Maps for an Uncertain Future: Teaching AI and Machine Learning Using the ATLAS Concept

Thilo Stadelmann[*], [1] **Christoph Würsch**[2]
[1]ZHAW School of Engineering, Obere Kirchgasse 2, 8400 Winterthur ZH, Switzerland
[2]OST Institute for Computational Engineering, Werdenbergstrasse 4, 9471 Buchs SG, Switzerland
stdm@zhaw.ch, christoph.wuersch@ost.ch

Abstract

Every student seems to have an opinion on AI. This is arguably due to the fact that its assumed topic, "intelligence", is deemed to be one's very own possession, and hence an area of every individual's expertise. To turn this initial motivation into a stable foundation for life-long learning and working, the opposite of ready-made solutions must be made available by an educator. Additionally, the current hype needs to be exposed to thoroughly assess the real potential (for better or worse) of the technology. Hence, students need to be given an ATLAS: a collection of *analog maps* to the field of AI that (a) give an overview in this highly dynamic and complex environment; that (b) highlight the beauty of certain places therein; that however (c) don't restrict themselves to advocating only a single path. This paper outlines the concept behind the design and teaching of said "cartographical material" and evaluates it in the context of two curricula: an introduction to AI for undergraduate students of computer science, and an introduction to machine learning in an interdisciplinary masters in engineering programme. It further contributes a model assignment for teaching a fundamental lesson on AI: leveraging the right algorithms pays off way more than leveraging human insight. All course materials including slides, assignments and video lectures, are freely available online.

1 Introduction

Gerhard Mercartor's original "Atlas sive Cosmographicae Meditationes de Fabrica Mundi et Fabricati Figura" combined maps and associated explanations of the known world (Mercator 1595), which were used by generations to explore, push boundaries, and further trade and development (Schneider and Brakensiek 2015). Today, more than 400 years later, similar brave acts are expected from students as they enter their careers in a time that awaits nothing short of a digital disruption (Skog, Wimelius, and Sandberg 2018). The core of the disruptive potential through the "digital transformation" is provided by technological developments, foremost by artificial intelligence (AI) and its currently hottest branch, machine learning (ML) (Aoun 2017). Hence, future generations of professionals need the analogon to what Mercator gave to his contemporaries: an atlas to the world of AI. This is especially needed since teaching AI as a foundational aspect of technical education was until recently largely neglected in curricula according to surveys (Dessimoz, Köhler, and Stadelmann 2015), and the intermediately developed data science courses (Stadelmann et al. 2013) do not cover the same terrain.

In this paper, we report on the design of the ATLAS concept for teaching AI and ML and its implementation within two courses: "Artificial Intelligence 1" for undergraduate students of computer science, and "Machine Learning" within an interdisciplinary graduate programme in engineering. We evaluate results from teaching both for several terms qualitatively (based on relevant student feedback) and quantitatively (comparing exam results of individual tasks with the overall achievement of educational objectives). The quantitative results are based on the spring term 2020 that, due to moving fully online as a result of the COVID-19 situation, also sheds some light on the topic of online examination. An additional contribution of our experience report is a specific model assignment that is regularly highlighted by our students as being extraordinarily helpful. It is—together with all other course material including slides, labs and video recordings of all lectures—freely available online.

The rest of this paper is organized as follows: Section 2 provides the context for the conceptual discussion by introducing two specific curricula for AI and for ML and an elaborate model assignment as an example of their content. Section 3 then introduces the ATLAS didactic concept as their underlying foundation based on four key dimensions. Section 4 presents and discusses lessons learned we draw from the evaluation of the specific courses in comparison with the conceptual ideas. Section 5 concludes the paper with an retrospective on the use of our analog maps in teaching AI and ML, and the role of technology in relation to humans.

2 Context

Before introducing the abstracted didactic concept, we provide context for its choices and lines of discussion by briefly outlining the two concrete modules that have been designed based on the ATLAS concept. They shed light on the educational environment that forms a starting point for the conceptual considerations, and make the outcome in terms of educational objectives and syllabi tangible.

[*]Fellow, European Centre for Living Technology, Venice, Italy.

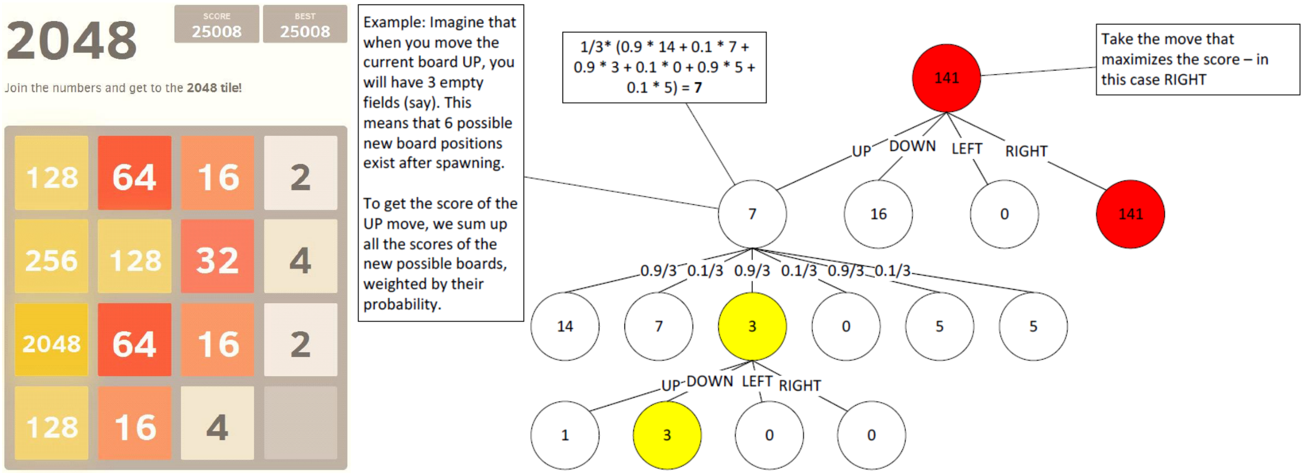

Figure 1: Left: screenshot from the 2048 number puzzle; goal of the game is to reach a 2048 tile by joining adjacent tiles of similar value through consecutive up/down/left/right movements of the whole board. Right: exemplary search tree as processed by Expectimax for a fictional board configuration, excerpted from the assignment description.

Topic (duration)	Key questions	Methods (excerpt)	Practice
Introduction to AI (2 weeks)	What is (artificial) intelligence?	The concept of a rational agent	AI for SciFi readers: formulating one's own opinion as a reply to a futuristic essay (Urban 2015)
Search (3 weeks)	How to find suitable sequences of actions to reach a complex goal?	Uninformed and heuristic search, (Expecti-)Minimax, constraint satisfaction problem solvers	AI for the game "2048": controlling a number puzzle game (see Section 2.2)
Planning (3 weeks)	How to represent knowledge that facilitates reasoning?	Propositional and first order logic, knowledge engineering and reasoning, Datalog for big data	AI for a dragnet investigation: finding potential fraudsters using inference over communication meta data
Supervised ML (3 weeks)	What is learning in machines? How to learn from examples?	From linear regression to decision trees and state of the art ensembles	AI for bargain hunters: data mining a dataset of used cars
Selected chapters (2 weeks)	What is the current hype about? How does AI effect society? How could society react?	Primer on deep neural networks and generative adversarial training for image generation	SciFi revisited: formulating a reply to the blog post from the first week

Table 1: The curriculum of the AI course, spanning a 14-weeks semester with 2 lectures and 2 labs (45 minutes each) per week. On successful completion, the students are awarded 4 ECTS, meaning they have invested ca. 120 hours into the coursework (i.e., they spent roughly twice the amount of time in self-study as in class, with most of this time invested into the lab assignments).

2.1 The AI course

"Artificial Intelligence 1"[1] is a practice-oriented elective course based on (Russell and Norvig 2010) in the final year of a B.Sc. computer science programme at a university of applied sciences, encompassing selected foundations of AI and ML and aiming at hands-on problem-solving competency for everyday software challenges. It is geared towards students who have a general curiosity for smartness in software but no aspirations towards research. Most of them, when starting the course, look forward to a career as software engineers, with some thinking about becoming data scientists or about further interdisciplinary studies in areas like information engineering, speech processing, computer vision or robotics.

The superior learning objectives are defined as (a) knowing the breadth of AI and particularly ML problem solving strategies, thus identifying such challenges in practice and developing corresponding solutions on one's own; (b) being able to explain the discussed algorithms and methodologies, thus being enabled to transfer the respective knowledge to the real world. The corresponding syllabus is depicted in Table 1. It is structured in five phases based on the main approaches to AI (symbolic and sub-symbolic) and an elaborate parenthesis dealing with overarching concerns.

2.2 Content example: AI model assignment

Summary, topics and audience. The lab "2048 game playing agent" (see Figure 1) is a four-week assignment at the beginning of the AI course to be approached by pairs of two students[2] . It is based on the game "2048" by Gabriele Cirulli[3] and covers the topics of rational agent development and adversarial search (heuristic search, Expectimax algorithm).

[1] See https://stdm.github.io/ai-course/

[2] See http://stdm.github.io/downloads/courses/AI/P02_2048.zip

[3] See https://play2048.co/

Topic (duration)	Key concept	Cross-cutting concerns	Methods (excerpt)
Introduction (2 weeks)	Convergence for participants with different backgrounds	Hypothesis space search, computational learning theory	VC dimensions; ML from scratch: implementing linear regression with gradient descent purely from formulae
Supervised learning (7 weeks)	Learn from labeled data	Feature engineering, ensemble learning, debugging ML systems	Cross-validation, learning curve & ceiling analysis, SVMs, bagging, boosting, probabilistic graphical models
Unsupervised learning (3 weeks)	Learning without labels	Probability and Bayesian learning	Dimensionality reduction, anomaly detection, k-Means and expectation maximization (EM)
Special chapters (2 weeks)	Reinforcement learning	-	AlphaZero

Table 2: The 3 ECTS curriculum of the ML course, spanning a 14-weeks semester of 2 lectures and 1 lab per week.

The assignment is divided into two distinct phases, each with the task of developing an artificial player that controls the game to win, but different strategies and learning objectives.

Phase one is about taking one's software development and problem solving skills, together with one's understanding of the game after a few hours of playing, and implement an agent ad hoc by designing useful heuristics (links to the literature and online forums are provided, where ideas abound). The usual experience of a student after phase one is that it is very difficult and not overly successful to try encoding one's own strategies purely ad hoc (and that it is impossible to exhaust the knowledge on the web and in the literature without a clear idea of how to conceptually approach the problem).

Phase two introduces the conceptual framework of adversarial heuristic search and the Expectimax algorithm. Students can leverage on their developed ideas of a heuristic function here, but thanks to the look-ahead provided by the search, reach scores usually an order of magnitude higher than their previous results (or manual play). This drives home the point that mapping the problem at hand to the best fitting conceptual/algorithmic approach from the literature pays off way more in AI than investing many hours of manual labor. It also reinforces Sutton's "bitter lesson" that leveraging compute through search is usually the smartest thing one can do (Sutton 2019).

Strengths, weaknesses and difficulty. This assignment's biggest strength is its addictiveness: students regularly report that they got so caught in the task that they worked through nights and weekends on the hunt for a better high score. This motivation carries over to trying other methods than search: we have seen deep reinforcement learning (RL) approaches developed during these four weeks, despite them not being part of the curriculum. Another strength is its accessibility: students on any skill level find something worthwhile to work on, be it improving their programming skills, understanding a recursive algorithm, or tapping into previously unknown scientific literature to understand RL.

A weakness of the assignment is its dependency on the pace of the corresponding lecture: it helps the educational objective of phase one that the students don't know search algorithms yet (so that they really try ad hoc solutions); it is however necessary for phase two that they are acquainted with adversarial search, so that the schedule of the lectures and labs needs to be tightly synced. Another weakness is that much of the initial motivation comes from the students knowing the 2048 game already from its viral history on the web; this effect is dying away over the years.

Dependencies and variants. Platform-independent code templates in Python are given for all technicalities like interaction with the game, so that students can focus purely on implementing the agent function $next\_move = f(current\_board)$. Students with a good command of any imperative programming language regularly take this as their first attempt to Python programming. Content-wise, the assignment is preceded by a general introduction to the field of AI as well as to search algorithms in the order of one lecture each. Before entering phase two of the assignment, students need to get an introduction to adversarial search and the Expectimax algorithm. An easy variation of the assignment would be to exchange the game by another version that might be more fashionable (and hence able to evoke interest with students) in a few years.

2.3 The ML course

"Machine Learning"[4] is an elective course in an interdisciplinary joint graduate program on engineering of different universities of applied sciences. It builds upon basic knowledge in math, programming and analytics/statistics as is typically gained in respective undergraduate courses of diverse engineering disciplines and draws on a respective diverse audience with heterogeneous backgrounds. The module teaches the foundations of modern machine learning techniques in a way that focuses on practical applicability to real-world problems. The complete process of building a learning system is considered: formulating the task at hand as a learning problem; extracting useful features from the available data; and choosing and parameterizing a suitable learning algorithm. The syllabus highlights cross-cutting concerns like ML system design and debugging (how to get intuition into learned models and results) as well as feature engineering, aspects typically cut short in previous courses these students took that touched on learning algorithms; covered algorithms include (amongst others) Support Vector Machines (SVM) and ensemble methods.

The corresponding educational objectives are designed as follows: (a) students know the background and taxonomy of machine learning methods; (b) on this basis, they formulate given problems as learning tasks and select a proper learn-

[4] See https://stdm.github.io/ml-course/

ing method; (c) students are able to convert a data set into a trained model by first defining a proper feature set fitting for a task at hand; then they evaluate the chosen approach in a structured way using proper design of experiment; they know how to select models, and "debug" features and learning algorithms if results do not fit expectations; finally, they are able to leverage on the evaluation framework to tune the parameters of a given system and optimize its performances; (d) students have seen examples of different data sources and problem types and are able to acquire additional expert knowledge from the scientific literature. The curriculum, depicted in Table 2, spends most time on first principles and illustrates them by specific learning algorithms, and is structured four-fold with an introduction followed by supervised, unsupervised and reinforcement learning.

3 The ATLAS concept

Our concept for teaching AI and ML is based on the well-known "AIMA" text book (Russell and Norvig 2010) (the much welcomed updates to the recent 4^{th} edition from April 2020 have not yet been adopted; they include a more timely selection and framing of the contents that has partly been anticipated by our curriculum design). What distinguishes it from the many other adoptions is an end-to-end focus on the applicability of the taught foundations that should lead to successful transfer into personal problem-solving skills and professional practice.

We choose to introduce the ATLAS concept by means of four key aspects: first, we establish our *understanding of AI and ML* as foundational parts of computer science and the *implications* this has on the teaching mode, in Section 3.1. This blends over into the *core attitudes* we seek to convey with our curricula besides technical content (see Section 3.2). We then discuss the *didactic design* to most efficiently enable our educational objectives in Section 3.3 and finally how this concept enables and motivates different *career paths* for our students in Section 3.4.

3.1 AI as a foundational subject

Our educational concept situates AI (and within it ML) as one of the five pillars of the discipline of computer science. As such, it shares specific characteristics with other foundational subjects that need to be accounted for in teaching (e.g., thorough establishment of basic ideas), but holds a peculiarity: different from foundations such as algebra, AI already builds upon a body of knowledge from computer science. This implies a later slot in respective programmes, and hence more mature students that can better judge the impact of AI on other aspects of their profession or society as a whole. Within this context, our concepts highlight the following three aspects of effective foundational teaching:

Canonization. The current hype (Stadelmann, Braschler, and Stockinger 2019) around AI and the daily growth of scientific literature on the topic (Perrault et al. 2019) make a proper selection of content a key aspect of teaching AI. Hence, a key aspect of the ATLAS concept is to give a timely selection that emphasizes topics with future relevance and their historic development, thereby making the overarching principles that stood the test of time stand out. This is given priority over intriguing details and formal derivations.

Deconstruction. Due to the current extensive media coverage of AI, many misconceptions about the field abound in prospective students (such as the focus of the field being to understand human intelligence or create conscious machines (Urban 2015)). An important aspect of the ATLAS concept thus is a form of demythologization that keeps the original motivation of the students and channels it into more realistic, sustainable paths.

Cross-linkage. Both aspects above—a stable body of knowledge in AI fundamentals and careful treatise of real and misguided excitement—become a firm foundation given the third ingredient: a dense network of cross-references to other subjects in the study programme that is compatible with the different occupations of a professional career in computer science. The ATLAS concept takes care to teach AI not only to future scientists, but also to software developers, data analysts or system administrators, acknowledging the future importance of AI methodology in any field.

We suggest that these three aspects ensure a proper mediation of AI foundations in the following way: by *canonization*, we ensure to teach the full canon of relevant methods (ranging from heuristic search and logical planning to machine learning). Yet, we link each of these areas to a practical example that the students can chose to work on as part of the accompanying assignments (e.g., controlling a fashionable browser game, building a dragnet investigation system, or decision support for second-hand vehicles). This way, students see for themselves that not only the currently most fashionable methodology, ML, has practical relevancy. The students themselves also play a major role in the *deconstruction* of myths by first forming a personal view through the occupation with scientific texts and programming tasks, which they then present in own write-ups or oral discussions. This leads to more effective *cross-linkage* of the AI topics as our lecturing gets enriched by personal, first-hand experience of each learner.

3.2 Imparting the discipline's core attitudes

The discipline of AI with its sub-field ML as we see it does not have a single goal ("creating intelligence"), but rather offers a methodical toolbox to approach multiple targets ("solving complex problems") (Luger 2005). At its core, it is thus not so much constituted by technology than by a specific attitude: since the discipline's inception in the 1950s, AI researchers notoriously approached the kind of problems with creativity and pragmatism that had been laid aside by fellow researchers from other disciplines as "too hard" (Nilsson 2009). They did so by employing an interdisciplinary oriented "let's do it" mentality. Today, this mentality distinguishes the work of the AI engineer from other modelling approaches used by software engineers, database designers or statisticians, although skills in all these areas are relevant for success in and through AI as well.

Our students get to experience this attitude in the programming labs that accompany the lectures: programming skills are only a means to an end here, while problem analysis and experimentation become the focus. The lab tasks are accompanied by pen and paper exercises embedded throughout the lectures. For example, AIMA exercise 3.9, extended and embedded into the self-study time of AI lecture $V03$, vividly shows the difference between AI (having a computer program *appear* intelligent) and human intelligence: a classical brain twister is approached here by efficient search through all combinations of possible solution steps, which constitutes an excellent AI solution for the problem at hand but typically gets labelled "just brute force" by the students at first sight.

Finally, the students read weekly portions of the AIMA book and articles like (Domingos 2012) as accompaniment to the lectures. The conveyed anecdotes and historical notes therein specifically contribute to the students' socialization in the discipline of AI that is being consolidated through the gamification elements provided in the labs (see Section 2.2).

3.3 Didactic settings

ATLAS builds upon the lecture plus lab pattern widespread in engineering education: weekly lectures are accompanied by lab exercises with roughly the same extent in supervised in-class time. The practice time thus infused into the curriculum is extended by interactive parts within the lectures that embed small group research tasks and discussion as well as thinking exercises, and labs that go beyond programming and development to accommodate essay writing or philosophical questions. This way, respective modules host educational objectives for professional and methodical competences on levels K1–K4 (Bloom et al. 1956) by presenting AI as socio-technically broadly understood. We highlight the following didactic means used to mediate them:

Reflection. For example, AI lab assignment $P01$ is concerned with the already mentioned diverse preconceptions of AI. It asks students to create a blog post that presents a well-founded and argued-for own opinion on the contents of a futuristic essay (Urban 2015). At the end of the semester, students can reflect on their initial statement with a second blog post (lab $P01b$) that can incorporate the insights gained throughout the course. While all opinions are welcome, the emphasis in grading is on self-reflection and reasoning.

Self-responsibility and motivation. Twenty percent of the final grade are acquired during the semester through the lab assignments. From the existing six assignments in the AI course for example that are distributed evenly throughout the semester, students can choose any two to get graded within a short colloquium between the student team and lecturer during the in-class time (students usually work on all assignments, but put considerably more time into the two graded ones). This way, students get empowered to prioritize own learning goals and take ownership of their investment of time and its distribution over the semester.

Cooperative competence development. Said lab assignments are usually to be worked on in teams of two students. This way, students can strategically pair up competencies and effort as well as learn from each other. Teams are allowed to help each other as long as any help is disclosed (according to good scientific practice), and competitive elements as withing AI lab assignment $P02$ (see Section 2.2) increase the appeal of and the necessity for good team work.

Activation of students. Each 90 minutes lecture block contains a part of up to 30 minutes that assigns an active role to the students rather than the lecturer. This comes in forms of jointly solving a puzzle ("escape from the Wumpus world" in AI-lecture $V06a$), individually applying learned principles (logic training in AI-lecture $V06b$), computing results in small groups ("help inspector Clouseau to probabilistically convict a murderer" in ML-lecture $V09a$), or sharing insights from individual research at tables (exploration of possibilities with OpenAI Gym (Brockman et al. 2016) in AI-lecture $V02$).

Social learning. A prominent place throughout each module is given to the research work and careers of course alumni and junior teaching staff. Linking course content to concrete outcomes of applied research projects with regional industrial partners known by the students creates a pull that contributes to the students' motivation and expanded vision for AI in practice as well as their role in it. Key to create this are the graduate students that teach part of the labs: closer in age and role to the course participants, these tutors are approached frequently by the class to give a second opinion on the more philosophical and career aspects of AI. Uncounted lunches, coffee invitations and after work beers have been realized this way between teaching staff and students.

Open educational resources (OER) and blended learning. All course materials, including lecture recordings, slides, and lab materials, are fully and freely available online (see links above). This should enable flexible deepened learning (e.g., for exam preparation), but does not compromise live lecture attendance in our experience. As an add-on, it supports the transition to live online teaching (as was required during the COVID-19-induced lock-downs in 2020) as content is already designed to be streaming-friendly.

What competencies are specifically facilitated through this didactic design? Learners (a) identify relevant AI (research) questions independently in their (practical) work as they trained this during several labs. They (b) independently dive deeper into respective methods to find practical solution as the lab descriptions incentivised own research work into more advanced methods and applications (solutions where impossible to attain without independently going beyond the lecture content). Students (c) are curious and unbashful to bring in their knowledge or expand it through questions as they trained this in the lab colloquia, the guest lectures and with their tutors (see next section). Finally (d), they can recap all details when needed on the job as all material is permanently and openly available as OER. This enhances the *map* of AI solution strategies they know by heart.

3.4 Paths towards a career in AI

Our students of computer science usually envisage a career in software engineering, not specifically AI. We meet them there by showing how different AI methods serve as puzzle pieces in numerous everyday situations of software engineering. The lab tasks and in-class exercises are strictly sourced from practical applications such as automatic university time tabling, biometric access control or data analysis to reinforce this point. More deeply entrenched students are regularly shown cross references to current research as well as ways to pursue a respective career by engaging with faculty at the university (the possibility to e.g. take up graduate studies is typically completely unknown to our students due to the setup of a "Fachhochschule" (Brodie 2019)).

Additionally, lectures end with an outlook called "where's the intelligence?" that explains in which way the weekly topic is a clever solution but what separates it from human-like intelligence. This not only shows the discrepancy between warranted enthusiasm (about a practically working methodology) and sheer hype, it also helps the students spot the kind of tasks they might approach in their future job using the conveyed foundations. Finally, we invite specialists from regional industrial partners for guest lectures and report on recent successes of course alumni. Learners are encouraged to actively use these opportunities to network and engage with those speakers and their ideas. In contrast to the culturally typical reticence of our students, the fresh setup with people on stage that might be considered peers age-wise opens them up in the direct (active participation) and metaphorical sense (opening up to the idea of other career options within the field of AI).

By connecting the practical coursework with typical tasks of a programmer or IT consultant, students clearly see how learning foundations of AI makes them better in their original career goal. By confronting them with new opportunities in and through data science and AI in business and research, they recognize new and viable career paths (e.g., data scientist (Stadelmann et al. 2019)) that only begin to take traction in public awareness. Our alumni's success with award-winning theses inspired by the courses or ongoing (research) careers in AI testify to the effectiveness of these measures.

4 Evaluation

This section, rather than reporting on a formal evaluation of the concept or derived courses, aims at supporting the design choices made in the ATLAS concept by grounding them in qualitative student feedback (next section) and / or figures from the latest taken exams (Section 4.1).

4.1 Qualitative assessment

The following select quotations are taken from AI and ML students' feedback at the end of different semesters. We group them by the arguments we think these short statements support as examples of many similar statements:

Canonization and deconstruction. *"Sustainable technologies are taught; in the process you are brought down to earth." "[The] module gives a good overview of the overall topic." "I welcome that [. . .] the question "where's the intelligence?" is answered each lecture."* Students seem to grasp that AI is rather a way to solve complex practical problems than a theory to explain how we think or create artificial life. The content is indeed perceived as a foundation for practice rather than a narrow specialization.

Motivation. *"The professors [. . .] enthusiastically explained it very precisely. I also had the feeling that the fun of the topic seemed very important to them. It was also important for them that everyone understood."* Student's perception of the module content is in our perspective strongly connected with and dependent on the person that teaches. Insofar, the concept, curriculum or OER availability alone is no guarantee for the intended outcome: enthusiastic teaching[5] is an integral part of ATLAS as it facilitates activation.

Activation. *"The labs support the learning process very much; similarly helpful are the exercises throughout the lectures." "Very handy are the labs where one implements hands-on what should be learned." "The lectures are very interactive." "Good lecture-style presentation, active presence of the lecturers during the labs that motivates students to listen even on Friday afternoons."* We increased the time for in-class exercises and interactivity over the years and received increasingly positive feedback on its effects. Despite the success of more modern teaching styles, lecture-style teaching still seems to be a very helpful didactic setting for technical education if mixed with practical and interactive aspects where applicable.

OER. *"The videos on Youtube are ideal for repeating." "The recording of the lectures is very helpful. It gives the students the possibility to review parts of the lecture for exam preparation or if you haven't understood everything during the lecture."* Students use video recordings as intended for repetition without getting distracted by the new flexibility (a real danger of digital transformation: procrastination due to everything being available anytime).

4.2 Quantitative assessment

Here we evaluate the extent of superior educational goal attainment of full cohorts by looking at final exam results from the two most recent implementations of the ML course:

ML course spring term 2019. Figure 2 shows histograms of the achieved relative scores for different topics in the final written exam. Most histograms show a unimodal, potentially slightly skewed distribution, whereas for question $Q9$, the distribution is notably bimodal—which was our intention: it manages to separate clearly those who grasped the respective concept correctly from those who did not. Here, most students understood how to apply the Naïve Bayes classifier to a dataset by hand very well. This was trained and exercised several times during the lecture and labs. Other questions, considered individually, do not seem to discriminate well between pass and fail, but still maintain an even higher correlation with the final grade (0.78 for $Q2$ and 0.65 for $Q8$) than $Q9$ (0.62). Overall, this result indicates satisfactory realization of educational objectives for passing students.

ML course spring term 2020. Due to the COVID-19 pandemic, the final assessment of spring term 2020 had to be taken in full distant mode over the Moodle (Dougiamas and Taylor 2003) learning platform. This opened up the opportunity to design a different kind of exam: open book, as online proctoring could not be extended far enough to meaningfully control the use of only permissible aids; and involving hands-on programming, as every participant would sit in front of a well set-up developer's machine (the personal laptop). Didactic underpinning for this comes from Johann Heinrich Pestalozzi's (1746–1827) maxim of holistic learn-

[5]We do not want to conceal the following minority report: *"Don't always stamp your foot, it wakes one up."*

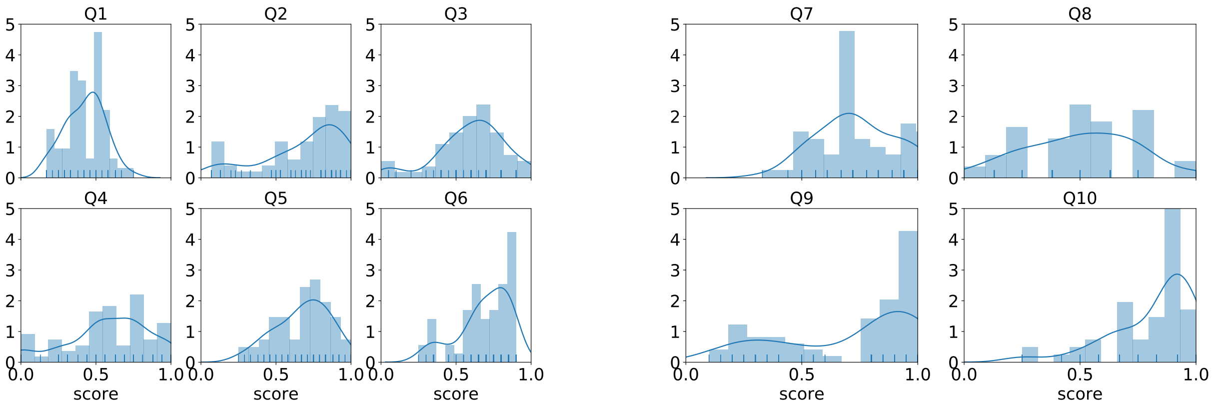

Figure 2: Histograms ($n = 60$) of the achieved relative scores for each of the ML course's spring term 2019 final exam questions $Q1$–$Q6$ (left, covering the following topics: *fundamentals* [MC]; *gradient descent* [P]; *SVMs* [FT]; *ensembles* [FT]; *advanced topics* [FT]; *debugging models* [PI]) and $Q7$–$Q10$ (right: *feature engineering* [P]; *Bayesian networks* [MC]; *Naïve Bayes* [C]; *clustering* [PI]) with task types MC=multiple choice, P=programming, FT=free text, PI=plot interpretation, C=calculation.

ing: *"hand, heart and head"* (Brühlmeier 2010), referring to a focus on the individual learning processes that considers cognitive-intellectual as well as physical and affective-emotional stimuli. Every concept that is taught should also be exercised by hand. For this reason, programming was important within the lab exercises, and the two programming tasks that would together make up 50% of the exam's content reflect this importance as well. Thus, participants uploaded Jupyter notebooks (Kluyver et al. 2016) containing all programming at the end of the 120 minutes long exam.

The result is noteworthy: most of the participants did very well in programming. Figure 3 shows the histograms of the results for the respective tasks $Q5$ and $Q6$ alongside a selection of other tasks. The histograms for the two programming tasks (in the bottom right) are left skewed, meaning that most of the students know now how to apply machine learning to solve tasks in real life (the many 0-point entries for $Q6$ might be the result of time problems with the exam as a whole, as this was the last task). This indicates that the overall educational objectives of the ML course—to apply ML algorithms—are met by the majority.

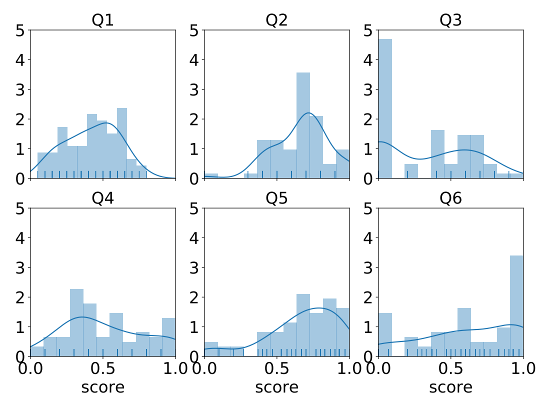

Figure 3: Histograms ($n = 68$) of the achieved relative scores for six selected ML course spring term 2020 exam questions (topics for $Q1$–$Q6$, respectively: *fundamentals* [MC]; *SVMs* [FT]; *ensembles* [FT]; *Naïve Bayes* [C]; *feature engineering* [P]; *debugging models* [P]).

5 Conclusions

The presented didactic concept promised *analog maps* for the digital world of AI. AI undoubtedly is a driver of the digital transformation, and digital methods accordingly play a large role in the teaching of respective courses built using the ATLAS concept (e.g. in the programming labs on a computer, in research exercises using educational videos and web search during lectures, in blogging assignments and in web-based leader boards for the gamification of labs). The benefit of presenting AI in form of "maps" rather than a "recipe" has been introduced above as stemming from the complexity of the use cases and the respective wide array of different methods to have in one's toolbox. But what makes this collection of maps "analog"?

The maps only emerge in the beholder's mind. They are thus created by analog means (through specifically designed didactic scenarios and enthusiastic presentations in the lectures) and stored in analog form (in natural neural networks). This analogy carries over to the application of the taught AI fundamentals: artificial intelligence is not primarily replacing human intelligence, just like digital does not primarily replace analog, but augments it (Ford et al. 2015). AI thus finds an optimal environment for application where human and machine complement each other with their respective strengths and weaknesses.

Acknowledgments

We are thankful for the recognition of the ATLAS didactic concept through the "best teaching—best practice" award (3^{rd} rank) of the Zurich University of Applied Sciences.

References

Aoun, J. E. 2017. *Robot-proof: higher education in the age of artificial intelligence*. MIT press.

Bloom, B. S.; Engelhart, M. D.; Furst, E. J.; Hill, W. H.; and Krathwohl, D. R. 1956. *Taxonomy of educational objectives: The classification of educational goals*, volume 1. Longman Group.

Brockman, G.; Cheung, V.; Pettersson, L.; Schneider, J.; Schulman, J.; Tang, J.; and Zaremba, W. 2016. Openai gym. *arXiv preprint arXiv:1606.01540* .

Brodie, M. L. 2019. On developing data science. In *Applied Data Science*, 131–160. Springer.

Brühlmeier, A. 2010. *Head, heart and hand: Education in the spirit of Pestalozzi*. Open Book Publishers.

Dessimoz, J.-D.; Köhler, J.; and Stadelmann, T. 2015. AI in Switzerland. *AI Magazine* 36(2): 102–105.

Domingos, P. 2012. A few useful things to know about machine learning. *Communications of the ACM* 55(10): 78–87.

Dougiamas, M.; and Taylor, P. 2003. Moodle: Using learning communities to create an open source course management system. In *EdMedia+ Innovate Learning*, 171–178. Association for the Advancement of Computing in Education (AACE).

Ford, K. M.; Hayes, P. J.; Glymour, C.; and Allen, J. 2015. Cognitive orthoses: toward human-centered AI. *AI Magazine* 36(4): 5–8.

Kluyver, T.; Ragan-Kelley, B.; Pérez, F.; Granger, B. E.; Bussonnier, M.; Frederic, J.; Kelley, K.; Hamrick, J. B.; Grout, J.; Corlay, S.; et al. 2016. Jupyter Notebooks-a publishing format for reproducible computational workflows. In *ELPUB*, 87–90.

Luger, G. F. 2005. *Artificial intelligence: structures and strategies for complex problem solving*. Pearson education.

Mercator, G. 1595. *Atlas sive Cosmographicae Meditationes de Fabrica Mundi et Fabricati Figura*. Rumold Mercator, Duisburg, Germany.

Nilsson, N. J. 2009. *The quest for artificial intelligence*. Cambridge University Press.

Perrault, R.; Shoham, Y.; Brynjolfsson, E.; Clark, J.; Etchemendy, J.; Grosz, B.; Lyons, T.; Manyika, J.; Mishra, S.; and Niebles, J. C. 2019. The AI Index 2019 Annual Report. *AI Index Steering Committee, Human-Centered AI Institute, Stanford University, Stanford, CA* .

Russell, S.; and Norvig, P. 2010. *Artificial Intelligence: A Modern Approach*. Pearson Education, Inc., New Jersey, United States, 3 edition.

Schneider, U.; and Brakensiek, S., eds. 2015. *Gerhard Mercator: Wissenschaft und Wissenstransfer*. WBG, Darmstadt. ISBN 9783534264513.

Skog, D. A.; Wimelius, H.; and Sandberg, J. 2018. Digital disruption. *Business & Information Systems Engineering* 60(5): 431–437.

Stadelmann, T.; Braschler, M.; and Stockinger, K. 2019. Introduction to applied data science. In *Applied Data Science*, 3–16. Springer.

Stadelmann, T.; Stockinger, K.; Braschler, M.; Cieliebak, M.; Baudinot, G.; Dürr, O.; and Ruckstuhl, A. 2013. Applied data science in Europe: Challenges for academia in keeping up with a highly demanded topic. In *9th European Computer Science Summit, Amsterdam, The Netherlands, 8-9 October 2013*.

Stadelmann, T.; Stockinger, K.; Bürki, G. H.; and Braschler, M. 2019. Data scientists. In *Applied Data Science*, 31–45. Springer, Cham, Switzerland.

Sutton, R. 2019. The bitter lesson. *Incomplete Ideas (blog), March* .

Urban, T. 2015. The AI revolution: the road to superintelligence. *Wait But Why blog* .