AI in Architectural Design: Adaptive Tabular Model for Classifying Intelligent Tools

Louis Roobaert[1,2*], and *Damien* Claeys[1,2]

[1]TSA-LAB, LAB, UCLouvain, Rue Wafelaerts 47/51, 1060 Bruxelles, Belgique
[2]LOCI, UCLouvain, Rue Wafelaerts 47/51, 1060 Bruxelles, Belgique

Abstract. This research proposes an adaptive model in the form of a periodic table to classify intelligent tools used in architectural design. Inspired by Mendeleev's periodic table, this model aims to organize various artificial learning algorithms based on their complexity, learning method, and the phases of artificial intelligence (AI) processes. The objective is to make the functions and potential of design assistance tools accessible to non-experts while facilitating the understanding of possible combinations between these algorithms. The proposed periodic table is structured around eight periods representing increasing levels of algorithmic complexity, six families based on learning methods and their depth, and four blocks representing the typical phases of data processing: analysis, model development, decision-making, and causal inference. Each algorithm is identified by a label that provides information about the types of inputs and outputs it processes, its accuracy, its learning duration, and whether it models linear or nonlinear relationships. This categorization allows for a better understanding and comparison of various algorithms in relation to their specific applications in architectural design, where AI does not replace human cognition but enriches and extends it.

Keywords. artificial intelligence, architectural design, periodic table, machine learning, deep learning.

Reference. Roobaert, Louis & Claeys, D. (2024). "AI in architectural design: Adaptive tabular model for classifying intelligent tools". International conference AI & Automation in Architectural Research and Practice (AIARP 2024), ENSA Paris Malaquais, 10/10/2024, 11h15.

1. Introduction

As the digitization of reality profoundly affects all sectors of human activity, the increasing use of intelligent design assistance tools is changing the practices of designing the built environment. Recent comparative studies between the cognitive spectra of humans and machines analyze potential hybridizations within ecosystems of designers [1,2]. These studies explore collaborative working modalities in which artificial cognition does not replace human cognition but extends and enriches it. Indeed, tools equipped with artificial

* Corresponding author: louis.roobaert@uclouvain.be

intelligence (AI) possess unprecedented capabilities in specific tasks, such as mass data processing, performing complex calculations, or making predictions from a dataset. The thoughtful integration of intelligent tools into architectural design processes would therefore enrich current practices while leading them to reinvent themselves.

However, the initial training of architects includes little on the subject of AI. Moreover, practitioners rarely benefit from continuing education in this area. AI is often viewed with distrust as a black box, designed and implemented by others, whose complexity and constant evolution stifle the autonomy and creativity of architectural designers. More precisely, knowledge of AI forms and their potential hybridizations is hardly accessible to non-experts. When used, it is mostly limited to automating repetitive tasks, rather than as a genuine design partner.

Assuming the risk is not that AI will replace humans, but that a limited number of humans with AI knowledge will dominate all others, repeated efforts are necessary to ensure equitable access for both the initiated and the uninitiated. Specifically, the goal is to facilitate the knowledge of intelligent design assistance tools by proposing a model of adaptive tabular structure, allowing for their objective classification. Inspired by the periodic table of chemical elements, the table proposed here aims to reinforce, among non-experts, an intuitive understanding of the potentialities of these tools and their possible combinations, before making them partners to collaborate with in architectural design processes.

In this text, AI and its sub-parts are first defined globally, based on questions related to data management and learning. Next, a model of tabular classification is established (periods, families, blocks, labels). Selected for their links to the field of architectural design, intelligent tools are placed within the table's structure. Finally, the relevance of the model is discussed.

2. Artificial Intelligence

Given that the concept of intelligence lacks a definitive definition, the term artificial intelligence (AI) has been controversial since its introduction [3]. Rather than attempting to define what AI is, it is preferable to study the constitution of the discipline, its objectives, and the various approaches used to achieve them. As a problem-solving tool based on data, AI aims to "emulate intelligent behaviors through a computer program without replicating the corresponding functioning of the human being" [4].

Two historical approaches are regularly distinguished: (1) connectionism, which models cognition as an emergent process, inspired by the brain's physiological structure to create a universal neural machine, giving rise to the first artificial neural networks (NN) [5,6]; and (2) symbolism, leading to cognitivism, modeling mental processes on the symbolic functioning of human language through the manipulation of signs based on logical rules, to establish a universal symbolic machine, setting up the first conversational agents and expert systems [7–9]. Moreover, the two approaches are regularly hybridized [9].

AI has gradually formed through the convergence of several fields [10] (see figure 1): algorithmics (Alg); computer science (CS); (3) statistics (Sta); (4) neurosciences (NS) and (5) cognitive sciences (CS). Many other sub-domains are clearly distinguished but are not covered here. AI is divided into two main categories: machine learning (ML), and deep learning (DL).

In response to the exponential growth of Artificial Intelligence Systems (AIS) [11], initiatives such as the Foundation Model Transparency Index (FMTI) [12] or the Open Ended Electronic Intelligent Robot Operating System (ONEIROS) program [13] aim to classify and compare these systems. Complementing these studies, this article offers a different approach. The goal is to establish a mapping of the various techniques specific to the conceptual level of artificial intelligence—when it is not transposed into a system—. The idea is to establish a periodic table of the different algorithms related to artificial learning in architectural design, which can be combined to generate AIS.

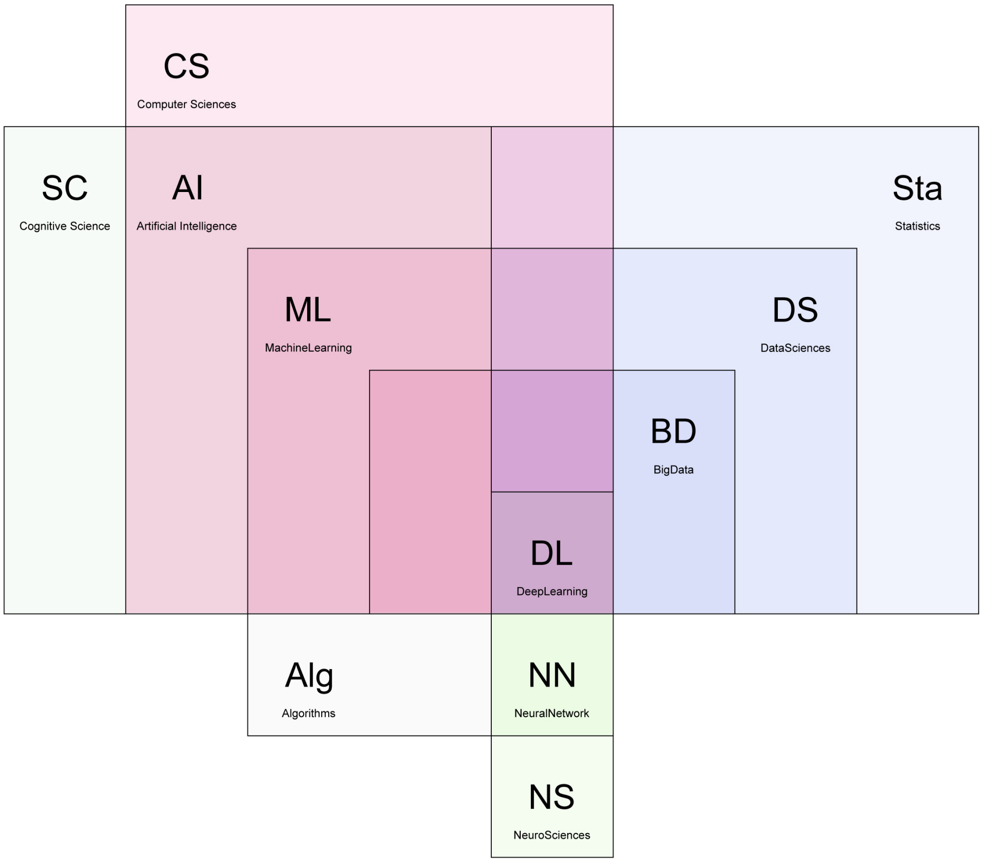

Fig. 1. The Discipline of Artificial Intelligence and its major subdivisions.

3. The Structure of the Periodic Table

A structure is proposed to create a table for classifying various intelligent tools for architectural design assistance.

Attributed to the Russian chemist Dmitri I. Mendeleev [14], the periodic table of elements has been adapted numerous times before appearing in its current form. The known chemical elements are classified based on their physical and chemical properties, while allowing for the future addition of yet unknown elements. The structure of the current table includes: seven horizontal periods (horizontal alignment of elements with the same electronic energy level), eighteen vertical families (vertical alignment of elements with the same number of valence electrons), and four blocks (groups of elements whose valence electrons belong to the same electronic sub-shells).

By analogy, the periodic table of artificial learning adopts a similar configuration: (1) eight horizontal periods provided by a vertical scale describing the algorithmic complexity level of the available tools; (2) six vertical families distinguished by a basic stratification, corresponding to the different learning methods utilized by the tools (supervised, unsupervised, reinforcement) and the intensity of this learning (shallow, deep); (3) four blocks horizontally repeat this stratification by reproducing the sequence of the four usual phases of data processing during an AI process (analysis, modeling, decision, inference); (4) a label individually identifies each element by providing additional information. The following four sections explain the various structural elements of the table: the periods, families, blocks, and labels.

3.1. Algorithmic Complexity Levels (Periods)

The efficiency of intelligent tools is formally distinguished here according to their level of algorithmic complexity, as it is calculated independently of their execution environment [15]. In theoretical computer science, the complexity of a problem measures its difficulty, both spatially (memory space) and temporally (computation time). Each algorithm can be categorized into an efficiency class, each rigorously described by asymptotic orders of magnitude. Introduced in the late 19th century through number theory [16] and later widely adopted in computer science to analyze algorithmic complexity [17,18], Big O notation designates the complexity order of an algorithm, while the data complexity is expressed by the variable n. It provides a standardized framework for evaluating the efficiency of algorithms based on their temporal or spatial complexity, without needing to implement and empirically measure their performance.

In the table, intelligent tools are classified into eight periods according to increasing complexity: (1) constant $O(1)$; (2) logarithmic $O(\log n)$; (3) linear $O(n)$; (4) linearithmic $O(n \log n)$; (5) quadratic $O(n^2)$; (6) cubic $O(n^3)$; (7) exponential $O(2n)$; (8) factorial $O(n!)$. Thus, the complexity level of an $O(1)$ algorithm does not vary with the size of the data, whereas an $O(n^2)$ algorithm sees its execution time quadruple if the size of the data doubles. By plotting the data size n on the x-axis and the processing time t on the y-axis, algorithms can be compared graphically: the steeper the curve, the higher the algorithmic complexity [19].

3.2. Learning Methods (Families)

In the table, intelligent tools are classified based on the type of learning method, organized into families. Two criteria determine these families: (1) the learning mode of the tool ((I) supervised, (II) unsupervised, (III) reinforcement); (2) the intensity of this learning ((a) shallow, (b) deep). As a result, the families are arranged into six categories:

(1) supervised and shallow learning (I.a – N): This uses labeled datasets to precisely associate inputs with outputs but requires human intervention [20];
(2) unsupervised and shallow learning (II.a – S): This allows algorithms to detect patterns in datasets without explicit directives, such as through clustering or dimensionality reduction [21];
(3) reinforcement and shallow learning (III.a – R): This enables algorithms to learn from experience using a system of rewards and penalties [22];
(4) supervised and deep learning (I.b – NP): This relies on neural networks to improve prediction accuracy, for example in voice recognition [23];
(5) unsupervised and deep learning (II.b – SP): These methods are used to discover structures in unlabeled data, allowing the learning of multiple levels of features or data representations [24];
(6) reinforcement and deep learning (III.b – RP): This combines deep learning with reinforcement learning, enabling an agent to make optimal decisions based on rewards [25].

3.3. Phases of Intelligent Processes (Blocks)

Problem-solving processes using intelligent tools go through four fundamental phases related to data management and learning. From this, based on the six learning families, four blocks are defined:

(1) Data analysis (input processing and collection): The designer collects and processes available data, whether from past experiences or specifically gathered for the task. This phase includes classifying, filtering, and organizing data to facilitate visualization and recognition of recurring patterns.

(2) Model development (creation and adjustment): The analyzed data is used to develop predictive models capable of making forecasts. These models rely on past data characteristics to generate scenarios and adjust proposals.

(3) Decision-making (based on model results): Using the model results, this phase involves evaluating various options and selecting the best strategy, balancing exploration (search for new solutions) and exploitation (use of proven strategies) [22].

(4) Causal inference (verification and interpretation): This phase consists of verifying and interpreting the decision results by simulating the effects of the choices made. It tests the robustness and generalization of the models, linking causes to observed effects to improve decision-making [26].

These steps are comparable to the methods used by data scientists seeking to build an algorithmic system to solve a given problem. By analogy, a partial parallel can be drawn between these phases and the tasks encountered in the architectural design process. Architectural design can be seen as an attempt to solve a particular range of "ill-defined problems" [27], requiring specific resolution methods. Rather than being linear, this process is more "circular," "iterative," and "reflexive," alternating between phases of "convergence" and "divergence" before arriving at a "suboptimal solution" [27,28]. In this context, the designer operates a dynamic combination of intelligent design tools, constantly navigating between them to refine the project being conceived.

3.4. Properties of Intelligent Tools (Labels)

To better understand the use of algorithms in architectural design, each algorithm is identified by a label providing five additional pieces of information:

(1) Type of inputs: Describes the nature of the inputs processed by the algorithm, which can be numerical (N), audio (A), video (V), or written (E), with some algorithms capable of handling multiple types of data [29].

(2) Type of outputs: Outputs can be continuous (numerical values), categorical (classification), sequential (ordered values), structured (creation of a new label), or in the form of rankings or probabilities [30,31].

(3) Accuracy (or "positive predictive value"): Measures the proportion of correct predictions among the positive predictions (true positives compared to false positives), with a scale from 1* to 4*.

(4) Learning duration: Indicates the speed at which the algorithm learns from training data before reaching a sufficient performance or accuracy threshold, rated from 1* (slow) to 4* (very fast) [32].

(5) Linearity: Specifies whether the algorithms model linear or nonlinear relationships, depending on whether changes in the inputs are directly proportional to those in the outputs, although some algorithms can model both types of relationships [32].

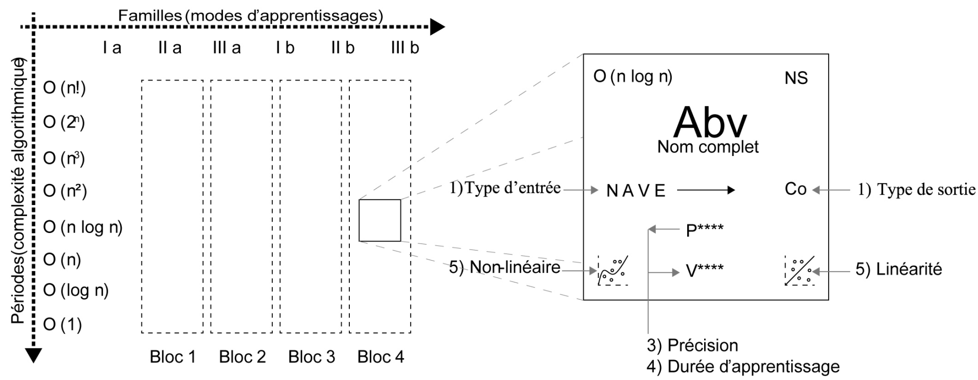

Fig. 2. (a) Structure of the table using periods, families, and blocks. (b) Structure of a label.

4. Placement of Algorithmic Varieties in the Table

After establishing the general structure of the tabular model of intelligent tools (periods, families, blocks, labels), the relevance of this model is tested by a non-exhaustive positioning of various algorithms. The adaptive nature of the tabular model includes the possibility to adapt and add other algorithms, thus reflecting the dynamic and cyclical nature of design processes and the constant implementation of new intelligent tools. The algorithms presented below are described block by block, successively covering the four phases of data processing that constitute intelligent problem-solving processes: (1) data analysis; (2) model development; (3) decision making; (4) causal inference.

4.1. Data Analysis (Block 1)

This block contains the set of algorithms used for data processing, facilitating their visualization. There are three main categories of algorithms, each comprising a variety of sub-algorithms [33]: (1) Dimensionality reduction, which allows for the processing of high-dimensional data where the number of variables, labels, and attributes prevents human cognition from identifying correlations [34]; (2) Clustering, which groups data into homogeneous subsets, revealing internal structures and patterns; (3) Classification, which assigns labels to unlabeled data based on learned models [35]. The choice of algorithm class depends on the structure of the data available to the designer, while the type of data (text, numbers, videos, images) determines the most appropriate tool for analysis [20]. List of algorithms: Principal Component Analysis (PCA), Clustering (Clu), K-Means (K-M), Topic Model (TM), Latent Dirichlet Allocation (LDA), Support Vector Machine (SVM) and Rules Learning (RuL).

4.2. Model Development (Block 2)

This block contains the networks and models used for the development and adjustment of predictive models. The purpose of these tools is to make decisions based on processed data. This section typically includes three families: (1) Regression, which comprises methods for estimating relationships between variables and predicting continuous values [36]; (2) Classification, which includes techniques for assigning discrete categories to data instances; (3) Neural networks, which are systems that model complex nonlinear interactions between inputs and outputs, simulating the behavior of the human brain to recognize complex patterns [37]. List of algorithms: Regression (Reg), Linear Regression (LR), Ridge

Regression (RR), Logistic Regression (LoR), Classification (Cla), Quadratic Discriminant Analysis (QDA), Neural Network (NN), Complex Neural Network (CNN), Biological Network (Bio), Generative Adversarial Network (GAN), Generative Modeling (GM), Transformers (Tra), Generative Pretrained Transformer (GPT), Variational Autoencoders (VAE), Dropout (Dro), Max Pooling (MP).

4.3. Decision-Making (Block 3)

Decision-making largely depends on the nature and quantity of available data. After an autonomous learning phase, AI models are capable of generating predictions (Phase 2). Based on these predictions, decisions can be made depending on data availability (from low to high) and the state of the environment (fixed or dynamic). The environment is considered fixed when it does not change or changes very slowly (e.g., an existing built environment) and dynamic when it changes with every action, interaction, or unit of time (e.g., during the design of an architectural project). By combining data availability with the state of the environment, four distinct scenarios are defined to optimize decision-making, which in turn influences the choice of intelligent tools. List of algorithms: Model Predictive Control (MPC), Random Forest (RD), Multi-Armed Bandit (MAB), Lasso Regression (Las), Markov Decision Process (MDP), Dynamic Bayesian Networks (DBN), Actor-Critique (AC), Q-Learning (QL), Epsilon-Greedy (EG).

4.4. Causal Inference (Block 4)

Causal inference focuses on identifying and attributing causes to observed phenomena. Its goal is to confirm the existence of cause-and-effect relationships between various events, thereby allowing the evaluation of the effectiveness of the decisions made. List of algorithms: Causal Forest (CF), Instrumental Variable Regression (IVR) and Causal Decision Tree (CDT).

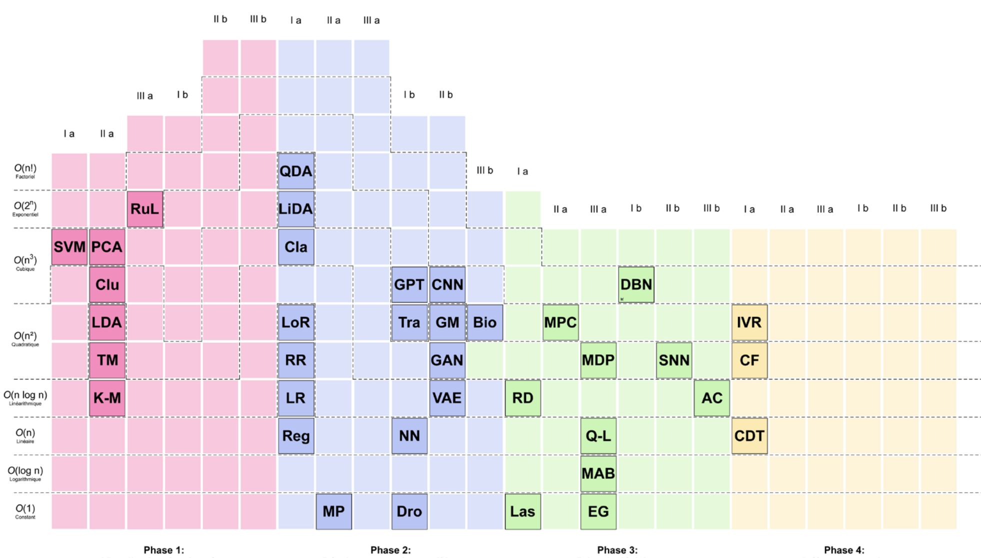

Fig. 3. The periodic and evolving table of intelligent design assistance tools.

5. Discussion and Conclusions

Similar to Mendeleev's periodic table, which classifies chemical elements and predicts the combinations forming various molecules, or visual scripting software that allows combinations of intelligent components to construct scripts capable of solving specific problems (like Grasshopper or Dynamo), the periodic and evolving table of design assistance tools presented here aims to help non-expert designers locate and consider the interoperability of the various available tools by visually comparing the algorithms they employ.

This combinatorial approach allows for at least the following: (1) partially demystifying the AI processes that can be utilized in architectural design ; (2) understanding the individual functions of each algorithm, as well as their inputs and outputs ; (3) identifying possible synergies between the classified algorithms to solve complex problems ; (4) imagining processes that combine multiple tools before their implementation and identifying the solutions best suited to the available computing power and data ; (5) anticipating interoperability between all phases of a design process using AI.

The positioning of intelligent tools in the proposed table demonstrates the relevance of its structure. The aim was not to definitively place all existing algorithms in the table (only algorithms with direct relevance to architectural design were prioritized), but to provide a reference model situating each algorithm relative to others within a coherent structure.

Finally, the reflection on a tabular model for classifying intelligent tools contributes to a deeper understanding of the cognitive spectrum within hybrid networks in architectural design—encompassing humans, machines, and cyborgs. This research introduces a structure to apprehend the interrelations between AI and architectural tools. This instrument serves as both an educational tool that deciphers and makes concepts of machine and deep learning accessible, and facilitates their integration into both professional practice and architectural education. This research will continue by experimenting with the periodic table of machine learning, combining different algorithms to create AI systems. It will also be compared to the periodic table of human intelligence [2], enabling the establishment of synergistic connections between the two types of cognition in architectural design.

References

[1] D. Claeys, *Entre cogitation et computation : comment déjouer l'obsolescence programmée de l'architecte ?* in *Anticrise architecturale : analyse d'une discipline immergée dans un monde numérique*, edited by D. Claeys (Presses universitaires de Louvain, Louvain-la-Neuve, 2021), pp. 157–176

[2] L. Roobaert, D. Claeys, and S. Cleven, *Espaces d'hybridation en conception architecturale : modalités collaboratives entre cognitions naturelles et artificielles*, SHS Web Conf. **203**, 04002 (2024)

[3] J. McCarthy, M. L. Minsky, N. Rochester, and C. E. Shannon, *A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence, August 31, 1955*, Artificial Intelligence 12 (1955)

[4] J.-P. Haton and M.-C. Haton, *L'Intelligence artificielle* (Presses universitaires de France, Paris, 1989)

[5] W. McCulloch and W. Pitts, *A logical calculus of the ideas immanent in nervous activity*, Bulletin of Mathematical Biology **65**, 99 (1943)

[6] F. Rosenblatt, *The perceptron: A probabilistic model for information storage and organization in the brain.*, Psychological Review **65**, 386 (1958)

[7] A. Turing, *Computing Machinery and Intelligence*, Mind **LIX**, 433 (1950)

[8] J. Von Neumann, *First Draft of a Report on the EDVAC*, Moore School of Electrical Engineering, Univeristy of Pennsylvania (1945)
[9] J. Weizenbaum, *ELIZA—a computer program for the study of natural language communication between man and machine*, Communications of the ACM **9**, 36 (1966)
[10] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed (Pearson, Upper Saddle River, 2016)
[11] C. S. R. Silva and J. M. Fonseca, *Artificial Intelligence and Algorithms in Intelligent Systems: Advanced Analytics: Moving Forward Artificial Intelligence (AI), Algorithm Intelligent Systems (AIS) and General Impressions from the Field* in *Artificial Intelligence and Algorithms in Intelligent Systems*, edited by R. Silhavy (Springer International Publishing, Cham, 2019), pp. 308–317
[12] W. Bauer and C. Vocke, *Work in the Age of Artificial Intelligence – Challenges and Potentials for the Design of New Forms of Human-Machine Interaction* in *Advances in Human Factors, Business Management and Leadership*, edited by J. I. Kantola and S. Nazir (Springer International Publishing, Cham, 2020), pp. 493–501
[13] F. Chollet, *Deep Learning with Python*, Second edition (Manning Publications, Shelter Island, 2021)
[14] M. Dmitri, *La dépendance entre les propriétés des masses atomiques des élémentsL*, (1869)
[15] J. Gustafson, D. Rover, S. Elbert, and M. Carter, *The design of a scalable, fixed-time computer benchmark*, Journal of Parallel and Distributed Computing **12**, 388 (1991)
[16] P. Bachmann, *Die Analytische Zahlentheorie*, Zahlentheorie, Band 2 (B. G. Teubner, Leipzig, 1894)
[17] Z. Bimurat, Y. Kim, R. Ismailova, and B. Sagindykov, *Methods of navigating algorithmic complexity: Big-Oh and small-oh notations*, Scientific Journal of Astana IT University 160 (2023)
[18] F. A. Mala and R. Ali, *The Big-O of Mathematics and Computer Science*, Journal of Applied Mathematics and Computation **6**, 1 (2022)
[19] S. Bae, *Big-O Notation* in *JavaScript Data Structures and Algorithms* (Apress, Berkeley, CA, 2019), pp. 1–11
[20] Y. LeCun, Y. Bengio, and G. Hinton, *Deep learning*, Nature **521**, 436 (2015)
[21] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed (Springer, New York, NY, 2009)
[22] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, Second edition (The MIT Press, Cambridge, Massachusetts, 2018)
[23] M. R. B. Emdad, *Deep Learning is the Core Method of Machine Learning*, International Journal of Engineering Research and Technology **6**, (2017)
[24] Z. Han, M. Hong, and D. Wang, *Signal Processing and Networking for Big Data Applications*, 1st ed. (Cambridge University Press, 2017)
[25] V. François-Lavet, P. Henderson, R. Islam, M. G. Bellemare, and J. Pineau, *An Introduction to Deep Reinforcement Learning*, Foundations and Trends® in Machine Learning **11**, 219 (2018)
[26] D. P. Kingma and M. Welling, *Auto-Encoding Variational Bayes*, (2013)
[27] W. Reitman, *Heuristic decision procedures, open constraints, and the structure of ill-defined problems* in *Human Judgments and Optimality* (John Wiley & Sons, New-York, 1964), pp. 282–315
[28] D. Claeys, Architecture et complexité : un modèle systémique du processus de (co)conception qui *vise* l'architecture (Presses universitaires de Louvain, Louvain-la-Neuve, 2013)

[29] T. Oladipupo, *Types of Machine Learning Algorithms* in *New Advances in Machine Learning*, edited by Y. Zhang (InTech, 2010)
[30] C. M. Bishop, *Pattern Recognition and Machine Learning* (Springer, New York, 2006)
[31] K. P. Murphy, *Probabilistic Machine Learning: An Introduction* (The MIT Press, Cambridge, Massachusetts, 2022)
[32] F. Osisanwo, O. Awodele, J. Hinmikaiye, O. Olakanmi, and A. J. Akinjobi, *Supervised Machine Learning Algorithms: Classification and Comparison*, International Journal of Computer Trends and Technology **48**, 128 (2017)
[33] G. E. Hinton and R. R. Salakhutdinov, *Reducing the Dimensionality of Data with Neural Networks*, Science **313**, 504 (2006)
[34] L. van der Maaten and G. Hinton, *Visualizing Data using t-SNE*, Journal of Machine Learning Research **9**, 2579 (2008)
[35] C. Cortes and V. Vapnik, *Support-vector networks*, Machine Learning **20**, 273 (1995)
[36] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to Linear Regression Analysis* (John Wiley & Sons, 2012)
[37] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (The MIT Press, Cambridge, Massachusetts, 2016)