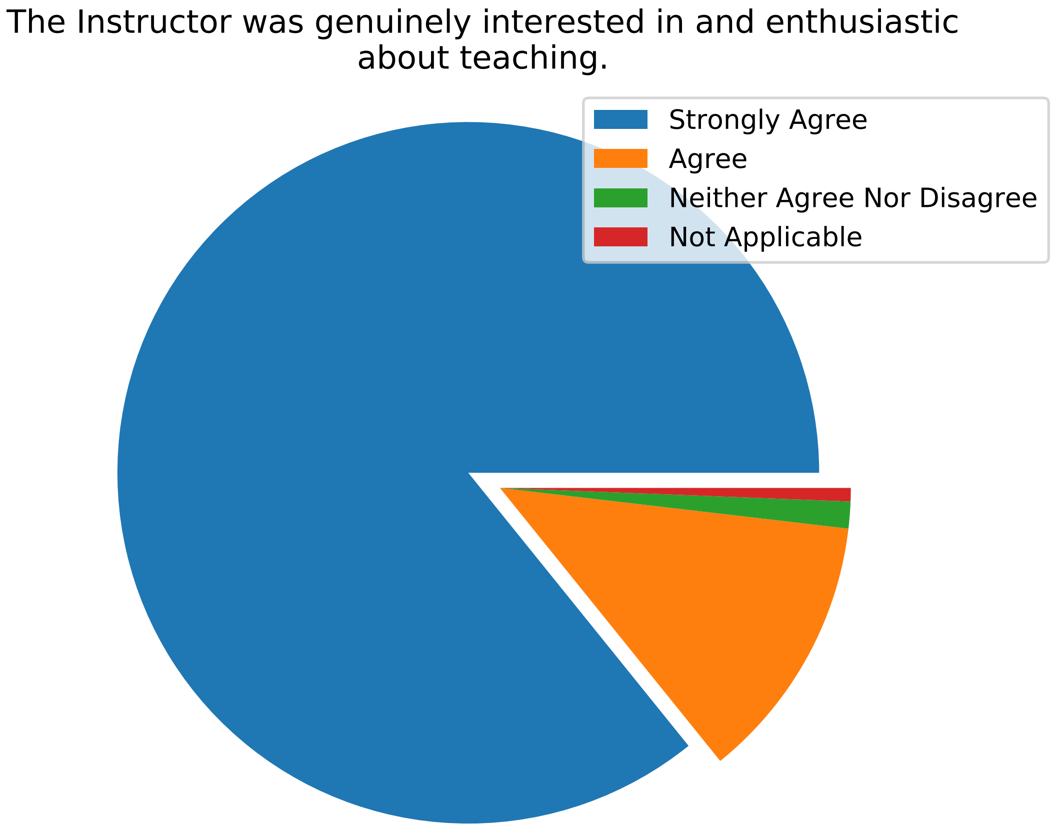
The Instructor helped develop my thinking skills on the subject.
Strongly Agree
Agree
Neither Agree Nor Disagree
Disagree
Not Applicable


Feedback from the Instructor on assignments, exams and/or papers was helpful and constructive.
Strongly Agree
Agree
Neither Agree Nor Disagree
Not Applicable

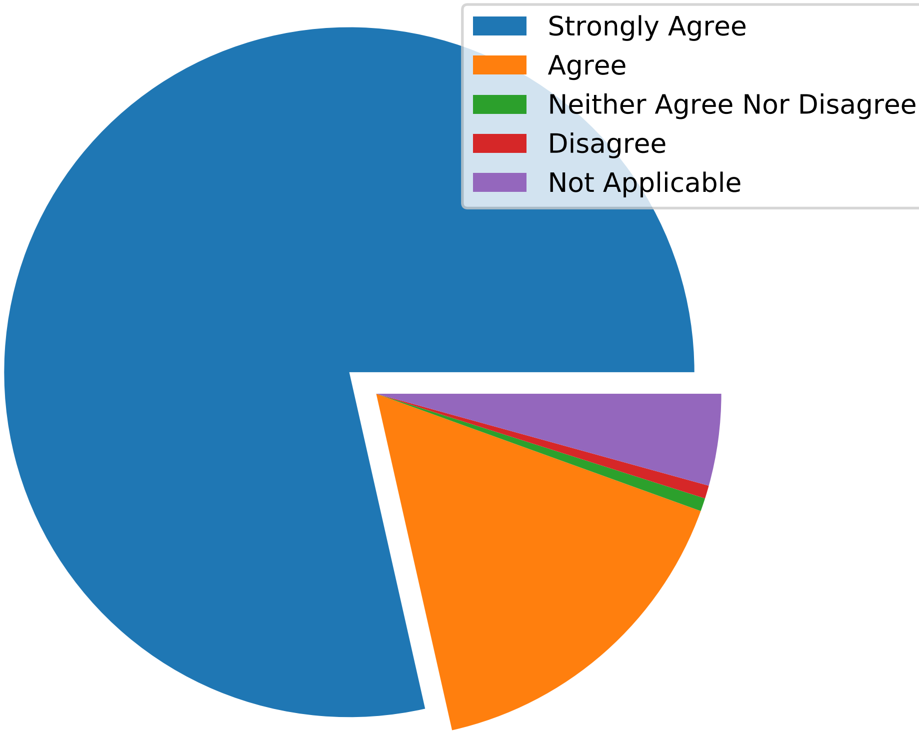

The Instructor's explanations were appropriate, being
neither too complicated nor too simple.
Strongly Agree
Agree
Neither Agree Nor Disagree
Disagree
Not Applicable

The Instructor answered questions clearly and effectively, helping students to make connections among the course readings, assignments, and lectures.

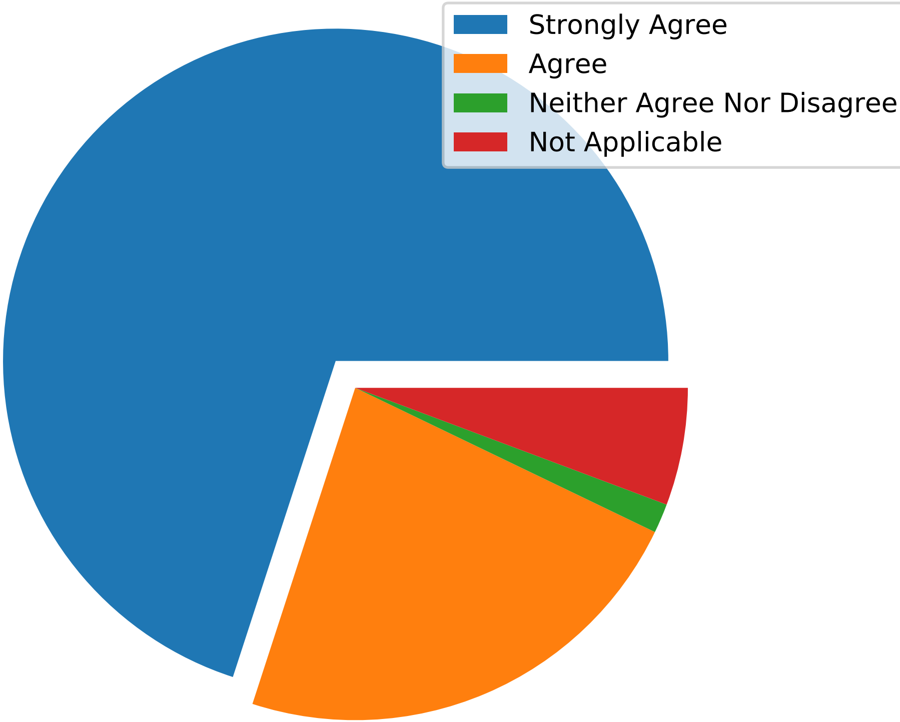


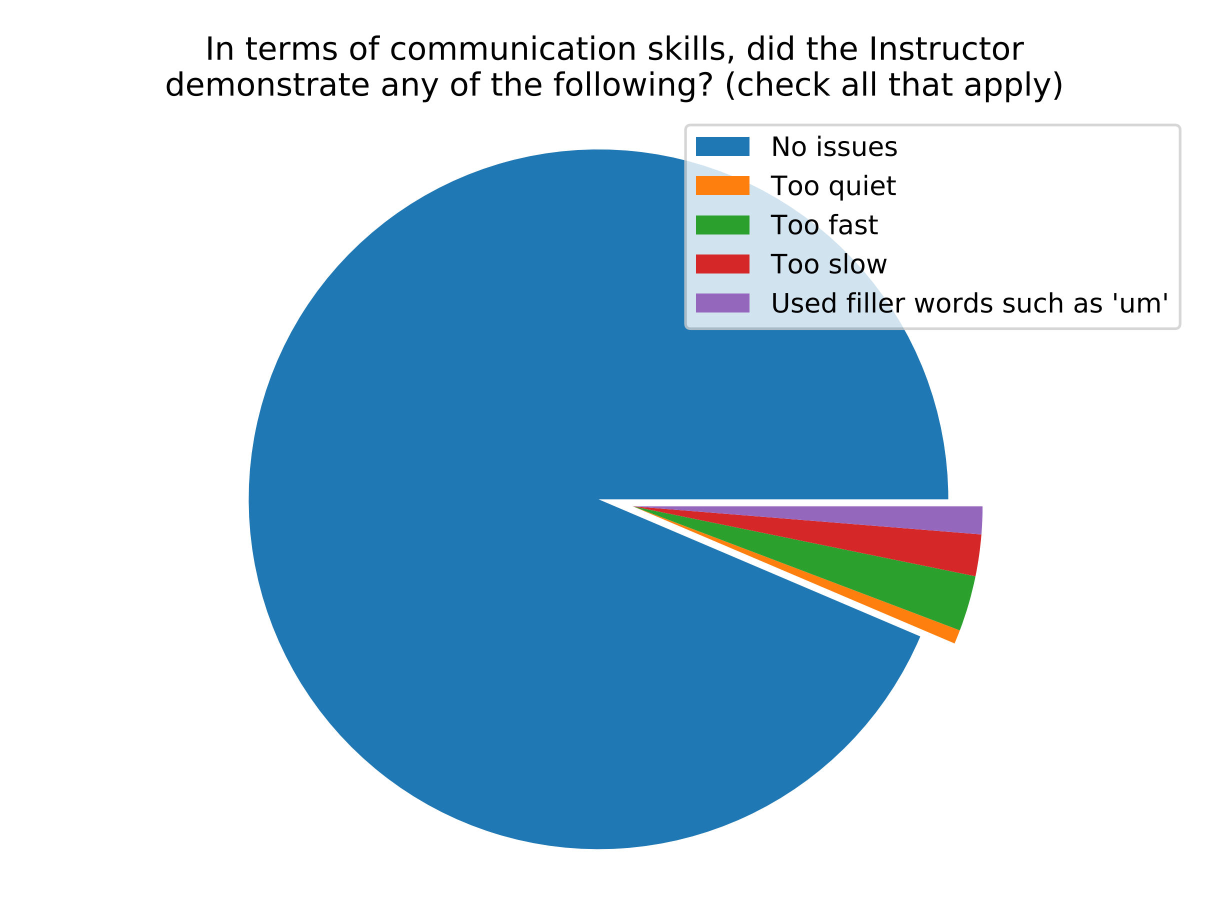

The Instructor was genuinely interested in and enthusiastic about teaching.
Strongly Agree
Agree
Neither Agree Nor Disagree
Not Applicable

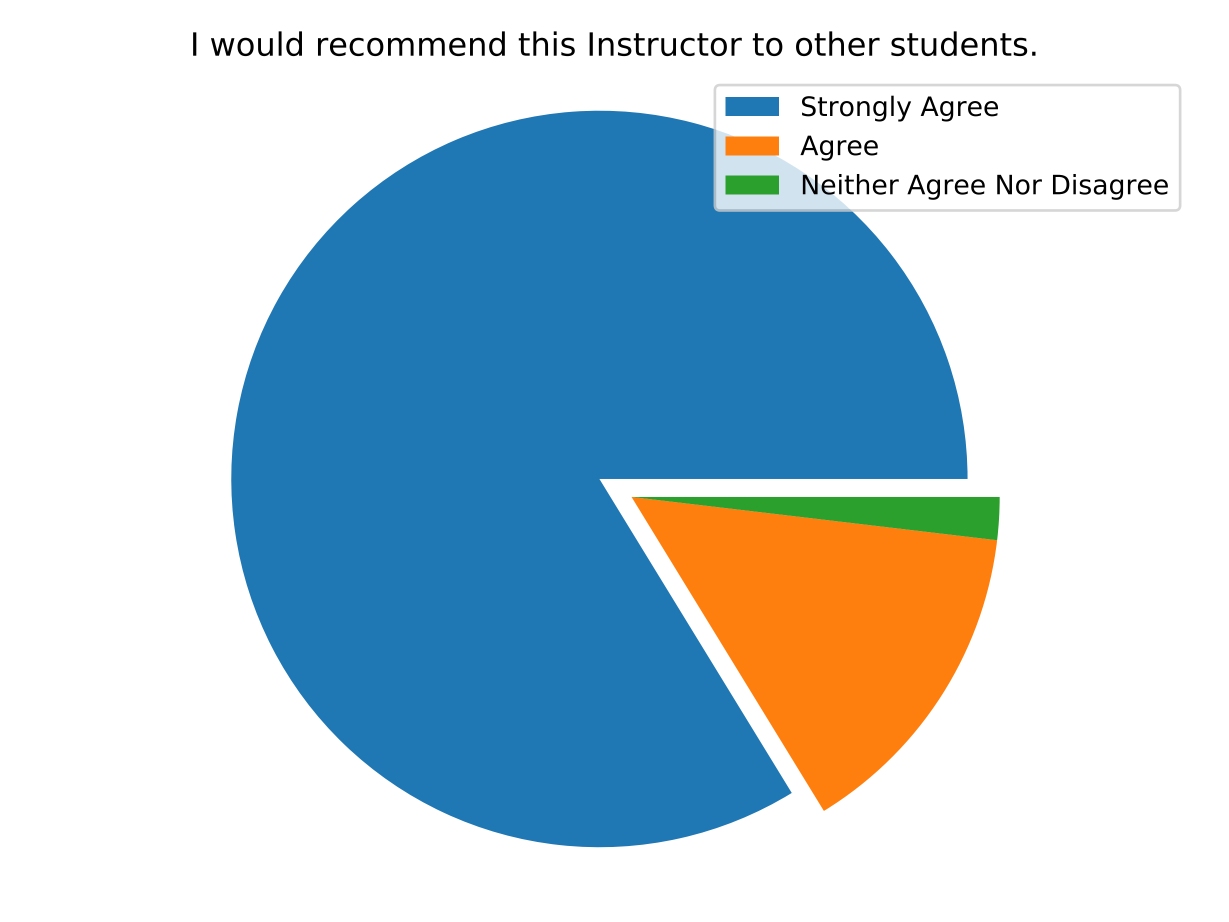
The Instructor was accessible to students outside of class
(office hours, e-mail, etc.).



The Instructor effectively connected the section/lab exercises with the material covered in lecture.




In terms of communication skills, did the Instructor demonstrate any of the following? (check all that apply)
No issues
Too quiet
Too fast
Too slow
Used filler words such as 'um'


I would recommend this Instructor to other students.
Strongly Agree
Agree
Neither Agree Nor Disagree