Overview of Machine Learning

Ioannis Stefanou

Nantes Université, Ecole Centrale Nantes, CNRS, GeM, UMR 6183, F-44000, Nantes, France

1 What machine learning is?

One of the best methods for making a puppy to learn its name is to play the name game:

1. Take your puppy to a quiet place and gain its attention by calling its name in a happy and reassuring tone (e.g. LoupI look at me or LoupI come here!).
2. As soon as you've gained its attention, reward it with a treat to encourage its good behavior.
3. Repeat this process several times until your puppy has learned to respond to its name.

In the above iterative process the puppy learns from experience that reacting when you call its name is a good thing. By repeating this process, the puppy recognizes the pattern between similar, but not identical vocal signals of its name (training data) and associates them with a specific behavior (task).

Many cognitive processes of humans involve a similar way for associating patterns with specific reactions, see for instance how we learn to drive a car, speak a foreign language or even understanding how a geomaterial behaves!

Machine learning (ML) is an anthropomorphic extension of the above process of "learning" to a computer program, i.e. to the machine. By learning, the machine will progressively improve its performance in executing a specific task. According to Tom Mitchel [Mit97], ML can be defined as follows:

> *A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.*

The whole set of examples that the computer will use during learning is the *training set*, while each training example is called a training instance or sample. The whole process of learning is called *training*. The training set is usually distinguished into *training data*, *validation data* and *test data*. More details about those sets and their importance will be given in section 5 and in Chapters 2 and 7.

A popular example showing the power of ML was *AlphaGo*, a ML program, which won the human world champion of Go (see also [SHM^{+}16] and Chapter 10 for *Reinforcement Learning*). Go is a strategy board game for two players in which the aim is to cover more territory than the opponent (see Figure 1). Despite its relatively simple rules, Go is extremely complex. The *state-space complexity* of Go, i.e. the number of legal game positions reachable from the initial position of the game [All94], is of the order of 10^{170} [TF07], which is much higher than that of chess, which is of the order of 10^{44} [Sha50], and certainly higher than the number of atoms in the universe. This makes Go a great challenge for ML algorithms.

Applying the above mentioned definition in a ML program for Go, the performance, P, will be measured by the ability of the machine to win its opponent by gaining experience, E, by executing multiple times the task, T, which is playing the game with itself. Experience can be also obtained by processing existing games of other (human) players.

Figure 1: Eighteen times Go world champion Lee Sedoll playing with the machine (courtesy: Lee Jin-man / AP, The New Yorker, `https://www.newyorker.com/tech/annals-of-technology/alphago-lee-sedol-and-the-reassuring-future-of-humans-and-machines`).

Another recent popular example of machine learning is *ChatGPT* chatbot developed by OpenAI. ChatGPT was released in November 2022 and provoked a massive public reaction. Let us see what it answers when we ask it to describe itself and what it knows about ALERT Geomaterials.

Ioannis: *Describe your self.*

ChatGPT: *I am ChatGPT, an AI language model developed by OpenAI. I have been trained on a vast amount of text data from the internet, books, and other sources. My purpose is to assist users in generating human-like text responses and providing information on a wide range of topics.*

I don't have a physical form or personal experiences like humans do, but I can understand and generate text in multiple languages. I strive to provide helpful and accurate responses to the best of my abilities, drawing upon the knowledge I have acquired during my training.

I can engage in conversations, answer questions, offer explanations, and assist with various tasks. However, please keep in mind that while I strive for accuracy, I may occasionally generate incorrect or incomplete information. It's always a good idea to verify important details from reliable sources.

Overall, I am here to assist and provide information to the best of my abilities. Feel free to ask me anything you'd like to know or engage in a conversation on any topic of your interest.

Ioannis: *How do you work?*

ChatGPT: *As an AI language model, my functioning can be divided into two main phases: training and inference.*

Training: During the training phase, large amounts of text data are used to train the model. In the case of GPT models like ChatGPT, this training involves a process called unsupervised learning. The model learns to predict what comes next in a given sequence of text, based on the patterns and relationships it discovers in the training data. By doing this over a vast dataset, the model gradually learns grammar, facts, reasoning abilities, and the nuances of human language.

Inference: Once the training is complete, the model moves into the inference phase, where it can generate responses and interact with users. Here's a simplified overview of how it works during inference:

Input: When you provide a prompt or a message, the model receives it as input. The input can be a question, a statement, or any text you want to generate a response to.

Tokenization: The model breaks down the input text into smaller units called tokens. These tokens can represent words, subwords, or characters. Tokenization helps the model understand the structure and meaning of the input.

Processing: The tokenized input is then passed through several layers of neural networks. These layers are designed to understand the relationships between the tokens and capture the context and meaning of the text.

Generation: Once the input is processed, the model generates a response. It predicts the most likely next tokens based on the context and the patterns it has learned during training. The generated tokens form the model's response.

Iteration: The process of tokenization, processing, and generation can happen in an iterative manner. The generated response can be further used as input, creating a back-and-forth conversation or a continuous interaction.

It's important to note that while ChatGPT can generate impressive responses, it may sometimes produce incorrect or nonsensical answers. It doesn't have real-world experiences or access to current information beyond its knowledge cutoff date. It's always a good idea to fact-check and verify important information from reliable sources.

Ioannis: *What is ALERT Geomaterials in summary?*

ChatGPT: *In summary, ALERT Geomaterials is an international scientific association dedicated to the study of the mechanics and behavior of geomaterials, including soil, rock, and other geological materials. It brings together researchers and professionals in the field of geomechanics and geotechnical engineering to promote collaboration, knowledge exchange, and scientific advancements. ALERT Geomaterials organizes workshops, conferences, and training courses, providing a platform for researchers to share their findings and discuss topics related to soil and rock mechanics, slope stability, geotechnical engineering, experimental testing, and numerical modeling. The association also publishes a journal focused on energy geotechnics and geoenvironmental engineering. Overall, ALERT Geomaterials plays a crucial role in fostering research and innovation in the field of geomaterials.*

Impressive! At this point it becomes quite tempting to let ChatGPT to finish this chapter or even the whole book! However, I assure you, we took the necessary time to entirely draft this volume word by word. Read carefully also ChatGPT's answers. Are all of them correct?

It is straightforward to see how the above definition for ML applies to ChatGPT and to many other applications of ML, such as in speech, handwriting, face and object recognition, voice-to-text and vice versa, translation, text auto-correction and auto-completion, spam filtering, computer games, self-driving cars, medicine, forecasting, banking, security, marketing, control problems, engineering and, of course, to (geo-)mechanics.

ML is an evolving field of knowledge and involves a plethora of methods and combinations of those. In the next section we will try to categorize and classify them in groups.

Having described and defined what ML is, it is natural to ask what *Artificial Intelligence (AI)* is? ML and AI are closely related and the latter is considered to include the former. The exact definition of AI seems to be a bit foggy for the time being and depends on how we define the terms "artificial" and "intelligence". To the author's opinion, it is easier to describe the characteristics of AI, rather than give a unique and exact definition of the term. Another example, in a totally different domain, that definitions are hard to make is what is justice. It seems easier and more important to describe the characteristics of justice (e.g. equality for all) rather than give a precise definition of the term.

2 Classification of ML methods

There are numerous ML methods in the literature. Therefore, it is useful to classify them into different categories. Here we follow the classification of Géron [Gé19], who categorizes ML to: *supervised learning* vs *unsupervised learning*, *batch learning* vs *online learning* and *instance-based learning* vs *model-based learning* methods. Of course, this is a rough classification and one method can combine different categories, as shown in Figure 2.

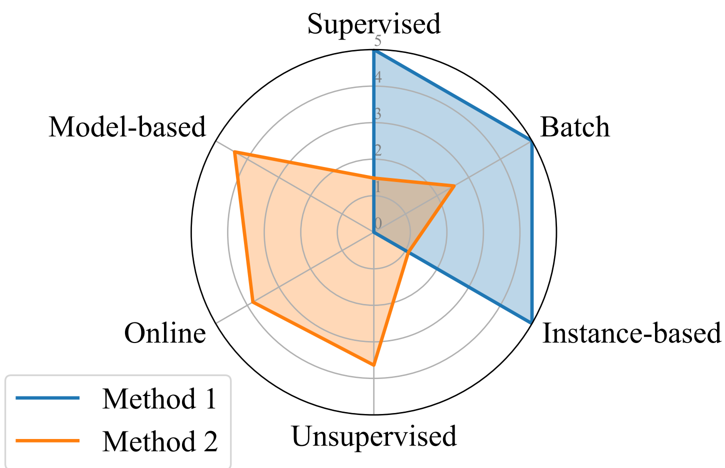

Figure 2: Classification of ML methods. Method 1 is a supervised, batch, instance-based method, while Method 2 has components belonging to different categories.

2.1 Supervised vs unsupervised ML

In *supervised learning* the training set includes the desired solutions/predictions, which are called *labels*.

For instance, imagine that your training set is hundreds of images with each one containing either a dog or a cat and that the training task is to distinguish the photos of dogs from those of cats. This is called a *classification* problem (see Chapter 4). If we give to the computer the information which of the photos show dogs and which cats, or in other words, if we label the training set then the learning is called supervised. If the training is successful, then the computer will have learned to identify cats from dogs even in a new set of photos that it haven't processed during its training.

On the contrary, if the training set is unlabeled, then the learning is called *unsupervised learning*. In this case the computer will eventually understand the pattern of the two different animals shown in the photos and it will be able to distinguish dogs from cats in an unsupervised manner (see also *clustering* in Chapter 3). Of course, the machine won't have learned to call a cat, cat and a dog, dog, because we haven't given this additional information, but it will have identified their differences and separate the data into two different classes.

Another machine learning problem is *regression* (see Chapters 2, 7, 8 and 9). In regression the training set contains one or several numerical inputs, also called *features*, and the task is to predict one or several numerical outputs, also called *predictors* that depend on the inputs. Consider as an example the prediction of the stress response of a geomaterial, which as we know depends on several input parameters, such as the applied strain, available information about the evolution of its microstructure (e.g. the position and the velocity of the grains of a sand obtained by a Discrete Element Method (DEM) analysis), history and/or other features. As the training set contains both the output and the input, the training for predicting the stresses based on the above mentioned features is supervised. However, the identification/extraction by the machine of a representation of the most important features of the microstructure that are related to the prediction of the stress response is unsupervised. Examples of unsupervised methods in ML are *feature extraction*, *anomaly detection*, *dimensionality reduction*, in which the aim is to reduce the the size of the training set without loosing important information with respect to a specific task or measure and *data compression* (among others).

In Table 1, we provide a list of important supervised and unsupervised methods in ML. However, not all ML methods can be categorized to supervised and unsupervised. A notable example is *Reinforcement Learning (RL)*, which does not require labeled data or a even training set. For more details on RL we refer to Chapter 10. Finally, when labels are not available for all the samples in the training set, we refer to *semisupervised learning*. Most of semisupervised methods are a combination of supervised and unsupervised algorithms. An example of semisupervised learning is *Active Learning* (see Chapter 4 for more details).

2.2 Batch vs online ML

Another manner to classify ML algorithms is based on whether they can improve their predictions by providing them with new data that may become available after the first training.

In *batch learning* the machine has to be trained over a fixed training data set, without being able to add more data to the training set. Therefore, batch ML methods cannot

Table 1: Classification of some important methods in ML. The asterisk denotes that not all variations of the method fall into this category.

ML Method	Supervised	Unsupervised	Online	Presented in this volume
Linear regression	✓			✓
Logistic regression	✓			✓
Polynomial regression	✓			✓
Lasso, Ridge	✓			✓
k-Nearest neighbors	✓			✓
Support vector machines (SVM)	✓			✓
Decision trees	✓			
Random forests	✓			
Artificial Neural Networks (ANN)	✓		✓	✓
Autoencoders		✓	✓	✓
Clustering		✓		✓
Principal Component Analysis (PCA)		✓	✓*	
Locally Linear Embedding (LLE)		✓		
Reinforcement Learning (RL)			✓	✓

improve their performance in a specific task with providing them with more data after the end of the training.

On the contrary, ML methods that support *online learning* allow to modify and increase the initial training set. Consequently, they are more flexible and suitable for large training sets, as they can be partially loaded into the memory of the computer and used whenever needed. Old data, over which the computer has been already trained, can be also erased to save space. Online learning is also very convenient when a constant flow of information exists, contrary to batch learning, in which the machine must be retrained over the whole data set. In other words, in batch learning, when new data become available the training set has to be updated and the training has to be repeated from scratch. This can be fine for some applications, but in many others it could have a very high computational and data storage cost.

Not all ML algorithms support online learning, see Table 1.

2.3 Instance-based vs model-based ML

One more way to categorize ML methods is based on the way data is learned in order to make predictions for data outside the training set. Two ML categories can be distinguished, *instance-based* (also known as *memory-based* or *lazy*) learning and *model-based* learning (also known as *physics-based* in some applications).

In instance-based learning, training data is simply interpolated in a high (usually) dimensional space. Then, new predictions are made based on how close or how similar new data are to those used for training. Similarity is measured on the basis of a distance measure, depending on the data and the problem at hand. In this sense data is "memorized" by the machine, thus the term memory-based. The quality of the predictions of data outside the training set is determined by how well new data are represented by the data of the training set. Instance-based approaches are straightforward to apply in any data set and they don't require any particular knowledge about the structure of the data or other characteristics that they might have. The hope of the user of instance-based methods is the machine to eventually identify by itself the hidden patterns in the data and give correct predictions even for data outside the training set. Examples of instance-based methods are *Artificial Neural Networks*, *Decision trees*, *Random forests*, *k-Nearest neighbors* and many *clustering* techniques, *Locally Linear Embedding (LLE)* and *Principal Component Analysis (PCA)*.

Despite the versatility and the many advantages of instance-based ML methods, they have an important drawback. In physics and engineering, we know that data have to respect at least some fundamental principles, like for instance the conservation of mass, of the energy and of the linear and angular momentum. Therefore, instance-/memory-based predictions that do not respect these conservation laws are unacceptable and can be even dangerous for applications. Data in this case has to be processed and learned by the machine under a model that includes the laws of physics.

Model-based machine learning adopts a model with some model parameters, which are optimized during training in order to optimally represent the data of the training set. Then the model is used to make predictions for unseen input data. Of course, if the adopted model is poor, the predictions will be poor as well.

Both instance-based and model-based approaches can introduce bias. The former because of limited data over which they were trained and the latter because of the model chosen. To fix the ideas, an example of instance-based and model-based ML is given in Figure 3. The training set contains the shear stress at failure (*predictor*) for a given normal stress (*feature*) of a series of experimental tests of a frictional interface. An instance-based method could give very poor predictions for unseen data, while a model-based prediction will be as good as the model is for describing the data. In this simple example, a Coulomb model was adopted for the model-based approach.

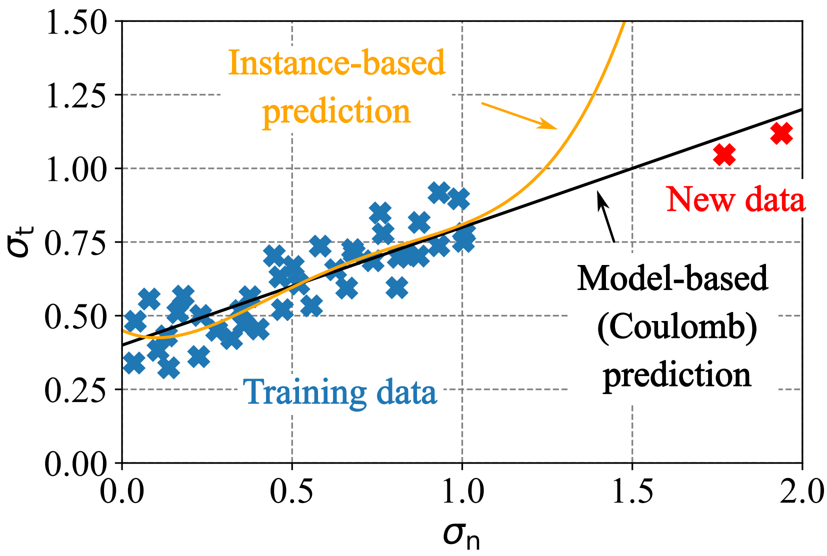

Figure 3: Instance-based vs model-based ML for predicting failure (slip) of a frictional interface based on experimental data (fictitious).

3 ML and Geomechanics

The applications of ML in science are nowadays numerous and increasing. The same holds for applications of ML in geomechanics. It is out of the scope of this chapter to provide a comprehensive literature review, but it is worth mentioning some research directions in *constitutive modeling*, *geotechnics*, *geophysics* and *image corelation* that can be a starting point for the interested reader.

Geomaterials are among the most complex materials to study and model. The main reason is that their mechanical response is governed by multiphysics couplings at multiple spatial and temporal scales, which result in a macroscopic strongly non-linear and dissipative response.

Empirical constitutive models are often used to describe the mechanical behavior of geomaterials. Classification and regression methods can be used for choosing the appropriate constitutive model that fits the best the experimental data (see [MPRP22, MH19, GBL+21, SKOM23, PEW20, ZYJ21] and references therein). Another challenging task is to capture the macroscopic behavior of geomaterials based on the the behavior of their microstructure and its evolution due to loading. Multiscale approaches can be employed for this purpose, requiring the solution of a boundary value problem (BVP) of an elementary cell of the microstructure at each point and time increment of the macroscopic analysis. Then, based on homogenization, the intrinsic constitutive behavior of the microstructure can be upscaled to the macroscopic level. For this purpose, mixed numerical schemes, such as the FE^2 method [Fey03, LVRSHOO19, EBC+16] and FEM$\times$DEM [NMCDD+11, NCCD14] are often employed, depending on the nature of the microstructure. However, the computational cost of these methods is extremely high for real-scale applications, if not prohibitive. ML and ANN are one of the most promising ways to speed-up this multiscale process.

Recent works have shown that ANN can successfully encapsulate several aspects of the constitutive behavior of the underlying microstructure and provide the necessary information to the macroscopic scale with reduced calculation cost (see [GGW91, LS03, MBC+19, LW19, HXFD20, VS21, ZHX21, WSD19, RKVDM21, BDMJ22, WXW23, SBV+21, ZZJ+23, PABT+21] to mention few). Going a step further, ANN can be designed in such a way to respect, by construction, the laws of physics [KKL+21, RPK19], symmetries [HWS20] and thermodynamics [MSVMB21, MS22, MS23, HBG+21] (see also Chapters 6 and 9). Once trained, these approaches can tremendously speed-up the solution of difficult multiscale problems, they can guarantee the respect of the thermodynamic restrictions in their predictions and enable the extraction of the hidden state variables of the material. The latter can, in turn, shed light on the importance and the role of specific micromechanisms to the overall macroscopic behavior of complex (geo-)materials. The above collection of ML approaches is enriched by the so-called "data-driven." methods, which present an alternative formulation, whereby optimal material states are sought within a dataset that most closely satisfy momentum and energy conservation principles [KSOA20, KO16, KO18, KOA21] (see also Chapter 5).

Moving to applications of ML in geotechnics, according to the recent review of Baghbani et al. [BCCR22] (see also [ZLL+21]), more than 1200 articles can be found in the literature starting from the early 90's. According to the same source, a net burst of production of scientific articles is observed after 2017. Notice, that more than half of these works use ANN. Several areas of geotechnical engineering are covered (see Figure 4), such as frozen soils and soil thermal properties, rock mechanics, subgrade soils and pavements, landslides (see [TCL+22] for a recent review), liquefaction, slope stability, shallow foundations, piles, tunneling and tunnel boring machines, dams, and unsaturated soils, among others. In 2018, the increasing interest of the geotechnical community in ML led [ZL22] to the creation of a new technical committee (TC) in the International Society for Soil Mechanics and Geotechnical Engineering (ISSMGE), entitled as TC309 "Machine Learning and Big Data" (`https://www.issmge.org/committees/technical-committees/impact-on-society/machine-learning`, see also TC304).

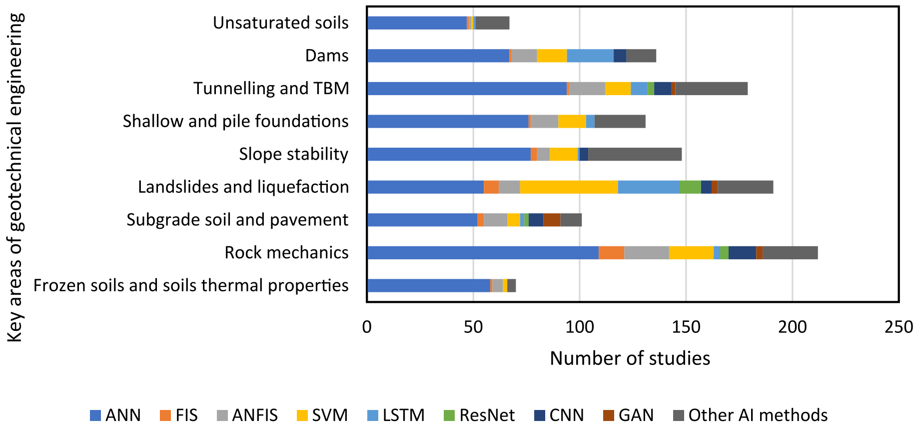

Figure 4: Number of published articles using ML for geotechnical applications (slightly modified from [BCCR22]).

ML has also promising applications in geophysics. For instance, it finds applications in geophysical exploration, reservoir engineering and drilling (see [SYR+21] and references therein). Moreover, it was used for creating synthetic accelerograms based on numerical simulations or databases of real earthquake signals [GC20, LGBC22]. ML was used as well in an attempt for earthquake prediction —the holy grail in seismology— showing that earthquake-like events in the laboratory could be predicted by identification by the machine of seismic precursor patterns [JRLPN+21, BSMM21, LTG+22, RY23]. RL was also used for controlling earthquake-like events [PS21] (see Chapter 10) for which more exact mathematical theories show that are controllable [Ste19, GOTSP].

Given the striking advances of ML in self-driving cars, object and face recognition [Bal15], it is natural to expect ML methods to also find numerous applications in image processing in experimental geomechanics and geotechnics. *Convolutional Neural Networks (CNNs)* (see Chapters 7 and 8 for more details) is the basic ingredient of most machine learning techniques used in image and video processing, without forgetting more traditional compression techniques that are also considered as ML (e.g. PCA). Boukhtache et al. [BAB+21] presents a review of Digital Image Correlation (DIC) with deep learning. We also refer to [CZX+23, DXD+23, BAB+23] for some recent developments. Accuracy is one of the main issues for those methods in order to outperform the current state of the art (see ALERT Doctoral School 2022 [AMHD22]). Focusing on granular materials, Stefano Buitrón et al. [CJM+23] propose a CNN to automatically distinguish properly segmented digital grains with up to 90% of accuracy, while Cheng et al. [CWX23] present a machine learning-based strategy to estimate the contact force chains of uniformly sized spherical granular materials using particle kinematics and inter-particle contact evolution data measured by X-ray micro-tomography.

4 Libraries for ML

Today, many libraries exist for machine learning (see `https://en.wikipedia.org/wiki/Machine_learning#Software` for an updated list). Most of the existing libraries provide a Python interface.

Some general purpose libraries for ML that are extensively used in this doctoral school are Numpy [HMW+20], Pandas [McK10], SciPy [VGO+20] and Scikit-learn [PVG+11]. For *Artificial Neural Networks*, TensorFlow [MAP+15] and PyTorch [PGM+19] are equally popular today (see Figure 5). Both offer parallelization and GPU support for training large ANN and handling large collections of data. In this doctoral school we use PyTorch (see Chapter 7 for more details). As always, the best library is the library that we know the best, provided that it is open-source and it allows us do what we want!

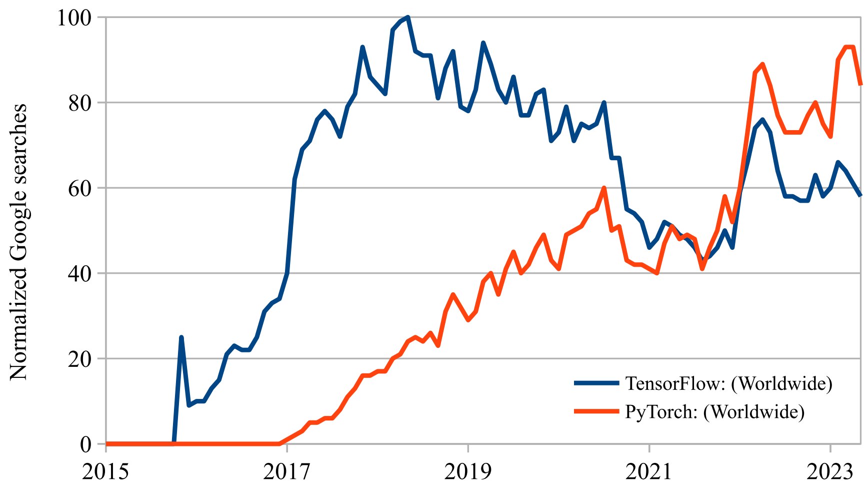

Figure 5: Normalized Google searches over time for TensorFlow and PyTorch ANN libraries.

5 Bias in ML and limitations

The success of ML algorithms in performing complicated and complex tasks makes them convenient tools for many applications. However, we have to be aware of their limitations. In other words, we have to use the right screwdriver for the right screw! In section 2.3 (see Figure 3), we showed how the choice of the ML method and the quality of the data can lead to biased predictions, even in a very simple problem.

Applications of ML algorithms have demonstrated gender [Kel19], racial [DF18, Per21], hiring [VVNR21] and other biases [Var23]. ML methods are algorithms which should be used with care and knowledge of the the underlying limitations. This is not different from the application of other methods. For instance, we know how wrong Finite Element predictions can be when the appropriate finite elements are not used, when convergence analyses are not performed or when we want to model a softening material without regularizing the underlying mathematical problem. The inappropriate use of specific methods has traditionally led to spectacular failures with uncountable casualties and economic loss. ML methods will not be an exception, unless we understand them better and use them with caution.

As far as it concerns ML and geomechanics, luckily, we have at our disposal established and undeniable principles that have to be respected in any application. Conservation principles, the laws of thermodynamics and other physics at various spatiotemporal scales should be incorporated into the ML algorithms in order to assure adequate and safe predictions. Physics-based approaches in ML (see Chapters 5, 6 and 9) gain more and more attention from the scientific community and can become the natural environment for marrying the established know-how of decades of research in geomechanics with ML. This could give fresh ideas and an opportunity to push further the current state of the art in our fields. The incorporation of physics in ML could eventually inspire new ML methods in other domains too, which today suffer from inevitable bias (see social sciences where data are always limited).

Another limitation of most of the available ML methods today is their greediness for data. We do not need thousand of photos to teach to a child what a car is! With ML though we need tons of data! In many applications there is abundance of data (see for instance the data that are produced, but not saved, at each increment at each Gauss point during a non-linear Finite Element analysis in a geomechanics problem). In some other applications though, data may not be enough (see for example X-ray scans of thousand of specimens [TLA$^+$20]).

Noise in the data, overfitting and underfitting are some other points that we have to pay attention to. Data with a lot of noise can make hard the learning process and render the predictions unreliable and of poor generalization (see also Chapters 2 and 7 to 9). Data preparation to assure good quality is of paramount importance then.

The choice of a ML method allowing to fit data in a very high dimensional space can lead to overfitting. In this case the training data can be very well represented, but predictions for unseen data can be far off. The opposite happens with underfitting, where the space for fitting is too low for identifying and reproducing the inherent patterns in the data. A classical example of overfitting and underfitting in polynomial regression is shown in Figure 6. Thankfully, there are ways for estimating in practice how well a ML model performs. This is mainly achieved by splitting the training set into training, validation and test data. *Regularization* methods can also help in avoiding this problem. For more details we refer to Chapters 2, 4 and 7.

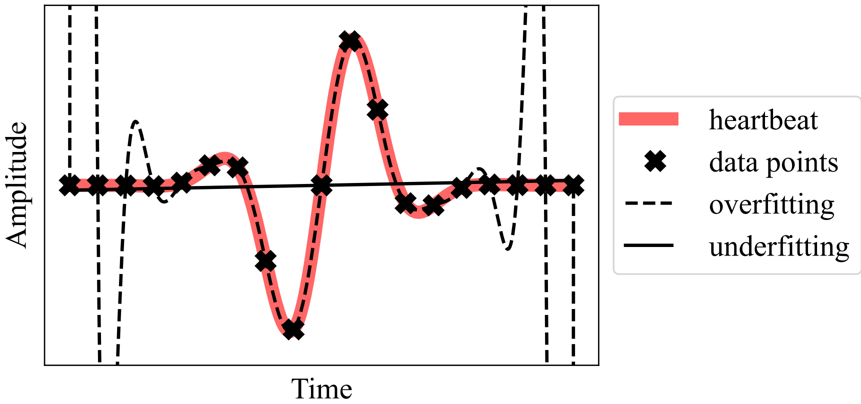

Figure 6: Data points of a synthetic heartbeat, interpolated by two polynomial functions. The high degree polynomial fits exactly the data points, but overfits the signal. The low degree polynomial fits poorly the data points and underfits the signal. Both regressions are not acceptable, because they either predict a superhuman heartbeat or a dead person!

6 What to expect from this volume?

This volume aims at explaining what Machine Learning is, what its main methods are and how they can be used for solving problems in (geo-)mechanics. Most of the chapters were written having in mind to provide a pedagogical introduction to the most important methods in machine learning and the fundamental notions behind them.

It is not possible to cover all the available ML methods in the existing literature and, without any doubt, many important methods were inevitably left out. For instance we won't discuss about genetic algorithms, principal component analysis and related methods, particle swarm optimization, fuzzy logic algorithms, ML methods based on control theory and many others. We hope, however, to have provided a good selection of ML methods for an introductory course.

By the end of this school we expect the students to have:

- demystified and understood what ML is;
- be conscious of the fundamental notions of the most important ML methods;
- used ML in simple examples, got aware of pitfalls and understood the need for physics- and (geo-)mechanics-based ML methods for solving problems in (geo-)mechanics.

The courses are addressed to undergraduate and graduate level. The minimum requirements for accessing them are:

- knowledge of Python programming language[1];
- basic concepts in mathematics (calculus, elements of differential calculus and of numerical analysis).
- have some nice problems in mind that could combine ML and geomechanics!

Updated versions of the chapters of this volume and the python scripts supporting this volume and hands-on sessions are available at:
`https://github.com/alert-geomaterials/2023-doctoral-school`.

Acknowledgments

The author would like to thank all the authors of the chapters of this volume for their time and effort to provide comprehensive and detailed lecture notes for the ALERT 2023 Doctoral School.
Moreover, the author would like to acknowledge the support of the European Research Council (ERC) under the European Union's Horizon2020 research and innovation program (Grant agreement no. 757848 CoQuake).

References

[All94] L. Victor Allis. *Searching for solutions in games and artificial intelligence*. 1994. OCLC: 1110390182.

[AMHD22] Edward Ando, Benjy Marks, Ryan Hurley, and Joshua A Dijksman. *Advanced Experimental Geomechanics*. Doctoral School lecture notes. The Alliance of Laboratories in Europe for Education, Research and Technology, 2022.

[BAB+21] S. Boukhtache, K. Abdelouahab, F. Berry, B. Blaysat, M. Grédiac, and F. Sur. When Deep Learning Meets Digital Image Correlation. *Optics and Lasers in Engineering*, 136:106308, January 2021.

[1]For learning Python, we propose the book "Python Crash Course: A Hands-On, Project-Based Introduction to Programming" by Eric Matthes and/or many excellent tutorials that can be found on the internet

[BAB+23] S. Boukhtache, K. Abdelouahab, A. Bahou, F. Berry, B. Blaysat, M. Grédiac, and F. Sur. A lightweight convolutional neural network as an alternative to DIC to measure in-plane displacement fields. *Optics and Lasers in Engineering*, 161:107367, February 2023.

[Bal15] Stephen Balaban. Deep learning and face recognition: the state of the art. page 94570B, Baltimore, Maryland, United States, May 2015.

[BCCR22] Abolfazl Baghbani, Tanveer Choudhury, Susanga Costa, and Johannes Reiner. Application of artificial intelligence in geotechnical engineering: A state-of-the-art review. *Earth-Science Reviews*, 228:103991, May 2022.

[BDMJ22] Sébastien Becker, Emilie Devijver, Rémi Molinier, and Noël Jakse. Unsupervised topological learning approach of crystal nucleation. *Scientific Reports*, 12(1):3195, February 2022.

[BSMM21] Gregory C. Beroza, Margarita Segou, and S. Mostafa Mousavi. Machine learning and earthquake forecasting—next steps. *Nature Communications*, 12(1):4761, August 2021.

[CJM+23] Stefano Buitrón Cevallos, Alex X. Jerves, Utkarsh Mital, David A. Medina, V. Santiago Quinteros, Maurizio Mulas, and Øyvind Torgersrud. Towards a more accurate characterization of granular media 2.0: Involving AI in the process. *Computers and Geotechnics*, 160:105510, August 2023.

[CWX23] Zhuang Cheng, Jianfeng Wang, and Wei Xiong. A machine learning-based strategy for experimentally estimating force chains of granular materials using X-ray micro-tomography. *Géotechnique*, pages 1–13, February 2023.

[CZX+23] Xiangnan Cheng, Shichao Zhou, Tongzhen Xing, Yicheng Zhu, and Shaopeng Ma. Solving digital image correlation with neural networks constrained by strain-displacement relations. *Optics Express*, 31(3):3865, January 2023.

[DF18] Julia Dressel and Hany Farid. The accuracy, fairness, and limits of predicting recidivism. *Science Advances*, 4(1):eaao5580, January 2018.

[DXD+23] Xiaocen Duan, Hongwei Xu, Runfeng Dong, Feng Lin, and Jianyong Huang. Digital image correlation based on convolutional neural networks. *Optics and Lasers in Engineering*, 160:107234, January 2023.

[EBC+16] A P Van Den Eijnden, P Bésuelle, F. Collin, René Chambon, and Jacques Desrues. Modeling the strain localization around an underground gallery with a hydro-mechanical double scale model ; effect of anisotropy. *Computers and Geotechnics*, 2016.

[Fey03] Frédéric Feyel. A multilevel finite element method (FE2) to describe the response of highly non-linear structures using generalized continua. *Computer Methods in Applied Mechanics and Engineering*, 192(28-30):3233–3244, 2003. ISBN: 0045-7825.

[GBL+21] Zhenfei Guo, Ruixiang Bai, Zhenkun Lei, Hao Jiang, Da Liu, Jianchao Zou, and Cheng Yan. CPINet: Parameter identification of path-dependent constitutive model with automatic denoising based on CNN-LSTM. *European Journal of Mechanics - A/Solids*, 90:104327, November 2021.

[GC20] Filippo Gatti and Didier Clouteau. Towards blending Physics-Based numerical simulations and seismic databases using Generative Adversarial Network. *Computer Methods in Applied Mechanics and Engineering*, 372:113421, December 2020.

[GGW91] J Ghaboussi, J H Garrett, and X Wu. Knowledge Based Modeling of Material Behavior with Neural Networks. *Journal of Engineering Mechanics*, 117(1):132–153, 1991.

[GOTSP] Diego Gutiérrez-Oribio, Georgios Tzortzopoulos, Ioannis Stefanou, and Franck Plestan. Earthquake Control: An Emerging Application for Robust Control. Theory and Experimental Tests. *Accepted for publication in IEEE Transactions on Control Systems Technology*. arXiv: 2203.00296.

[Gé19] Aurélien Géron. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media, 2019.

[HBG+21] Quercus Hernandez, Alberto Badías, David González, Francisco Chinesta, and Elías Cueto. Deep learning of thermodynamics-aware reduced-order models from data. *Computer Methods in Applied Mechanics and Engineering*, 379:113763, June 2021.

[HMW+20] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020. Publisher: Springer Science and Business Media LLC.

[HWS20] Yousef Heider, Kun Wang, and WaiChing Sun. SO(3)-invariance of informed-graph-based deep neural network for anisotropic elastoplastic materials. *Computer Methods in Applied Mechanics and Engineering*, 363:112875, 2020.

[HXFD20] Daniel Z Huang, Kailai Xu, Charbel Farhat, and Eric Darve. Learning Constitutive Relations from Indirect Observations Using Deep Neural Networks. *Journal of Computational Physics*, page 109491, 2020. Publisher: Elsevier.

[JRLPN+21] Paul A. Johnson, Bertrand Rouet-Leduc, Laura J. Pyrak-Nolte, Gregory C. Beroza, Chris J. Marone, Claudia Hulbert, Addison Howard, Philipp Singer, Dmitry Gordeev, Dimosthenis Karaflos, Corey J. Levinson, Pascal Pfeiffer, Kin Ming Puk, and Walter Reade. Laboratory earthquake forecasting: A machine learning competition. *Proceedings of the National Academy of Sciences*, 118(5):e2011362118, February 2021.

[Kel19] Leo Kelion. Apple's 'sexist' credit card investigated by US regulator. *BBC*, 2019.

[KKL+21] George Em Karniadakis, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, May 2021.

[KO16] Trenton Kirchdoerfer and Michael Ortiz. Data-driven computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 304:81–101, 2016. arXiv: 1510.04232 Publisher: Elsevier B.V.

[KO18] Trenton Kirchdoerfer and Michael Ortiz. Data-driven computing in dynamics. *International Journal for Numerical Methods in Engineering*, 113(11):1697–1710, 2018. Publisher: Wiley Online Library.

[KOA21] K Karapiperis, M Ortiz, and J.E. Andrade. Data-Driven nonlocal mechanics: Discovering the internal length scales of materials. *Computer Methods in Applied Mechanics and Engineering*, 386:114039, December 2021. Publisher: Elsevier B.V.

[KSOA20] K. Karapiperis, L. Stainier, M. Ortiz, and J.E. Andrade. Data-Driven Multiscale Modeling in Mechanics. *Journal of the Mechanics and Physics of Solids*, (December):104239, 2020. Publisher: Elsevier Ltd.

[LGBC22] Fanny Lehmann, Filippo Gatti, Michaël Bertin, and Didier Clouteau. Machine learning opportunities to conduct high-fidelity earthquake simulations in multi-scale heterogeneous geology. *Frontiers in Earth Science*, 10:1029160, November 2022.

[LS03] Marek Lefik and Bernhard A. Schrefler. Artificial neural network as an incremental non-linear constitutive model for a finite element code. *Computer Methods in Applied Mechanics and Engineering*, 192(28-30):3265–3283, 2003. Publisher: Elsevier ISBN: 3949827560.

[LTG+22] Laura Laurenti, Elisa Tinti, Fabio Galasso, Luca Franco, and Chris Marone. Deep learning for laboratory earthquake prediction and autoregressive forecasting of fault zone stress. *Earth and Planetary Science Letters*, 598:117825, November 2022.

[LVRSHOO19] Oriol Lloberas Valls, Marcelo Raschi Schaw, Alfredo Edmundo Huespe, and Xavier Oliver Olivella. Reduced Finite Element square techniques (RFE2): towards industrial multiscale FE software. In *COMPLAS 2019: XV International Conference on Computational Plasticity: Fundamentals and Applications*, pages 157–169. International Centre for Numerical Methods in Engineering (CIMNE), 2019.

[LW19] Zeliang Liu and C. T. Wu. Exploring the 3d architectures of deep material network in data-driven multiscale mechanics. *Journal of the Mechanics and Physics of Solids*, 127:20–46, 2019. arXiv: 1901.04832 Publisher: Elsevier.

[MAP+15] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, 2015.

[MBC+19] M Mozaffar, R Bostanabad, W Chen, K Ehmann, Jian Cao, and M A Bessa. Deep learning predicts path-dependent plasticity. *Proceedings of the National Academy of Sciences*, 116(52):26414–26420, 2019. Publisher: National Acad Sciences.

[McK10] Wes McKinney. Data Structures for Statistical Computing in Python. In Stéfan van der Walt and Jarrod Millman, editors, *Proceedings of the 9th Python in Science Conference*, pages 56 – 61, 2010.

[MH19] Lukas Morand and Dirk Helm. A mixture of experts approach to handle ambiguities in parameter identification problems in material modeling. *Computational Materials Science*, 167:85–91, September 2019.

[Mit97] Tom M. Mitchell. *Machine learning*. McGraw-Hill series in Computer Science. McGraw-Hill, New York, nachdr. edition, 1997.

[MPRP22] Armando Marques, André Pereira, Bernardete Ribeiro, and Pedro A. Prates. On the Identification of Material Constitutive Model Parameters Using Machine Learning Algorithms. *Key Engineering Materials*, 926:2146–2153, July 2022.

[MS22] Filippo Masi and Ioannis Stefanou. Multiscale modeling of inelastic materials with Thermodynamics-based Artificial Neural Networks (TANN). *Computer Methods in Applied Mechanics and Engineering*, 398:115190, August 2022.

[MS23] Filippo Masi and Ioannis Stefanou. Evolution TANN and the identification of internal variables and evolution equations in solid mechanics. *Journal of the Mechanics and Physics of Solids*, page 105245, February 2023.

[MSVMB21] Filippo Masi, Ioannis Stefanou, Paolo Vannucci, and Victor Maffi-Berthier. Thermodynamics-based Artificial Neural Networks for constitutive modeling. *Journal of the Mechanics and Physics of Solids*, 147:104277, February 2021. arXiv: 2005.12183.

[NCCD14] Trung Kien Nguyen, Gaël Combe, Denis Caillerie, and Jacques Desrues. FEM × DEM modelling of cohesive granular materials: Numerical homogenisation and multi-scale simulations. *Acta Geophysica*, 62(5):1109–1126, August 2014.

[NMCDD+11] Gaël Nitka Michałand Combe, Cristian Dascalu, Jacques Desrues, Michał Nitka, Gaël Combe, Cristian Dascalu, and Jacques Desrues. Two-scale modeling of granular materials: a DEM-FEM approach. *Granular Matter*, 13(3):277–281, March 2011.

[PABT+21] Grace C. Y. Peng, Mark Alber, Adrian Buganza Tepole, William R. Cannon, Suvranu De, Savador Dura-Bernal, Krishna Garikipati, George Karniadakis, William W. Lytton, Paris Perdikaris, Linda Petzold, and Ellen Kuhl. Multiscale Modeling Meets Machine Learning: What Can We Learn? *Archives of Computational Methods in Engineering*, 28(3):1017–1037, May 2021.

[Per21] Sidney Perkowitz. The Bias in the Machine: Facial Recognition Technology and Racial Disparities. *MIT Case Studies in Social and Ethical Responsibilities of Computing*, February 2021.

[PEW20] Mehdi Pouragha, Mahdad Eghbalian, and Richard Wan. A mote on applicability of Artificial Intelligence to constitutive modeling of geomaterials. *Journal of Machine Learning for Modeling and Computing*, 1(2):157–170, 2020.

[PGM+19] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[PS21] Efthymios Papachristos and Ioannis Stefanou. Controlling earthquake-like instabilities using artificial intelligence, April 2021. arXiv:2104.13180 [physics].

[PVG+11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[RKVDM21] I.B.C.M. Rocha, P. Kerfriden, and F.P. Van Der Meer. On-the-fly construction of surrogate constitutive models for concurrent multiscale mechanical analysis through probabilistic machine learning. *Journal of Computational Physics: X*, 9:100083, January 2021.

[RPK19] Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019. Publisher: Elsevier Inc.

[RY23] Nurafiqah Syahirah Md Ridzwan and Siti Harwani Md. Yusoff. Machine learning for earthquake prediction: a review (2017–2021). *Earth Science Informatics*, 16(2):1133–1149, June 2023.

[SBV+21] Xiao Sun, Bahador Bahmani, Nikolaos N. Vlassis, WaiChing Sun, and Yanxun Xu. *Data-driven discovery of interpretable causal relations for deep learning material laws with uncertainty propagation*. Springer Berlin Heidelberg, 2021. arXiv: 2105.09980 Publication Title: Granular Matter ISSN: 1434-7636.

[Sha50] Claude E. Shannon. XXII. Programming a computer for playing chess. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 41(314):256–275, March 1950.

[SHM+16] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, January 2016.

[SKOM23] Robin Schulte, Cavid Karca, Richard Ostwald, and Andreas Menzel. Machine learning-assisted parameter identification for constitutive models based on concatenated loading path sequences. *European Journal of Mechanics - A/Solids*, 98:104854, March 2023.

[Ste19] Ioannis Stefanou. Controlling Anthropogenic and Natural Seismicity: Insights From Active Stabilization of the Spring-Slider Model. *Journal of Geophysical Research: Solid Earth*, 124(8):8786–8802, August 2019.

[SYR+21] Anirbid Sircar, Kriti Yadav, Kamakshi Rayavarapu, Namrata Bist, and Hemangi Oza. Application of machine learning and artificial intelligence in oil and gas industry. *Petroleum Research*, 6(4):379–391, December 2021.

[TCL+22] Faraz S. Tehrani, Michele Calvello, Zhongqiang Liu, Limin Zhang, and Suzanne Lacasse. Machine learning and landslide studies: recent advances and applications. *Natural Hazards*, 114(2):1197–1245, November 2022.

[TF07] John Tromp and Gunnar Farnebäck. Combinatorics of Go. In David Hutchison, Takeo Kanade, Josef Kittler, Jon M. Kleinberg, Friedemann Mattern, John C. Mitchell, Moni Naor, Oscar Nierstrasz, C. Pandu Rangan, Bernhard Steffen, Madhu Sudan, Demetri Terzopoulos, Doug Tygar, Moshe Y. Vardi, Gerhard Weikum, H. Jaap Van Den Herik, Paolo Ciancarini, and H. H. L. M. Donkers, editors, *Computers and Games*, volume 4630, pages 84–99. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007. Series Title: Lecture Notes in Computer Science.

[TLA+20] Alessandro Tengattini, Nicolas Lenoir, Edward Andò, Benjamin Giroud, Duncan Atkins, Jerome Beaucour, and Gioacchino Viggiani. NeXT-Grenoble, the Neutron and X-ray tomograph in Grenoble. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 968:163939, July 2020.

[Var23] P. S. Varsha. How can we manage biases in artificial intelligence systems – A systematic literature review. *International Journal of Information Management Data Insights*, 3(1):100165, April 2023.

[VGO+20] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy

1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

[VS21] Nikolaos Napoleon Vlassis and Waiching Sun. Component-based machine learning paradigm for discovering rate-dependent and pressure-sensitive level-set plasticity models. *Journal of Applied Mechanics*, pages 1–13, October 2021.

[VVNR21] Alexis Megan Votto, Rohit Valecha, Peyman Najafirad, and H. Raghav Rao. Artificial Intelligence in Tactical Human Resource Management: A Systematic Literature Review. *International Journal of Information Management Data Insights*, 1(2):100047, November 2021.

[WSD19] Kun Wang, WaiChing Sun, and Qiang Du. A cooperative game for automated learning of elasto-plasticity knowledge graphs and models with AI-guided experimentation. *Computational Mechanics*, 64(2):467–499, August 2019.

[WXW23] Mengmeng Wu, Zhangqi Xia, and Jianfeng Wang. Constitutive modelling of idealised granular materials using machine learning method. *Journal of Rock Mechanics and Geotechnical Engineering*, 15(4):1038–1051, April 2023.

[ZHX21] Xu-Hui Zhou, Jiequn Han, and Heng Xiao. Learning nonlocal constitutive models with neural networks. *Computer Methods in Applied Mechanics and Engineering*, 384:113927, October 2021.

[ZL22] Wengang Zhang and Zhongqiang Liu. Editorial for machine learning in geotechnics. *Acta Geotechnica*, 17(4):1017–1017, April 2022.

[ZLL+21] Wengang Zhang, Hongrui Li, Yongqin Li, Hanlong Liu, Yumin Chen, and Xuanming Ding. Application of deep learning algorithms in geotechnical engineering: a short critical review. *Artificial Intelligence Review*, 54(8):5633–5673, December 2021.

[ZYJ21] Pin Zhang, Zhen-Yu Yin, and Yin-Fu Jin. State-of-the-Art Review of Machine Learning Applications in Constitutive Modeling of Soils. *Archives of Computational Methods in Engineering*, 28(5):3661–3686, August 2021.

[ZZJ+23] Ning Zhang, Annan Zhou, Yin-Fu Jin, Zhen-Yu Yin, and Shui-Long Shen. An enhanced deep learning method for accurate and robust modelling of soil stress–strain response. *Acta Geotechnica*, February 2023.