Qualitative Findings from an Online Course on Machine Learning

Steve Chenoweth, Ph.D.
Computer Science & Software Engineering Dept.
Rose-Hulman Institute of Technology
Terra Haute, Indiana
Email: chenowet@rose-hulman.edu

Panagiotis K. Linos, Ph.D.
Computer Science & Software Engineering Dept.
Butler University
Indianapolis, Indiana
Email: linos@butler.edu

ABSTRACT

This is a full paper in the Innovate Practice category. It reports experiences while teaching a largely online course about Machine Learning at two separate Universities. We targeted our course for a much wider than usual audience -- as "Computer Science (CS) for All," with undergraduate non-CS majors learning the same material alongside CS majors. We discuss why the majority of the students appreciated the flexibility of online classes designed for this wide group, and how they welcomed the opportunity to learn together about a "hot" topic such as Machine Learning. We explain our handling of challenges coordinating diverse and remote teams working with realistic big data of their own interest. Moreover, we describe how we engaged students in stimulating discussions about their readings and team projects, and how we balanced keeping everyone on the same pace while providing opportunities for learning ahead. Finally, we explain how we were able to attract the non-CS majors to take a CS special topics course and how we plan to use their constructive suggestions to improve future offerings of this course.

Keywords: Active learning, machine learning, data science, course management, learning environment. Teamwork and collaboration skills

1. INTRODUCTION

Software engineering jobs are already at 1,256,200 and have an expected growth greater than any other engineering field through 2024 [1]. At the same time, the "CS for all" K-12 computer literacy movement [2] needs to be supplemented by "CS for all engineers" and perhaps "CS for all professions," as software grows pervasively in value within these fields. Yet CS, like math, has a reputation in colleges which scares-off many students who will need to know it in their careers. Will they be ready, say, when half the work in other engineering fields has become software-based?

Our original goal was to create a CS course with widespread appeal which could be taken by beginners, and which would provide them with practical skills and spur their interest in software. Perhaps we could increase the number of people studying CS? Starting novices with an attractive, more advanced class is not a new idea. Richard Feynman and other senior physics faculty used this approach at Cal Tech in the 1960's [3]. This ruse is, however, somewhat unorthodox in CS. We believe the idea may be novel, of drawing students from other majors because of something outside CS that they already are interested in, and for which sophisticated programs, ML in this case, could provide a solution.

We also wanted the course to have value for more seasoned CS students looking for an intro to this particular topic. In addition, we hoped for a course in which CS majors and other majors would be able to exchange their perspectives, using the vector of learning a common subject. This is an aggressive try at "CS for All" -- attracting new audiences to learn about developing software because of some other interest, and using their working with CS majors as a further vehicle for that. These motivations led to three initial course design decisions: a) Machine Learning (ML) as the course subject, b) Online as the principal course format, and c) Designated points of interaction and reflection, about their diverse interests.

Throughout the rest of this paper, we explain our experiences and lessons learned from an effort to teach and coordinate this online course on Machine Learning, at two separate institutions. Section 2 below notes related work. Section 3 discusses the rationale behind offering such a course, whereas section 4 entails a detailed description of the course's logistics. Section 5 presents an empirical study, followed by our conclusions and future steps.

2. RELATED WORK

Online education has now been around for decades. The application of online to teaching Computer Science in particular is seasoned, as well. Specifically, online courses on Machine Learning (ML) already exist and are worth exploring. For example, some popular online ML courses including Stanford's Open Classroom course focus on statistical ML methods [4]. This course includes 30 video lectures and 9 student exercises. Other similar courses exist from vendors like Coursera, DataCamp,

978-1-5386-1174-6/18/$31.00 ©2018 IEEE

edX, Udacity, and udemy, primarily designed to be run as MOOCs (Massively Open Online Courses), without direct student support from faculty.

Remote education has helped to introduce and improve practices upon which it now relies, perhaps more than has face-to-face education. Online forums are an example of this. O'Sullivan and Zevallos cast online discussions as "patterns," and proposed different ones to be most effective, depending on objectives like class participation, collaboration, providing course feedback, and grading [5]. Ghosh and Kleinberg's study showed that the optimal rate of instructor participation is in some middle ground, to maximize student participation [6]. Mihail, *et al*, concluded that certain posting practices taken from online education also benefitted face-to-face classes. The posting of questions, and engaging in Devil's Advocacy had positive effects on learning; strictly informational posts and explanations did not! [7]

Online engagement techniques proposed by Linda Craig include "gamification," micro-podcasts, consistent feedback, focus on students' professional needs, calling on students during live webcasts, and storytelling [8]. Merrilea J. Mayo described video games, in particular, as a generalizable way to draw more young people into STEM fields. [36] Other authors, such as Hsin and Cigas, replicated the desirable outcomes from short videos [9]. Krause, et al found that "social games" especially improved retention in MOOCs -- their example being competitions between students [10].

At the same time, some of the factors in online learning seem to play stronger roles than in face-to-face. Kizilcec and Halawa found that successful learning in MOOCs was much related to goal striving and growth mindset, as well as to feelings of belonging. They suggested targeted interventions to narrow achievement gaps in online learning [11]. Krichen recommended that online courses need targeted communication based on evaluation of the individual learners -- that is, tailoring of the interpersonal aspects of the course [12]. Lee and Ferwerda went so far as to recommend personalizing online courses based on input from social media. This could be accomplished if students used their Facebook Login, for example. [13] MOOCs for learning programming now regularly employ web-based or other API's to guide the user's development steps [14].

The thrust of including programming in the liberal arts has been tried. For example, Peter Andersen, et al. discussed the importance of programming to college students studying multimedia. At the same time, they devised a special learning environment to "help alleviate the problems we have experienced in teaching programming to liberal arts students." [35]

Prior experience in teaching courses in data science, online at a university, date from Kalathur (2008) [15]. In university degree programs, standard classroom courses in ML have been proliferating, but almost all of these are upper division (for juniors and seniors), intended for computer science majors with significant prior coursework. Many are graduate courses. There is an emphasis on theory and on advanced techniques like boosting, semi-supervised learning, active learning (by the machine), and deep learning. Examples include Carnegie-Mellon (graduate level) [16], Georgia Tech (knowledge of matrix algebra and AI search is assumed) [17], Cornell (placement exam must be passed) [18], and Washington University (prerequisites include parallel computing and theory of computing) [19].

Finally, on the technical high end, Google has a stake in having as many people as possible understand "deep learning," which would provide a base of technical users for their commercial services based on these ML methods. They have a GitHub based project toward that end, and a free "advanced" course via Udacity [20][21]. Universities are beginning to provide courses at this advanced level, too [22].

One might ask why either classroom or online approaches are preferred for a particular audience learning ML. Meta-studies comparing web-based and classroom instruction, for procedural subjects like ML show that they can be equally effective, and that students are equally satisfied with their learning [23] [24]. In the next section, the authors describe how they arrived at a mixed but predominantly online format. Some studies have shown that a hybrid online/classroom approach can provide optimal learning [25] [26]. Finally, a professional factor favoring online education for engineering students is that the life-long learning we all claim to be preparing them for is going to be largely from online resources. [27]

3. RATIONALE

Regarding the first point in our Introduction, we expected that ML would draw students from other majors like a magnet -- upperclassmen in non-CS fields often are faced with "big data" problems where even basic skills in ML could be a benefit. If they only learned ML fundamentals, they still would have the marketable skill of knowing how to interact, as subject-matter experts, with ML professionals. As to the second point, an online course can sweep past the usual issue that people with different backgrounds need to study at different rates, with some requiring more background or explanation to understand topics. This advantage especially occurs if course content is designed explicitly for delivery at varying paces -- for example, skipping "talking head lectures" in favor of student pace-controlled delivery and searchable media. On the third point, we knew that students with majors other than CS would come in with alternative perspectives and interests.

We did not attempt to replace more advanced courses in ML, or ML as part of an upper-division artificial intelligence course. At the schools where our course is taught, one or both of those alternatives exist for teaching theory-intensive and more programming-intensive ML.

We recognized that we needed to find ways for learning to be shared, such as putting students with different majors on the same project team, and having targeted opportunities for those working on various projects to exchange information.

For our course, we chose a slightly unusual "mostly online" delivery variant. We wished for our course to be offered to an on-campus or off-campus group of students. And we wanted not to lose the advantage of having significant social interaction among students, something which can be a problem with online courses [28]. Thus, our course is a hybrid, mostly online, providing a high degree of motivation to contribute to forums, with the option of team projects to stimulate even deeper student interactions. The course is intentionally more online during summer offerings, because of the remoteness of all parties. However, in both this and regular school year sessions, instructor contact with students of at least an hour a week is designed-in. During the school year we could offer a live weekly meeting to answer questions and review project progress for each student or team. During the summer, this would be done remotely.

The value of student collaboration in online courses is well known [29]. As noted, we also included regular synchronized group sessions, live or online, where students would exchange information about progress and problems on their projects. At the same time, ML is a topic for which most students are highly motivated to learn on their own, a factor favoring online learning [30]. We had suitable materials, including a very good book and an easy-to-use software development environment, which could form the basis of a highly accessible online course. The instructors strove to hit a "sweet spot" in their own level of intervention, such as how often they contributed to the forums. We were aware that an online course also needs to promote the sense of self-reliance in its students [31].

The target audience was the same, for the engineering school and the liberal arts school, namely, a mixture of CS and SE majors with students having other majors. The hoped-for distribution of these two groups was 50-50, so that students forming teams would more likely be diverse but also include some programming expertise.

We did not attempt to include automated delivery or assessment features in our course because it is designed for a small audience. We did include portability considerations since it is delivered at two schools.

4. COURSE DESCRIPTION

We created an online course to be taught during the regular school year and as a summer class. It can be considered "hybrid" because it includes a weekly short "synchronous discussion" where students interact, asking questions but primarily exchanging information about their projects. These live sessions would be on-campus during the school year, but done remotely (using a videoconferencing tool called *Zoom*) in the summer.

The course learning outcomes listed in our syllabus are as follows:

- Be able to use the principal machine learning techniques on practical problems.
- Have a working knowledge of the AI-based variants of these.
- Possess a wide view of the places where ML systems vary in their results and in value.
- In a term (semester) project, be able to run different ML algorithms on data of your choice, data in some domain of personal interest to you.
- Exchange ideas and techniques with students in other disciplines.
- Describe ethics issues in ML, and experience applying standards and judgment to these situations.

These outcomes differ from the goals of upper-division CS or math department ML courses, which would be more like a list of the algorithms studied.

We begin the course with ethical considerations for ML, which is then woven into the rest of the course as an underlying theme.

The nature of this course is that it is closely tied to the textbook, *Machine Learning With R*, by Brett Lantz [32]. The book guides students through chapters that cover different machine learning algorithms, step-by-step using the R programming language. All of our course materials supplement this approach. For example, the week's homework is to turn-in snapshots of the activity in a chapter. Lectures are "slide guides" developed to build on a chapter's content, such as describing applications of the ideas and algorithms. Discussion Forums, where students share their own ideas, spring from this build, and give them a chance to interact about applying the content to their own interests.

Thus, our approach was to "scaffold" the learner's path, beginning with readings that allowed them to confirm each step of their learning via practice demonstrated in their book. The book provides practice with introductory theory; this was reinforced by no-fault quiz testing, then by concept extension in the slide guides, and by participation in the forums. Learners were reminded of goals and objectives throughout the course, including regarding the relative value of different ML approaches. Their final goal was solving a real-world problem in their project. In terms of theories of learning, we used a strongly constructivist approach.

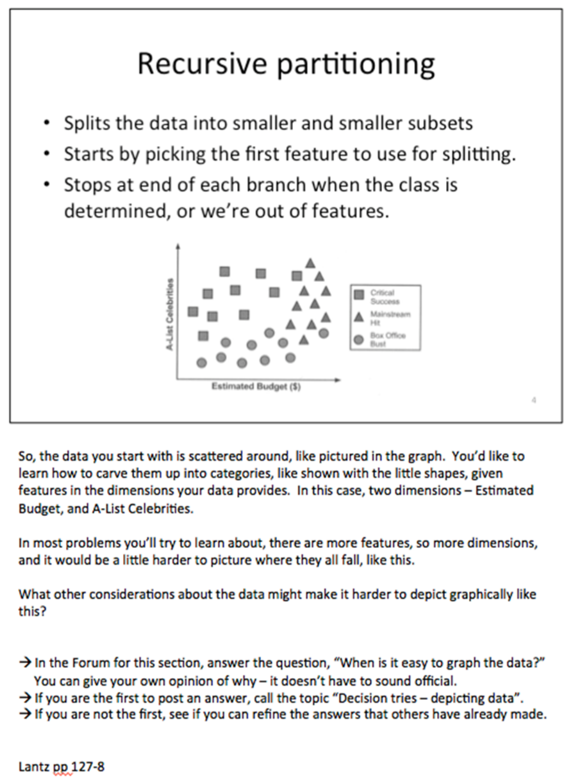
Student teams for projects were self-formed, based on interest in projects proposed by students early in the class. A number of students arrived at the class with these ideas already well in mind, and some had signed-up together based on a project idea. We tried to amend these teams so that students with more CS background were included on each team. This was important, for example, when data cleanup went beyond what was easily covered by tools embedded in the R-Studio environment used. In the course offerings where everyone was remote, students on a team coordinated their work via email and team meetings on Skype or Google Hangouts.

We invented the "slide guide" idea to replace video lectures. Since instructors would have "face time" with students anyway, either in a weekly meeting or online, and because there were two different primary instructors creating the course (and perhaps more to follow), we did not believe a steady diet of one particular instructor talking at the camera would fit our needs. The slide guides are PowerPoint files set up to be shown in notes page mode, with a slide above, typically showing a graphic and key words, and notes below emulating what an instructor might say about that slide to guide students. We pitched the level of this material to assume students had just done the reading, quiz and homework on the same subject -- similar to the transition point in a "flipped classroom." These slide guides have the advantage that students can go through them easily at their own desired pace, in contrast to most videos, they can jump around in them as desired, and they can search for content easily. The volatility of the ML field also made the heavier investment in video creation riskier.

In Figure 1 we see an example Slide Guide page showing the format in which students study it. As on lecture slides, key points are shown with figures in the slide part, while the "guide" beneath gives commentary as one would do in lecturing. Appropriate book pages are referenced, and the notes lines starting with arrows are recommended places for students to make forum contributions or execute related assignments.

In our search for the broadest student base, the two authors collaborated to teach the course at their differing Universities -- One being a liberal arts school and the other an engineering school. The course was developed in fall, 2016, and taken by one student at one school (the liberal arts college) as this occurred. A second "test subject" took the completed course during the winter term at the other school. A full on-campus class was offered at the engineering college in the spring term, 2017. In addition, remote classes took it during summer, 2017, one each at both institutions. Our paper is based on these experiences.

Figure 1 - An example Slide Guide page, this one from the course module on Decision Trees and Rules.



We used a common syllabus asking for a prerequisite of a minimal background in a programming or scripting language (e.g. MATLAB). The course covered topics mainly taken from our textbook including ML algorithms such as Lazy Learning, Naïve Bayes, Decision Trees and Rules, Regression, Neural Nets, Support Vector Machines, Market Basket Analysis, and Clustering with k-means. Further subjects included general concepts of ML, data management and cleaning, evaluating and improving model performance, and advanced topics of ML.

The primary support tool used during this online course was the Moodle course management system. We had the same system at both schools, so we were able to share course materials rather easily. The forums were posted on Moodle, as well. For remote meetings, we used *Zoom*, a state-of-the-art videoconferencing tool [33]. R was chosen as the programming language -- it is open-source and the RStudio development environment that accompanies this has a free version [34].

We took varying approaches to teamwork. We allowed the students to choose whether they preferred to work in a team, with a partner or alone. We did have guidelines for teamwork, but these were somewhat loosely enforced. The textbook comes with sample data files, taken from real studies the book's author did. Some of these datasets have thousands of lines. However, for projects, students found their own datasets, and most of these had tens or hundreds of thousands of lines. Sometimes, they needed to cut back on the data set they started with, because of the limitations of the laptops or other machines they were running these on.

The material on Moodle was organized in alignment with the chapters in the textbook. As sketched above, each Module in the online course has this expected flow of activities, once students got beyond the opening, intro module to the course: a) *Read the next chapter while following the author's programming*, b) *Take the associated online quiz*, c) *Contribute to the forum* (a place to talk about topics related to that Module), d) *Study the slide guides* (with extensive instructor's notes about the subject), e) *Try the "homework" problems*, f) *Try applying the algorithm to your team project data* (thus allowing students to reflect on their own experience trying the algorithms, and also to compare that with how others fared), g) *Contribute to the class's weekly "synchronous discussion".* This same flow was repeated each week to give the course an expected pace. Note in particular that every topic included (g) a short "live instruction" segment, as a part of this flow.

In the first five offerings of the course, we had 55 students, just slightly over half of which were CS or SE majors. The rest came from a variety of other fields, at both schools. Only two students dropped out. All participated in individual or team projects. Most of the students participated fully in the online forums. Almost all succeeded in testing a variety of ML algorithms on their project data, and course completion grades reflected that success.

5. EMPIRICAL STUDY

The main purpose of this study was to solicit, collect and analyze feedback from students who took this course. To this end, a survey was given to a total of 40 students at the engineering school, which was posted on Moodle during the spring 2017, summer 2017, fall 2017 and spring 2018 quarters. In addition, the same survey was given to 13 students during the summer 2017 semester at the liberal arts college. Out of 53 students total, we received 29 responses (i.e. about 73% response rate). The survey was completely optional and anonymous, with no impact whatsoever on the students' grades. The survey was posted during the middle of the semester/quarter and the students were given time to complete it until the last day of classes.

The survey entails a questionnaire with 35 brief-answer questions, which are grouped in five categories: *Motivation* (i.e. why take this elective class), *Course Rating* (i.e. course strengths, weaknesses etc.), *Course Content* (i.e. material covered), *Course Delivery* (i.e. online format, pace, etc.), *Instructor* (i.e. availability, helpfulness, response rate, etc.) and *Learning Assessment* (i.e. exams, quizzes, assignments, team project etc.). All responses were compiled in a single table and are analyzed in the next section. Due to space limitations, we have not included the table in this paper with all the collected data but we can make it available upon request.

5.1 Data Analysis

In this section, we summarize and analyze all gathered responses to each category of questions separately.

5.1.1 Motivation: With this category of questions, we wanted to know why students were interested in taking this elective course. The majority of students acknowledged that ML is a "hot" topic, which will make them more marketable and/or help them with pursuing graduate studies. For instance, one student said "*I would like to go into the sports analytics field after college and Machine Learning using R is very prominent and a required skill in this field*" while another indicated that "*I am interested in pursuing a PhD in Cognitive Science and Computer Science with a focus in Artificial Intelligence. Machine Learning is one of the few AI oriented classes at –this University.*" In addition, some responses showed that this course did change students' appreciation and understanding of ML. Specifically, one student said "*Big time! It changed my understanding/appreciation. Machine learning isn't magic.*"

5.1.2 Course Rating: Here, we asked questions about the overall impression of the course and how it felt different from other courses. Some of the differences mentioned between this course and others include a) its online nature, b) the learning and application of algorithms (i.e. using scripting instead of "traditional" coding), and c) the application of such algorithms on one's own real data of interest. For example, one student said, "*I love that the class has an ongoing project. I also love that as we learn the different algorithms, we are given the chance to apply it to our own sets of data*".

Regarding strengths of the course, students pointed out the timely and "hot" topic nature of ML, the opportunity to apply it to real big data as well the choice of textbook. Among others, here is a response supporting such claim: "*The topic is a growing field, so I appreciate that a topics course is being so timely. The number of algorithms we are learning to implement is fantastic.*"

Some challenges and areas of improvement were suggested including the need to learn more about the R language earlier in the course and improving the coordination of teams in an online course. Some related responses include: "*We kinda jumped right into ML without too much R experience. A more thorough introduction to R would be nice*" and "*While I love the idea of the group project, it is extremely hard to coordinate online with people coming from different areas, jobs and backgrounds*".

5.1.3 Course Content: By asking some pertinent questions, we wanted to know what students thought about the amount and nature of the material covered as well as the tools used during the course. The majority of the responses indicated that the course covers the right amount of material. We also received a very positive unanimous feedback from the students about the textbook. Some comments include "*The textbook is excellent. The walkthrough examples really make everything easy to understand, and it is very easy to read*". In addition, all students (except one) found R and R-Studio to be useful and easy to work with saying things like "*R is extremely user friendly. I have experience programming in multiple languages and R and RStudio have been some of my favorites that I have ever implemented*".

Several students found the chapter on Neural Networks to be fascinating. One student said, "*I found the Artificial Neural Networks (ANN) to be really interesting. It is fascinating that ANNs are based on something that is biological*". The chapters that included math formulas (e.g. Naïve Bayes) were the most challenging to a few students. Finally, we found some good suggestions with respect to additional material including "*I would like to see more information regarding the R programming language itself, either through required or optional lectures, or helpful links online. I think this would help me better prepare my data before apply ML algorithms*".

5.1.4 Course Delivery: We asked questions with respect to the way the material was delivered to the students. Firstly, the majority of the students agreed that this course is well suited to be offered either as a fully online or hybrid style course. It is worth mentioning that a few students were skeptical about going fully online saying "*I like the hybrid style but I think completely online should be an option as I don't think we are getting as much as we could out of the in-class meetings*". Secondly, we received mostly positive feedback regarding the posted material on Moodle. Thirdly, the majority of the students acknowledged the usefulness of the discussion forums saying, "*Discussions can be helpful if several individuals are experiencing a common problem then they can solve things together*". However, we also received comments that brought up some issues with the participation in discussions. For example, one student said, "*There are too many forums. I am confused as to which to use, which ones everyone sees, which ones only my team sees. This needs to be revamped in my opinion*".

Fourthly, several students liked the guided slides saying, "*I think that the slides are a good review after reading the material. They sum up all important topics in the text effectively*". In addition, most of the students liked the overall course pace but they also appreciated the option to follow their own speed saying, "*I think the course is not too slow or fast. I like being able to follow my own pace. The hard deadlines for the exams do keep me on track*". Finally, most students liked the overall idea of using online videoconferencing tools during the course.

5.1.5 Teamwork: Here, we were looking for student preferences and experiences with respect to working as a team. We found that the majority of the students preferred to work with a partner, a few of them in a team, and a small number of them chose to work alone. Here is a related response: "*In this class, I've enjoyed working in a team. We have very different interests and skill sets which I think complement each other for our project*". Regarding the size of the team, some students indicated that it was the right size, whereas a few students thought it was too small/large.

Students that took this course had different majors including Computer Science, Software Engineering, Computer Engineering, Electrical Engineering, Mechanical Engineering, Pharmacy, Business Administration, Finance, Math and Actuarial Science. It is worth mentioning that among the students we had two professors taking this course for credit. More specifically, one Pharmacy professor and another from the Business school. The first professor wanted to use this knowledge to leverage his own research activities on Molecular Biology and the second to enhance his own version of a similar course on Big Data Analytics, respectively.

Overall, students saw the need and appreciated the diversity of majors in their teams saying, "*I like having a mix of majors on the team*". However, coordination was a challenge for most teams. Here is a related comment: "*This is my first exposure to working on a team in an online class. I am concerned* *about having a cohesive final deliverable with so many people on the team*". We received some good suggestions to remedy the coordination of teams including "*I think there needs to be more constant communication between members to insure everyone is on the same page*" and "*Smaller teams and more direction from the instructor as to what the team milestones are throughout the semester*".

5.1.6 Instructor: This section entails questions about how to improve the teacher's effectiveness. The majority of students acknowledged the availability of the instructor(s) and their prompt response rate mentioning things like "*The instructor gives all students the opportunity to ask questions and discuss project ideas/issues, so he definitely facilitates the learning process effectively*". In addition, almost all students (except three with no responses) indicated that their instructor was available when needed. For instance, one student said, "*He is the fastest email responder I have seen*".

5.1.7 Assessment: We asked questions about the way we evaluated and graded the students' learning process relative to the degree of effort they invested in this course. Based on all responses, we calculated that students spent an average of 8-10 hours per week working on this course. The majority of the students liked the assignments and found them easy to complete saying things like "*I think they are well organized and in a good level of difficulty*". A few students however found them easy and said "*Most are just following along with the book, so they were pretty easy. Understanding the theory is more difficult, and applying the algorithms to your project can be very frustrating. That said, I think that this is perfectly okay*". In addition, all of the students agreed that the exams were reasonable and covered all the needed material. For instance, we received comments like this "*Yes, they covered most of what I felt we should have been tested on*".

All of the responses mentioned that students did not have any background in R (some did not have any coding background at all). Therefore, they thought that it would have been helpful if they had some before taking this class. Some responses include "*I haven't taken many programming courses. For me it would be helpful if I had a more solid foundation of programming in R*". Finally, the majority of students thought that the overall grading process was fair saying things like "*It is fair. I have nothing to complain about*". In addition, most students expressed appreciation about having the opportunity to take this course saying things like "*This class is very interesting thank you for bringing this course to our University!*"

5.2 Summary of Data Analysis

This course was taken by a variety of majors, as noted earlier. Our survey data indicates that the majority of those students had a strong interest in learning about Machine Learning (ML) regardless of their major. More specifically, they considered ML a very current and "hot" topic and their responses included comments about how helpful this course can be when looking for a job or pursuing further graduate studies. On the other hand, a few students said that although this course made them appreciate the power of ML, it is not something they would be interested in pursuing further as a career. Based on such results, we will continue advertising this course to a wide range of majors in our Universities.

Some of the course benefits students mentioned in their responses include: a) the flexibility provided by the online nature of the course, b) the opportunity they had to apply various ML algorithms on their own real big data of interest, c) the experience of working within a diverse team with different majors and domain experts and d) the exposure to powerful implementation tools such R and R-Studio.

Some of the main challenges described include the team coordination difficulties as well as the lack of programming background with R. Therefore, based on what we learned from such feedback, we plan to continue offering this course online, with a team project component. However, we will pay closer attention on how to improve the coordination and management of the teams.

Regarding the course content, the majority of the responses indicated that it was "just the right amount" of material. One unanimous point that stood out from all the responses was the effectiveness of the textbook. In addition, we received mostly positive feedback about R (especially R-Studio). Therefore, we plan to continue using the same textbook and cover all twelve chapters during future semesters/quarters. We will also resume using R and R-Studio as the vehicle implementation language and development environment respectively. However, we will introduce more syntax related material on R very early in the course.

In addition, students were challenged by the heavy math/statistical content of some chapters. Although Lantz (textbook author) is doing a great job with presenting such material in a simple way, we will plan to consider other options on how to remedy this. We do not know at this point, what is best. It is worth noting that some students were more skeptical about the online nature of the course perhaps due to their intimacy/familiarity of the face-to-face instruction typically offered by smaller private institutions like ours. More specifically, a couple of students preferred the hybrid style.

Moreover, students provided positive feedback regarding the format of the course and the way material was organized on Moodle. However, we plan to use some useful logistical observations and constructive suggestions to improve such material. In addition, we plan to stay with the same pace of delivering the material and continue using videoconferencing tools such as Zoom for regular and mandatory online meetings.

Overall, students mostly appreciated their experience working in a diverse team while learning and applying various ML algorithms on realistic big data of their interest. At the same time, they admitted that they faced many coordination/scheduling issues, which turned out to be an impediment on their progress and overall learning experience. Therefore, based on such feedback we plan to apply a more rigorous "hands-on" approach when it comes to teamwork. More specifically, we will a) require the posting of weekly progress reports and b) incorporate more mandatory online conference calls with each team. In addition, despite the fact that we received positive feedback about the overall instructor's availability and assistance, we still plan to get more involved from the beginning of the semester, especially with each team's progress.

Another surprising observation (mostly at the liberal arts college) was the varying amount of student participation and engagement to the online discussions, despite the prompts and reminders from the instructor that this was part of their grade. Therefore, we plan to incorporate other creative ways to address this shortcoming.

Finally, students indicated that they invested a reasonable amount of time (i.e. an average of 8-10 hours per week) to learn new material effectively and keep up with the pace of the course. They thought that their evaluation, progress assessment and grading rubrics were all fair and effective. We will address some constructive logistical suggestions that were made including changing the order of the material on Moodle as well as making some of the assignments more challenging. We will keep all the exams and quizzes the same though since everybody provided a solid positive feedback about them. Many students expressed great appreciation upon making such a current course available to their University. Therefore, for the time being, we plan to continue offering such course for elective credit to various majors.

6. CONCLUSIONS & FUTURE STEPS

Our experiences from teaching an online course on Machine Learning (ML) have led to the following observations about such course: a) it appears to be an effective vehicle for attracting non-CS majors and, b) it may be a promising platform for CS students to learn how to collaborate effectively with non-CS majors by working on the same project.

In addition, based on our course participants' positive responses we intend to continue offering such a course as almost fully online during the summers and in as a hybrid style including weekly live sessions during regular quarters/semesters because of the closer proximity of students and the instructor. We will continue promoting it to not only CS and SE majors but also any other major who has an interest in learning about using ML and wishes to make actionable predictions from assessing big data of their own interest. Finally, we are considering several constructive suggestions and comments mentioned in the previous section from our students in order to improve future course offerings.

7. REFERENCES

[1] Bureau of Labor Statistics, "Software Developers." Available: https://www.bls.gov/ooh/computer-and-information-technology/software-developers.htm, and "Employment outlook for engineering occupations." Available: https://www.bls.gov/opub/ted/2016/employment-outlook-for-engineering-occupations-to-2024.htm.

[2] "CS for all," Available: http://www.csforall.org.

[3] *The Feynman Lectures on Physics*. Available: www.feynmanlectures.caltech.edu/.

[4] A. Ng, "Machine learning," Stanford University Open Classroom Course. Available: http://openclassroom.stanford.edu/MainFolder/CoursePage.php?course=MachineLearning.

[5] J. O'Sullivan and R. Zevallos, "Patterns for Online Discussions." Presented at PLoP, 2010.

[6] A. Ghosh and J. Kleinberg, "Incentivizing participation in online forums for education." *EC '13*, pp 525-542.

[7] R. Mihail, Beth Rubin, and Judy Goldsmith. "Online discussions: Improving education in CS?" *SIGCSE '14*.

[8] L. Craig, "Engagement techniques for online education," *eLearn Magazine*, October, 2015. Available: http://elearnmag.acm.org/archive.cfm?aid=2836033.

[9] Wen-Jung Hsin and J. Cigas, "Short videos improve student learning in online education." *JCSC 2013*, 253-259.

[10] M. Krause, M. Mogalle, H. Pohl, and J. Williams, "A playful game changer: fostering student retention in online education with social gamification." *Proceedings of the Second ACM Conference on Learning @ Scale*, pp 95-102.

[11] R. Kizilcec and S. Halawa, "Attrition and achievement gaps in online learning." P*roceedings of the Second ACM Conference on Learning @ Scale,* 2015, pp 57-66.

[12] J. Krichen, "Evolving online learning: Can attention to learning styles make it more personal?"

Proceedings of the 10th ACM conference on SIG-information technology education, pp 8-12].
[13] M. Lee and B. Ferwerda, "Personalizing online educational tools." *Humanize '17*, pp 27-30.
[14] M. Ben-Ari, "MOOCs on introductory programming: a travelogue." *ACM Inroads*, 4(2), June 2013, pp 58-61.
[15] S. Kalathur, "Enriching student experience with student driven content while teaching an online data mining class." *SIGITE '08: Proceedings of the 9th ACM SIGITE conference on information technology education,* pp 125-130.
[16] "Machine Learning," CMU. Available: http://www.cs.cmu.edu/~ninamf/courses/601sp15/lectures.shtml.
[17] "CS 7641: Machine Learning," Georgia Tech. Available: https://www.omscs.gatech.edu/cs-7641-machine-learning.
[18] "Machine Learning," Cornell ML. Available: http://www.cs.cornell.edu/courses/cs4780/2017sp/.
[19] "CSE446: Machine Learning," Washington University. Available: https://courses.cs.washington.edu/courses/cse446/.
[20] "Intro to Deep Learning by Google," Udacity Nanodegree Program, https://www.udacity.com/course/deep-learning--ud730.
[21] V. Vanhoucke. "Teach yourself deep learning with TensorFlow and Udacity," Google Research Blog. Available: https://research.googleblog.com/2016/01/teach-yourself-deep-learning-with.html.
[22] J. Brownlee, "Deep Learning Courses." Available: https://machinelearningmastery.com/deep-learning-courses/.
[23] T. Sitzmann, K. Kraiger, D. Stewart and R. Wisher, "The comparative effectiveness of web-based and classroom instruction: A meta-analysis." *Personnel Psychology*, 59(3), Autumn 2006.
[24] L. Kirtman, "Online versus in-class courses: An examination of differences in learning outcomes." *Issues in Teacher Education,* Fall, 2009, pp 103-116.
[25] G. Black, "A comparison of traditional, online, and hybrid methods of course delivery." *Journal of Business Administration Online*, 1(1), Spring, 2002.
[26] A.P. Rovai and H. Jordan, "Blended learning and sense of community: A comparative analysis with traditional and fully online graduate courses." *The International Review of Research in Open and Distributed Learning*, 5(2), 2004.)
[27] B. Dasarathy, K. Sullivan, D.C. Schmidt, and D.H. Fisher, "The past, present and future of MOOCS and their relevance to software engineering." *FOSE 2014*, Proceedings of the on Future of Software Engineering, pp 212-224.
[28] L. Kirtman, "Online versus in-class courses: An examination of differences in learning outcomes," *Issues in Teacher Education*, Fall 2009, pp 103-116.
[29] S. Young and M.A. Bruce, "Classroom community and student engagement in online courses." *Journal of Online Learning and Teaching*, 7(2), Jun 2011, p 219.
[30] D. Xu and S.S. Jaggars, "Performance gaps between online and face-to-face courses: Differences across types of students and academic subject areas." *The Journal of Higher Education*, 85(5), 2014, 633-660.
[31] V.P. Dennen, A.A. Darabi and L.J. Smith, "Instructor-learner interaction in online courses: The relative perceived importance of particular instructor actions on performance and satisfaction." *Distance Education* 28(1), 2007.
[32] B. Lantz. *Machine Learning with R*, Second Edition, 2015. Packt Publishing.
[33] Zoom US. Available: https://zoom.us.
[34] RStudio. Available: https://www.rstudio.com.
[35] P.B. Andersen, J. Bennedsen, S. Brandorff, M.E. Caspersen, and J. Mosegaard. "Teaching programming to liberal arts students -- a narrative media approach." *ITiCSE'03*, 2003, 109-113.
[36] M.J. Mayo. "Video games: A route to large-scale STEM education?" *Science*, 323, 2 January 2009, 79-82.