When Achilles met the tortoise

Towards the problem of infinitesimals in machine learning[1]

Clemens Apprich

I would like to begin with a little story, a story that you probably already know. It's the story of Achilles and the tortoise.

One day, the hero of the Iliad met a tortoise whose mind was quicker than its legs. She challenged Achilles to a race, but asked him for a head start. Achilles willingly – and rather arrogantly – agreed to do so. The turtle crawled away. Achilles took his time, laced his sandals and finally started to run. In no time he covered the distance that had separated him from the turtle. In the meantime, however, the tortoise had also crawled on, and, while Achilles was catching up, she had again made a little progress. To cut the story short: no matter how fast Achilles ran, the tortoise always stayed a little way ahead – and so the famous hero could never catch up with the animal.

The story was told in this or a similar way – there is no exact record – by the Eleatic philosopher Zeno (around 490 to 430 BC) in order to present one of his paradoxes. The dichotomy paradox goes as follows: Because the world is one, movement is impossible. Every distance that a moving object has to cover can be broken down into an infinite number of partial distances (e.g. by continuous bisection: $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, and so on), with one distance always remaining. As a consequence, no movement can ever be carried out completely, because there is always a distance remaining, no matter how small it may be.[2]

1 Parts of the argumentation in this article were developed in context of Simon Denny's collaborative exhibition *Proof of Stakes: Technological Claims* (Denny et al. 2022).

2 With this paradox, which exists in different variations (e.g., in the form of an infinite regression), Zeno wanted to (at least if you follow common introductions into philosophy) defend the teachings of his mentor Parmenides of Elea (born around 515 BC). According to Plato (1997), Parmenides was accompanied by Zeno, when he met Socrates around 445 BC in Athens and confronted him with the astonishing claim that beings (reality) are a holistic, unchangeable and unified entity (i.e. ontological/ontic monism).

Of course, such an idea completely contradicts our everyday experience, as it declares the immediate perception to be an illusion. Nevertheless – or precisely because of this – Zeno's paradox, passed down via Plato's dialogue "Parmenides" (1997), would not let go of Western philosophy for the next two and a half thousand years (from Archimedes to Giovanni Benedetti, to Isaac Newton, David Hume, Gottfried Wilhelm Leibniz, to Georg Cantor, Alfred North Whitehead and Gilles Deleuze – and most recently Gregory Chaitin with his algorithmic information theory). What it introduced and has since then haunted the history of science, in particular mathematics, is the problem of infinitesimals – with infinitesimals being distances in space or time that denote a smallest possible unit. It is assumed that an infinitesimal quantity is so close to zero that it has no numerical effect; it simply eludes any attempt to measure it, like sand trickling through your fingers.

Infinitesimals were crucial for the development of differential and integral equations – also known as calculus. As is well established, Gottfried Wilhelm Leibniz (1646 – 1716) and Isaac Newton (1642 – 1726) developed the mathematical branch of infinitesimal calculus independently of each other (or so the story goes) in the late 17th century.[3] Defining a systematic method for the calculation of surfaces and motion, it soon became a 'killer application' in modern mathematics as it geared to solve practical problems (e.g. ballistic calculations, motion of planets, the design of bridges). Calculus, eventually, turned out "to be the richest lode that the mathematicians have ever struck" (Kline 1977: 4). The development of calculus marked a new era in mathematics and its uses within the sciences have continued to the present day.

Not surprisingly, calculus is also at the heart of today's machine learning processes. Understood as optimization problems, machine learning-algorithms, in particular in the field of artificial neural networks, draw on calculus and, as a consequence, entail some of the paradoxes that come with it. Hence, by addressing the "quality issues" brought up in this volume, I want to argue that a machine learning-model, precisely because it is built on an exhaustive approximation as part of its optimization process, can never fully converge, and as a consequence does not yield any final result. This is of relevance because it shows that – contrary to widespread belief – machine learning is deeply entangled with mathematics and logics. What's more, such a paradoxical take on machine learning, which can also be seen as yet another iteration

3 In fact, the question of who invented calculus first became the subject of a huge controversy, now known as the calculus controversy (cf. Hall 1980).

of the “halting problem” (Turing 1936), resonates with recent debates around the incomputability of reality (Parisi 2013; Fazi 2018; Galloway 2021) as well as speculative attempts to overcome modern computation altogether (Amaro 2022). The goal of these interventions and by consequence the following article, is to highlight the necessity of moving beyond the limited imagination of (statistical) probability with regard to machine learning models in order to search for new “politics of possibility” (Amoore 2013).

1. Forever converging

In the beginning of Google’s Machine Learning Crash Course,[4] Peter Norvig, Head of Google Research, makes the remarkable statement that – with machine learning – we are now moving from mathematics to natural science, from logics to statistics, and from coding to *growing* models:

> Machine learning changes the way you think about a problem. Software engineers are trained to think logically and mathematically […]. With machine learning, the focus shifts from a mathematical science to a natural science: we’re making observations about an uncertain world, running experiments, and using statistics, not logic, to analyze the results of the experiment. The ability to think like a scientist will expand your horizons and open up new areas that you couldn’t explore without it. (Norvig 2020)

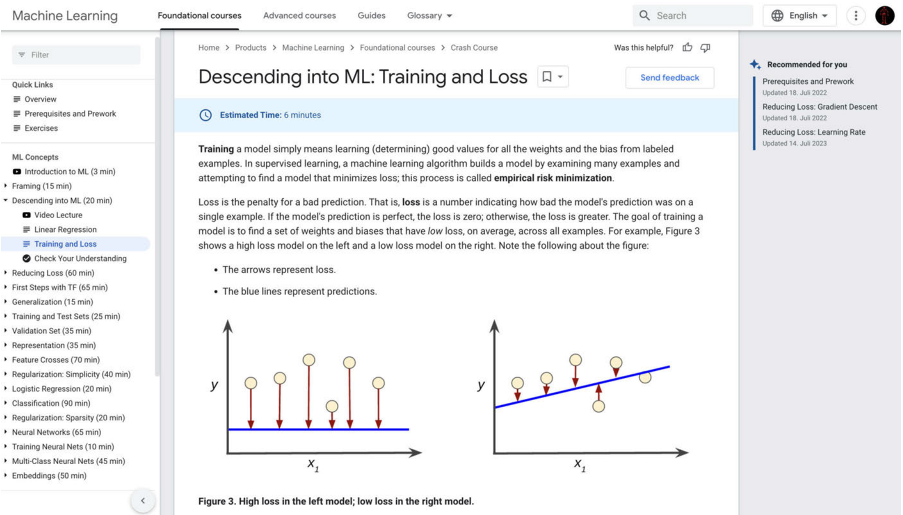
It is worthwhile to consider some of the deeper implications of Norvig’s statement: What does it mean to move from mathematics to natural science? And, in the process, do we really leave logic behind? What the statement implies, is the fact that with machine learning, and respectively neural networks as the most recent implementation of machine learning systems, we are moving from deductive to inductive methods of data processing: the model learns a correlation pattern between input and output data in order to make predictions on unseen data. To do so, a loss function is calculated for each instance, which shows “how bad the model’s prediction was on a single example” (Google Developers 2020).[5] Similar to the hot and cold play, the iterative strategy constitutes

4 Google’s MLCC is one of the most popular machine learning (online) course with tens of thousands of users (Rosenberg 2018).

5 In the context of machine learning, unsupervised learning is often spoken of. However, when training a model (e.g. a recommendation system) most commonly a supervised

the essential thing of this learning approach, which, of course, corresponds to the aforementioned optimization process.[6]

Figure 1: Machine Learning Crash Course (screenshot from Google Developers 2020).



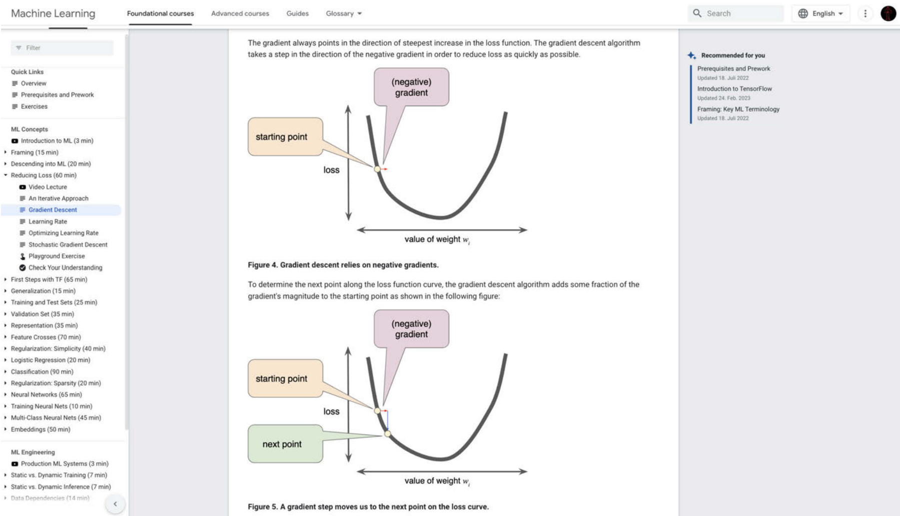
Suppose we had the time and the computing power to calculate the loss function for all possible learning parameters: the result would be a convex curve in which the rate of loss moves towards zero, that is the limit value to which the model converges. However, since the calculation of the loss function

procedure – or at least a mix of supervised and unsupervised learning – is used; this means that a data set is used containing both features (e.g. age, gender, search history of the user, temporal or geographical features in the data) and labels (what we want to predict).

6 The search for the optimal parameters constitutes machine learning. As Adrian Mackenzie writes in *Machine Learners* (2017): “[O]ptimization techniques are the operational underpinning of machine learning. Without their iterative process, there is no machine in machine learning” (95).

for every instance of a training set would take too long, a statistical method is used to solve the optimization problem: the gradient descent.[7]

Figure 2: Machine Learning Crash Course (screenshot from Google Developers 2020).



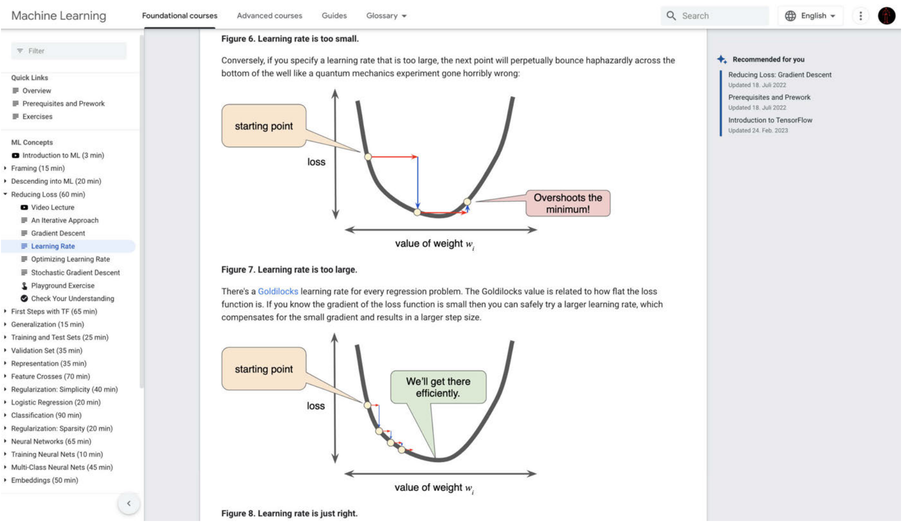
As is the case with mountaineering, the trick is to choose a descent direction and a step length in order to reach the valley of the curve as quickly as possible or, in other words, to reduce the loss as quickly as possible. The starting point is set arbitrarily, because it usually has no effect on the end result. To find the next point along the loss function curve, the learning algorithm then multiplies the gradient by a scalar quantity called the *learning rate* (sometimes also step length, although this can be misleading because the length of the step changes relative to the scalar). For example: if the amount of the (negative) gradient is 5 and the learning rate is 0.1, then the algorithm selects the next point

7 Gradient descent is one of many examples of optimization problems within machine learning systems. Others include coordinate descent, coordinate ascent or convex optimization.

0.5 units from the starting point and the next but one point at 0.05 units from this point.[8]

The correct setting of the learning rate makes up a good part of machine training. However, in practice, it is not necessary or possible to find the 'perfect' (or near-perfect) learning rate for successful training. The goal simply is to find a learning rate large enough for the model to converge in a timely manner, but not so large that it overshoots the target.

Figure 3: Machine Learning Crash Course (screenshot from Google Developers 2020).



Hence, the idea behind gradient descent is to tweak the parameters iteratively until the algorithm converges to a minimum, that is to repeat the process "until the difference between the old value and the new value is very small" (Kansal 2020). Now, as you can already guess, behind the seemingly innocuous notion *very small* lurks the two and a half thousand-year-old paradox of Zeno. Because the learning steps gradually get smaller as the parameters approach

8 Another inspiration for this approach might be "fitness landscapes" (Wright 1932), a concept developed as part of evolutionary biology in the 1930s. I thank Claus Pias for this reference.

the minimum, each step can be divided into an infinite number of sections with the result that the model, at least in theory, never fully converges.

Peter Norvig argues that machine learning is no longer a logical problem, but an experimental one. That might be true if we follow the premise that with machine learning we are moving from a mathematical (i.e. deductive) to a natural (i.e. inductive) science. However, given the central role of mathematics in the natural sciences when converting observations into measurements (not to mention the creation of measurement itself), the statement seems to be at odds with its own premise. What it does though, is play right into the hands of similar attempts to *biologize* AI and machine learning (i.e. to *naturalize* and thus *normalize* the labor processes, the material infrastructures, but also the data politics behind it). The apologists of the new machine learning paradigm want to make us believe that the world of data is simply a natural phenomenon that does away with logical, that is theoretical, explanations (cf. Anderson 2008).

What is not mentioned in Norvig's statement, but is definitely an issue in computer or the data sciences, is the fact that a machine learning algorithm "must embody some knowledge or assumptions beyond the data it is given in order to generalize beyond it" (Domingos 2012: 81). A machine learning algorithm cannot see, hear, or perceive input examples (images, text, audio files, etc.) directly. Instead, a representation of the data has to be created in order to allow the model to *see* it. In other words, for the model to train, features have to be selected (often even created) which, in the eyes of the still very human trainer, best represent the data.[9]

Now this basic insight contradicts the common idea that we, and the models respectively, simply have to look at the data to get the desired outcome. What gets omitted, if not to say oppressed, in this rather naïve view, is the fact that the desired outcome (together with its logics) is always already inscribed in the process. With each iteration, the model gets more and more tweaked towards *good* property values (also called identity values) in order to filter out the *right* information from the data set.[10]

9 A common practice in machine learning is actually called 'feature engineering.'

10 This is in particular true for 'reinforcement learning from human feedback' (RLHF), a technique to train a reward model from human feedback that is central to current generative AI-systems such as GPT.

2. Never being

With the alleged shift from deductive to inductive reasoning in machine learning, a new kind of *identity politics* has entered the field. The problem is that the hidden assumptions about the data, which directly inform the machine learning models, correspond in so many ways to the rather old, historically grown social categories (e.g. race, class or gender). Reintroduced as *natural* representations, these categories bring about the much-discussed issues of data bias and algorithmic discrimination (cf. Apprich et al. 2019). In this process, normalizing standards such as Whiteness in algorithmic filtering and face recognition, become the default setting of machine learning models (cf. Katz 2020: 172f.).

Due to the fact that models learn from past data in order to be able to make predictions about the future, machine learning turns into a self-fulfilling prophecy. In her new book *Discriminating Data* (2021), Wendy Hui Kyong Chun makes that point clear when she explains how

> predictive algorithms […] are verified as correct if they predict the past correctly, for they are usually cross-validated using past data that are hidden during the training period or out of sample data, similarly drawn from the past. (ibid.: 46)

By becoming the *ground truth* of (inductive) machine learning, limited and biased data from the past foreclose, rather than enhance, the future, with the effect that existing (racial, social and sexual) discrimination is perpetuated.

The usual answer to this problem is a call for better data or better models. However, as Ramon Amaro (2022) has shown, those well-intended attempts do not break away from the epistemic violence of current machine learning models. Instead, they merely *optimize* discriminatory practices. He writes, "What we experience today as algorithmic prejudice is the materialization of an overriding logic of correlation and hierarchy hidden under the illusion of objectivity" (ibid.: 61). Given the eugenic and biometric roots of correlation techniques, the past truly *overrides* our present and future by propagating a natural (i.e. eternal) truth through machine learning. Yet, to insist on the fact that machine learning models never fully converge, implies that they do not determine an ultimate truth or identity (cf. Cheney-Lippold 2011).

Beyond Quantity then also means that there always remains a surplus that cannot be calculated, because it does not fit into the (normalizing) norm of machine learning models. It means to – as Amaro and Khan (2020) propose – deploy a "calculus of variations", able to explore the liminal space between algorithmic calculations, the gaps and cracks that might open up to other, in particular non-white, versions of reality. Hence, exposing the internal limits of machine learning systems by confronting them with indeterminacy, incompatibility, as well as a "Black totality, always already in the process of transformation" (Amaro 2022: 62), might provide a way to work through those systems and put them to different ends.

The goal is to come up with machinic logics that break the shackles of merely inductive, but also deductive, reasoning. Rather than confirming what was already there, a generative (abductive) approach might allow for infinite possibilities. Machine learning, in this perspective, exposes the limits of computability in a productive way: To the same extent that learning algorithms are contingent on infinitesimals, the models themselves are not fixed by any preset identities or categories. On the contrary, the indeterminacy, in particular its inclusion in the calculation process, is what constitutes the ability to learn (cf. Parisi 2018). Hence, if the goal of machine learning is to *generalize* a model based on data, then generalization, when understood as an ongoing, open process, is at the core of machine learning; this concerns the central idea that concepts are not merely some abstract content that can be learned, but actually develop through learning as a discursive (i.e. social) practice.[11]

For machine learning to transform (and not merely repeat) the world, it is thus necessary

> to move from seeing an inert model as the machine learner to seeing the human researcher or developer – *along with*, and not separate from, his or her model and surrounding relations – as the machine learner. (Reigeluth/Castelle 2021: 104)

Because humans and machines are part of the same symbolic realm, they are, as learners, contingent on the same "*regular, discrete framework*" (Galloway 2021: 123). Acknowledging the social (not merely mechanical or cognitive) aspect of machine learning can help us better understand its ambiguity and contingency – moving back and forth between the formalization of real-word

11 The idea of 'concept-learning' as a social practice goes back to the Soviet psychologist Lev Vygotsky (1986).

problems and the actual implementation of such models to process those problems.

3. The incomputable

According do Hartmut Winkler, processing as the third media function of computing (besides storing and transmitting) implies to recognize the double character of regularity or repetition and innovation:

> Das Prozessieren – als Eingreifende Veränderung – scheint von vornherein auf die Seite des Neuen zu fallen, insofern es eben Eingriff und Veränderung, und mit Blick auf die Wiederholung die Verschiebung, betont. (Winkler 2015: 107)

The idea of an "interfering transformation" is crucial for machine learning as well. Precisely because it is characterized by variability and indeterminacy, it relies on repetitive steps. In this sense, machine learning, which is defined by the processing of data, also necessitates a formalization by means of programming (Python) and mathematics (Calculus).

In contradiction to Peter Norvig's statement, machine learning is deeply logical and heavily relying on mathematical science. To claim otherwise would be to promote a version of machine learning that is fetishized as a natural thing and, therefore, hides its inner workings (i.e. the processing steps) from its users. Consequently, Google TensorFlow as well as all the other machine learning-infrastructures, such as Amazon Web Services or Azure Machine Learning, depict themselves as mere *services*.[12] Similar to the Internet's client/server architecture (cf. Krajewski 2018), these hidden infrastructures are essential to how machine learning is presented to us and how these representations influence our understanding of it (cf. Luchs/Apprich/Broersma 2023).

Contrary to the common belief that machine learning algorithms simply process data until a final result is found, the actual process is rather messy. In fact, contingency, indetermination and uncertainty are at the center of modern mathematics and, therefore, computing. Luciana Parisi, by invoking

12 This becomes apparent in Google's MLCC itself, when the (Python) code to run the models is literally hidden in foldout boxes.

Gregory Chaitin's algorithmic information theory (Chaitin 2004),[13] explains that "[s]ince there are infinities that cannot be compressed into simpler postulates, theories, truths, it follows that there are realities that are logically irreducible" (Parisi 2021: 82). Accordingly, there are realities that cannot be computed, because they cannot be captured by today's algorithms. What we experience with machine learning is not simply a shift from deduction to induction, from mathematics and logics to natural sciences, but rather the introduction of the incomputable (i.e. negativity) at the heart of computation.

With Turing's "halting problem" (Turing 1936), which basically says that no algorithm (i.e. a finite step-by-step procedure) exists, which can determine in advance whether a machine will finish running a program, a fundamental shift within the logic of calculation has occurred. The inherent limit of the discrete-state machine opens it to dynamic forms of computation. Once more Parisi (2015): "the calculation of randomness or infinites has now turned what was defined as incomputables into a new form of probabilities, which are at once discrete and infinite." "In other words," she continues,

> whereas algorithmic automation has been understood as being fundamentally Turing's discrete universal machine, the increasing volume of incomputable data (or randomness) within online, distributive, and interactive computation is now revealing that infinite, patternless data are rather central to computational processing. (ibid.: 131)[14]

Applied to machine learning, this means that we are dealing with both, patternless data being processed *and* symbolic learning systems feeding on trial-and-error. Instead of a mere step-by-step procedure, those systems are adaptive, precisely because they have to deal with the contingency of messy data. Hence, the discrete framework of computation gets tainted by real-world applications with its infinite variations. Randomness, in this perspective, is not outside of computation or machine learning, but the very core of them.

13 With his Algorithmic information theory Chaitin wants to prove that there is no such thing as absolute certainty in mathematics. There are truths that cannot be proven, problems that are impossible to solve.

14 In a similar way, M. Beatrice Fazi (2018) argues for the incompleteness and, therefore, contingency of computation. Both see Kurt Gödel as a progenitor of the incompleteness problem and its productive application in mathematics and, consequently, computational thinking.

Rather than following claims about the end of logics and theoretical explanation, machine learning includes realities that cannot be proven, but are yet to be discovered. The immanent logic of those machines, therefore, offers a radical break with the inductive explanation of natural sciences, without falling back into a deductive predictability of classic form of computation. Allowing for "a *computational thought* that is contingent, and yet does not break away from structure" (Fazi 2018: 210) could yield a machinic logic that actually might take us by surprise. A new mode of thinking about the machinic based on its learning capacities and not as a one-sided solution for or against inductive or deductive reasoning.

This brings us back to the beginning of this article. What if Zeno did not simply use his paradoxes to confirm the ontological monism (i.e. the static identity of all things) taught by his teacher Parmenides,[15] but rather sought to defend the idea of motion by putting it to a test? In other words, what if he did not try to prove the one, but to problematize the many? That would bring him very close to the here discussed problematization of incomputability, in the sense that reality is less a question of true or false, but rather an affirmation of its (infinite) possibilities. A paradox, after all, always contains more than one perspective.

List of references

Amaro, Ramon (2022): The Black Technical Object. On Machine Learning and the Aspiration of Black Being, London: Sternberg Press.

Amaro, Ramon/Khan, Murad (2020): "Towards Black Individuation and a Calculus of Variations." In: e-flux 109 (https://www.e-flux.com/journal/109/330246/towards-black-individuation-and-a-calculus-of-variations).

Amoore, Louise (2013): The Politics of Possibility. Risk and Security Beyond Probability, Durham and London: Duke University Press.

Anderson, Chris (2008): "The End of Theory: The Data Deluge Makes Scientific Method Obsolete.", June 23, 2008 (https://www.wired.com/2008/06/pb-theory).

15 In fact, Plato (1997) writes about the encounter with Socrates that Zeno felt misunderstood by him – "you do not fully apprehend the true motive of the composition."

Apprich, Clemens/Chun, Wendy H. K./Cramer, Florian/Steyerl, Hito (2019): Pattern Discrimination, Minneapolis and London: The University of Minnesota Press, and Lüneburg: meson press.

Chaitin, Gregory J. (2004): Algorithmic Information Theory, Cambridge: Cambridge University Press.

Cheney-Lippold, John (2011): "A New Algorithmic Identity: Soft Biopolitics and the Modulation of Control." In: Theory, Culture & Society 28/6, pp. 164–181.

Chun, Wendy H. K. (2021): Discriminating Data. Correlation, Neighborhoods, and the New Politics of Recognition, Cambridge, MA: The MIT Press.

Denny, Simon/Beyes, Timon/Holt, Robin/Pias, Claus/Steinbrügge, Bettina (eds.) (2022): Proof of Stake: Technological Claims, Milano: Lenz Press.

Domingos, Pedro (2012): "A Few Useful Things to Know About Machine Learning." In: Communications of the ACM 55/10, pp. 78–87.

Fazi, M. Beatrice (2018): Contingent Computing. Abstraction, Experience, and Indeterminacy in Computational Aesthetics, Lanham: Rowman & Littlefield.

Galloway, Alexander (2021): Uncomputable. Play and Politics in the Long Digital Age, London: Verso.

Géron, Aurélien (2019): Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, London: O'Reilly Media.

Google Developers (2020): "Machine Learning Crash Course.", last modified September 26, 2022 (https://developers.google.com/machine-learning/crash-course/descending-into-ml/training-and-loss).

Hall, A. Rupert (1980): Philosophers at War. The Quarrel Between Newton and Leibniz, Cambridge: Cambridge University Press.

Kansal, Sahdev (2020): "Quick Guide to Gradient Descent and Its Variants.", August 18, 2020 (https://towardsdatascience.com/quick-guide-to-gradient-descent-and-its-variants-97a7afb33add).

Katz, Yarden (2020): Artificial Whiteness. Politics and Ideology in Artificial Intelligence, New York: Columbia University Press.

Kline, Morris (1977): Calculus. An Intuitive and Physical Approach, New York: John Wiley & Sons.

Krajewski, Markus (2018): The Server: A Media History from the Present to the Baroque, New Haven, CT: Yale University Press.

Luchs, Inga/Apprich, Clemens/Broersma, Marcel (2023): "Learning Machine Learning: On the Political Economy of Big Tech's AI Online Courses." In: Big Data & Society 10/1 (https://doi.org/10.1177/20539517231153806).

Norvig, Peter (2020): "Introduction to Machine Learning Crash Course.", last modified September 27, 2022 (https://developers.google.com/machine-learning/crash-course/ml-intro).
Parisi, Luciana (2013): Contagious Architecture. Computation, Aesthetics, and Space. Cambridge, MA: The MIT Press.
Parisi, Luciana (2015): "Instrumental Reason, Algorithmic Capitalism, and the Incomputable." In: Matteo Pasquinelli (ed.), Augmented Intelligence Traumas, Lüneburg: meson press, pp. 125–137.
Parisi, Luciana (2018): "Das Lernen lernen oder die algorithmische Entdeckung von Informationen." In: Christoph Engemann/Andreas Sudmann (eds.), Machine Learning. Medien, Infrastrukturen und Technologie der Künstlichen Intelligenz, Bielefeld: transcript, pp. 93–113.
Parisi, Luciana (2021): "Artificial Critique." In: Erich Hörl/Nelly Y. Pinkrah/Lotte Warnsholdt (eds.), Critique and the Digital, Zürich: diaphanes, pp. 75–108.
Plato (1997): The Dialogues of Plato, Volume 4: Plato's Parmenides, New Haven, CT: Yale University Press.
Reigeluth, Tyler/Castelle, Michael (2021): "What Kind of Learning Is Machine Learning?" In: Jonathan Roberge/Michael Castelle (eds.), The Cultural Life of Machine Learning. An Incursion Into Critical AI Studies, Cham: Palgrave Macmillan, pp. 79–115.
Rosenberg, Barry/Google Engineering Education Team (2018): "Machine Learning Crash Course", March 2, 2018 (https://developers.googleblog.com/2018/03/machine-learning-crash-course.html).
Turing, Alan. M (1936): "On Computable Numbers, with an Application to the Entscheidungsproblem." In: Proceedings of the London Mathematical Society, Volume 42/1, pp. 230–265.
Vygotsky, Lev (1986): Thought and Language, Cambridge, MA: The MIT Press.
Winkler, Hartmut (2015): Prozessieren. Die dritte, vernachlässigte Medienfunktion, Paderborn: Wilhelm Fink.
Wright, Sewall (1932): "The Roles of Mutation, Inbreeding, Crossbreeding and Selection in Evolution." In: Proceedings of the Sixth International Congress of Genetics 1, pp. 356–366.