Machine Learning for Business Students: An Experiential Learning Approach

Linus Wunderlich
l.wunderlich@qmul.ac.uk
Queen Mary University of London
United Kingdom

Allen Higgins
allen.higgins@ucd.ie
University College Dublin
Ireland

Yossi Lichtenstein
UCD Centre for Innovation,
Technology & Organisation
Ireland

ABSTRACT

This paper reports on the design and teaching of a course on Machine Learning (ML) for business students, focusing on the interaction between content and teaching approach. We demonstrate that the nature of ML technology is amenable to experiential learning and is well suited for business students as users of the technology.

The community-based tools, documentation and datasets enable non-programmers to use and adapt open, public-domain ML examples. Students learn how to select algorithms for specific data and tasks, experiment with hyper-parameters and neural-network structures, evaluate results and interpret their business implications.

We conclude that business students, at least in our school, often have good abstract understanding of computing and may be ready for deeper learning of digital technology. The general-purpose nature of ML makes it suitable for real-world business problems, making it business-relevant technology that should be introduced into business schools.

CCS CONCEPTS

• **Applied computing** → **Education**; • **Computing methodologies** → **Artificial intelligence**.

KEYWORDS

Non-CS Majors; Experiential learning; Machine Learning; Business Analytics and Big Data; Higher Education

ACM Reference Format:

Linus Wunderlich, Allen Higgins, and Yossi Lichtenstein. 2021. Machine Learning for Business Students: An Experiential Learning Approach. In *26th ACM Conference on Innovation and Technology in Computer Science Education V. 1 (ITiCSE 2021)*, June 26–July 1, 2021, Virtual Event, Germany. ACM, New York, NY, USA, 7 pages. <https://doi.org/10.1145/3430665.3456326>

1 INTRODUCTION

The challenge of educating business students as users and managers who wish to deploy the new digital industrial infrastructure [14] has attracted the interest of a variety of academic disciplines, including Information Systems [2], Data Science [4] and Entrepreneurship [12]. Central to this literature is the identification of fundamental

concepts, such as digital innovation [8] and digital disruption [2], or analytical thinking that combines predictive and inferential methods [5], and rendering these concepts into practical curricula. However, pedagogical approaches for delivering these curricula have attracted less attention. While the current pedagogical literature emphasises interactivity, communities, experience and exploration as crucial for today’s learners [7], these notions are not as well developed in the literature on digital business education.

The current paper addresses this gap by reporting on the design, teaching and outcomes of a Machine Learning (ML) course for business students. We are informed by Experiential Learning Theory [16] as the underlying pedagogic approach, and the emerging literature on teaching ML to non-majors in Computer Science or related areas [9, 11, 24, 25]. The resulting course consists of several weeks of online programming classes and exercises (Act in the theory’s terminology), two intensive weeks of daily conceptual lectures (Think) and coding sessions (Act again), culminating in several weeks of work on team projects. The team projects define business problems, implement ML solutions using large datasets (Experience), and present results in written project reports (Reflect).

Addressing the need for ML education research [23], we report on an exploratory study of the effectiveness of the various activities through interviews with students. This study shows that learning experiences from coding and project work were more effective than lectures and essay writing. Students also expressed confidence in their newly gained ability to act as effective clients for ML technology. Confidence was stronger for students with a programming background, yet still significant for non-programmers. Novice programmers gained a practical appreciation for how technical work is performed and overcame their fear of programming. Students with prior programming experience gained a better understanding of the culture of ML and could discuss with confidence different perspectives of the technology.

The following sections describe the design of the course, the actual teaching and a discussion of students’ perceptions of the learning experience.

2 THE COURSE

2.1 Machine Learning

ML, a branch of Artificial Intelligence, is a collection of analytical techniques underlying recent advances in digital technology, such as the widely used face recognition systems and virtual personal assistants. The technology is also seen as central to a new industrial infrastructure [14]. The ultimate scale and scope of ML may eventually rival that of earlier general-purpose technologies like

This work is licensed under a Creative Commons Attribution-ShareAlike International 4.0 License.

ITiCSE 2021, June 26–July 1, 2021, Virtual Event, Germany.

© 2021 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-8214-4/21/06.

<https://doi.org/10.1145/3430665.3456326>

electricity [1]. To use this potential, managers require more technical knowledge about ML as laid out in a recent Sloan Management Review executive guide [18] that forecasts that finance functions will be staffed by fewer people, but all of whom will understand ML. Consequently organizations should reskill employees to consume or use ML technologies, and decision makers moreover should understand which problems ML can and cannot solve.

As a result of these technologies and expectations, programmes aiming to educate business clients for ML are taking shape. For example, MIT Sloan School of Management offers a six-week online course titled Machine Learning in Business. It aims to help business leaders to incorporate ML into their business strategy, and to empower them to ask whether ML applications will benefit particular business problems [18, 22]. Similarly, the London School of Economics (LSE) offers an eight-week online course titled Machine learning: Practical Applications. It focuses on using ML as an aid for analysing data to solve real-world problems and the application of the technology within the business context [19]. These two examples differ somewhat as MIT focuses on strategy and does not require technical knowledge, while LSE prerequisites state that ‘tertiary level statistics and knowledge of a functional or object-oriented language is advantageous’ and teaches to code in R. These and similar business school programmes should be seen also in the wider context of teaching ML to non-majors and the many open questions on effective ways to do so [9, 21, 24, 25].

2.2 Teaching Approach

The ML for business course was designed to be taught to graduate business students over two months, including online programming classes, intensive in-class teaching (prior to the pandemic) and project work. The starting point for the design of this course was a pedagogical approach influenced by Experiential Learning Theory [17]. Kolb’s experiential learning cycle describes learning as a cycle of four states: experiencing, reflecting, thinking and acting (Figure 1).

Figure 1: The experiential learning cycle [15]

Kolb’s experiential learning is a popular and adaptable framework employed extensively for structuring courses and other educational programmes [15], including the teaching of ML to non-majors [9]. The literature suggests organising courses via a series of experience-centred learning cycles that expand in complexity and application; starting with concrete examples rather than abstract principles, and to allow the time and space for all four learning modes to occur. A second central concept is the learner’s learning style which is a habit of learning that is formed when one or more of the learning cycle stages are preferred over others. Eleven styles have been identified and their related measurement tools, known as Kolb Learning Style Inventory. Given differences between learners, Kolb et al. [15] suggest that each learner should engage the learning cycle stages in their own way regardless of the educator’s plan.

2.3 Course Design

The course consists of three interconnected phases: A preparation phase of a month, two-weeks on-campus teaching, and a total of five weeks of group-wise project work. Due to the diverse backgrounds of the students, some of them with very little computer science experience, the course is designed to be self-contained with only minimal prerequisites. The course is mainly based on open source and community-based tools, and thus enables and encourages learning beyond the module’s taught content.

During the crucial preparation phase, the students are required to complete four to six Python programming classes offered by DataCamp [3]. The DataCamp courses (Table 1) do not require prior programming experience and are estimated to take four hours each to complete. Access is provided for free to universities and, as they are delivered via online systems, allow teachers to readily monitor students’ progress. We found these courses easy to follow, useful, even fun, but more importantly they allowed students to learn (or refresh) the foundation programming skills needed to participate in the ML course.

Introduction to Data Science in Python	Python syntax; Pandas data-frames; Matplotlib plotting
Statistical Thinking in Python	Exploratory data analysis; Discrete and continuous variables; Parameter estimation; Confidence intervals; Hypothesis testing
Intermediate Python for Data Science	Matplotlib (again); Dictionaries and Pandas; Logic and control flow; Loops
Python Data Science Toolbox	Functions; Arguments and scope; Lambda functions and error handling; Iterators; List comprehensions and generators

Table 1: Required DataCamp online courses

The second phase of the course is on-campus teaching. The goal of this phase is to familiarize the students with ML practices and algorithms. Namely, students should select correctly algorithms for specific data and tasks, be able to experiment with hyper-parameters and neural-network structures following existing examples, evaluate results and interpret their business implications. We do not expect deep knowledge or sophisticated manipulation of the algorithms. These learning objectives and the assumption that students can make sensible decisions about algorithms and parameters through experimentation are similar to those stated by Fiebrink [9].

The campus teaching phase consists of ten immersive teaching days. Each day includes a morning lecture (10am to 12 noon) that covers conceptual material and introduces a small piece of code applied to a dataset. At the end of the lecture, an assignment that helps to get familiar with the code is given and briefly discussed. The afternoon session takes part in a computer lab (2pm to 4pm) and includes working on the assignment (individually by each student), discussing the answers (full class), reworking the code for a new dataset (in teams), and discussing the initial results (full class). From the second day, we open each morning lecture with a short discussion of the previous day's assignment.

Day 1	Motivation and basic concepts of ML; End-to-End Project example (Chapter 2)
Day 2	Business-related End-to-End Projects; Locating reasonable datasets; Classification (Chapter 3)
Day 3	Business-related datasets; Regression and Optimisation (Chapter 4)
Day 4	Support Vector Machines (Chapter 5)
Day 5	Decision Trees (Chapter 6); ML applications in different companies
Day 6	Neural Networks and TensorFlow (Chapters 9 and 10)
Day 7	Deep Neural Networks (Chapter 11)
Day 8	Convolutional Neural Networks (Chapter 13)
Day 9	Reinforcement Learning (Chapter 16)
Day 10	The ethics of ML; Discussion of projects and essays

Table 2: Intensive teaching structure and corresponding textbook chapters [10]

This structure benefits from ML's underlying logic of identifying a problem, selecting analytical tools and applying them on data, often iteratively and not necessarily in this order. A second characteristic of ML which is also crucial is the public domain availability of thousands of datasets, often with examples of their analysis, in a variety of community web resources, such as Kaggle.com. With these characteristics, the assignment typically reuses the same problem and tools presented during the lecture and asks the students to apply them to a new dataset. The later part of the afternoon session repeats this but asks the students to locate other public domain datasets to fit the problem and tools. The final project is again about the identification of datasets, defining business problems, cleaning and preparing the data and applying a variety of analytical tools.

The content of the course is based on a popular ML textbook by Aurélien Géron [10] aimed at programmers. The textbook focuses on hands-on learning which fits our pedagogic approach. We choose a subset of the subjects, usually a small number of sections from the main chapters, and small segments of the code that is presented in the book. Table 2 summarises the structure of the ten teaching days, referring to the textbook's chapters. In addition to the selected chapters, best-practices of ML are taught and applied, such as data handling, data cleaning and the relevance of data selection biases. With business students as the target audience, particular emphasis was placed on visualising and interpreting results of an analysis. Acknowledging the importance of ethical considerations in ML applications [18, 21], one day's class is devoted to a critical discussion of this topic.

Géron [10] uses the Python programming language and the Scikit-Learn and Tensorflow packages which are free popular tools with strong community support available online. We encourage students to use this support as much as possible to enable learning beyond the course.

The third phase of the course is based on projects that require teams of students to select public domain business-related datasets, define analysis and prediction tasks, apply ML techniques and tools to accomplish these tasks and write detailed project reports¹. The teams are required to work independently and submit the reports which are then graded.

As described above, we designed the course as an implementation of experiential learning cycles. The exact cycles are seen differently by different learners: A student with no programming background acquires concrete experience of programming by doing the DataCamp courses; the lectures during the block-teaching provide abstract conceptualisations of data science; repeated active experimentation occurs in the daily assignments; the project provides a concrete experience of applying ML holistically; and writing the final report requires reflective observation of both the project and the course. In contrast, while a student with a background in programming may not gain as much from the DataCamp courses, their knowledge of statistics will benefit from active experimentation with the daily application of analytic tools to new datasets and business problems; they will learn abstract conceptualisation in the lectures; acquire concrete experience with the project; and benefit from reflection by discussing and writing the report.

3 THE CLASS

The instance of the course described here was taught during July 2019 at the graduate business school of University College Dublin as an elective subject of its MSc Digital Innovation, a twelve-month full-time programme that aims to equip business leaders with the knowledge and skills to succeed in the digital age [6]. We had 33 students from various business or management programmes at the school. The class included 19 male and 14 female students. Fourteen students were from European countries, nine from India and other south Asian countries, eight from China and Taiwan, and two from the United States. Eighteen students were recent graduates with no prior professional work experience, while the rest had between one and ten years of experience. Seventeen of the students had business

¹A list of projects is available online <https://github.com/LWunderlich/MLforBusiness>

or management degrees, eleven held various engineering degrees including Computer Engineering, Computing, Telecommunication and Electronics, and the remainder held degrees in the Arts and Humanities.

The course was taught roughly as planned (Table 2). The students were asked to copy code examples, apply them to different datasets and change some of the details as necessary. Using the browser-based Jupyter development environment made the work interactive and streamlined web-search, copy and paste. Although most of the students had completed the DataCamp courses (Table 1), we realised that further simplification of our own example code was needed as we ended up dedicating more time than planned explaining it in class.

The group projects that form the third phase of the course were completed by teams of three. During class we discussed the datasets (typically from Kaggle [13]), the business problems, and appropriate analytical tools. Example problems included: the identification of amenities that influence rating and attractiveness of 74,000 AirBnB properties; fraud detection in 280,000 credit card transactions that included 492 fraud cases; and the identification of good/bad drivers with seven million time-stamped sensor-recordings in commercial trucks.

We did not encounter the applied or conceptual problems reported by others (e.g. in a course on business analytics for law [23]) where students had difficulty understanding notions like parameter tuning or overfitting. The main difficulty our teams encountered was in formulating a holistic business response; typically, students appreciated the value of predicting a business variable such as identifying customers who may respond positively to a marketing campaign, however they tended to overlook the knock-on managerial consequences, for example, of allocating marketing and operational resources according to the prediction.

4 EXPLORATORY STUDY

Addressing the need for ML education research [21], we conducted an exploratory study of students' perceptions in order to assess the effectiveness of our approach. Our main goal was to get an impression how well the hands-on method worked for programmers and for non-programmers.

4.1 Method

A detailed consent form for students to opt-in to the research was presented by a member of faculty not involved in the course. Students completed an online questionnaire based on Experiential Learning Theory². Given a number of constraints (limitations of research scope, timing and class size) and preferring to allow time for interviews, we did not attempt to employ the Learning Style Inventory questionnaire as it takes around thirty minutes for respondents to complete [17].

Following the submission of final reports, but prior to the publication of grades, one-to-one interviews were conducted with approximately half of the class. The interviews were conducted by the MSc programme director (second author) who was not directly involved in teaching or grading the course.

²The questionnaire is available online at <https://github.com/LWunderlich/MLforBusiness>

4.2 Perceived Learning Experience

In the following we analyse responses from the online questionnaire relating to the effectiveness of learning activities and the overall effectiveness of achieving the learning outcomes.

In the first set of questions, students were asked to reflect on the course and rank the effectiveness of the various activities. These activities were described as the conceptual lectures during the mornings, coding during the afternoons, the project work, including data analysis and programming and discussing and writing the final report. The twenty answers cover eleven different ranking orders, almost half of the possible orders; namely, we see even in this small group considerable variety. The most popular preference ranking orders are Project, Coding, Lectures, Essay; and Coding, Project, Lectures, Essay with three students for each of these orders. Table 3 presents the number of students ranking each of the activities.

	Most effective		Least effective		
	Rank=1	Rank=2	Rank=3	Rank=4	Mean
Lectures	2	2	9	7	3.05
Coding	9	4	2	5	2.15
Project	8	7	5	0	1.85
Essay	1	7	4	8	2.95

Table 3: Students' ranking of activity effectiveness (n=20)

These rankings are correlated with the ranking of effectiveness of conceptual activities in classes taken in the past: lectures (spearman=0.60, $p<0.01$) and essays (spearman=0.59, $p<0.01$); however, for hands-on activities the correlation is statistically insignificant. There is also no other variable that explains the ranking of Table 3. For example, when the rankings are cross-tabulated contrasting students with and without programming experience, the numbers of students are similar: 5 students with programming experience and 4 students without such experience rank coding as the most effective activity, and 3 students with experience and 2 students without experience rank lectures as least effective.

Students who ranked either coding or the project as most effective, expressed both positive and negative explanations. For example, a positive explanation:

“Coding during the afternoons [was] most effective because for me personally it works best to learn with trial and error. The alternation with the lectures and the hands-on coding was very beneficial to my learning.”

Another example is the following somewhat negative explanation:

“Lectures gave a good basic understanding prior to coding classes. I enjoyed the initial project work, however, once the course had ended, we struggled with our coding.”

The group project was expected to bring all of the learning together. It linked students understanding of the concepts with practical outputs and hopefully began the process of embedding them within the ML community of practice.

“The exercise of having to actually complete the essay and attempting to make the models work with the dataset [were] the best means of learning. . . What was also invaluable was the Kaggle resource which had examples of other people who solved similar problem and similar dataset.”

However, the intensive workshop style of teaching could be overwhelming. It was particularly challenging for novice programmers who had not engaged fully with the advance preparation in DataCamp.

“I found it very difficult to understand most of the lectures. The quizzes were very good but in such a short time frame I felt rushed with the learning. Too little time for the project and report.”

“As I didn’t have any experience of ML before, it made it difficult for me to understand during the morning lectures. I feel the module should be transferred either into Semester Two or could be divided into two parts in two semesters.”

One student thought that coding was the least effective learning activity:

“I think the hands-on session provided us opportunities to make more sense of the concepts. However, those students with less or none programming experience could not really make sense of the code. The daily exercise did help a lot as we discuss the questions together with our classmates. [But] I rank the lecture as the first since the concepts are basic.”

	Agree		Disagree			
(Outcome confidence)	(5)	(4)	(3)	(2)	(1)	mean
Apply methods	6	11	2	2	0	4.1
Apply to problem	8	10	1	1	0	4.25
Effective client	9	7	1	3	0	4.1
Effective manager	5	8	5	2	0	3.8

Table 4: Students’ confidence in learning outcomes (n=20)

The second set of questions asked about the effectiveness of the course in delivering its overall learning outcomes. Students were asked to agree or disagree with statements about their confidence in various tasks. These are: applying ML methods, produce results and interpret them; apply methods to a ML problem, for example, a problem from Kaggle; effective client for a team working on ML problems; and effective manager for a team working on ML problems. The answers are summarised in Table 4 and generally show that the students are quite confident. Some of the variability in this table is related to the student programming experience. For example, reacting to the statement about their confidence to apply ML methods to problems, eight students with programming experience strongly agree, while eight students with no programming experience somewhat agree. Similar slight shifts between the two groups are observed for all the course effectiveness questions.

In follow-up interviews students reflected on the programming experience and what they learned from the course overall.

“I don’t know if I feel confident or not enough to add it to my CV or not. What I know now about ML and AI was definitely an eye opener. I hadn’t really realised before how technical work is performed. I would have an idea but not practically, now I do.”

Many of the novice programmers commented on their newly acquired programming skills.

“Writing reports is the main method in other modules, but this class, I learned a new skill. I used Python to a deal with problems. And I did write some code, something that I’ve never think I could do before.”

Many acknowledged an appreciation for the process and application of Data Science and ML techniques.

“The project really helped make sense of things. The project tied it all together. After the course, I guess I’ve kind of gotten over the fear of programming. I also got a good overview of ML in business. The project was really good way to get practical experience because it was real data.”

And for some the skills and knowledge gained had a significant career impact.

“After this class I got a lot of confidence basically, overcoming my fear of programming. Now I am actually planning to learn SQL as well. In the job market there is a lot of expectation from management students to have some coding knowledge so this course in ML has given me all I know. The Python is a skill to add to my CV which is quite good. From what I learnt in class I landed an interview with Google, after it I also got an interview from Amazon.”

Students with prior programming or statistics experienced a slightly different range of learning outcomes. Some saw the class as an opportunity to develop ML as new area of professional competence.

“I felt that ML is an important area, but I felt like it would be really complicated, but the class has shown me both how useful it is but also that I can actually do it. I feel more confident. I feel like I could handle, you know, take part in a project, work with the data part of a project. I find I’m a lot more confident in being able to talk to people in industry now, and I did have conversations with a couple of friends from back home. They’re working on the coding aspects, but I couldn’t talk with them before about these concepts. Now I recognise what they are talking about. Take, for example, when they say that training data is not enough. Now I understand why.”

Others with experience appreciated the opportunity to refresh their knowledge and apply new techniques and skills to current challenges.

“I’m working in a really large company and we have what we call Data Lakes. ML is directly applicable to a problem we have right now, today! So, I’m applying what I learnt in class to a project inside the company.

Well, you know, for me I’ve [now] got a huge appreciation for the principles of ML. I really can see how it applies in reality, today. I think it’s an incredible shift to move from packaged analysis in the sort of business applications we buy. In this class we do it in a hands-on way, with these great open technologies.”

The learning gained by experienced technologists was qualitatively different to that of the novices. Of these some felt they had attained breakthrough knowledge gains.

“I think I gained a comprehensive overview of ML techniques. For me a breakthrough realisation moment came, probably when we got to the neural network section of the course, which I hadn’t encountered before that stage I felt I was really learning new things.”

Or they mentioned a growing affinity with the culture, and confidence in using the language of data science.

“Now I think I have a very good grasp of the concepts of ML, The language of ML. So, I think by the end of the course, we were definitely getting into the culture of ML. And from what I’ve learned I now definitely have a, you know, confidence, that I can go into the ML domain and talk about ML from various different perspectives. Maybe not as in depth as the lectures would have hoped but competent.”

5 DISCUSSION AND CONCLUSIONS

This paper reports on the design and teaching of a course on ML for business students, focusing on the interaction between content and teaching approach. We demonstrate that the nature of ML technology is amenable to experiential learning [15, 16] and that it allows to address both technical issues and business problems in an engaging way. Our experience fits the wider advice about teaching ML to non-CS majors [23, 24], namely that data integral to the topic (public domain business data in our case) is central to the success of such teaching.

An online survey completed immediately after the course shows that the effectiveness of “acting and experiencing”, namely coding and project, are ranked higher than the effectiveness of “thinking and reflection” activities which are lectures and essay writing (Table 3). This conclusion holds for both students with and students without prior programming experience. The importance of acting and experience is also consistent with Fiebrink’s report on teaching ML to creative practitioners [9]. The survey also shows that students have confidence in their ability to apply ML methods to appropriate problems and to be effective clients to this technology (Table 4). For the overall effectiveness of the course, confidence is stronger for students with prior programming background, but is still significant for non-programmers.

Somewhat stronger confidence in the effectiveness of the course is demonstrated in the interviews. Several of the students with no prior programming experience claimed gaining a greater understanding for how technical work is performed. They acquired an appreciation for what programming was and some talked of overcoming their fear of programming.

Different learning outcomes were mentioned by students with a programming background. They talked about personal transformations of understanding and gaining deeper knowledge. For example, better understanding of ML and practical approaches to analysing data, and that they better understand the distinctive open culture of the ML community. They can now discuss with confidence the different perspectives of ML applied to real business problems.

Given the limited number of students and time constraints, our exploratory study does not include in-depth measures of the individual learning style and preferences [17]. While we did ask about individual preference of learning activities prior to the course, these answers are not a substitute or valid measure of individual learning style. Similarly, the fact that the ranking of the effectiveness of conceptual activities are significantly correlated, but hands-on activities are not, cannot be interpreted with confidence and should be only a base for further research.

Further insights from the interviews with students related to the effectiveness of current pedagogic techniques described in the general pedagogic literature [7]. Students found the interactive approach of the DataCamp courses [3] an appealing way to prepare for this kind of intensive learning experience. Students also gained an appreciation of the open, community-based culture surrounding the tools and datasets as exemplified by Python, Scikit-Learn, Keras, TensorFlow, Jupyter and other open technologies. Many students valued the online Q&A forums of communities like Kaggle [13] that enabled them to explore and experiment with different technical solutions and to solve problems. A practical conclusion from these interviews is that the two-week block teaching was too intensive for students of all backgrounds and that more time is required for the second phase of this course.

Following the experience of designing and teaching this course, we offer two conclusions. Firstly, business students, at least in our school, are prepared for deeper learning of digital technology, as they usually understand the abstract notions of computing and their applications in finance, marketing, supply chain, project and innovation management. The various misleading anthropomorphic notions of computing [9], the difficulties in comparing and tuning models [23] were not visible in our class. For example, an abstract notion of a pipeline [9] was not needed, as similar sequences of operation are common in management disciplines. Namely, although business students are non-majors, many of them are not very remote from Computer Science majors in abstract understanding of computing. Secondly, ML offers a good platform for introducing business-relevant digital technology knowledge and skills into business schools. Strong grounding in information technology has never been more important to business students [8], however teaching of either programming or specific system categories, such as ERP or CRM, has in the past been deemed either too trivial or too specific or both. The large body of business-relevant public domain datasets and the open, democratic [20] easy to copy and manipulate nature of ML programming makes it neither trivial nor very specific. The data and tools allow learners to experience the new digital industrial infrastructure [14] which underlies marketing, supply chain, production, accounting and other business and management areas.

REFERENCES

- [1] Erik Brynjolfsson and Tom Mitchell. 2017. What can machine learning do? Workforce implications. *Science* 358, 6370 (Dec. 2017), 1530–1534. <https://doi.org/10.1126/science.aap8062>
- [2] Thomas Case, Geoffrey Dick, Mary J Granger, and Asli Y Akbulut. 2019. Teaching Information Systems in the Age of Digital Disruption. *Journal of Information Systems Education* 30, 4 (2019), 287–297.
- [3] DataCamp. 2020. Data Scientist with Python Track | DataCamp. <https://learn.datacamp.com/career-tracks/data-scientist-with-python>
- [4] Richard D. De Veaux, Mahesh Agarwal, Maia Averett, Benjamin S. Baumer, Andrew Bray, Thomas C. Bressoud, Lance Bryant, Lei Z. Cheng, Amanda Francis, Robert Gould, Albert Y. Kim, Matt Kretchmar, Qin Lu, Ann Moskol, Deborah Nolan, Roberto Pelayo, Sean Raleigh, Ricky J. Sethi, Mutiara Sondjaja, Neelesh Tiruvilumala, Paul X. Uhlig, Talitha M. Washington, Curtis L. Wesley, David White, and Ping Ye. 2017. Curriculum Guidelines for Undergraduate Programs in Data Science. *Annual Review of Statistics and Its Application* 4, 1 (March 2017), 15–30. <https://doi.org/10.1146/annurev-statistics-060116-053930>
- [5] David Donoho. 2017. 50 Years of Data Science. *Journal of Computational and Graphical Statistics* 26, 4 (Oct. 2017), 745–766. <https://doi.org/10.1080/10618600.2017.1384734>
- [6] University College Dublin. 2021. MSc in Digital Innovation, UCD Michael Smurfit Graduate Business School, University College Dublin. <https://www.smurfitschool.ie/programmes/masters/mscindigitalinnovation/>
- [7] Rebecca Ferguson, Sarit Barzilai, Dani Ben-Zvi, Clark A Chinn, Christothea Herodotou, Yotam Hod, Yael Kali, Agnes Kukulska-Hulme, Haggai Kupermintz, Patrick McAndrews, Bart Rienties, Ornit Sagy, Eileen Scanlon, Mike Sharples, Martin Weller, and Denise Whitelock. 2017. *Innovating pedagogy 2017*. The Open University, Walton Hall, Milton Keynes, MK7 6AA, United Kingdom.
- [8] Robert G Fichman, Brian L Dos Santos, and Zhiqiang Zheng. 2014. Digital innovation as a fundamental and powerful concept in the information systems curriculum. *MIS quarterly* 38, 2 (2014), 329–A15. Publisher: Management Information Systems Research Center, University of Minnesota.
- [9] Rebecca Fiebrink. 2019. Machine learning education for artists, musicians, and other creative practitioners. *ACM Transactions on Computing Education (TOCE)* 19, 4 (2019), 1–32.
- [10] Aurélien Géron. 2019. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media.
- [11] Piper Jackson. 2019. Casting a Wider (Neural) Net: Introducing Data Science and Machine Learning to a Larger Audience. In *Proceedings of the Western Canadian Conference on Computing Education*. ACM, New York, NY, USA, 1–2.
- [12] Christopher G Jones and David Liu. 2019. Approaches to incorporating it entrepreneurship into the information systems curriculum. *Journal of Information Systems Education* 28, 1 (2019), 4.
- [13] Kaggle. 2020. Kaggle: Your Machine Learning and Data Science Community. <https://www.kaggle.com/>
- [14] Brenna Killmeyer and Jason Sniderman. 2019. Navigating a New Industrial Infrastructure. <https://sloanreview.mit.edu/article/navigating-a-new-industrial-infrastructure/> Accessed 30 December 2020.
- [15] Alice Kolb and David Kolb. 2018. Eight important things to know about the experiential learning cycle. *Australian educational leader* 40, 3 (2018), 8. <https://learningfromexperience.com/downloads/research-library/eight-important-things-to-know-about-the-experiential-learning-cycle.pdf>
- [16] Alice Y Kolb and David A Kolb. 2005. Learning styles and learning spaces: Enhancing experiential learning in higher education. *Academy of management learning & education* 4, 2 (2005), 193–212. Publisher: Academy of Management Briarcliff Manor, NY 10510.
- [17] David A Kolb. 1985. Learning-style inventory: Self-scoring inventory and interpretation booklet.
- [18] MIT. 2020. The AI & Machine Learning Imperative. <https://sloanreview.mit.edu/series/the-ai-machine-learning-imperative/>
- [19] London School of Economics and Political Science. 2021. Machine Learning: Practical Applications. <https://www.lse.ac.uk/study-at-lse/Online-learning/Courses/Machine-Learning-Practical-Applications/Home.aspx>
- [20] Daniel Rock. 2019. Engineering Value: The Returns to Technological Talent and Investments in Artificial Intelligence. *SSRN Electronic Journal* (2019). <https://doi.org/10.2139/ssrn.3427412>
- [21] R Benjamin Shapiro and Rebecca Fiebrink. 2019. Introduction to the special section: Launching an agenda for research on learning machine learning.
- [22] MIT Sloan and MIT CSAIL. 2021. Machine Learning In Business. <https://executive-education-online.mit.edu/presentations/lp/mit-machine-learning-online-short-course/>
- [23] Elisabeth Sulmont, Elizabeth Patitsas, and Jeremy R Cooperstock. 2019. What is hard about teaching machine learning to non-majors? Insights from classifying instructors' learning goals. *ACM Transactions on Computing Education (TOCE)* 19, 4 (2019), 1–16.
- [24] Thomas Way, Mary-Angela Papalaskari, Lillian Cassel, Paula Matuszek, Carol Weiss, and Yamini Praveena Tella. 2017. Machine learning modules for all disciplines. In *Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education*. ACM, New York, NY, USA, 84–85.
- [25] Xiaochun Yang and Jiawei Liang. 2018. Machine Learning Online Education Experience for Non-technical People. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. ACM, New York, NY, USA, 1075–1075.