Lato Sensu
REVUE
DE LA SOCIÉTÉ
DE PHILOSOPHIE
DES SCIENCES

Vol 8 N°2 2021 http://dx.doi.org/10.20416/LSRSPS.V8I2.5

Domenico Napoletani, Marco Panza et Daniele Struppa

The Agnostic Structure of Data Science Methods

SOCIÉTÉ DE PHILOSOPHIE DES SCIENCES (SPS)
École normale supérieure
45, rue d'Ulm
75005 Paris
www.sps-philoscience.org

lato sensu
REVUE
DE LA SOCIÉTÉ
DE PHILOSOPHIE
DES SCIENCES

Domenico Napoletani, Marco Panza and Daniele Struppa

The Agnostic Structure of Data Science Methods

Sommaire

In this paper we argue that data science is a coherent and novel approach to empirical problems that, in its most general form, does not build understanding about phenomena. Within the new type of mathematization at work in data science, mathematical methods are not selected because of any relevance for a problem at hand; mathematical methods are applied to a specific problem only by `forcing', i.e. on the basis of their ability to reorganize the data for further analysis and the intrinsic richness of their mathematical structure. In particular, we argue that deep learning neural networks are best understood within the context of forcing optimization methods. We finally explore the broader question of the appropriateness of data science methods in solving problems. We argue that this question should not be interpreted as a search for a correspondence between phenomena and specific solutions found by data science methods; rather, it is the internal structure of data science methods that is open to precise forms of understanding.

Mots-clés : philosophie des mathématiques, philosophie des sciences, big data.
Keywords : Philosophy of mathematics, Philosophy of science, Big Data.

1 – Introduction

In this paper we discuss the changing role of mathematics in science, as a way to discuss some methodological trends at work in big data science. Classically, any application of mathematical techniques requires a prior understanding of the phenomena and of the mutual relations among the relevant data. Modern data analysis, on the other hand, does not require this. It rather asks mathematics to re-organize data in order to reveal possible patterns uniquely attached to the specific questions we may ask about the phenomena of interest. These patterns may or may not provide further understanding *per se*, but nevertheless provide an answer to these questions.

It is due to this diminished emphasis on understanding that we suggested in (Napoletani, Panza and Struppa 2011) that this approach should be denoted using the label 'agnostic science', and we speak of 'blind methods' to denote individual instances of agnostic science. These methods usually rely only on large and diverse collections of data to answer questions about a phenomenon. As we will see in Section 3, a reliance on large amounts of data is, however, not sufficient in itself to make a method in data science blind.

The lack of understanding of phenomena in agnostic science makes the solution to any specific problem dependent on patterns identified automatically through mathematical methods. At the same time, this approach calls for a different kind of understanding of what makes mathematical methods and tools well adapted to these tasks.

One current explanation of the power of data science is that it succeeds only because of the relevance and size of the data. Accordingly, the use of mathematics in data science results in disconnected methods devoid of a common structure. This view would amount to a new Wignerian paradox of 'unreasonable effectiveness', where such effectiveness is assigned to data and not to mathematics. This affirmation of the exclusive primacy of data could be thought of as revenge of facts against their mathematization.

We reject such an answer as both immaterial and unsupported. In our rejection we do not argue that any exploration of data is doomed to fail without some previous understanding of the phenomenon. In other words, we do not oppose data science's methods by defending the classical approach. Rather, we observe the effectiveness of agnostic science in the absence of previous understanding, and consider what makes this possible.

Whilst we do not advance here any comprehensive reason for the effectiveness of data science, we observe that no account is possible unless we engage in a technical inquiry of how these algorithms work, and suggest a largely schematic account of their *modus operandi*. This account relies on the results of our previous work (Napoletani, Panza and Struppa 2011, 2014, 2017), which we reorganize in a comprehensive way. Furthermore, we identify a possible direction for future research and a promising perspective from which to approach the question.

In (Napoletani, Panza and Struppa 2011) we discussed the lack of understanding of the current big data methods. In doing so, we did not borrow any general and univocal notion of understanding for an empirical (physical, biological, biomedical, social, etc.) phenomenon. We simply observed that big data methods do not typically apply to the study of a certain empirical phenomenon, but rather they apply to a pair comprising a phenomenon and a question about it.

Such methods are used when it is impossible to identify a small number of independent variables whose measurement suffices to describe the phenomenon and to answer the question at hand.

We have argued that when this happens, no appropriate understanding of the phenomenon is available, regardless of how one could conceive of the notion of understanding. This highlights the need to understand why (and when) mathematics is successful in agnostic science, despite the blindness of its methods.

Let us now briefly describe the structure of the paper.

In Section 2, we discuss what we consider to be the basic trend of agnostic science: the 'microarray paradigm', as we called it in (Napoletani, Panza and Struppa 2011). This name was chosen to reflect the fact that this trend first became manifest in biology and biomedical sciences, though it is now pervasive in all data science. It is characterized by the handling of large amounts of data, whose specific provenance is often unknown, and whose modes of selection are often disconnected from any previous identification of a relevant structure in the phenomenon under observation. This feature is intended as the most notable virtue of the paradigm, since it allows investigation of the relevant phenomena, and specifically the data that have been gathered, without any previous hypothesis on possible correlations or causal relationships between the measured variables.

As already noted, there is an important distinction to be made between the use of powerful and often uncontrollable algorithms on large amounts of data and real agnostic science. This distinction will be investigated in Section 3 with the help of a negative example, the PageRank algorithm used by Google to weight web pages. The key point is that the lack of local control over the algorithm being used here, is not the same as a lack of understanding of the relevant phenomenon. The former is true of any algorithm working on large amounts of data, such as PageRank; the latter is, by definition, the characteristic feature of agnostic science.

In Section 4, we will go further in our analysis of agnostic science, by investigating the relations between optimization and 'forcing', a term we first introduced in this context in (Napoletani, Panza and Struppa 2011). More specifically, by forcing we referred in (Napoletani, Panza and Struppa 2017) to the following methodological practice:

> *The use of specific mathematical techniques on the available data is not motivated by the understanding of the relevant phenomena, but by the ability of such techniques to structure the data to be amenable to further analysis.*

For example, we could impose continuity and smoothness on the data, even in the case of variables that can only take a discrete value, just to be able to use derivatives and differential equations to analyze them. In cases such as these, we say that we are forcing mathematics over the available data.

In our terminology, optimization itself can be seen as a form of forcing when we carefully consider the ways it is used within agnostic science.

By better describing this *modus operandi* in relation to deep learning techniques, we will also make clear that the reasons for the effectiveness of optimization cannot be regarded as the rationale for the success of agnostic science. In short, this is because optimization is itself a form of forcing and does not ensure that the optimal solutions found by such methods correspond to anything of significance in the evolution or state of the phenomenon. Having made this clear, in Section 5 we outline a tentative answer to the question of the success and the appropriateness of agnostic science, by indicating a possible direction for further reflection.

Our suggestion is based on the observation that blind methods can be regarded as complying with a simple and general prescription for the development of an algorithm, what we called 'Brandt's principle' in (Napoletani, Panza and Struppa 2017). We then discuss the way in which data science (Hastie, Tibshirani and Friedman 2016, Chapter 16) relies on entire families of methods ('ensemble methods'), and we interpret this in light of the microarray paradigm, ultimately showing how this forces some implicit analytic constraints on blind methods. Finally, we propose that these analytic constraints, when taken together with Brandt's principle, exert strong restrictions on the relevant data sets that agnostic science can deal with.

2 – The Microarray Paradigm and Supervised Learning

We now describe in some detail a biological problem and a corresponding experimental technique. This provides a paradigmatic example of a blind method, and illustrates the typical way agnostic science works.

One of the great advances in biology and medical practice has been the understanding of the relevance of genes in the development of several diseases such as cancer. The impact of genetic information, as encompassed by DNA sequences, is primarily mediated in an organism by its expression through corresponding messenger RNA (mRNA) molecules. We know that the specific behavior of a cell largely depends on the activity, concentration, and state of proteins in the cell, and the distribution of proteins is, in turn, influenced by the changes in levels of mRNA. This opens up the possibility of understanding diseases and their genetic basis through the analysis of mRNA molecules.

The mechanism that leads from a certain distribution of mRNA molecules to the manifestation of a certain disease is, however, rarely understood. In addition, it is also unclear which specific mRNA molecules are relevant in particular diseases. Biologists developed a technique, called 'DNA microarray', that can to some extent bypass this lack of understanding, and enable the identification of patterns within mRNA distributions, that may be markers for the presence of some diseases.

We first briefly describe the experimental structure of the DNA microarray and the way it can be used in diagnostics (we refer to Napoletani, Panza and Struppa 2011 for a list of references on this technique, or to Amaratunga, Cabrera and Shkedy 2014 for a broad introduction).

A DNA microarray is essentially a matrix of microscopic sites where several thousand different short pieces of a single strand of DNA are attached. Messenger RNA (mRNA) molecules are extracted from specific tissues of different patients, then amplified and marked with a fluorescent substance and finally dropped on each site of the microarray.

This makes each site show different levels of florescence according to the amount of mRNA that binds with the strands of DNA previously placed in it. The intensity and distribution of the fluorescence provides a way to evaluate the degree of complementarity of the DNA and the mRNA strands.

This provides a correspondence between the information displayed by a DNA microarray and the behavior of a cell. This correspondence, however, is by no means exact or unequivocal, since the function of many proteins in the cell is not known, and several strands of DNA are complementary to the mRNA strands of all protein types. Nevertheless, thousands of strands of DNA are checked on a single microarray, so that one might expect this method to offer a fairly accurate description of the state of the cells, even if it does not offer any understanding of what is happening in the relevant tissues. The microarray is, indeed, particularly valuable for a huge number of variables, whose relation to each other and to the state of the cell we ignore.

This does not forbid, for example, the use of microarrays for the diagnosis of many illnesses. By measuring the activity of proteins, one may be able to distinguish patients affected by a certain pathology from those patients that are not, even without knowing the reason for the differences.

From a mathematical point of view, this process can be described as follows: let us label ‘X_i’ the vector of expression levels of mRNA strands associated with a patient i: the hope is to be able to find a function F such that $F(X_i) = 0$ if the patient does not have a particular disease, and $F(X_i) = 1$ if the patient does have the disease. The question of how to find such a function F is at the core of agnostic science, and we will come back to it shortly.

This brief description of DNA microarrays should be enough to justify why this technology can be taken as a paradigmatic example of the way agnostic science works. In short, this point of view can be summarized as follows:

> *If enough and sufficiently diverse data are collected regarding a certain phenomenon, we can answer all relevant questions about it.*

This slogan, which we refer to as the ‘microarray paradigm’, pertains not only to this specific example, but also applies to agnostic science as whole.[1] The question we want to tackle here is what makes this paradigm successful (at least in a relevant number of cases).

To better grasp the point, let us sketch the general scheme that agnostic science applies under this paradigm: data are processed through an appropriate algorithm that works on the available data independent of their specific nature and of any knowledge concerning the possible relationships between the relevant variables. The process is subject to normalization constraints imposed by the data, rather than by the (unknown) structure of the phenomenon. This treatment produces an output which is taken as an answer to a specific question about this phenomenon.

This approach makes it impossible to generalize the results or even deal with a change of scale; different questions require different algorithms whose structure is general and applied uniformly across different problems. Moreover, the specific mathematical way in which a question is formulated depends on the structure of the algorithm which is used, and not the other way around.

To illustrate how blind methods work, we will provide a quick overview of supervised machine learning (we shall come back to this description in more detail later). While this is just an example, it will show the strong similarity between blind methods and interpolation and approximation theory.

The starting point is a training set (X,Y), constituted by M pairs (X_i,Y_i) (i = 1,2, … ,M), where each X_i is typically an array $(X_i^{[j]})$ (j = 1,2, … , N) of given values of N variables. For example, in the DNA microarray example, each array X_i is the expression of mRNA fragments detected by the microarray, while the corresponding Y_i indicates the presence ($Y_i = 1$) or absence ($Y_i = 0$) of a given disease.

By looking at this training set with the help of an appropriate algorithm, the data scientist looks for a function F such that $F(X_i) = Y_i$ or $F(X_i) \approx Y_i$ (i = 1,2, … ,M). This function is usually called a ‘classifier’ if the output is categorical, and a ‘model’ or a ‘learned function’ for continuous outputs. In the following cases, we refer to F as a ‘fitting function’, to stress the connection of supervised machine learning with approximation theory.

In general, one looks for a function F that satisfies these conditions for most indices i. Moreover, it is helpful if the function belongs to a functional space $\mathcal{F}$ selected because of its ability to approximate general regular functions in a computationally efficient way. For example, if the regular functions of interest are analytical functions over an interval, $\mathcal{F}$ can be taken to be the space of polynomial functions defined on the same interval.

Let $\mathcal{A}$ be the space of parameters that define a function in $\mathcal{F}$, and denote its elements as $F_a(X)$, where a is an array of parameters in $\mathcal{F}$. The standard way to find the most suitable $F_a \in \mathcal{F}$ for a supervised learning problem is akin to functional approximation that can be described as follows. One de-

[1] - *We introduced the term ‘microarray paradigm’ in (Napoletani, Panza and Struppa 2011) to refer to a specific attitude towards the solution of data science problems, and not only to denote the technique underlying DNA microarrays. The microarray paradigm, as an attitude to the solution of problems, should not be confused with agnostic science, which is a general scientific practice implementing this attitude. It should also not be confused with any particular blind method, namely a specific way to implement the microarray paradigm.*

fines a 'fitness function' $E(a)$ by comparing the output $F_a(X_i)$ to the corresponding value Y_i at each X_i, and by setting

$$E(a) = \sum_i (Y_i - F_a(X_i))^2.$$

The most suitable $F_{\bar{a}}$ is then identified, by seeking a value $\bar{a}$ that minimizes $E(a)$. The function $F_{\bar{a}}$ selected in this way is then tested on a testing set (X_i,Y_i), $(i = M+1, M+2, \dots M+M')$, and, if it is found to be accurate on this set as well, it is used by analytical continuation to forecast $\tilde{Y}$ when a new instance of argument $\tilde{X}$ is taken into account.

Though brief, this description shows that supervised machine learning consists of analytically continuing a function found by constraining its values on a discrete subset of points defined by the training set. Moreover, supervised learning gives a mathematical form to the basic classification problem; given a finite set of classes of objects, and a new object that is not labelled, find the class of objects to which it belongs. What is relevant, however, is that each of the numbers M, N, and M' may be huge, and we have no idea of how the values Y_i depend on the values X_i, or how the values in each array X_i are related to each other. In particular, we do not know whether the variables taking these values are reducible (that is, depend on each other in some way) or whether it is possible to apply suitable changes of scale or normalization on the variables.

Supervised learning is therefore emblematic of agnostic science since we have no way to identify a possible interpolating function F_a, except the use of appropriate algorithms. Our lack of understanding of the phenomenon ensures that there is no effective criterion to guide the choice of the vector of parameters a, which are instead initially taken to be arbitrary values, and eventually corrected by successive reiterations of the algorithm, until some sort of stability is achieved.

Note that not all data science algorithms fall directly under the domain of supervised learning. For example, in unsupervised learning, the goal is not to match an input X to an output Y, but rather to find patterns directly in the set of inputs $\{X_1, \dots, X_M\}$. The most recent consensus is that unsupervised learning is most efficiently performed when conceived as a particular type of supervised learning (we shall come back to this point later). Another important modality of machine learning is reinforcement learning, a sort of semi-supervised learning strategy, where no fixed output Y is attached to X, and the fitting function F_a is evaluated with respect to a system of 'rewards' and 'penalties' such that the algorithm attempts to maximize the first and minimize the second. This type of machine learning is most often used when the algorithm needs to make a series of consecutive decisions to achieve a final goal (as for example when attempting to win in a game such as chess or Go). Similar to unsupervised learning, it has been shown (Mnih et al. 2015) that reinforcement learning works best when implemented in a modified, supervised learning setting.

Given the possibility of reducing both unsupervised and reinforcement learning to supervised learning schemes, we continue our analysis of supervised learning algorithms.

We now consider another important question suggested by the uncontrolled parameter structure of the supervised fitting function. Is it possible that, by working on a large enough data set, one can find arbitrary patterns in the data set that have no predictive power? The question can be addressed with the help of combinatorics, namely through Ramsey's theory. This makes it possible to establish, in many cases, the minimal size for a set $\mathcal{S}$ in order to enable the identification of a given combinatorial pattern in a subset of $\mathcal{S}$ (Graham, Rothschild and Spencer 2015). By adapting Ramsey's theory to data analysis, Calude and Longo (2017) have shown that with large enough data sets any possible correlation among the data can be established.

This might suggest that asking how much data is enough is only part of the problem.[2] To avoid the possibility of making the result of blind methods perfectly insignificant, another, possibly more important question is: how much data is too much?

There are several comments to be made on this matter.

To begin with, we note that Ramsey's theory proves the existence of lower bounds on the size of sets of data that ensures the appearance of correlations. However, these lower bounds are so large as to be of little significance for the size of data sets one usually handles.

More importantly, Ramsey's theory enables the presence of patterns in subsets of the initial data set to be established. In supervised learning, on the other hand, we require that every element of X matches with an appropriate element of Y; this is essentially different from seeking correlations in a subset of the data set. In other words, Ramsey's theory would only show that it is possible to write $F(X_i) = Y_i$ for some specific subset of elements of X and of Y. This would have no useful application in practice for supervised learning, where the totality of the available data must be properly matched.

Hence, as long as finding patterns within a data set X is tied to supervised learning, there is no risk of uncontrolled and spurious correlations. Instead, any such correlation will be strongly dependent on its relevance in finding the most appropriate fitting function F. Moreover, we will see in Section 4 that even when blind methods do not seem to fall within the structural constraints of supervised learning, they can still be reinterpreted as such.

We should add that agnostic science enters the game not in opposition to traditional, theoretically bound methods, but as another mode of exploration of phenomena, and it should in no way discourage, or inhibit the search for other methods based on previous understanding. Any form of understanding of the relevant phenomena is certainly welcome. Still, our point here is that there is no intrinsic methodological weakness in blind methods that is not, in one way or another, already implicit in those methodologies with a theoretical bent. At their core, they all depend on some sort of inductive infer-

[2] - On this question, note that there are also ways to apply data science to small data sets, if we accept strong limitations on the type of questions and we impose strong regularization restrictions on the type of solutions (Napoletani, Signore et al. 2011).

ence: the assumption that a predictive rule, or a functional interpolation of data, either justified by a structural account of phenomena, or by analytical continuation of interpolating functions, will continue to hold true when confronted with new observations.

Supervised learning shows that we can succeed, despite the obvious (theoretical and/or practical) risks, in using data to find patterns useful for solving specific problems with the available resources (though not necessarily patterns universally associated with the relevant phenomena). The degree of success is manifest in disparate applications such as face recognition (Schroff, Kalenichenko and Philbin 2015), automated translation algorithms (Wu et al. 2016), playing (and beating humans) at difficult games such as Go (Silver et al. 2016), and even the noteworthy progress in self-driving cars (Rao and Frtunikj 2018). This makes agnostic science both useful and welcome.

However, the intrinsic and unavoidable risks of agnostic science mean that it is important to understand why it frequently works well, and what makes it successful. We should not be blind as to why blind methods succeed! Lack of understanding of phenomena does not necessarily require lack of understanding of agnostic science itself. Rather, it demands such an understanding in order to provide some sort of indirect (scientific, methodological, political or ethical) control. This is the aim of an informed philosophy of data analysis, which shows not only its intellectual interest, but also its practical utility and necessity.

3 – Agnostic Science versus Lack of Control

Before continuing our search for such a (meta-)understanding, we observe that agnostic science is not equivalent to the use of data-driven algorithms on large amounts of data. As we will see in this section, we can single out computationally efficient algorithms that can be applied to extremely large data sets. Yet these very algorithms can be proven to converge. Since we fully understand their output and the structure of the data that makes them useful, they cannot be considered as blind algorithms and their use is not an example of agnostic science. To better illustrate this point, we will describe PageRank: the algorithm used by Google to weight web pages (Brin and Page 1998; Page et al. 1998).

Let A be a web page with n other pages T_i $(i = 1,2, \dots , n)$ pointing to it. We introduce a damping factor d_A $(0 \leq d \leq 1)$ that describes the probability that a random web surfer landing on A will leave the page. If $d_A = 0$, no surfer will leave the page A; if $d_A = 1$, every surfer will abandon the page. One can chose d_A arbitrarily or on the basis of any possible a priori reason. Such choice does not affect the outcome of the algorithm in the limit of a sufficiently large number of iterations of the algorithm itself. The PageRank of A is given by this formula:

$$PR(A) = \frac{1-d_A}{n} + d_A \left(\sum_{i=1}^{n} \frac{PR\,(T_i)}{C\,(T_i)} \right).$$

where $PR(T_i)$ and $C(T_i)$ are respectively the PageRank of T_i and the number of outgoing links starting at T_i.

This formula is very simple, but it is recursive; in order to compute $PR(A)$, one needs to compute the PageRank of all the pages pointing to A. In general, this makes it impossible to directly compute it, since if A points to some T_i, then in turn $PR(T_i)$ depends on $PR(A)$. However, this does not make the computation of $PR(A)$ impossible, since we can compute it by successive approximations: *(1)* we begin by computing $PR(A)$ choosing any arbitrary value for $PR(T_i)$, *(2)* the value of $PR(A)$ computed in step *(1)* is used to provisionally compute $PR(T_i)$, *(3)* next $PR(A)$ is recalculated on the basis of the values of $PR(T_i)$ found in *(2)*, and so forth for a sufficient number of times.

It is impossible to say a priori how many times the process has to be reiterated in order to reach a stable value for any page of the Web. Moreover, the actual complexity and dimension of the Web make it impossible to follow the algorithm's computation at any of its stages, and for all the relevant pages. This is difficult even for a single page A, if the page is sufficiently connected within the Web. Since the Web is constantly changing, the PageRank of each page is not fixed and has to be computed again and again such that the algorithm needs to be run continuously. Thus, the impossibility of any local control on this process is obvious.

Still, it can be demonstrated that the algorithm converges to the principal eigenvector of the normalized link matrix of the Web. This makes the limit PageRank of any page, namely the value of the PageRank of the given page in this vector, a measure of the centrality of this page in the Web.

Whether this actually measures the importance of the page is a totally different story. What is relevant is that the algorithm has been designed to compute the principal eigenvector, under the assumption that the value obtained in this way is an index of the importance of the page. Given any reasonable definition of importance, and under suitable conditions, it has been recently been proved (Masterton, Olsson and Angere 2016, Theorem 2) that PageRank will asymptotically (in the size of the Web) rank pages according to their importance.

This result confirms the essential point: the algorithm responds to a structural understanding of the Web, and to the assumption that the importance of any page is proportional to its centrality in its normalized link matrix. Then, strictly speaking, there is nothing blind in this approach, and using it is in no way an instance of agnostic science, although the Web is one of the most obvious examples of Big Data. So what makes blind methods blind, and agnostic science agnostic?

Agnostic science appears when, for the purpose of solving specific problems, one uses methods to search patterns which, unlike PageRank, correspond to no previous understanding.

This means we use methods and algorithms to find problem-dependent patterns in the hope that, once discovered, they will provide an apparent solution to the given problem. If this is so, then agnostic science is not only a way to solve problems from data without structural understanding, but also a family of mathematically sophisticated techniques to learn from experience by observation of patterns. Still, attention to invariant patterns is ultimately what Plato (*Theaetetus*, 155d) called 'astonishment [ϑαυμάζειν]', and considered to be 'the origin of philosophy [ἀρχὴ φιλοσοφίας]'. What happens with agnostic science is that we have too much data to be astonished by our experience as guided by the conceptual schemas we have at hand. So we use blind methods to look for sources of astonishment deeply hidden within these data.

4 – Forcing And Deep Learning Neural Networks

4.1 Forcing Optimality

We will now try to understand the features of an algorithm that make it suitable for identifying appropriate patterns within a specific problem.

The question has two facets. On the one hand, it consists of asking what makes these algorithms successful. On the other hand, it consists of wondering what makes them so appropriate (for a specific problem). The difficulty is that what appears to be a good answer to the first question seems to contradict, at least at first glance, the possibility of providing a satisfactory answer to the second question.

Indeed, with regard to the first question, we would say in our terminology, that the algorithms perform successfully because they act by forcing, i.e., by choosing interpolation methods and selecting functional spaces for the fitting functions in agreement with a criterion of intrinsic (mathematical) effectiveness, rather than conceiving these methods in connection with the relevant phenomena.

This answer seems to be in contrast with the possibility of providing a satisfactory answer to the second question, since it appears from the start to negate the possibility of understanding the appropriateness of methods in agnostic science. However, we do not think that this is the case. In this section we refine the notion of forcing by looking more carefully at the use of optimization in data science, and more specifically for a powerful class of algorithms, the so-called deep learning neural networks.

Finally, in Section 5, we explore several ways to make the answer to the question regarding the success of data science algorithms compatible with the existence of an answer to the appropriate question.

As we showed in (Napoletani, Panza and Struppa 2011, 2017), boosting algorithms is a clear example of forcing. Forcing is designed to improve weak classifiers, generally just slightly better than random ones, and to transform them, by iteration, into strong classifiers. This is achieved by carefully focusing each iteration on the data points that were misclassified in the previous iteration. Boosting algorithms are particularly effective in improving the accuracy of classifiers based on sequences of binary decisions (so called `classification trees'). Such classifiers are easy to build, but on their own are relatively inaccurate. Boosting can, in some cases, reduce error rates for simple classification trees from 45% to about 5% (Hastie, Tibshirani and Friedman 2016, Chapter 10).

Regularization algorithms offer a second example. If the data are too complicated and/or rough, these algorithms render them amenable to being treated by other algorithms, for example, by reducing their dimension. Despite the variety of regularization algorithms, they can all be conceptually equated to the process of approximating a non-necessarily differentiable function by a function whose derivative absolute value is bounded from above everywhere on its domain.

The use of these algorithms reveals a double application of forcing: forcing on the original data to smooth them, and then forcing on the smooth data to treat them with a second set of algorithms. For example, after a regularization that forces the data to be smooth, some data science methods advocate the forcing of unjustified differential equations in the search for a fitting function (Ramsery and Silverman 2005, chapter 19). These methods have been very effective in recognition of the authenticity of handwritten signatures and they depend essentially on the condition that the data are accounted for by a smooth function.

Since in virtually all instances of forcing, the mathematical structure of the methods is either directly or indirectly reducible to an optimization technique, we claim that optimization is a form of forcing within the domain of agnostic science.

In a sense, this is suggested by the historical origins of optimization methods (Panza 1995, 2003). When Maupertuis, then President of the Berlin Academy of Sciences, first introduced the idea of least action, he claimed to have found the quantity that God wanted to minimize when creating the universe. Euler, at the time a member of the Berlin Academy of Sciences, could not openly criticize the President, but clearly adopted a different attitude, by maintaining that action was nothing more than what was expressed by the equations governing the system under consideration. In other terms, he suggested that one should force the minimization (or maximization) of an expression like

$$\int F(x)dx$$

on any physical system in order to find the function *F* characteristic of it. *Mutatis mutandis*, this is the basic idea that we associate today with the Lagrangian of a system. Since that time, optimization became the pre-eminent methodology in solving empirical problems. One could say that the idea of a Lagrangian has been generalized to the notion of a fitting function, whose optimization characterizes the dynamics of a given system.

Though this might be seen as a form of forcing within a classical setting, one should note that, in this case, the only thing that is forced on the problem is the form of the relevant condition and the request that a certain appropriate integral reaches a maximum or minimum. In this classical setting, however, the relevant variables are chosen on the basis of a preliminary understanding of the system itself, and the relevant function is chosen so that its extremal values yield those solutions that have already been found in simpler cases.

Things change radically when the fitting function is selected within a convenient functional space through an interpolation process designed to make the function fit the given data. In this case, both the space of functions and the specific fitting procedure (which makes the space of functions appropriate) are forced on the system. These conditions are often not enough to select a unique fitting function or to find or ensure the existence of an absolute minimum, so that an additional choice may be required (forced) to this purpose.

There are many reasons why such an optimization process can be considered effective. One is that it matches the microarray principle; enough data, and a sufficiently flexible set of algorithms, will solve, in principle, any scientific problem. More concretely, optimization has shown to be both simple and relatively reliable, not necessarily to find the actual solution of a problem, but rather to obtain, without exceeding time and resource constraints, outcomes that can be taken as useful solutions to the problem. The outcomes of optimization processes can be tested in simple cases and shown to be compatible with solutions that had been found with methods based on a structural understanding of the relevant phenomenon.[3] In addition, the results of an optimization process may turn out to be suitable for practical purposes, even when it is not the best possible solution. An example is provided by algorithms for self-driving cars. In this case, the aim is not to mimic human reactions, but rather to simply have a car that can autonomously drive with sufficient attention to the safety of the driver and all other cars and pedestrians on the road.

This last example makes it clear that we can conceive optimization as a motivation for finding algorithms without being constrained by the search for the best solution. Optimization becomes a conceptual framework for the development of blind methods.

Blind methods are disconnected from any consideration of actual optimality; this sets them apart from methods in perturbation theory, where a solution to a more complex problem is derived from a (small) deformation of a solution of a simpler problem. On the one hand, there is no doubt that looking at supervised learning as a case of interpolation leads naturally to a comparison with such a theory, and the sophistication of its most modern versions (including perturbation methods in quantum field theory[4]) and may provide fundamental contributions to data science in this respect. On the other hand, blind methods place the process of interpolation at the center, rather than any correspondence between existing instances of solutions (i.e., simpler problems) and those to be determined (that we can equate to more complex problems).

Optimization as forcing also raises some important issues beyond the obvious one, which is typical of blind methods, namely the absence of an a priori justification.

One issue is that, in concrete data science applications such as pattern recognition or data mining, optimization techniques generally require fixing a large number of parameters, sometimes millions of them, which not only makes control of the algorithms hopeless, but also makes it difficult to understand the way algorithms work. This large number of parameters often results in a lack of robustness, since different initial choices of parameters can lead to completely different solutions.

Another issue is evident when considering the default technique of most large-scale optimization problems, the so-called point-by-point optimization. Essentially, this is a technique where the search for an optimal solution is done locally, by gradually improving any currently available candidate for the optimal choice of parameters. This local search can be done, for example, by using the gradient descent method (Goodfellow, Bengio and Courville 2016, Section 4.3), which does not guarantee that we will reach the desired minimum, or even a significant relative minimum. Since virtually all significant supervised machine learning methods can be shown to be equivalent to point-by-point optimization (Napoletani, Panza and Struppa 2017), we will briefly describe the gradient descent method.

If $F(X)$ is a real-valued multi-variable function, its gradient ∇F is the vector that gives the slope of its tangent oriented towards the direction in which it increases most. The gradient descent method exploits this fact to obtain a sequence of values of F which converges to a minimum. Indeed, if we take K_n small enough and set

$$x_{n+1} = x_n - K_n \nabla F\left(x_n\right) \quad (x = 0, 1, \ldots)$$

then

$$F\left(x_0\right) \geq F\left(x_1\right) \geq F\left(x_2\right), \ldots$$

we hope to see this sequence of values converge towards the desired minimum. However, this is only a hope, since nothing in the method can guarantee that the minimum it detects is significant.

4.2 Deep Learning Neural Networks

Let us now further illustrate the idea of optimization as forcing, by considering the paradigmatic example of deep learning neural networks (we follow here (Hastie, Tibshirani and Friedman 2016, Section 11.3; Goodfellow, Bengio and Courville 2016)) as it applies to the simple case of classification problems.

The basic idea is the same as anticipated above for the search of a fitting function F by supervised learning. One starts with a training set (X,Y), where X is a collection of M arrays of

[3] - *This is different from choosing the fitting function on the basis of solutions previously obtained with the help of an appropriate understanding.*

[4] - *For example, in (Napoletani, Petricoin and Struppa 2012) a general classification problem from developmental biology is formulated as a path integral akin to those used in quantum mechanics. Such integrals are usually analyzed with the help of perturbative methods such as WKB approximation methods, see (Schulman 2005, Chapter 18).*

variables:

$$X = (X_1, \dots, X_M) \qquad X_i = \left(X_i^{[1]}, \dots X_i^{[N]}\right)$$

and *Y* is a corresponding collection of *M* variables:

$$Y = (Y_1, \dots, Y_M)\,.$$

What is specific to deep learning neural networks is the set of specific steps used to recursively build *F*.

1. We build *K* linear functions
$$Q^{[k]}(X_i) = A_{0,k} + \sum_{n=1}^{N} A_{n,k} X_i^{[n]} \qquad (k = 1, \dots, K),$$
where $A_{n,k}$ are *K*(*N*+1) parameters chosen on some a priori criterion, possibly even randomly, and *K* is a positive integer chosen on the basis of the particular application of the algorithm.

2. One then selects an appropriate non-linear function *G* (we will say more about how this function is chosen) to obtain *K* new arrays of variables
$$H^{[k]}(X_i) = G\left(Q^{[k]}(X_i)\right) \qquad (k = 1, \dots, K).$$

3. One chooses (as in step 1) a new set of *T* (*K*+1) parameters $B_{k,t}$ in order to obtain *T* linear combinations of the variables $H^{[k]}(X_i)$
$$Z^{[t]}(X_i) = B_{0,t} + \sum_{k=1}^{K} B_{k,t} H^{[k]}(X_i) \qquad (t = 1, \dots, T),$$
where *T* is a positive integer appropriately chosen in accordance with the particular application of the algorithm.

If we stop after a single application of steps 1-3, the neural network is said to have only one layer (and is, then, 'shallow' or not deep). We can set *T*=1 and the process ends by imposing that all the values of the parameters are suitably modified (in a way to be described shortly) to ensure that:

$$Z^{[1]}(X_i) \approx Y_i \qquad (i = 1, \dots, M),$$

For any given new input $\tilde{X}$, we can then define our fitting function *F* as $F(\tilde{X}) = Z^{[1]}(\tilde{X})$.[5] In deep networks, steps 1-3 are iterated several times, starting every new iteration from the *M* arrays $Z^{[t]}(X_i)$ constructed by the previous iteration. This iterative procedure creates several 'layers', by choosing different parameters *A* and *B* (possibly of different sizes as well) at each iteration, with the obvious limitation that the dimension of the output of the last layer *L* has to match the dimension of the elements of *Y*. If we denote by $Z_L^{[1]}(X_i)$ the output of the last layer *L*, we impose, similar to the one layer case, that $Z_L^{[1]}(X_i) \approx Y_i$.

In other words, the construction of a deep learning neural network involves the repeated transformation of an input X by the recursive application of a linear transformation (step 1) followed by a non-linear transformation (step 2) and then another linear transformation (step 3).

The algorithm is also designed to facilitate learning, in the absence of *Y*, by using *X* itself, possibly appropriately regularized, in place of *Y* (auto-encoding). When an independent *Y* is used, the learning is called 'supervised' and provides an instance of the setting described in Section 3. In its absence, the learning is instead, called 'unsupervised' (Goodfellow, Bengio and Coureville 2016, chapter 14), and its purpose is to find significant patterns and correlations within the set *X* itself. The possibility of using an algorithm designed for supervised learning, for unsupervised learning is an important shift of perspective. It constrains the exploration of patterns within *X*, for the sole purpose of regularizing the data themselves. Whichever correlations and patterns are found, they will be instrumental in this specific aim, rather than in the ambiguous task of finding causal relationships within *X*.

Two things remain to be explained. The first concerns the non-linear function *G*, called 'activation function' (because of the origin of the algorithm as a model for neural dynamic). Such function can take different forms. Two classical examples are the sigmoid function

$$G(u) = \frac{1}{1 + e^{-u}}$$

and the ReLU (Rectified Linear Unit) function

$$G(u) = max(0, u).$$

This second function is composed of two linear branches and therefore is, mathematically speaking, much simpler than the sigmoid function. While also the ReLU function is not linear, it has uniform slope on a wide portion of its domain, and this seems to explain its significantly better performance as activation function for deep networks. The use of an appropriate activation function allows the method to approximate any function that is continuous on the compact sets in $\mathbb{R}^n$. This result is known as the universal approximation theorem for neural networks (Hornik 1991)

The second thing to be explained concerns the computation of the parameters according to the condition:

$$Z_L^{[1]}(X_i) \approx Y_i \qquad (i = 1, \dots, M).$$

This is typically achieved through the gradient descent method by minimizing a fitness function such as:

$$\sum_{i=1}^{M} \left[Y_i - Z_L^{[1]}(X_i)\right]^2.$$

The gradient is computed by an appropriate fast algorithm adapted to neural networks known as backpropagation (Hastie, Tibshirani and Friedman 2016, Section 11.4). As we have already noted in (Napoletani, Panza and Struppa 2013), the effectiveness of neural networks (both deep and shallow)

[5] - For classification problems, one often imposes $P(Z^{[1]}(X_i)) \approx Y_i, (i = 1, \dots, M)$, where P is a final, suitably chosen, output function, see (Hastie, Tibshirani and Friedman 2016, Section 11.3). For any new input $\tilde{X}$, the fitting function is, then, $F(\tilde{X}) = P(Z^{[1]}(\tilde{X}))$.

seems to depend more on the specific structure of the backpropagation algorithm than on the universal approximation properties of neural networks. Note also that the minimization of the fitness function is equivalent to a regularization of the final fitting function, if we stop the iterative application of the backpropagation algorithm when the value of the fitness function does not significantly decreases any further (Goodfellow, Bengio and Courville 2016, Section 7.8).

When dealing with deep networks, one can go as far as to consider hundreds of layers, though it is not generally true that increasing the number of layers always improves the minimum of the corresponding fitness function. Nevertheless, in many cases, taking more layers often allows up to a tenfold reduction of errors. For example, it has been shown that in a database of images of handwritten digits, classification errors go from a rate of 1.6% for a two-layer network (Hastie, Tibshirani and Friedman 2016, Section 11.7) to a rate of 0.23% with a network of about 10 layers (Ciresan, Meier and Schmidhuber 2012).

This short description of the way in which deep learning neural networks operate should be enough to clarify why we have taken them as an example of optimization by forcing. Above all, both the dependence of the effectiveness of neural networks on the structure of the backpropagation algorithm, and their interpretation as regularization, are clear indications that the way neural networks are applied is an instance of forcing.

More broadly, the opacity of the recursive process that creates the layers of the network is matched by computationally driven considerations that establish the specific type of gradient descent method to be used, and by a criterion to stop the whole iterative process that is simply based on the inability to find better solutions. However, this same description should be enough to clarify the second question mentioned at the start of this section: how can methods like deep learning neural networks be appropriate for solving specific problems when the methods themselves do not in any way reflect the particular features of the problems? We explore this question in the next section.

5 – On the Appropriateness of Blind Methods

5.1 Understanding Methods rather than Phenomena

A simple way to elude the question of the appropriateness of blind methods is by negating its premise. One can argue that, in fact, blind methods are in no way appropriate, that their success is nothing but appearance and that the faith in their success is actually dangerous, since such faith provides an incentive to the practice of accepting illusory forecasts and solutions.

The problem with this argument is that it ultimately depends on arguing that blind methods fail to succeed because they do not conform with the pattern of classical science. However, an objective look at the results obtained in data science should be enough to convince ourselves that this cannot be a good strategy. Of course, to come back to the example of DNA microarrays, grounding cancer therapy on microarrays alone is as inappropriate as it is dangerous, since, in such a domain, looking for causes is as crucial as it is necessary. At the same time, we cannot deny the fact that microarrays can be used as an evidential basis in a search for causes. In addition, we cannot deny that, in many successful applications of blind methods, such as handwriting recognition, the search for causes is much less crucial.

So we need another justification of the effectiveness of blind methods, which, far from negating the appropriateness question, takes it seriously and challenges the assumption that classical science is the only appropriate pattern for good science. Such an approach cannot depend, of course, on the assumption that blind methods succeed because they perform appropriate optimization. This assumption merely displaces the problem, since optimization is only used by these methods as a device to find possible solutions.

A more promising response to the question of the appropriateness of blind methods might be that they succeed for the same reason as classical induction does; blind methods are indeed interpolation methods on input/output pairs, followed by analytical continuation, which is how induction works. Of course, one could argue that induction itself is not logically sound, but should one really reject it as an appropriate method in science because of this? Is there another way to be empiricist other than trusting induction? Can one really defend classical science without accepting some form of empiricism, as refined as it might be? We believe that all these questions should be answered in the negative, and therefore that the objection itself is immaterial.

There are, however, two other important objections to this response.

The first is that it applies only to supervised methods, that is, methods based on the consideration of a training set on which interpolation is performed. It does not apply, at least not immediately, to unsupervised methods, where no sort of induction is present. However, this objection is superseded by noting that it is possible to reduce unsupervised methods to supervised ones through the auto-encoding regularization processes described above.

The second objection is more relevant. It consists of recognizing that, when forcing is at work, interpolation is restricted to a space of functions which is not selected by considering the specific nature of the relevant phenomenon, and that cannot be justified by any sort of induction. Rather, the choice of the functional space corresponds to a regularization of the data and it often modifies those data in a way that does not reflect, mathematically, the phenomenon itself.

This objection is not strong enough to force us to completely dismiss the induction response, but it makes it clear that advocating the power of induction cannot be enough to explain the success of agnostic science. This general response is, at least, to be complemented by a more specific and stronger approach.

In the remainder of this section, we would like to offer the beginning of a new perspective, consistent with our interpretation of the structure of blind methods.

The basic idea is to stop looking at the appropriateness question as a question concerning some kind of correspondence between phenomena and specific solutions found by blind methods. The very use of forcing makes this perspective illusory. We should instead look at the question from a more abstract, and structural perspective. Our conjecture, already advanced in (Napoletani, Panza and Struppa 2014, 2017), is that we can find a significant correspondence between the structure of the algorithms used to solve problems, and the way in which phenomena of interest in data science are selected and conceived. We submit, indeed, that the (general) structure of blind methods, together with the formal features of the microarray paradigm, exert strong restrictions on the class of data sets that agnostic science deals with.

5.2 Brandt's Principle and the Dynamics of Blind Methods

To justify the existence of these restrictions, we start by recalling a result of (Napoletani, Panza and Struppa 2017), that all blind methods share a common structure that conforms to the following prescription:

> *An algorithm that approaches a steady state in its output has found a solution to a problem, or needs to be replaced.*

In (Napoletani, Panza and Struppa 2017) we called this prescription 'Brandt's principle', to reflect the fact that it was first expounded by Achi Brandt for the restricted class of multi-scale algorithms (Brandt 2002).

As simple as Brandt's principle appears at first glance, in (Napoletani, Panza and Struppa 2017) we showed that this principle allows a comprehensive and coherent reflection of the structure of blind methods in agnostic science. First of all, Brandt's principle is implicit in forcing, since an integral idea in forcing is that if an algorithm does not work, another one must be chosen. More specifically, the key to the power of this principle is that the steady state output of each algorithm, when it is reached, is chosen as input to the next algorithm if a suitable solution to the initial problem has not yet been found.

Notably, deep learning architecture matches with Brandt's principle, since the iteration of the gradient descent algorithm is generally stopped when the improvement of parameters reaches a steady state and, then, either the function that has been obtained is accepted and used for forecasting or problem-solving, or the algorithm is replaced by a new one, or at least re-applied starting from a new assignation of values to the initial parameters. Moreover, all local optimization methods satisfy this principle and since most algorithms in agnostic data science can be rewritten as local optimization methods, we can say that virtually all algorithms in agnostic data science can be written in this way.

In (Napoletani, Panza and Struppa 2017) we argued that thinking about algorithms in terms of Brandt's principle often sheds light on those characteristics of a specific method that are essential to its success. For example, the success of deep learning algorithms, as we have seen in the previous section, relies in a fundamental way on two advances: (1) the use of the ReLU activation function that, thanks to its constant slope for non-zero arguments, enables the fast exploration of the parameter space with gradient descent, and (2) a well-defined regularization obtained by stopping the gradient descent algorithm when error rates no longer improve significantly. Both these advances took a significant amount time to be identified as fundamental to the success of deep learning algorithms, perhaps exactly because of their deceptive simplicity, and yet both of them are naturally derived from Brandt's principle.

There is, however, a possible objection to ascribing a decisive importance to this principle (one that is in the same vein as that discussed in Section 2, considering an argument from (Calude and Longo 2017)). This objection relies on the observation that, in practical applications, agnostic science works with floating-point computations which require a finite set of floating-point numbers. The point, then, is that any iterative algorithm on a finite set of inputs, reaches a limit cycle in a finite period of time, in which case steady states satisfying Brandt's principle become trivial and uninformative regarding the nature of the subjacent phenomenon.

However, blind methods that satisfy Brandt's principle, such as boosting algorithms and neural networks, will usually converge to steady states after just a few thousand iterations. Limit cycles in an algorithm's output due to the limitations of floating-point arithmetic, will instead appear after a very large number of iterations, comparable to the size of the set of floating-point numbers. Any practical application of Brandt's principle needs to take this into consideration by imposing, for example, that the number of iterations necessary to reach a steady state is at most linear in the size of the training set.

Regardless of this practical limitation in recognizing steady states, the real significance of Brandt's principle for supervised learning algorithms is that it shifts the attention from the optimization of the fitting function F to the study of the dynamics of the algorithm's output. In this perspective, applying Brandt's principle depends on building sequences of steady-state fitting functions $\{F_j\}$ that are stable in their performance under the deformations induced by the algorithms chosen during the implementation of the principle itself. This generates a space of fitting functions that are mapped into each other by the different algorithms.

A clear example of this process is provided by boosting, where the entire family of fitting functions (that is recursively found) is robust in its performance, once the classification error on the training set stabilizes. Moreover, fitting functions found by the algorithm at each recursive step are built by combining all those that were found earlier (by weighting and adding them suitably). Indeed, boosting is an instance of ensemble methods, where distinct fitting functions are combined to find a single final fitting function. It can be argued that a lot of the recent progress in data science has been due to the recognition of the central role of ensemble methods (such as boosting), in greatly reducing error rates when solving problems (Hastie, Tibshirani and Friedman 2016, Chapter 16).

5.3 Ensemble Methods and Interpolation

Note that for ensemble methods to be trustworthy, eventually they must stabilize to a fitting function that does not significantly change with the addition of more sample points. Therefore, if we take the limit of an infinite number of sample points, generically distributed across the domain where the relevant problem is well defined, the fitting function is forced to be unique.

To understand the significance of this instance of forcing, we first rephrase the basic slogan of the microarray paradigm in more precise terms as a quantitative microarray paradigm:

> *Given enough sample data points, and for a large and suitably diverse set of variables X, the value of any other variable Y relevant for the solution of a given problem can generally be calculated from the value of X (via the fitting function F(X)=Y).*

Once this assumption is admitted, the unicity of *F* in the limit entails a form of analyticity on the fitting function which we call ‘asymptotic sample-analyticity’:

> *Let N be the dimension of X; then, for N sufficiently large, F(X)=Y is uniquely determined, on an appropriate domain, by a generic infinite set of sample points.*

In application, we will always have finite data, and we will not be able to choose *F* uniquely to solve a problem. However, suppose that the same solution is given by the entire class of asymptotically sample-analytic functions that are compatible with the available data. Do we trust that such a solution reflects a relevant actual property of the phenomenon at hand[6]? The question shifts attention from the nature of data sets to the nature of the space of functions defined on them, and on their assemblage by appropriate algorithms.

Our suggestion is that blind methods succeed when (and because) they select, in agreement with Brandt’s principle, appropriate classes of asymptotically sample-analytic functions, apt to robustly provide uniquely determinate solutions for the data problems at hand.

Despite this shift from data sets to functions on data sets, the properties of such functions enforce some general conditions on the data as well. First of all, the quantitative microarray paradigm essentially requires variables in the data set to be strongly interdependent in the limit of large data sets. Second, we claimed at the end of Section 5.2 that Brandt’s principle identifies a space (ensemble) of fitting functions that are stable in their ability to solve a given problem. Such stability is possible only if the functional relations that can be defined on the variables are themselves robust, so that they persist across the large data sets that are required by the microarray paradigm. It is not important for interdependence and robustness to be apparent, that is, we do not need to be able to identify the specific dependence of variables from each other. Nor it is important for robustness to warrant a long-term conservation of specific relations among the variables. What is relevant is that the interdependence is strong and persistent enough to allow iterative algorithms conforming to Brandt’s principle to subsequently correct their outputs by building more and more convenient fitting functions.

The requirements of interdependence of variables and robustness of functional relations among them, can now be used to discriminate data sets most suitable for the application of blind methods. For example, such requirements are believed to be satisfied by developmental biological systems, see (Minelli 2003). Moreover, in (Napoletani, Panza and Struppa 2017) we gave evidence that social and economical systems satisfy a generalization of the “principle of developmental inertia”, an organizing principle for developmental biology, first proposed in (Minelli 2011). It is therefore likely that the same interdependence and robustness satisfied by biological systems holds for most social and economical systems as well.

We conclude that the microarray paradigm and Brandt’s principle enforce specific requirements on data sets and that these requirements are likely to be satisfied by data arising from biological, social and economical systems. Blind methods would then be most appropriate when applied to such systems.

6 – Conclusion

In this paper we reviewed and extended a perspective on the methodological structure of data science which we have been building in a series of papers (Napoletani, Panza and Struppa 2011, 2013, 2014, 2017). The basic assumption of our approach is that data science is a coherent approach to empirical problems that in its most general form does not build understanding about phenomena. It is due to this characteristic that we labelled this approach to empirical phenomena ‘agnostic science’, and called the methods that make up agnostic science ‘blind methods’.

The basic attitude underlying agnostic science is the belief that if enough and sufficiently diverse data are collected regarding a certain phenomenon, it is possible to answer all relevant questions about it. In Section 2, we referred to this belief as the microarray paradigm and we explored the specific ways it is used in the practice of machine learning.

We noted in Section 3 that not all computational methods dealing with large data sets are properly within the domain of agnostic science, and we gave the example of PageRank, an algorithm used to weight web pages. The convergence of

[6] - *Note that sample-analyticity can be forced on any discrete variable in the problem after imposing continuity on such variables.*

this algorithm and the significance of its output are readily intelligible and therefore we argued that PageRank is not a blind method.

In Section 4.1 we explored how the microarray paradigm calls for a new type of mathematization in agnostic science, where mathematical methods are forced on the problem, i.e., they are applied to a specific problem only on the basis of their ability to reorganize the data for further analysis by general purpose techniques that are selected only on the basis of the richness of their mathematical structure, rather than by any particular relevance for the problem at hand. We then showed that optimization methods are used in data science as a form of forcing. This is particularly significant since virtually all methods of data science can be rephrased as a type of optimization method. In particular, in Section 4.2 we argued that deep learning neural networks are best understood within the context of forcing optimality.

In Section 5 we moved to the broader question of the appropriateness of blind methods in solving problems. In Section 5.1 we argued that this question should not be interpreted as a search for a correspondence between phenomena and specific solutions found by blind methods. Rather, it is the internal structure of blind methods that should be understood, and its implications on the structure of the data sets that are most appropriate for such methods.

In Section 5.2 we reviewed a simple prescription on algorithms, Brandt's principle, which asserts that an algorithm that approaches a steady state in its output has found a solution to a problem, or needs to be replaced. One of our main claims in (Napoletani, Panza and Struppa 2017) was that Brandt's principle is ideally suited for the understanding of the dynamical structure of blind methods. For example, in Section 5.2 we used Brandt's principle to understand two of the significant innovations of deep learning neural networks: the use of the ReLU activation function in the network, and an efficient criterion for early stopping of the algorithm. Ensemble methods, where distinct fitting functions are combined to find a single final fitting function, can also be interpreted within the context of Brandt's principle.

In Section 5.3 we showed that Brandt's principle and the microarray paradigm force a specific type of analytical structure, which we call 'sample-analyticity', on the final fitting function found by ensemble methods. We argued that sample-analyticity forces a shift from data sets to functions on data sets. In turn, the properties of such functions enforce two general conditions on the data sets: a strong interconnectedness of the variables of the data set, and the robustness of the functional relations of such variables.

Finally, we speculated that blind methods are most appropriate for the solution of problems in biological, social and economical systems, since data sets arising from these systems are likely to satisfy the two conditions above.

ACKNOWLEDGMENTS

We thank Maxime Darrin for reading and commenting on a preliminary version of this paper and the anonymous referees for many detailed and useful remarks.

RÉFÉRENCES

Dhammika Amaratunga, Javier Cabrera, Ziv Shkedy. 2014. *Exploration and Analysis of DNA Microarray and Other High-Dimensional Data*. Hoboken: John Wiley & Sons.
https://doi.org/10.1002/9781118364505

A. Brandt. 2002. Multiscale Scientific Computation: Review 2001. In T. J. Barth, T. F. Chan, R. Haimes (eds.) *Multiscale and Multiresolution Methods: Theory and Applications*. Berlin-Heidelberg: Springer Verlag, 3-95.
https://doi.org/10.1007/978-3-642-56205-1_1

S. Brin & L. Page. 1998. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30, 107-117.
https://doi.org/10.1016/j.comnet.2012.10.007

C. Calude, G. Longo. 2017. The Deluge of Spurious Correlations in Big Data. *Foundations of Science*, 22, 595–612.
https://doi.org/10.1007/s10699-016-9489-4

D. Ciresan, U. Meier, J. Schmidhuber. 2012. Multi-column deep neural networks for image classification. *CVPR '12 Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE Computer Society, 3642-3649.
https://doi.org/10.1109/cvpr.2012.6248110

I. Goodfellow, Y. Bengio, A. Courville. 2016. *Deep Learning*. The MIT Press.
https://doi.org/10.1007/s10710-017-9314-z

R. L. Graham, B. L. Rothschild, Joel H. Spencer. 2015. *Ramsey Theory*. Hoboken: John Wiley & Sons.
https://doi.org/10.1038/scientificamerican0790-112

T. Hastie, R. Tibshirani, J. Friedman. 2016. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Second Edition, Springer Series in Statistics, Springer.
https://doi.org/10.1007/978-0-387-84858-7

K. Hornik. 1991. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2), 251–257.
https://doi.org/10.1016/0893-6080(91)90009-t

G. Masterton, E. J. Olsson & S. Angere. 2016. Linking as voting: how the Condorcet jury theorem in political science is relevant to webometrics. *Scientometrics*, 106(3), 945-966.
https://doi.org/10.1007/s11192-016-1837-1

A. Minelli. 2003. *The Development of Animal Form: Ontogeny, Morphology, and Evolution*. Cambridge: Cambridge University Press.
https://doi.org/10.1017/CBO9780511541476

A. Minelli, 2011. A principle of Developmental Inertia. In B. Hallgrimsson and B. K. Hall (eds.) *Epigenetics: Linking Genotype and Phenotype in Development and Evolution*. Berkeley, CA: University of California Press.

V. Mnih, K. Kavukcuoglu, . Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg and D. Hassabis. 2015. Humanlevel control through deep reinforcement learning. *Nature*, 518, 529-533
https://doi.org/doi:10.1038/nature14236

D. Napoletani, M. Panza, and D.C. Struppa. 2011. Agnostic science. Towards a philosophy of data analysis. *Foundations of Science*, 16, 1-20.
https://doi.org/10.1007/s10699-010-9186-7

D. Napoletani, M. Panza, and D.C. Struppa. 2013. Processes rather than descriptions? *Foundations of Science*, 18(3)3, 587-590.
https://doi.org/10.1007/s10699-013-9332-0

D. Napoletani, M. Panza, and D.C. Struppa. 2014. Is big data enough? A reflection on the changing role of mathematics in applications. *Notices of the American Mathematical Society*, 61(5), 485-490.
https://doi.org/10.2307/j.ctvc778jw.29

D. Napoletani, M. Panza, and D.C. Struppa. 2017. Forcing Optimality and Brandt's Principle. In J. Lenhard and M. Carrier (ed.), *Mathematics as a Tool*. Boston Studies in the Philosophy and History of Science 327, Springer.
https://doi.org/10.1007/978-3-319-54469-4_13

D. Napoletani, E. Petricoin, D. C. Struppa. 2012. Geometric Path Integrals. A Language for Multiscale Biology and Systems Robustness. In *The Mathematical Legacy of Leon Ehrenpreis*. Springer Proceedings in Mathematics, 16, 247-260.
https://doi.org/10.1007/978-88-470-1947-8_16

D. Napoletani, M. Signore, T. Sauer, L. Liotta, E. Petricoin. 2011. Homologous Control of Protein Signaling Networks. *Journal of Theoretical Biology*, 279(1), 21.
https://doi.org/10.1016/j.jtbi.2011.03.020

L. Page, S. Brin, R. Motwani, & T. Winograd, The PageRank citation ranking: bringing order in the Web. Manuscript to be found at http://ilpubs.stanford.edu:8090/422/1/1999-66.pdf.

M. Panza. 1995. De la nature épargnante aux forces généreuses. Le principe de moindre action entre mathématiques et métaphysique : Maupertuis et Euler (1740-1751). *Revue*

d'Histoire des sciences, 48, 435-520.
https://doi.org/10.3406/rhs.1995.1240

M. Panza. 2003. The Origins of Analytical Mechanics in 18th century. In H. N. Jahnke (ed.) *A History of Analysis, American Mathematical Society and London Mathematical Society*, s.l., 137-153.

J. Ramsay, B. W. Silverman. 2005. Functional Data Analysis. 2nd edition, Springer.
https://doi.org/10.1007/b98888

Q. Rao and J. Frtunikj. 2018. Deep Learning for Self-Driving Cars: Chances and Challenges. *2018 IEEE/ACM 1st International Workshop on Software Engineering for AI in Autonomous Systems (SEFAIAS),* 35-38.
https://doi.org/10.1145/3194085.3194087

F. Schroff, D. Kalenichenko, J, Philbin. 2015. FaceNet: A Unified Embedding for Face Recognition and Clustering. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 815-823.
https://doi.org/10.1109/cvpr.2015.7298682

L. S. Schulman. 2005. *Techniques and Applications of Path Integration*. New York: Dover.
https://doi.org/10.1063/1.2914703

D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel and D. Hassabis. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529, 484-489.
https://doi.org/10.1038/nature16961

Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, L. Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, J. Dean. 2016. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. Preprint available at https://arxiv.org/abs/1609.08144, 2016.

HISTORIQUE
Article initialement soumis le 26 novembre 2018.
Article accepté le 23 juin 2020.

SITE WEB DE LA REVUE
ojs.uclouvain.be/index.php/latosensu
ISSN 2295-8029
DOI http://dx.doi.org/10.20416/LSRSPS.V8I2.5

SOCIÉTÉ DE PHILOSOPHIE DES SCIENCES (SPS)
École normale supérieure
45, rue d'Ulm
75005 Paris
www.sps-philoscience.org

CONTACT ET COORDONNÉES :

Domenico Napoletani
University Honors Program and Institute for Quantum Studies, Chapman University, Orange (CA)

Marco Panza
CNRS, IHPST (CNRS and Univ. of Paris 1, Panthéon-Sorbonne) & Chapman University, Orange (CA)

Daniele Struppa
The Donald Bren Presidential Chair in Mathematics, Chapman University, Orange (CA)