MACHINE LEARNING AND THEORY-LADENNESS: A PHENOMENOLOGICAL ACCOUNT

Alberto Termine, IDSIA USI-SUPSI, Lugano

Emanuele Ratti[1], University of Bristol

Alessandro Facchini, IDSIA USI-SUPSI, Lugano

Abstract. In recent years, the dissemination of machine learning (ML) methodologies in scientific research has prompted discussions on theory-ladenness. More specifically, the issue of theory-ladenness has re-emerged as questions about whether and how ML models (MLMs) and ML modelling strategies are impacted by the domain theory of the scientific field in which ML is used and implemented (e.g., physics, chemistry, biology, etc). On the one hand, some have argued that there is no difference between 'traditional' (pre-ML) and ML-assisted science. In both cases, theory plays an essential and unavoidable role in the analysis of phenomena and the construction and use of models. Others have argued instead that ML methodologies and models are theory-independent and, in some cases, even theory-free. In this article, we argue that both positions are overly simplistic and do not advance our understanding of the interplay between ML methods and domain theories. Specifically, we provide an analysis of theory-ladenness in ML-assisted science. We do so by constructing an account of MLMs based on a comparison with phenomenological models (PMs), and we show that seeing MLMs through the lens of the debate on PMs, can shed light on the subtle roles that domain-theory plays in the various steps of the construction and use of MLMs. Our analysis reveals that, while the construction of MLMs can be relatively independent of domain-theory, the practical implementation and interpretation of these models within a given specific domain still relies on fundamental theoretical assumptions and background knowledge. Based on our analysis, we introduce new categories of theory-ladenness - 'theory indifference,' 'theory-aid,' and 'theory-infection' - to capture the varying degrees of influence of domain-theory on MLMs. This analysis of theory-ladenness has far-reaching consequences for understanding the role of ML practices in contemporary science, and the relation between ML specialists and scientists of a given field.

1. Introduction

Philosophers of science have discussed the role of human expertise in building and implementing machine learning models (MLM) in science[2]. There is a consensus that some kind of human expertise is always required in building and using machine learning (ML) methods (see e.g. Hansen and Quinon 2023). However, underneath this agreement, it remains unclear what kind of human expertise is needed in the context of ML, whether scientific and/or purely engineering-based. For this reason, there is much confusion on the extent to which ML methods are theory-laden, where by 'theory' here we refer to the domain knowledge of the given scientific field in which a ML method is implemented (e.g. physics, biology, etc). In the philosophical debate, some emphasize the a-theoretical nature of ML methods (Napoletani et al 2022; Pietsch 2015), while others have shown that there are significant aspects of ML requiring domain knowledge expertise (Ratti 2020; Hansen and Quinon 2023). This issue is

---

[1] Corresponding author, mnl.ratti@gmail.com

[2] Our focus is only in the use of machine learning in the sciences, in particular in the natural and life sciences.

among the most relevant ones in the epistemology of ML-based science. Identifying the extent to which ML methods are theory-laden and need domain knowledge expertise to be built and used, has repercussions on the kind of expertise needed for ML-based science (whether purely engineering-based, or else), the way curricula for training future ML-expert in the sciences should be shaped, and whether ML, as a practice, is a tool for the unification of the sciences beyond disciplinary and theoretical differences.

The aim of this article is to shed light on the nature of theory-ladenness in ML methods when implemented in the sciences. After a brief introduction on the *status quaestionis* for theory-ladenness of ML (Section 1.1), we show that the investigation of this topic requires a comprehensive analysis of the interactions between MLMs and scientific theories based on a more precise account of what the former are. Specifically, we construct an account of MLMs by comparing them to phenomenological models (PMs), and we show that this comparison can illuminate the role that theory plays in ML-based science (Section 2). As the debate on PMs in philosophy of science has emphasized a distinction between *model construction* and *model applicability* and its consequences for claims of degrees of theory-independence, here we use this insight to show that MLMs can have a high-degree of theory-independence in the way they are constructed and trained, but they are less so in the way they are used in scientific practice (Section 3). This analysis of the construction and use of MLMs will reveal new categories of theory-ladenness that are not covered by the classical literature on theory-ladenness, nor by that on 'models as mediators' (Morgan and Morrison 1999).

1.1 Machine learning, blind methods, and theory-ladenness

Theory-ladenness - which is sometimes seen as a challenge and other times as productive (Aktunk 2021) - has been referred originally to 'observation' (Kuhn 1970; Hanson 1958), and later to experimentation (Schindler 2013). It is well-known that there are different ways in which scientific practices such as experimentation are impacted by theories. For instance, exploratory experiments have been seen as only loosely guided by theories (Steinle 1997). This could be expressed by saying that experiments are theory-informed (Waters 2007), or that they are loosely guided by the theoretical background of scientists involved (Heidelberg 2003). This is unlike stronger theory-directed experiments (Waters 2007), where "a theory generates expectations about what will be observed" (p. 277). This is not the space to recall the nuances of theory-ladenness formulated in the philosophy of science literature. However, it is important

to point out that it is now a given that scientific activities like experimentation cannot be theory-free (Radder 2003).

But ML is a modelling practice, and not a case of material experimentation[3]. The strict connection between models and theories has been a topic of interest at least since the semantic view of theory. Even in the debate on models as mediators (Morgan and Morrison 1999), where models maintain a partial independent status from theory and data, models are still theory-laden, since models "typically involve some of both [i.e. theory and data]" (p. 11). Therefore, even here the question is not whether models are theory-laden, but rather how, and which role the theory plays[4]. One might be tempted to simply be satisfied with the claim that MLMs are inevitably theory-laden, but the peculiarities of this context deserve a more careful analysis.

Within the philosophical debate, discussions of theory-ladenness in ML methods have emerged first in the context of the much older rhetoric on the promises of Big Data and its role in the sciences. Beyond the early seriously flawed hypes on the 'data deluge' which would dispense science from theory in the first place (as popularized by Anderson in 2008[5]), the discourse on hypothesis/theory-free nature of the new scientific methodologies brought forth by Big Data gradually shifted to the relation between theory and ML. For instance, in talking about data-intensive science, Pietsch (2015) refers explicitly to ML methods. He notices that these methods are *externally* theory-laden, in the sense that elements of theory must be presupposed in the framing of a scientific problem, in particular considered parameters must "be those (...) relevant for a phenomenon (...) [and in general] relevant with respect to a given research question" (p. 913). However, ML methods and MLMs can be considered theory-independent *internally*, in the sense that "no hypotheses are made about causal connections (...) no assumptions are presupposed about the functional dependencies between different quantities" (p. 913). Napoletani et al. (2022) argue in favour of a stronger characterization of theory-freedom, by arguing explicitly that no preconceived understanding of a phenomenon to investigate plays any role in how ML methods are used to build MLMs. In fact, data are processed independently of their specific nature and of any background knowledge (including theoretical knowledge) of the possible relationships between variables of the phenomenon under scrutiny. In other words, the "process is subject to normalization constraints imposed by

---

[3] Although, to paraphrase Parker (2009), one can also say that models are types of representation, and experiments are investigative activities involving intervention - as such, there can be experiments in modelling (Peschard and Van Fraassen 2018)

[4] The only exception, as we will see, is in the discussion on phenomenological models, where under a certain interpretation of PMs as derived "purely from measurement" (Cartwright and Suarez 2008, p 70), some have proposed that these can be not only independent, but even theory-free.

[5] The End of Theory: The Data Deluge Makes the Scientific Method Obsolete | WIRED

the data, rather than by the (unknown) structure of the phenomenon" (2022, p 46). This is as close to theory-freedom that we found: the domain knowledge characterizing our preconceptions of a phenomenon under investigation will play no role in the construction of a MLM. As a contrast, there have been works making the opposite claim, namely that ML methods or MLMs are *inevitably* theory-laden. For instance, Hansen and Quinon (2023) argue that expert knowledge is needed in all phases and stages of MLM construction, as well as in their application. However, they do not really distinguish between the scientific domain knowledge of the context in which MLMs are implemented and engineering expertise. While these are important insights, these authors seem to talk past each other. On the one hand, claims of complete theory-freedom are likely to be exaggerations. On the other hand, just noticing that everything is theory-laden in a general sense does not do justice to the fact that theory-ladenness of *uses* of a model is different than theory-ladenness of *practices of building* a model, and that even in each of these cases, important nuances are relevant to understand the role of ML and theory in contemporary science.

2. Machine learning models: a phenomenological account

In order to grasp the relation between domain knowledge/theories and MLMs, it is important to characterize more precisely what MLMs are in the first place. In this section, we first narrow down the meaning we attribute to 'model' in ML, and then provide an account of MLM based on the characteristics typically attributed to PMs in philosophy of science.

2.1 Machine Learning Models and Machine Learning Systems

In literature, the term 'machine learning model' is used quite ambiguously to denote a variety of different things. To solve this ambiguity, it is fundamental to make an explicit distinction between this term and the term *ML system* (see, e.g., Facchini and Termine 2021). Here, we propose to use the latter to refer to any computational artifact capable of learning information from data and adapt its behaviour accordingly. ML systems share all a common general architecture, which encompasses three main components:

1. the training sample;
2. the training engine; and
3. the learned MLM (or, more simply, MLM).

The interaction between these components makes the ML system autonomously capable of extracting information from data and adapts its behaviour accordingly.

The *training sample* is the repository of observational/synthetic data that the system uses as the source of information to learn and adapt its behaviour. The individual *data-points* of this sample denote specific instances of the system's target-phenomenon and are composed of *features*, i.e., mathematical representations encoding specific measurable magnitudes of the target-phenomenon.

Selecting and constructing the proper features have a crucial impact on the predictive performances of a ML system, notably as predictions are generated by analysing the correlations between the single features in the training sample and occurrence, or the probability of occurrence, of the specific target-phenomenon. This process is usually referred to as *features engineering* and requires the analysis of different possibilities and a suitable combination of statistical techniques and background knowledge. The process starts with sampling relevant properties of a target-phenomenon from pre-processed data, which are thus mapped into measurable variables called *raw-features*. The latter are further processed through the iterative application of several data-transformations, which eventually lead to *derived features* better suitable for prediction. More specifically, the process of selecting and constructing derived features can be 'hand-made' or performed automatically through appropriate *feature learning algorithms*, this being mostly the case for big-size contemporary deep learning systems (including, for instance, *Generalised Pre-trained Transformers*) (Baldi 2021). In Section 3.1, we will see how this progressive shifting from hand-made to automatic features engineering has relevant implications for the theory ladenness of MLM-building and deserves a careful analysis. For the moment, however, let us finish our brief discussion on ML technicalities by considering the other two fundamental components of the architecture of a ML system, i.e., the *training engine* and the *learned MLM*.

The *training engine* is the computational machine that allows a ML system to 'learn' from the training data. The learning process consists of an iterative adjustment of the system's input-output behaviour with the goal of optimizing its predictive performance at test time. The training engine performs this iterative adjustment by updating a vector of parameters $\vec{W}$ that governs the overall system's behavior. This updating is driven by an *optimisation function f* related to the parameters vector $\vec{W}$ and that accounts for the predictive performance of the system. The goal of the training engine is thus to approximate the global maximum/minimum of f by iteratively tuning the parameters in $\vec{W}$ via a suitable optimisation procedure. Both the specific nature of f and of the optimisation procedures used to maximize/minimize it varies

depending on the *learning paradigm* adopted[6]. For example, in supervised learning, f is usually a *loss function* that measures the predictive error of the system over the training sample, and that has to be minimized. A common example of loss function is the *mean squared error* of the system, which measures the distance between the correct prediction y associated with an instance x (according to the information included in the training sample) and the prediction $m(x)$ that the system m assigns to x as the (average) squared difference between y and $m(x)$. There exists a variety of optimization procedures to minimize loss functions: one of the most widely adopted, especially in *deep learning*, is the *stochastic gradient descent*, an heuristic of search based on the computation of the gradient of the loss with respect to the parameters $\vec{W}$. The procedure exploits a basic concept of differential calculus, namely the equivalence between the partial derivative and the slope degree of the tangent line to the loss function at each of its points. As one approaches the point of minimum (see Fig. 1), the derivative will tend to decrease (i.e. the tangent line gradually decreases its slope) until it reaches a point of minimum.

The third component of an ML system we consider in our analysis is the *learned model*. This is a mathematical representation of the statistical patterns that the system learns from data and uses to formulate predictions on the target-phenomenon. Technically, this representation is a *fitting curve* (or alternatively a *probability distribution*) defined in a n-dimensional space, called *features space*, whose axes codify the values of the features.
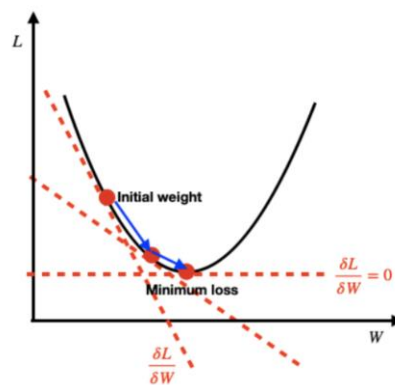

Figure 1: A simplified representation of the gradient descent procedure obtained by assuming the loss function to be defined over a single-element parameter vector. The curve represents the value of the loss for different values of the weights. Dotted lines represent the tangents of the loss for different values of the gradient of the loss with respect to. the weight. Starting from the initial weight, the gradient descent algorithm adjusts the values of the weight in order to iteratively reduce the gradient slope until reaching the minimum of the loss.

---

[6] The ML literature typically distinguishes among three main 'learning (or *training*) paradigms': *supervised learning*, *unsupervised learning*, and *reinforcement learning*. A fourth-paradigm, *self-supervised learning*, which can be in some sense located halfway between supervised and unsupervised learning, has been increasingly emerging in the last years with the advent of large language models and generative AI (Rani et al. 2023).

This curve is what scientists commonly denote with the term *MLM* in the context of scientific research. Understood in these terms, a MLM does not substantially differ in nature from the more 'traditional' kind of statistical models commonly used in scientific practice, such as linear or multiple regression (Dobson & Barnett 2018). The only relevant differences are the usual high dimensionality of MLMs (compared to that of traditional statistical models), and their inherent nonlinearity. *Dimensionality* possesses a very specific meaning in this context, i.e., it refers to the number of dimensions of the model's feature space. In 'traditional' statistical models, the number of features considered is typically limited and allows these models to be easily represented graphically as lines or planes in low-dimensional spaces. Instead, for contemporary MLMs, and particularly in deep learning, the number of features considered is typically very large, making it impossible to represent the learned models in a suitable and humanly understandable graphical manner (see Fig. 2). *Linearity*, on the other hand, refers to the possibility of representing a statistical model analytically as a linear equation $y = a_1 x_1 + \cdots + a_n x_n$ where y denotes the model's target prediction, the $x_i's$ represent the features, and the a_i's are the *statistical weights* assigned to each feature. While more 'traditional' statistical models, such as *linear regression*[7], are typically representable in this format, the several nonlinear transformations MLMs encompass make it impossible to express them as weighted linear sums.

High dimensionality and nonlinearity severely limit the transparency and interpretability of MLMs (Selbst and Barocas 2018), notably as they prevent them to be representable in suitable graphical (e.g., curves in a plane) or analytic (e.g. linear equations) formats that make it easy for scientists to get access to and survey the statistical information these models include[8]. Consider a simple example (Fig. 2) for the sake of clarity. Take a traditional linear regression model that analyses the correlation between *age* and *cancer risk*. This model can be easily represented graphically as a curve in a two-dimensional plane (Fig. 2a), or analytically as a linear equation $Y = rX + b$ (where r measures the correlation strength between the two features and b is a *bias* parameter). Both these two formats of representation make it easy for scientists to grasp the statistical information the model embeds.

---

[7] Notice that linear *regression* is also sometimes considered in literature a kind of MLM. In general, a clear distinction between mere 'statistical' models and MLMs is not given. Here, we use the term MLM with a narrow sense, including complex automatically-learned models such as *decision trees*, *random forests, clustering, support vector machines, deep and reinforcement learning models*, whereas we exclude more traditional kinds of statistical models such as linear and multiple regressions or Markov models.

[8] This issue has been widely discuss in the philosophical literature, especially within the debate on the *opacity* of MLMs (Burrel 2016, Duran and Formanek 2018, Paez 2019, Creel 2020, Sullivan 2022, Zednik 2021, Zednick and Boelsen 2022, Boge 2022, Facchini and Termine 2022).

One has just to observe the slope of the regression line in Fig. 2 to realize that the model identifies a positive correlation between the feature *age* and the target-phenomenon *cancer risk* (the greater the age, the greater the cancer risk). Similarly, it is sufficient to observe the *weight* (parameter r) of the variable X (representing *age*) to understand the degree of statistical correlation existing between this feature and the target-phenomenon of interest. Now consider the graphical representation of an MLM provided in Fig. 2b. In this case, it is clearly challenging to capture any statistical pattern between features and target-phenomena by looking at this type of graphical representation, since it is far too complex. Likewise, it is impossible to identify the correlation strength of each feature by observing its weight a_i within a linear sum, since no such linear representations of the model is given. As we shall see in the course of the paper, these issues have profound epistemological implications for the integration of MLMs in scientific practices and mark a substantial gap between the latter and other types of statistical models commonly used in scientific research.
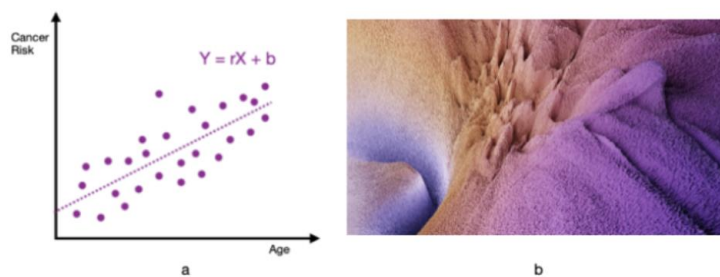

Figure 2: A graphical representation of a *linear regression model* (a) compared with a low-dimensionalised graphical representation of a MLM learned by a deep neural network (b). Notice that *a* gives an immediate access to the statistical information that the model captures: it only takes a quick glance to realize that the model represents a positive correlation between the features *age* and *cancer risk*. Instead *b* provides a much more complicated representation of the statistical pattern the deep network has learnt, which makes grasping the statistical correlations between features substantially impossible. Image (b) borrowed from https://losslandscape.com/gallery/.

2.2 Machine learning models and phenomenological models

Remember that one of the goals of this article is to show that understanding MLMs through the lens of the debate on PMs[9] can shed light on the intricate theory-ladenness in ML methods, and specifically of MLMs themselves. Now, what are the similarities between MLMs and PMs? Let us start by clarifying the nature of PMs first.

In philosophy of science, PMs have been seen as either models of phenomena, or models that are not-theory-driven, or simply as models derived from measurements (Cartwright and Suarez 2008). Underneath these disagreements, philosophers of science seem to agree on four basic characteristics of PMs. The first characteristic (C1) is to be found in an article by

---

[9] We are not claiming that MLMs are identical to PMs. As we will show, we just point to some notable similarities between MLMs and PMs, and we argue that these similarities can shed light on the relations between MLMs and theory.

McMullin (1968), where he distinguishes between theoretical laws or explanations, and PMs. A theory, in his account, is not just a description of the evidence, but it goes beyond the evidence by entertaining the existence of "a postulated physical structure that could provide a causal account of the data to be explained" (1968, p. 388). The theoretical model is the (representation of the) postulated structure. A PM is different: it is "an arbitrarily chosen mathematically-expressed correlation of physical parameters from which the empirical laws of some domain can be derived" (p. 391). As such, PMs account for evidence "in convenient [mathematical] form" (p. 391), but they do not postulate any physical structure, like theoretical models. As an example, he considers a data set of extensive cosmic ray showers. In order to bring the data into a single array, one can just hypothesize that they follow a general distribution function of, let's say, nucleon collisions, and then try to fit the data by varying the parameters – the model will account for the data in an arbitrarily-chosen mathematical form (that is, C1). Next, the second characteristic (C2) pertains to *how PMs are built*. Cartwright et al (1995) claim that paradigmatic examples of phenomenological model-building are characterized by "independence from theory, in methods and aims" (p. 148). PM building, in their opinion, is mostly based on phenomenological considerations[10], and on *ad hoc* varying of the mathematical conditions to fit the data adequately, where these 'moves' are not licensed by theory, nor follow from theory de-idealisation. The third characteristic identifies the *origin of PMs* (C3), and was explicitly pointed out by Wilholt (2005), when defining PMs as models that are built starting from measurements and observations, with little theoretical input or information. He looks at the provenance of models as one reason to retain McMullin's distinction between theoretical and PMs, where PMs' history starts with a mathematical description of observed properties/behaviour. Finally, a fourth characteristic (C4) is about the *goals of PMs*. Bokulich (2011) has more recently characterized PMs (built through ad hoc fitting to empirical data) as being useful for predictions, but not for explanations (C4).

MLMs, we claim, fulfil all conditions C1-C4, and can be treated as akin to PMs[11], in particular as derived from measurements and *ad hoc* adaptations of some mathematical formalism for a given (non-explanatory) purpose.

Consider C3 first. A key step in building a MLM is to collect training data sets. Because the latter are usually constructed from measurements (except for certain types of synthetic

---

[10] These are 'phenomenological' in the sense of being purely descriptive lacking any theoretical justification (Wilholt 2005)
[11] Again, we are not claiming that MLMs are identical – see the footnote at the beginning of this section.

data), then the starting point for constructing MLMs - their *provenance* - is the same as the one of PMs: measurements. Hence, C3 is indeed central.[12]

Condition C1 is also central in the construction of MLMs. As we have said, MLMs are mathematical representations of the statistical regularities that a ML system learns from its training data, and that it uses to formulate predictions on the target-phenomenon of interest. In other words, the goal of building a MLM is to describe the relation between selected features in a convenient mathematical form (one allowing predictions/classifications).

Moreover, the construction of MLMs is shaped by technical and engineering-related considerations that are justified not theoretically, but rather by the proximate goal of 'selecting' the model that fits the data better. For instance, whether one chooses between discriminative (e.g. nonlinear kernels, decision tree, convolutional neural network, etc) or generative (e.g. variational autoencoders, transformers, etc) algorithms will depend not just by the problem, but also by the type of data you are dealing with. We mean this not in terms of the data domain (e.g. biomedical, financial, etc), but rather in the most technical (and theory-neutral) sense of *modality* (image, text, etc). Each ML algorithm will come 'pre-packaged' with a number of assumptions about the nature of the dataset it can be applied to, which are independent from the theory of the domain in which ML is applied (e.g. the kind of function family that will characterize the model, the parameters, etc). Put it differently, MLMs describe a function mapping input labels to output labels. But the mapping, *per se*, receives little input from theoretical considerations, and it is related to the mathematical nature of the used algorithmic tools (i.e., they are 'ad hoc'). For instance, in early cases of cancer genomics using support vector machines (SVM), classifiers were often built to distinguish between cancer-causing vs cancer-neutral somatic mutations. Those classifiers (e.g. Capriotti and Altman 2011) had continuous outputs from 0 to 1, where 0 was 'cancer-neutral' and 1 was 'cancer-causing'. Given the continuous values, thresholds for classification had to be chosen. But the choice of thresholds (e.g. 0.5) was usually motivated on the basis of technical considerations, and from the point of view of 'theoretical' justification can be considered *ad hoc*. This means that MLMs also possess C2.

---

[12] It is true that these measurements undergo various layers of pre- and -intra processing (Leonelli 2016; Tal 2020), but this does not change the fact that they are intended to be measurements of properties of a given target system (even when model-based).

Finally, central to MLMs are *predictions*. These are essential to measure the performance of models on the test set, and they are taken to be one of the things that ML can do very well. The emphasis on predictions aligns well with C4.

3. Machime learning models and the role of theory

Seeing MLM through the lens of PMs is not just an empty exercise in characterizing MLMs more precisely. The PM-lens, we argue, is also useful for understanding the theory-ladenness of MLM. Let us see how.

In (1999), Morrison distinguishes between aspects of *PM construction* from aspects of *PM use*. This distinction is used to disentangle the complex and intricate relation between theory and PMs. Unlike in C1-C4 where there is agreement on the main points, this relation has been a topic of a heated disagreement. For instance, McMullin (1968) has an extreme position, by conceiving PMs as derived only from measurement and being theory-free, while drawing a sharp separation from theoretical models, which are explanatory and theory-laden[13]. Cartwright et al. (1995) sees PM-building as *not theory-driven*, where this would grant a high-level of independence from theory. Wiholt (2005), as we saw, claims that PMs are built with little theoretical input, and this seems to weaken a claim of theory-independence, even though the point is not clear. But the relation between PMs and theory, Morrison says, is much more complicated than 'clear-cut' separations. PMs should not be seen as *completely* independent from theories: even though they provide a model of a phenomenon, and they seem to be based on fitting data to various mathematical formulations, they "can also be reliant on high level theory" (1999, p 46) in the way they are *applied*. In discussing the model of the boundary layer describing the motion of a fluid, she notices that two different theories are required to solve the hydrodynamic nonlinear equations.

In this section – and despite the differences between the context of the original debate on PMs and the present context - we analyse the relation between domain-theory and MLM by using this suggestion of separating model construction from its use. Distinguishing between *model construction* and *model use* in the context of MLMs will allow us to identify new categories of 'theory-ladenness', which we refer to as 'theory indifference', 'theory-aid', and 'theory-infection':

---

[13] Though this is unclear as, in his opinion, theory is derived from models, and not the other way around. So it might be more accurate to say from his perspective that theory is model-laden.

- *Theory indifference*: a specific activity *x* in the process of *building* a model *y* within a domain *D* is *theory-indifferent* when the execution of tasks prescribed by *x* requires no reference to the theoretical background of *D*.

- *Theory-aid*: a specific activity in the process of *using* a model *y* within a domain *D* is *theory-aided* by theory *T* (where *T* is the theory of the domain *D* in which *y* will be implemented), when *T* is used to actively helping scientist-users to interpret *y*, and direct its uses as a result.

- *Theory-infection*: a model *y* is *theory-infected* with theory *T* (where *T* is the theory of the domain *D* in which *y* will be implemented), when *T* is slipped into one of *y*'s components in ways unrelated to *y*'s construction. This is an attribute *of a model* rather than of the process of building or using a model. One might think that, if the process of building a model is theory-indifferent, then the model is theory-free. However, this is not the case; models can be 'infected' with theoretical repertoires in various ways, even though their contributions to the model are not clear or tangible.

A few remarks are in order about these definitions. First, they have been inferred from the analysis of the practice of building and using MLM, which we illustrate in Sections 3.1 and 3.2, and they are not *a priori* categories fitted to scientific practice. Second, theory-indifference is indeed a rejection of most senses of theory-ladenness as discussed in the literature on scientific experimentation. For instance, if an activity is theory-indifferent, modelers need not make any assumption that is informed by the theory (Waters 2007), nor they need to bring any theoretical repertoire that will be used explicitly to execute or plan that activity (Franklin 2005). Moreover, modelers need not use theory as a starting point or as a foil for that activity (Elliott 2007). However, it should be noted that theory-indifference does not imply that the activity is necessarily theory-independent - theory can still inform, but *it needs not to*: it is not explicitly and fundamentally required. In other words, theory-indifference does not imply 'theory-freedom'. Finally, theory-aid might remind of some uses of theory-ladenness in experimentation (especially weak theory-directedness). However, notice that theory-aid, as we define it, is limited to models and not to experiments, because models can be 'used', while 'experiments' are usually executed and/or built, so that a specific category for theory-ladenness in the context of 'using' may be required[14].

In what follows, we illustrate the extent to which activities in *building* (Section 3.1) and *using* (Section 3.2) MLMs are theory-laden, and we will show that sometimes theory-ladenness

---

[14] Even though one may say that, while we do not 'use' experiments, we do 'use' experimental systems (Rheinberger 1997).

fits the typical categories laid out in the literature of scientific experimentation, but in other cases the new categories defined above are necessary.

3.1 The role of theory in the construction of machine learning models

To understand the level of theory-ladenness in the construction process of a MLM, and clarify the various 'new' senses of theory-ladenness we previously introduced, it is useful to examine such process in comparison with the construction process of other kinds of models commonly used in scientific practice, such as *mechanistic models* and/or *computer simulations*. Consider for example the well-known SIR model used in epidemiology to study the dynamic evolution of an infectious disease in a population (Milgroom 2023). The model relies on three *state variables*, namely S (i.e. individuals in the population susceptible to contracting the infectious agent, such as a virus), I (i.e. individuals in the population who have contracted the infectious agent and can transmit it), and R (i.e. individuals in the population who have contracted the infectious agent and can no longer either contract or transmit it). In addition, the model introduces two parameters of precise biological significance, which are the *average infectious rate β* (denoting how many susceptible individuals in the population get infected daily on average), and the *average removal rate γ* (denoting how many infected individuals in the population recover or die daily on average). These variables and parameters are combined to obtain a compact mathematical description of the system's dynamic in terms of a system of three differential equations:

$$\frac{dS}{dt} = -\beta \frac{S}{N} I \qquad (1)$$

$$\frac{dI}{dt} = \beta \frac{S}{N} I - \gamma I \quad (2)$$

$$\frac{dR}{dt} = \gamma I \qquad\qquad (3)$$

The specification of such an equation system is a *theory-directed* process where experimental data play a marginal role. In particular, the choice of variables and parameters to be considered and the mathematical form to be given to the equations are essentially the result of theoretical considerations based on background knowledge of the target phenomenon at stake, i.e. the spread of epidemics. For instance, modelers do not introduce an equation describing the *infected*-to-*susceptible* transition because they assume that once an individual has contracted the infectious agent they can no longer be susceptible to it because they develop immunity. This assumption is not induced from data but *it is the result of theoretical considerations* that

follow from the theory and background knowledge of the specific domain of interest, namely *immunology*. The only relevant task where data play a role is the estimation of the parameters β and γ, which is based on experimental observations and measurements. However, note that this parameter estimation is usually obtained by performing experimental tasks designed on the basis of hypotheses that are drawn from domain theory, hence it remains a theory-directed task in essence.

Different considerations emerge instead if we examine the construction process of MLMs. The latter, we argue, is not sharply theory-laden. In addition to some classical theory-informed aspects, we claim that constitutive aspects of this process should be regarded as *theory-indifferent*. This is not to say that domain-theoretical considerations are always, or mostly, absent; but that they are not necessary to the practice of modelling, in contrast to what we have just seen with more 'traditional' kinds of scientific models (e.g. the SIR model), which necessarily require domain-theoretical considerations for their specification.

3.1.1 Theory indifference in the selection of parameters and hyperparameters

A MLM is specified by two sets of mathematical entities called *hyper-parameters* and *weight-parameters* (or simply *parameters*). Hyper-parameters are the parameters that determine the skeleton of the model, thereby constraining its possible final structure within certain given borders. The term is taken from Bayesian statistics, where a hyper-parameter is a parameter of the *prior distribution* fixing the set of possible *posterior distributions* that a model can fit (Bovens and Hartmann 2004). In the context of ML, the nature of hyper-parameters vary depending on the specific kind of architecture and framework considered. In the case of *deep neural networks* (Baldi 2021), for instance, hyper-parameters describe the topology of the network (fully connected, convolutional, recurrent, etc.), the kind of activation function used (linear, sigmoid, tan-h, etc.), the kinds of optimization function and the related algorithm used for training, and so on. Beyond their specific nature, hyper-parameters play the same very specific role in all ML contexts, i.e., they fix the set of all possible models (i.e., predictive function/distribution) that the ML system can learn from the training data. Given a class of possible MLMs, determined by the hyper-parameters, the *actual model* is specifically determined by the *weight-parameters*.

The construction that leads to a MLM requires a sharp specification of both the weight-parameters and the hyper-parameters. For what concerns the former, as we have seen in Section 2, their specification is a fully-automated process that ultimately consists of solving an *optimisation task*, i.e., finding the minimum/maximum of a function accounting for some

statistical magnitude (e.g., prediction error, variance, expected cumulative reward etc.) relative to the interaction between the model and the training sample – this is, in essence, the characteristic C1 that MLMs share with PMs. The performance of this optimization task requires no reference to the theoretical background of the specific domain to which the model is applied, and it can be therefore qualified as substantially *theory-indifferent* in the sense of the term we have outlined above. On one hand, this is proved by the fact that the same optimisation functions and procedures can be exported and applied in a variety of different domains without requiring any theoretical adaptation and without implications for the model's predictive performances. For example, the loss function *means absolute error* can be identically applied in all the application domains that require learning a ML regression model, independently of whether this model describes the relation between *age* and *cancer risks* or the relation between the *financial hazard* and the *long-term income* of an economic agent. Theory-indifference is also evident if we consider the heuristic strategies used to implement the optimisation procedures that underlines the learning of weight-parameters. In 'traditional' (i.e., non ML-based) science, the heuristics that guide the scientific model-building process make a fundamental use of hypotheses that are formulated with the support of the existing corpus of domain-specific background knowledge. Consider the paradigmatic case of *decomposition* and *localisation*, the two main heuristic strategies that guide the construction process of mechanistic models according to Bechtel and Richardson (2010)[15]. Both rely on a fundamental contribution of domain-theory[16] for the formulation of hypotheses regarding the specific component-parts of a mechanism and the functions they perform. On the contrary, the heuristic strategies used in weight-parameters learning make no use of domain-theory: they just exploit fundamental mathematical properties of the optimisations task they are supposed to solve – and this is because of the characteristic C2 that MLMs share with PMs. Consider in this regard the *stochastic gradient descent* described in Section 2. This heuristic strategy exploits a fundamental mathematical property of differentiable functions, which guarantees that their global minimum can be effectively approximated by following the value path of its gradient: no domain-theory is required for the application of this heuristic strategy but all one has to know is the value path of the loss function.

Similar considerations hold for the hyper-parameters' specification. The latter, differently from the learning of weight-parameters, is not usually a fully automated task but

---

[15] See also Craver and Darden (2013) on different strategies for building mechanistic models.

[16] The contribution can be specified as a weakly directed theoretical contribution (Franklin 2005; Waters 2007)

requires a suitable combination of automation and hand-made work. In general, a hand-made pre-selection of the hyper-parameters of the model is performed before the training phase, while automatic optimisation procedures (analogous in nature to those used for the learning of parameters) are typically used at test time to fine-tune the hyper-parameter values and reach the best predictive performance (and avoid notorious problems, such as *overfitting*). Domain-theoretical considerations can (and sometimes do) come into play in the hand-made preselection of the hyper-parameters, however they are not strictly necessary to this task, as instead they are in the specification of the differential equations that govern the SIR model previously discussed. Necessary to the hyper-parameters' specification are instead considerations of mathematical and engineering nature, e.g., related to the specific predictive task at stake (e.g., regression, classification, etc) and the format of the data to be processed (e.g., tabular data, time series, etc). Again, attributing C2 to MLMs is central here, given that the emphasis is on 'ad hoc' moves not licensed by theory. For example, in the analysis of time-series with neural networks, modelers typically select the *recurrent* topology due to its ability to support the processing of sequential data (Goodfellow et al. 2016), independently on whether these data represent fluctuations of energy market or brain signal. Similarly, *convolutional* topology is commonly used in the analysis of images for its ability to process different regions of the image in parallel, independently on whether the images represent *cats* and *dogs* or *nevus* and *melanomas*. In support of the claim that hyper-parameters specification is substantially a theory-indifferent task, we can also mention the increasing diffusion of fully automated procedures for the pre-selection of the hyper-parameters based on the application of *meta-learning algorithms* (Vanschoren 2019). The latter are optimisation procedures substantially analogous to those adopted in the learning of weight-parameters and, as these, they can operate independently from domain-theory and domain-theoretical considerations.

Before going any further, please note that the claim that the specification of parameters and hyperparameters is a theory-indifferent task does not imply that this task is *always* (or even *often*) performed by modelers without doing any, more or less, explicit reference to domain theory. The ML literature is in fact replete with examples of MLMs whose hyper-parameters have been selected (also) based on domain-theoretic considerations, or whose training is executed via optimisation algorithms opportunistically constrained with domain-specific background knowledge (and therefore partially *theory-directed*)[17]. However, what we claim

---

[17] One prominent example is *AlphaFold* (for an updated overview of the model's architecture, see (Yang et al. 2023)), the deep learning model developed by Google DeepMind for solving the protein-folding protein. The topological structure of this model has been blueprinted explicitly considering the fact that the protein-folding process consists of three consequent

here is that domain-theoretical considerations *are not necessary*, as instead they are in more 'traditional' model-building practices. Predictively accurate MLMs can be constructed also by doing little to no reference to domain-theory, and this has significant consequences on the daily practice of science. If you walk into an AI centre nowadays, you will see the same ML engineers working in parallel on building MLMs applied to very different domains about which they know little or nothing; their interactions with domain experts are limited, and often do not concern model-building practices but other related tasks, such as data sampling (a task in which, as we have said, domain theory continues to play a fundamental role) or the use of models in practice (a task in which, as we will see in the next sections, theory is still necessary). In this regard, it should be noted that there are different trends and schools of thought between ML-oriented scholars and ML practitioners regarding whether and how to use domain theory in the specification of parameters and hyperparameters. Some scholars argue, indeed, that the explicit use of domain theory can help make MLMs more robust and/or generalisable in out-of-distribution contexts (Pearl 2019, Schölkhopf et al. 2021, Kaddour et al. 2022), while, on the other hand, the increasing use of meta-learning and the success of *generally purposed* and *domain non-specific* models suggests that theory-indifference could become a gold standard. This notwithstanding, it is not the aim of this paper to analyse the actual trends in science, nor it is to address the normative question on whether domain-theory ought to be used in making more 'epistemically virtuous' MLMs. What matters to us is to emphasize the mutated role of domain theory in MLM-building compared to more traditional model-building practices, and to underline the enormous consequences that this fact may have on scientific practice.

Coming back to our discussion, although the specification of the hyper-parameters and weight-parameters are substantially theory-indifferent tasks, one might claim that domain-theory still makes an appearance in two fundamental steps of MLM constructions: the construction of data sets from which training data are sampled, and the features engineering processes (see, Section 2).

3.1.2 Database curation as a theory-informed process that propagates to machine learning models.

As noticed in Section 2, MLMs are constructed by data sets, which are collections of measurements (that is, C3 applies here). But theory, in the form of domain-knowledge, is already present in the data sets acquired to construct the training sample before any data

---

prediction steps (primary-to-secondary, secondary-to-tertiary, and tertiary-to-quaternary structure of the protein), hence following explicit domain-theoretical considerations.

processing procedure is done by ML specialists. In this regard, the work by Leonelli (2016) has documented the epistemic subtleties behind the construction of databases in biology in depth, in particular for what concerns data curation practices. Since databases have to be used (in the context of her work) by biologists, and biologists need to be in the right position to judge whether a given data-set can be used to achieve a given research goal, then databases have to be constructed to reflect, at least partially, biological knowledge, in the sense that reflecting biological knowledge will increase the usability of those data-sets by biologists. Indeed, "terms used for data classification should be the ones used by biologists to describe their research interests - that is, terms referring to biological phenomena" (Leonelli 2016, p 116). This means that a theoretical, pre-conceived understanding of biological phenomena is already embedded in those data-sets that are drawn from biological databases. This is likely to apply to any database used in any scientific discipline. In this sense, database curation is a theory-informed process: theory provides constraints on how the data must be curated. However, the consequences of this are not as straightforward as it may seem.

First of all, the practices of working on weight-parameters and hyperparameters remain theory-indifferent anyway. In fact, the theory used in databases' curation is not used in any *explicit* way in setting the hyperparameters, nor in learning the weight-parameters. The training engine learns independently of those theoretical considerations implicit in the databases. Nonetheless, the theory is still there. In particular, it is implicitly embedded in the MLM itself. Hence the MLM is theory-laden after all, although it is unclear what kind of theory-ladenness we are dealing with. One can interpret it as yet another case of 'theory-informed' (Waters 2007). However, in theory-informed contexts, such as exploratory experimentation, theory is used to set up experiments *explicitly*. In other words, being theory-informed is an attribute of the practices, not of 'entities' like models. In the case of the trained MLM, theory is first *passively* passed to the data sets used by the training engine, and then eventually ('like an infection') propagates to influence the predictive behaviour of the trained model. This is the reason why we call *theory-infection* the kind of theory-ladenness associated with a MLM: theory is passed to the model, but it does not play any specific role in its construction (the infection is, so to speak, asymptomatic).

3.1.3 Features engineering: from theory-directed to theory-indifferent
The second step where theory seems to play an important role is features engineering, i.e., the construction process of the features composing the training sample. As we anticipated in Section 2, this process is based on the collection of large amounts of observational data, which

are finely pre-processed and sampled to obtain raw features. These data may come either from databases (as discussed in the previous paragraph), or from more direct measurements taken by a given scientific group. These are then subjected to various manipulation processes directly by ML specialists, which ultimately result in derived features that are used as input for training procedures. Theory seems to play a non-negligible role in the process of feature construction. After all, deciding which variables of a target-phenomenon to consider for predictive purposes, and how to combine them in suitable representation formats, is a task that requires an extensive knowledge of the phenomenon under investigation. This is true for more traditional kinds of MLMs, like decision-trees or random forests, which operate with hand-made features.

However, the advent of automatic feature learning algorithms, whose operation is essentially based on the execution of optimisation tasks similar to those used to train MLMs, are gradually reducing the prominent role of domain theory in feature construction. Examples of this path from theory-informed to theory-indifferent features engineering can be found in various domains of scientific investigations, in particular in those witnessing an increasing use of deep learning systems. These systems are in fact capable of generating predictions directly from 'raw-data' (e.g., *images*), and incorporate features engineering as a step of the predictive inferences they perform.

To illustrate and exemplify these considerations on features engineering, let us consider the case of MLMs in neuroimaging-based psychiatric research (Eitel et al. 2023). The detection of psychiatric disorders is notoriously a challenging task, notably because the underlying mechanisms of these pathologies, with a few exceptions such as Alzheimer's disease, remain mostly unknown or only partially understood. This makes traditional 'theory-directed' modelling techniques, such as mechanistic models and simulations, difficult to apply. On the contrary, MLMs have proven to be easier to implement, in particular thanks to the independence of their training from theoretical considerations.

The analysis of literature (see, e.g., Eitel et al. 2023) not only shows that neuroscientific theory has a limited influence on the MLM construction process, but it also displays a trend towards an increasing theory-indifference of model-building practices. This trend is particularly evident in the shift from more classical ML architectures (e.g., decision-trees), which require the use of high-level features, to deep learning systems, which can instead learn the features directly from raw-data (see, Fig 3) in a way akin to C2. Let us clarify this point a bit more in detail.
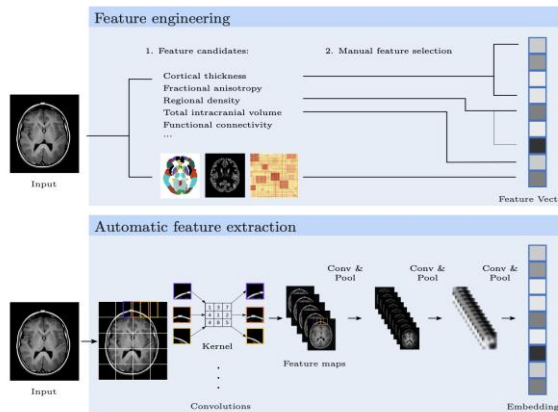
Figure 3 (from Eitel et al. 2023)

Both for 'classical' and deep learning models the model-building process starts with the collection of brain images, which are here represented as grids of pixels encoded in the form of numerical matrices. Each cell of the matrix (i.e., each pixel) represents a single low-level 'raw' feature. These features must be converted into high-level features that allow for predicting the target psychiatric disorder. This is the *features engineering* process. Differences between 'classical' and more contemporary deep learning models emerge exactly at the level of this process. In the case of 'classical' models, scientists select and extract manually from images a number of high-level features, such as *cortical thickness, fractional anisotropy* etc., and thus use 'raw-data' to determine the values of such features. The choice of the features to consider depends on explicit domain-theoretical considerations: for example, modelers focus on *cortical thickness* because they are aware (from domain-theory) of the relevance of this feature for the prediction of specific psychiatric disorders. In this case, the process can be qualified as (weakly) theory-directed. In more contemporary deep learning models things go differently. Consider for example the case of *Convolutional Neural Networks* (CNN), which represent a standard architecture for image-recognition tasks in deep learning. CNNs do not require hand-made high-level features but are able to automatically extract these features from 'raw-data' through the application of a mathematical operation known as *convolution* (Fig. 4).
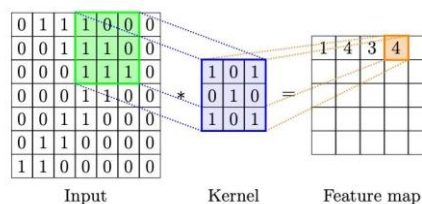

Figure 4: representation of a 2-dimensional convolution operation (image from Eitel et al. 2023)

The latter is a linear transformation based on the application of a kernel of parameters to the input. In practice, the various regions of the input are processed through a filter unit that computes the weighted sum of the input-features (i.e., the pixels of the image encoded in the matrix) in the region and kernel parameters, hence mapping the result into a feature map. In general, data are processed through iterative convolution operations, which eventually produce the high-level features the model uses for predicting the target, technically called 'embeddings'. Differently from hand-made high-level features, embeddings may not possess a clear interpretation and may represent magnitudes of the input that do not possess any clear meaning for the domain-experts. Furthermore, and more importantly, the construction and selection of embeddings do not require any theoretical considerations related to the specific application domain. On the contrary, they rely only on the execution of pure mathematical operations based on numerical parameters, which are learned via standard optimization procedures analogous to those used for MLMs training. With CNNs, feature engineering can be therefore qualified as a theory-indifferent process. This consideration can be generalized to the majority of the deep learning architectures used in the various domains of scientific research (see, Baldi 2021). In general, the analysis of literature reveals a trend towards the increasing use of theory-indifferent automatic features extraction procedures, a trend sustained also by the better predictive performances of these techniques compared to hand-made features, and their cross-domains applicability.

3.2 The role of theory in the implementation of machine learning models in scientific practice

As we saw, MLMs are theory-indifferent in how they are built. This creates a *structural semantic gap* between models and the scientific context in which a MLM is implemented: MLMs are constructed with little input from the context in which are implemented, such that their components do not straightforwardly match concepts of the given scientific context of implementation[18]. This gap promotes misunderstandings and misalignments on how MLMs can contribute to a given scientific context. It is in this context, indeed, that domain theory regains a primary importance for ML-based science. In particular, theory promotes an alignment of the model to its domain of implementation, which is *a necessary condition for using a MLM in a given scientific context* by experts of that field beyond typical classificatory/predictive uses. We call this alignment process 'theory-aid' (as defined at the

---

[18] This is particularly true for models that involve automatic features learning procedures, such as the CNNs mentioned in the previous section.

beginning of Section 3), and we claim that this takes typically the form of a *post hoc assessment*. In order to exemplify how this happens, we will use one case taken from genomics, a subfield of molecular biology. But since we want to avoid any ambiguity in what the contribution of theory amounts to, we first define what we mean by 'domain-theory' in molecular biology.

In molecular biology (which includes subsequent developments such as genomics, systems biology, cell biology, etc), we can think about theories along the lines of Suarez and Cartwright (2008): a theory is something that "offers tools for constructing representations" (p. 65). This fits nicely with the practices of representing and conceptualizing mechanistic models. Theory as a toolbox is what Darden calls the *store of a field* (2006). In the context of mechanistic sciences, "[f]or a given field at a given time, there is typically a store of established and accepted components out of which mechanisms can be constructed (…) [T]he store also contains accepted modules: organized composites of the established entities and activities" (Darden 2006, p. 51). Molecular biology (and subsequent developments) is a mechanistic science. In this field, components will be entities like RNA molecules, genes, enzymes, etc, while more complex modules will include ribosomes, nucleosomes, etc. The store of molecular biology (virtually) contains all those components that biologists use to conceptualize biological phenomena as well as to build mechanistic models. This can even be something formalized at a very precise level, via resources like the Gene Ontology (Gene Ontology Consortium al. 2000).

3.2.1 Theory aids the integration and use of machine learning in a given scientific context

In molecular biology, 'theory' understood as above becomes an important tool for scientists to integrate MLMs in that context. Domain knowledge is used by domain-experts to get a sense of *what* the MLM has learnt, where the 'what' is used to assess the potential relevance to a given scientific context (e.g biology). One might think about it as a 'validation' strategy, but this is not an epistemic process: using theory in this way does not ensure that the model is reliable; rather, it provides information to fill the semantic gap mentioned at the beginning of Section 3.2. We call this process *domain expert post hoc assessment*.

As an example, consider again the use of SVM in a field like cancer genomics mentioned at the end of Section 2.2. This class of algorithm has proven to be a powerful method to build classifiers. What SVMs do is to determine a decision boundary (i.e. the hyperplane) between two classes enabling prediction of labels starting from features vectors. The hyperplane is such that it is as far away as possible from what are called 'support vectors' (i.e.

the closest data points of the classes). SVM are also used as kernel methods, especially in the presence of higher-dimensional datasets with the goal of creating nonlinear models. In the specific case of cancer genomics, SVMs and similar algorithms should be understood as contributing to the grand research program of finding mechanistic evidence that can link somatic mutations to phases of cancer progression. One important goal for computational biology of cancer within this context is to find ways to automate the process of distinguishing between passenger mutations from driver mutations (that is, mutations that confer a growth advantage to cancer cells). Among the many MLMs constructed for this purpose, before the advent of deep learning, SVM has been popular for some time. The idea is that, as a geometric method, SVM discerns combinations of features that characterize mutations of different classes, so that it can distinguish between driver and passenger mutations. The study by Capriotti and Altman (2011), a classic of ML predicting cancer-causing variants, is an illustration of this tendency. They set the task as a binary classification for distinguishing single amino acid polymorphisms (SAP) that are cancer-causing from neutral polymorphisms. Input features include things like amino acid sequence mutation, local sequence environment, and others. The final input vector is complex, consisting of 51 values. The important aspect of this study is that the MLM is evaluated not only based on traditional quantitative metrics such as accuracy, sensitivity, AUC, etc. There is also an evaluation procedure related to the Gene Ontology (GO). In addition to GO terms associated with genes mutated with cancer-causing or neutral SAP, there is something called GO log-odds score (LGO). What this calculates is the correlation between the effect of a given SAP (be it cancer-causing or neutral) and the related GO term. In this way, one can get a score representing the strength of association between a SAP and typical biological processes, understood through the lens of molecular biology's store of the field (i.e. theory as a tool-box) formalized via GO. In other words, what the MLM has learnt is evaluated by means of a comparison with GO terms, and in this way, biologists who want to use this tool will get an understanding of *what,* at a general level, the MLM has learnt in a domain-knowledge friendly language. This is because, again, GO categories reflect the entities and processes that, within the store of the field, constitute the *theory* of molecular biology. Terms like 'Growth', 'Kinase Activity', and 'Transporter Activity' scored high for cancer-causing SAP, and this makes sense because these processes have been experimentally associated with cancer. As a result, in order to ascertain whether what MLM has learnt makes sense to a biologist, what the model has learnt must be connected to the 'theory' of that specific biological context, understood in the way explained above. After this connection is created, it is possible for domain experts to use the MLM for other exploratory goals other than just

prediction and classification (Zednik and Boelsen 2022). This is because after the post hoc assessment, domain experts will know *how* the MLM (and what it has learnt) is relevant to the context of implementation.

4. Conclusion

In this article, we have proposed an in-depth analysis of the relation between MLMs and the domain-theory of the scientific context in which they are implemented (see Table 1 for an overview). Looking at MLMs through the lens of the debate on PMs, we have identified new dimensions of theory-ladenness, partially vindicating both claims of theory-independence and theory-ladenness in the ways MLMs are constructed and used. Specifically, we have claimed that while domain-theory has a limited role in the construction processed of MLMs, which are increasingly characterized by theory-indifferent practices, it still retails a fundamental role in the way MLMs are used in daily scientific practice, for instance to support fundamental tasks of post-hoc assessment.

Despite the work we have done, there are still several open-questions that could be investigated in future works. First, it would be interesting to understand, via qualitative methods (Wagenknecht et al. 2015), which are the dominant trends in ML-based science regarding the relationship between domain theory and MLMs. On the one hand, although the construction processes of MLMs can be mostly regarded as theory-indifferent tasks, a growing number of scholars consider it essential to make greater use of domain theory in the construction and training phase, particularly in order to build domain-specific MLMs that are more *robust*, *reliable* and *explainable*. On the other hand, it is more and more popular to deploy and use 'generalist-purpose' models (a paradigmatic example are generative large language models such as OpenAI's GPTs) and domain-a-specific models that are constructed and trained in a completely theory-independent manner. Understanding what scientists think about these trends is certainly of central relevance for philosophy of science, as it is the normative question of whether a greater use of domain-theory in the MLMs' construction processes is epistemologically desirable, or instead the extensive use of theory-indifferent practices is compatible with the epistemic and methodological values of our science.

Finally, as mentioned in Section 3.2, post hoc assessment is a necessary process for using MLMs in a given scientific context. This notwithstanding, we have been quite vague about the 'uses' beyond the classificatory/predictive ones that can be done with MLMs. In general, one can use the information provided by a post hoc assessment to bridge the semantic gap so as to generate or refine hypotheses in order to, e.g., design experiments. For instance,

the information provided by the GO-score described in Section 3.2 can be exploited to explore the association between a certain biological process and cancer as identified by a MLM. It is often the case that Explainable AI (XAI) methods play this role (Zednik and Boelsen 2022), and we plan to explore the connections between XAI and domain-theory in another article.

ASPECT OF ML	MODELING CONTEXT	RELATION TO THEORY
Data collection and/or database curation	Preceding Model-Construction	Theory-Informed (practice) Theory-Infected (the resulting data-set, and thus the model trained on it)
Features Engineering	Model-Construction	Classically Theory-Informed, but increasingly Theory-Indifferent
Specification/tuning of Hyperparameters	Model-Construction	Theory-Indifferent
Learning of Weight-Parameters	Model-Construction	Theory-Indifferent
Post Hoc Model Assessment	Model-Use	Theory-Aided

Table1. Varieties of theory-ladenness in ML.

References

Anderson, C. (2008). The end of theory: The data deluge makes the scientific method obsolete. *Wired magazine*, *16*(7), 16-07.

Aktunc, M. E. (2021). Productive theory-ladenness in fMRI. *Synthese*, *198*(9), 7987–8003. https://doi.org/10.1007/s11229-019-02125-9

Baldi, P. (2021). *Deep Learning in Science*. Cambridge University Press.

Bechtel, W., & Richardson, R. (2010). *Discovering Complexity - Decomposition and Localization as Strategies in Scientific Research*. The MIT Press.

Boge, F. J. (2022). Two Dimensions of Opacity and the Deep Learning Predicament. *Minds and Machines*, *32*(1), 43–75. https://doi.org/10.1007/s11023-021-09569-4

Bokulich, A. (2011). How scientific models can explain. *Synthese*, *180*(1), 33–45. https://doi.org/10.1007/s11229-009-9565-1.

Bovens, L., & Hartmann, S. (2004). *Bayesian Epistemology*. Oxford University Press, Oxford.

Burrell, J. (2016). How the machine 'thinks': Understanding opacity in machine learning algorithms. *Big Data & Society*, *3*(1), 205395171562251. https://doi.org/10.1177/2053951715622512

Capriotti, E., & Altman, R. B. (2011). A new disease-specific machine learning approach for the prediction of cancer-causing missense variants. *Genomics*, *98*(4), 310–317. https://doi.org/10.1016/j.ygeno.2011.06.010

Cartwright, N., Shomar, T., & Suarez, M. (1995). The tool box of science. *Poznan Studies in the Philosophy of the Sciences and the Humanities*, *44*, 137–149.

Craver, C., & Darden, L. (2013). *In search of Mechanisms*. The University of Chicago Press.

Creel, K. A. (2020). Transparency in complex computational systems. *Philosophy of Science*, *87*(4), 568–589. https://doi.org/10.1086/709729

Darden, L. (2006). *Reasoning in Biological discoveries*. Cambridge University Press.

Durán, J. M., & Formanek, N. (2018). Grounds for trust: Essential epistemic opacity and computational reliabilism. *Minds and Machines*, *28*, 645-666.

Elliott, K. C. (2007). Varieties of Exploratory Experimentation in Nanotoxicology. In *Philosophy of the Life Sciences* (Vol. 29, Issue 3). https://www.jstor.org/stable/23334264

Eitel, F., Schulz, M. A., Seiler, M., Walter, H., & Ritter, K. (2021). Promises and pitfalls of deep neural networks in neuroimaging-based psychiatric research, *Experimental Neurology* (Vol. 339). https://doi.org/10.1016/j.expneurol.2021.113608

Facchini, A., & Termine, A. (2021). Towards a taxonomy for the opacity of AI systems. In *Conference on philosophy and theory of artificial intelligence* (pp. 73-89). Cham: Springer International Publishing.

Franklin, L. R. (2005). Exploratory Experiments. *Philosophy of Science*, *72*(December), 888–899.

Gene Ontology Consortium. (2000). Gene Ontology : tool for the unification of biology. *Nature Genetics*, *25*(1), 25–29. https://doi.org/10.1038/75556.Gene

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

Hansen, J. U., & Quinon, P. (2023). The importance of expert knowledge in big data and machine learning. *Synthese*, *201*(2). https://doi.org/10.1007/s11229-023-04041-5

Hanson, N. (1958). *Patterns of Discovery*. Cambridge University Press.

Heidelberger, M. (2003). Theory-Ladenness and Scientific Instruments in Experimentation. In H. Radder (Ed.), *The Philosophy of Scientific Experimentation*. University of Pittsburgh Press.

Jordan, E. J., & Radhakrishnan, R. (2015). Machine Learning Predictions of Cancer Driver Mutations, *Proc 2014 6th Int Adv Res Workshop In Silico Oncol Cancer Investig* (2014). 2014 Nov: 10.1109/iarwisoci.2014.7034632.

Kaddour, J., Lynch, A., Liu, Q., Kusner, M. J., & Silva, R. (2022). Causal machine learning: A survey and open problems. *arXiv preprint arXiv:2206.15475*.

Kuhn, T. (1970). *The Structure of Scientific Revolutions*. University of Chicago Press.

Leonelli, S. (2016). *Data-centric Biology*. University of Chicago Press.

Milgroom, M. G. (2023). Epidemiology and SIR Models. *Biology of Infectious Disease: From Molecules to Ecosystems*, 253-268.

Mcmullin, E. (1968). What do Physical Models Tell us? *Studies in Logic and the Foundations of Mathematics*, *52*(C), 385–396. https://doi.org/10.1016/S0049-237X(08)71206-0

Morgan, M., & Morrison, M. (1999). *Models as Mediators: Perspectives on Natural and Social Sciences*. Cambridge University Press.

Morrison, M. (1999). Models as autonomous agents. In M. Morgan & M. Morrison (Eds.), *Models as Mediators - Perspectives on Natural and Social Science*. Cambridge University Press.

Napoletani, D., Panza, M., & Struppa, D. (2021). The Agnostic Structure of Data Science Methods. *Lato Sensu: Revue de La Société de Philosophie Des Sciences*, *8*(2), 44–57. https://doi.org/10.20416/lsrsps.v8i2.5

Páez, A. (2019). The pragmatic turn in explainable artificial intelligence (XAI). *Minds and Machines*, *29*(3), 441-459.

Parker, W. S. (2009). Does matter really matter? Computer simulations, experiments, and materiality. *Synthese*, *169*(3), 483–496. https://doi.org/10.1007/s11229-008-9434-3

Pearl, J. (2009). *Causality*. Cambridge University Press.

Peschard, I., & van Fraassen, B. (Eds.). (2018). *The Experimental Side of Modeling*. University of Minnesota Press.

Pietsch, W. (2015). Aspects of Theory-Ladenness in Data-Intensive Science. *Philosophy of Science*, *82*(5), 905–916. https://doi.org/10.1086/683328

Pietsch, W. (2022). *On the Epistemology of Data Science - Conceptual Tools for a New Inductivism*. Springer. https://link.springer.com/bookseries/6459

Radder, H. (2003). Technology and Theory in Experimental Science. In H. Radder (Ed.), *The Philosophy of Scientific Experimentation*. University of Pittsburgh Press.

Rani, V., Nabi, S. T., Kumar, M., Mittal, A., & Kumar, K. (2023). Self-supervised learning: A succinct review. *Archives of Computational Methods in Engineering*, *30*(4), 2761-2775.

Ratti, E. (2020). What kind of novelties can machine learning possibly generate? The case of genomics. *Studies in History and Philosophy of Science Part A*, *83*, 86–96. https://doi.org/10.1016/j.shpsa.2020.04.001

Rheinberger, H.-J. (1997). *Toward a History of Epistemic Things: Synthetizing Proteins in the Test Tube*. Stanford University Press.

Schindler, S. (2013). Theory-laden experimentation. *Studies in History and Philosophy of Science Part A*, *44*(1), 89–101. https://doi.org/10.1016/j.shpsa.2012.07.010

Schölkopf, B., Locatello, F., Bauer, S., Ke, N. R., Kalchbrenner, N., Goyal, A., & Bengio, Y. (2021). Toward causal representation learning. *Proceedings of the IEEE*, *109*(5), 612-634.

Selbst, A. D., & Barocas, S. (2018). The intuitive appeal of explainable machines. *Fordham Law Review*, *87*(3), 1085–1139. https://doi.org/10.2139/ssrn.3126971

Steinle, F. (1997). Entering New Fields: Exploratory Uses of Experimentation. In *Philosophy of Science* (Vol. 64). https://www.jstor.org/stable/188390

Suárez, M., & Cartwright, N. (2008). Theories: Tools versus models. *Studies in History and Philosophy of Science Part B - Studies in History and Philosophy of Modern Physics*, *39*(1), 62–81. https://doi.org/10.1016/j.shpsb.2007.05.004

Sullivan, E. (2022). Understanding from machine learning models. *The British Journal for the Philosophy of Science*.

Tal, E. (2020) "Measurement in Science", *Stanford Encyclopedia of Philosophy*

Vanschoren, J. (2019). Meta-learning. *Automated machine learning: methods, systems, challenges*, 35-61.

Waters, C. K. (2007). The Nature and Context of Exploratory Experimentation. *History and Philosophy of the Life Sciences*, *29*, 1–9.

Wilholt, T. (2005). Explaining models: Theoretical and phenomenological models and their role for the first explanation of the hydrogen spectrum. *Foundations of Chemistry*, *7*(2), 149–169. https://doi.org/10.1007/s10698-004-5958-x

Yang, Z., Zeng, X., Zhao, Y., & Chen, R. (2023). AlphaFold2 and its applications in the fields of biology and medicine. *Signal Transduction and Targeted Therapy*, *8*(1), 115.

Zednik, C. (2021). Solving the black box problem: A normative framework for explainable artificial intelligence. *Philosophy & Technology*, *34*(2), 265-288.

Zednik, C., & Boelsen, H. (2022). Scientific Exploration and Explainable Artificial Intelligence. *Minds and Machines*, *32*(1), 219–239. https://doi.org/10.1007/s11023-021-09583-6