Original Research Article

BIG DATA & SOCIETY

Big Data & Society
July–December 2016: 1–17

DOI: 10.1177/2053951716664747
bds.sagepub.com

SAGE

Can we trust Big Data? Applying philosophy of science to software

John Symons and Ramón Alvarado

Abstract

We address some of the epistemological challenges highlighted by the Critical Data Studies literature by reference to some of the key debates in the philosophy of science concerning computational modeling and simulation. We provide a brief overview of these debates focusing particularly on what Paul Humphreys calls epistemic opacity. We argue that debates in Critical Data Studies and philosophy of science have neglected the problem of error management and error detection. This is an especially important feature of the epistemology of Big Data. In "Error" section we explain the main characteristics of error detection and correction along with the relationship between error and path complexity in software. In this section we provide an overview of conventional statistical methods for error detection and review their limitations when faced with the high degree of conditionality inherent to modern software systems.

Keywords

Big Data, epistemology, software, complexity, error, Critical Data Studies

Introduction

The surveillance and manipulation of individuals and populations through computing technologies for commercial or policy purposes raises a range of difficult philosophical questions. While the most pressing challenges have an obvious ethical and political component, we need to understand what levels of control and insight so-called Big Data allows before we can make informed decisions concerning its moral status. Thus, in the paper we argue for a careful assessment of the epistemic status of the computational methods that are currently in use. These technologies are deployed in pursuit of particular pragmatic ends in the service of corporate and political missions. The actions of corporations and political entities can be evaluated independently of the technology that they deploy. However, the extent to which users of Big Data can accomplish their goals depends on the epistemic status of those technologies.[1] In many contexts, moral and epistemic questions are inextricably intertwined, and our goal here is to help lay the necessary groundwork for moral and political engagement with Big Data by understanding as clearly as possible how the appearance of Big Data has changed the epistemic landscape over the past two decades. What can Big Data technologies allow users to know, what are the limits of these technologies, and in what sense is Big Data a genuinely new phenomenon? Answering these questions is essential for guiding our moral and political responses to Big Data.

Popular literature on Big Data is often dismissive of philosophy of science and epistemology. Popular authors and journalists frequently suggest that the rise of Big Data has made reflection on topics like causation, evidence, belief revision, and other theoretical notions irrelevant. On this view, the turn towards Big Data is a turn away from concern with a range of traditional questions in the philosophy of science.[2] Big Data, according to some, "represents a move away from always trying to understand the deeper reasons behind how the world works to simply learning about an association among phenomena and using that to get things done." (Cukier and Mayer-Schoenberger, 2013: 32)

Department of Philosophy, Lawrence, KS, USA

Corresponding author:
John Symons, Department of Philosophy, University of Kansas, 1445 Jayhawk Blvd., Wescoe Hall, Room 3090, Lawrence, KS 66045-7590, USA.
Email: johnsymons@ku.edu

This atheoretical turn makes the false assumption that more data means better inquiry. Worse than merely being a superficial view of knowledge and inquiry, the atheoretical stance is blithely uncritical towards the corporations and governments that use technology to "get things done".[3] The assumptions governing the atheoretical turn are false and, as we shall see, studying Big Data without taking contemporary philosophy of science into account is unwise (Frické, 2015). Some of the limitations and risks involved in the use of computational methods in public policy, commercial, and scientific contexts only become evident once we understand the ways in which these methods are fallible. Thus, in the broader social and political context, a precondition for understanding the potential abuses that can result from the deployment of Big Data techniques by powerful institutions is a careful account of the epistemic limits of computational methods. A clear sense for the nature of error in these systems is essential before we can decide how much trust we should grant them and what, if any, limits to their use we should impose.[4]

Coming to understand error and trust in these contexts involves a range of philosophical and social-scientific questions. No single scholarly or scientific discipline has the resources to respond to the questions and challenges posed by the rise of Big Data. Critical Data Studies is the interdisciplinary field that has begun to consolidate around the task of engaging with these questions. Critical Data Studies has, understandably, focused on the important political and social dimensions of Big Data. However, this work urgently requires attention to the assumptions governing the use of software in the manipulation of data and in the conduct of inquiry more generally.

We will argue that critical attention to the formal features of software is important if we are to get a proper understanding of the relationship between Big Data and reliable inquiry. We are friendly critics of existing work in Critical Data Studies: Our contention is that the field has neglected highly relevant recent work in philosophy of science. Critical Data Studies has correctly recognized that the technology underlying Big Data has changed the epistemic landscape in important ways, but has been unclear with respect to what these changes have been (Kitchin, 2014). Many of these changes have taken place with the advent of computational methodology in general, but more specifically with the integration of computer simulations into the toolkit of ordinary scientific practice. Thus, part of our purpose is to connect debates in philosophy of science concerning the status of computational models, simulations, and methods with the emerging field of Critical Data Studies. To this end, we explain the role of epistemic opacity in computational modeling and close with an example of a basic epistemological challenge associated with any software intensive practice, the problem of determining error distribution. Another feature of software intensive science (SIS) that philosophers have highlighted in recent years is the effect that errors in code can have for the reliability of systems. Horner and Symons (2014a), for example, explained the role of software error in scientific contexts. Although primarily epistemic in nature, such considerations have direct implications for policy, law, and ethics.

As several authors have noted, the term 'Big Data' does not refer strictly to size but rather to a range of computational methods used to group and analyze data sets (Arbesman, 2013; Boyd and Crawford, 2012). Thus one cannot responsibly address the epistemic status of 'Big Data' without understanding the implications of the use of software for inquiry. We are not arguing that philosophers of science have simply solved all the epistemic problems related to Big Data. In fact, given the central role of software in Big Data projects, traditional accounts of epistemic reliability drawn from philosophy of science are likely to prove inadequate for reasons we explain below.

For some philosophers, the increasingly dominant role of computational methods is not a matter of significant philosophical interest. On this view, there are no novel, philosophically relevant problems associated with the increased use of computational methods in inquiry (Frigg and Reiss, 2009). Others, like Eric Winsberg (2010) and Paul Humphreys (2009) have defended the view that computational modeling and simulation are associated with distinctive and novel strategies for inquiry. Another recent line of inquiry that has direct bearing on Big Data involves the problem of tackling error in large software systems. The effect that increasing software dependency has wrought with respect to the trustworthiness of scientific investigation carries over directly to Big Data. Big Data is part of a changed landscape of problems associated with the use of computational methods in scientific inquiry. While the term 'Big Data' rarely figures in the work of philosophers of science, there is now a large literature that discusses the role of software in science, particularly insofar as it relates to modeling and simulation (see for example Frigg and Reiss, 2009; Humphreys, 2009; Morrison, 2015; Winsberg, 2010). Symons and Horner have pointed, for example, to what they call the path complexity catastrophe in SIS (see 2014; Horner and Symons, 2014; Symons and Horner, forthcoming). In this paper, we will argue that the path complexity catastrophe will have consequences for Big Data projects. We will explain why Big Data, as a paradigmatic instance of SIS is especially vulnerable to intractably difficult problems associated with error in large software systems.

In ''Introduction'' section we introduce the many attempts to define Big Data and explain their limitations. This section has multiple aims. We begin by providing an overview of Big Data as currently practiced. Given the diverse uses of the term 'Big Data', in this section we stipulate a working definition that is precise enough for our purposes and that faithfully reflects the main features of current usage. The second aim of ''Introduction'' section is to show the unavoidable connection between the methods used in Big Data and the software dependence mentioned above. We conclude that Big Data is an example of what Horner and Symons call Software-Intensive Science. As such, Big Data epitomizes the kind of inquiry to which philosophical debates concerning the role of computers in science should apply.

In ''Big Data meets Critical Data Studies'' section, we do several things. First we provide an overview of recent criticisms of Big Data that originate from the Critical Data Studies literature. We provide reasons to think that although they may be important to the overall characterization of Big Data, the tools deployed by this interdisciplinary field of study are excessively anthropocentric and social in their orientation and are the product of debates in philosophy of science and social epistemology that have been largely superseded by the developments in recent decades. Notably, since they are generally related to science as a whole, the insights that derive from socially and historically oriented scholarship from the 1960s to 1980s shed relatively little new light on the use of software in scientific, corporate, and policy settings.

The best way to address some of the epistemological worries highlighted by the Critical Data Studies literature is to attend to debates in the philosophy of science concerning computational modeling and simulation. We provide a brief overview of the principal debates in ''The epistemic status of Big Data'' section. In particular, we focus on issues that relate to what Paul Humphreys (2009) calls epistemic opacity. ''The epistemic status of Big Data'' section concludes by noting that the existing debate in both Critical Data Studies and philosophy of science has neglected the issue of error management and error detection. This is an especially important feature of the epistemology of Big Data. In ''Error'' section we explain the main characteristics of error detection and correction along with the relationship between error and path complexity in software. In this section we provide an overview of conventional statistical methods for error detection and review their limitations when faced with the high degree of conditionality inherent to software systems used in Big Data. And finally, in ''Example'' section we offer an overview of the limitations exhibited by Google's Google Flu Trends (GFT). In particular, we focus on the ambiguity concerning the sources of such limitations. These limitations, we argue, exemplify the deficiencies of an atheoretical approach but most importantly they also clearly characterize the intrinsic epistemic challenges posed by large software systems conventional methods of error detection, correction, and general assessment.

What is Big Data?

The term 'Big Data' arose in the context of challenges facing engineers dealing with large data sets and limited computational resources. For example, as noted by Gill Press (2013), Cox and Ellsworth (1997) introduce the term ''Big Data'' in their discussion of challenges involving the limitations due to memory storage constraints and processing speed for data visualization at the NASA Ames Research Center. That paper focused on data sets that exceeded only 100 Gbytes. Attempts to partition those data yielded segments that were too large for any researcher to work with given the tools and techniques of the time. Specifically, desktop computers available to individual NASA engineers in the mid-1990s faced memory and processing constraints that limited their capacity to make good use of the data at their disposal. Cox and Ellsworth (1997) call this ''the problem of Big Data''. Contemporary usage of the term 'Big Data' differs in significant ways from this original context. It is common today for everyday data storage applications to reliably exceed 100 Gbytes. While there are significant technical challenges involved in managing large amounts of data, ''the problem of Big Data'' as characterized in the 1990s is not the pressing concern it once was.

Most, if not all, early definitions focused on resource constraints and data set size. This is not the case today. In fact, as Boyd and Crawford (2012) note, many data sets considered to be paradigmatic in the Big Data literature today are smaller than those used to coin the term. They cite, for example, the small size of the data sets involved in analyzing Twitter trends when compared to low-tech research into often very large-scale data sets generated by the US Census Bureau from the Nineteenth Century. So, although 'Big Data' connotes the use of large data sets, size is not an essential feature of current usage of the term.[5]

Other definitions (e.g. Chen et al., 2014) focus on the way the different elements of a data set relate and interact. In some cases this is described in terms of the dynamic interaction of the 3V's: velocity, variety, and volume. Whether a set is deemed to be a Big Data set has to do with the dynamical constraints of these three factors. Volume is of course size, but variety and velocity are less easy to define. Variety, for example has to do with the kind of data in the sets (i.e. pixels vs. nodes)

while velocity has to do with the physical and temporal resources required to economically process a set. Whether these three factors are sufficient to define Big Data is a topic of ongoing discussion. Some cite the extra V's of veracity, value, and visualization as necessary components of a working definition. However, regardless of the number of V's one includes, all of the definitions agree that analytical tools and methods are a core component of the definition of Big Data (Chen et al., 2014).

A working definition

In this paper we adopt what we think is the most faithful definition of what Big Data means in contemporary practice. Here, we follow the analysis provided by Chen et al. (2014) about the uses of the term in commercial contexts. They review the range of definitions of Big Data given by leading corporations in data management (for example, International Data Corporation (IDC), IBM, and Microsoft) before settling on IDC's 2011 definition. They preface their choice of definition by stating that ''Big Data is not a 'thing' but instead a dynamic/activity that crosses many IT borders.'' They cite an IDC report from 2011 defining Big Data as follows:

> ''Big Data technologies describe a new generation of technologies and architectures, designed to economically extract value from very large volumes of a wide variety of data, by enabling high-velocity capture, discovery, and/or analysis.'' (Gantz and Reinsel, 2011 as quoted in Chen et al., 2014)

This definition serves to highlight the most important and distinctive characteristics of Big Data, namely its use of statistical methods and computational tools of analysis. It will be particularly important to consider this definition in detail in ''The epistemic status of Big Data'' section. It is in this section that the epistemic status of Big Data is discussed and in which the case is made that Big Data, insofar as it is an intrinsically computer-based method of analysis deployed in inquiry, is a SIS par excellence. Thus, this definition is particularly apt since it clearly captures the interplay between the epistemic, normative, and economic dimensions of Big Data. Most importantly, this definition will highlight the limitations concerning error assessment characterized in ''Error'' section.

Big Data meets Critical Data Studies

This section presents and clarifies what we take to be some of the most significant critical studies of Big Data.[6] Although we agree with many of the observations made in the existing literature, we think that the critical scholarship to date has fallen short of addressing the distinctive epistemic features of Big Data. In part, this is because most criticisms are focused on the social level of analysis rather than on any distinctive features of the technology of Big Data per se. That is to say, the focus has been on limitations due to human-centered interactions such as inescapable cognitive and social biases and the overall value-ladenness of human inquiry. The basic conceptual point made in the field of Big Data studies is that data must be interpreted and that interpretation is subject to human bias. We agree that the processes by which data is selected and interpreted are important topics of study. However, they are not unique to Big Data. Thus, in this section the development of Critical Data Studies will be connected to its focus on the distinctive characteristics of Big Data rather than on considerations that could be addressed to human inquiry in general. In this spirit, and for the purpose of this paper, we focus on the analysis of error, error distribution assessment, testing and reliability, as they relate to the computational methods employed by Big Data.

Error is an epistemic concept and the treatment of epistemic questions arising from Big Data is in its early stage. In a recent article in this journal, for example, Rob Kitchin (2014) argues that there are three main types of account concerning the epistemic implications of Big Data. He contends that these derive from differing general perspectives on the nature of science held by scholars investigating Big Data. The three perspectives he identifies are the paradigmatic, the empirical, and the data-driven. Big Data theorists who follow a paradigmatic—or Kuhnian model—of scientific inquiry suggest that science normally functions within settled patterns and only occasionally advances via radical shifts in methodology. Advocates of this view contend that the advent of Big Data constitutes a paradigm shift of the sort described by Kuhn (Kitchin, 2014). That is, that Big Data has indeed revolutionized not only the methods by which we conduct science but also the goals of scientific inquiry per se. The second camp is that of the empiricist.[7] The motto of this camp is ''the death of theory'' (see Anderson, 2008; Cukier and Mayer-Schoenberger, 2013; Steadman, 2013). They regard the advent of Big Data and its capacity to detect patterns as replacing theoretical analysis with unrestricted sampling. On this view, raw data and correlation patterns are sufficient for scientific development. In this camp, terms such as causation, paradigmatic of scientific inquiry for centuries past, even in their conventional use in science, are regarded as being elusive and possibly even occult. The third camp, the data-driven one, is a hybrid of sorts in that ''it seeks to

generate hypotheses and insights 'born from the data' rather than 'born from theory'" (Kelling et al., 2009 as cited by Kitchin, 2014). According to Kitchin (2014), a data-driven science is one whose epistemological strategy is to use "guided knowledge techniques to identify potential questions (hypotheses) worthy of further examination and testing." (Kitchin, 2014) This last camp recognizes a role for conventional scientific terms and methods beyond mere pattern recognition, but its hypotheses are derived from the data itself and not "*just*" from guiding theoretical principles.

Kitchin (2014) criticizes the first two camps, focusing primarily on claims made by those advocating the end of theory.[8] According to Kitchin, the so-called empiricists have four main claims concerning the scope, reach, and assumptions of Big Data:[9]

1. full resolution ($N = \text{All}$),
2. no a priori (theory/model/hypothesis) needed,
3. agnostic data, and
4. domain transcendence (the assumption that unrestricted pattern recognition does away with scientific specialization).

Given that many problems involving Big Data techniques are of a dynamic nature, in real time and involving changing demarcations and inputs, the $N = \text{All}$ option is off the table[10] (Bollier and Firestone, 2010). That is to say, in a constantly dynamic landscape, like the ones often involved in Big Data problems, one can never be said to have all the data. However, for Kitchin the problem lies elsewhere. He thinks that the problem has rather to do with sampling bias that originates in the technology deployed, the collection methods, and the data ontology employed in the process. In other words, the problems with one above have to do with subjective limitations and biases of the agents conducting the inquiry. This argumentative strategy is not unique to Kitchin. It can be found in other widely cited authors in the Critical Data Studies literature (see for e.g. Boyd and Crawford, 2012) for whom the nature of the problems themselves (i.e. dynamic, real-time problem solving) is not recognized as a constraint on the quest for full resolution. Instead, they argue that constraints are due to the subjectivity inherent in the choice of discretization and the highly value-laden social aspects of inquiry that inevitably come into play.

Similarly, 'empiricist' assumptions 2 and 3 are rejected by Kitchin on the grounds that whatever methods allow us to collect and analyze data are already theory/model-laden to begin with. He explains that "data are created within a complex assemblage that actively shapes its constitution" and that ultimately, identifying patterns in data "does not occur in a scientific vacuum" and is "discursively framed" by theories, practitioners, and legacy methodology alike (Kitchin, 2014).

As mentioned above, other Critical Data studies' authors provide similar criticisms of Big Data. Take Boyd and Crawford (2012), for example. In their article (2012) they address the "death of theory" camp, or 'empiricists', by questioning their implicit claims to objectivity. They attack these claims because, according to them, they are "necessarily made by subjects and are based on subjective observations and choices." (Boyd and Crawford, 2012) They also criticize assumptions 1 and 2 by pointing that massive amounts of raw data are meaningless unless a question is posed, an experiment standardized and a sample curated (2012). All of which are subjective endeavors. This is an insight drawn from historically and socially oriented philosophy of science. Kuhn's work (1962) has been especially influential here, along with the critical work of philosophers like Longino (1990) and others.

While Kuhn, Longino, and other mid-to-late 20th century philosophers have helped shape the contributions of many in the Critical Data Studies community, the project of understanding Big Data can benefit from taking advantage of additional philosophical resources. Acknowledging that human bias influences inquiry is a reasonable, but relatively trivial philosophical observation.[11] Since it is applicable to all forms of inquiry at all levels (Longino, 1990), the recognition of bias is not a contribution that adds anything distinctive to the study of Big Data.[12] This is particularly the case considering the developments computer technologies deployed in the aid of science have undergone precisely in the last 70 years. Unfortunately, the influence of relativistic philosophy of science has impeded the development of analyses of the epistemic questions that arise in the context of Big Data.

Similarly, the emerging field of Software Studies, which attempts to develop critical perspectives on the development and use of software, often relies on philosophical literature that although interesting in its own right, is orthogonal to the core questions that arise from the use of software. This is particularly problematic since some in the field of Software Studies want to argue that the use of computational methods, in particular their capacity to deal with immense data sets in science and policy-making, does in fact bring about novel issues to explore (see Amoore, 2011, 2014; Berry, 2011). Take the following example. In his book *The Philosophy of Software*, David Berry (2011) defines software studies as a research field that includes disciplines as broad as *platform studies*, *media archaeology*, and *media theory*, all of which focus on the development, use, and historicity of hardware, operating systems, and even gaming devices (Berry, 2011). Berry argues that these technologies not only offer novel

insight into the human experience, but that they *are* also a novel part of it. However, the philosophical resources that he applies to these issues are restricted to authors like Kuhn and Heidegger. While these are deeply significant figures in the history of philosophy, they offer limited insight into the novel epistemic features of computational methods such as Big Data.

Consider another prominent figure in software studies: Louise Amoore. She addresses security risks in ways that are relevant to the discussion of Big Data. She argues that modern security risks' calculations can be understood by analogy with financial derivatives (Amoore, 2011). She offers an analysis of the implications of Big Data in risk assessment in the context of border security policy (Amoore, 2011). On her view, risk posed by individuals can be understood as a product of correlational patterns that derive from assorted data sets that include origin and destination of travel, meal choice, etc. Security risk, according to her, is construed as an emergent phenomenon, not reducible and frequently not directly related to the components from which it arises. Financial derivatives, she argues arise in the same manner (Amoore, 2011). What she means here is that derivatives are not mere aggregations of fluctuation in market stocks or patterns in debt, but are instead a financial instrument in their own right. Because of the fragmentation and manipulation of values derived from more conventional financial instruments, derivatives manage to have novel financial properties that are specific to them. According to her, the same can be said about risk assessment of individuals crossing borders that emerge from risk-based security calculations in contemporary security practice. The risk travelers pose, although derivative of certain specific choices and information about an individual, is often an independent feature that is not found in any of these choices and informational sets but as a product of an emergent whole. Although Amoore is indeed talking about the inherent features of Big Data systems here, like those involved in border-crossing security systems, we find that she relies heavily on an anthropocentric treatment of risk that focuses on policy and decision-making rather than on the distinctive features of those systems.[13] Big Data systems also involve risks that are due not only to the effects of design or policy choices, but also from the nature of the software systems themselves. While Amoore correctly points to the emergent features of large complex systems as important areas of inquiry, we think that the most important epistemic problems facing them are due to the characteristic features of software systems themselves and not mere contingent limitations on the part of agents.

Insofar as Critical Data Studies understands itself to be addressing a distinctive area of research, scholars in this field ought to recognize that Big Data, at its heart, involves the use of computational methods. The two principal areas of philosophical inquiry that have been missing from Critical Data Studies to date are contemporary philosophy of science and philosophy of computer science. Connecting these debates to philosophy of computer science is beyond the scope of the present paper.[14] Instead, for the remainder of this paper, we will demonstrate the relevance of more recent and growing literature on software, models, and simulations, in the philosophy of science to questions of reliability and error in Big Data.

The epistemic status of Big Data

The most distinctive aspect of Big Data, as we argued above, is the prominence of computational methods and in particular the central role played by software. What are the novel epistemic challenges brought about the use of computational methods? Although there is a broad debate in philosophical literature about the epistemic implications of the 'introduction of computers' into scientific inquiry[15] (see Barberousse et al., 2009; Frigg and Reiss, 2009; Humphreys, 2009; Winsberg, 2010), it is important to recognize, following the work of Evelyn Keller (2003) that this introduction took place gradually in a series of distinguishable stages from the end of the Second World War until relatively recently. Evelyn Keller (2003) argues that just as the introduction of computers was itself a gradual process that posed distinct challenges in distinct disciplines for different reasons, the epistemic challenges emerged in different disciplines at different times and at different stages of technological innovation.

Fox-Keller identifies three main stages. The first begins with the use of computers to overcome the problem of mathematically intractable equations in the context of research at Los Alamos in the years immediately following the Second World War.[16] This stage represents an important deviation from conventional analytical tools of the sciences at the time because it directly challenges the well-established use of differential equations as the main tool in the physical sciences (Keller, 2003). However, when computers were being used at this stage the primary concern was still to 'simulate' conventional differential equations and their probable solutions using Monte Carlo methods (Metropolis and Ulam, 1949). In this respect the Monte Carlo methods are directed towards the solution of equations and are removed in one step from the phenomena described by those equations. In other words, methods such as the Monte Carlo method were not deployed to simulate any system, but rather to provide a wide range of possible solutions to differential equations later deployed in order to understand a given system. With time, statistical approaches to problem solving (like Monte

Carlo) offered a practical alternative to the differential equations themselves (Keller, 2003).[17]

The second stage, according to Fox-Keller, has to do with the use of dynamic models as representations of a target system, or "approximate analogous systems" (Frigg and Reiss, 2009). That is to say, the use of computerized calculations was confined "to follow the dynamics of systems of idealized particles" (Keller, 2003). In this stage, scientists were no longer merely simulating possible solutions to differential equations but rather working under an assumed isomorphism between the observed behavior of a phenomenon and the dynamics expressed by the artificial system, or computer model, constructed to track its idealized development. In other words, the aim was to simulate "an idealized version of the physical system." (2003) Fox-Keller identifies two levels to the use of simulations in this second stage: (1) substitution of the natural for the artificial system, and (2) replacement of the differential equations at the first level for discrete, "computationally manageable", processes.[18] This second stage already posed a challenge to the conventional epistemic relation between theory construction and modeling. That is, while the mathematical formulations of the differential equations had strong and direct ties to theoretical principles to back them up, the discretized versions were now merely approximations without a direct link to the underlying theory (Winsberg, 2010). Nevertheless, what these simulations attempted to represent were entire theories and some would say that it is only in this second sense that the proper use of the term 'simulation' in its current usage enters the computational terminology (Hugues, 1999, as cited by Keller, 2003).[19]

Finally, the third stage, according to Fox-Keller, is a reliance on the analysis and model-building of particular and localized systems rather than generalized theoretical ones. Foregoing the wide scope of a full theoretical framework, this approach focused on the modeling of internally consistent mechanisms without generalizable principles or wide ranging laws at their core. As Keller (2003) notes, this change has important implications for scientific explanation (see also Symons, 2008).[20] This third stage, according to Keller, departs from the first two in that it "is employed to model phenomena which lack a theoretical underpinning in any sense of the term familiar to physicist." (2003)[21]

Big Data falls somewhere between first and second stage of Fox-Keller's taxonomy. Big Data, we will argue, is a software intensive enterprise that is focused on revealing patterns that can be used for commercial, political, or scientific purposes.[22] Unlike the third stage applications of computational models that Fox-Keller describes, applications of Big Data are intended to reveal features of natural or social systems. Big Data projects are generally not detached from specific practical applications, nor do they involve testing or demonstrating new theoretical frameworks.[23] Big Data is a relatively conservative and pragmatically motivated application of computational techniques, especially when compared with examples of the third part of Fox-Keller's taxonomy.

What is meant by calling Big Data *software intensive* is relatively straightforward. Computer scientists call a system software intensive if "software contributes essential influences to the design, construction, deployment, and evolution of the system as a whole." (IEEE, 2000) Given this definition, by almost any standard, Big Data, like much of contemporary science, is software intensive.[24]

One aspect of the heavy reliance on software by scientific or commercial enterprises is to say that the kinds of insights available via computational methods would not be available without the use of software. Embedded in many of the definitions of Big Data is the assumption that even just given the vast amount of information involved, no equation worked by paper and pencil could in practice be deployed to deal with it (Bryant et al., 2008). In other words, Big Data deals with problems where insights would be practically impossible without the help of computers.

Big Data can also address problems involving complex systems where the relevant dynamics are not obviously accessible except through surveying vast amounts of data (see Symons and Boschetti, 2013). In addition to those problems which would simply require raw computing power beyond our innate capacities there are also analytically intractable problems that require simulation by computer rather than admitting of analytic solutions.[25] Big Data is generally *not* deployed because the problems in question are analytically intractable. However, as we shall see below, computational models of the kind that are central to Big Data are of great interest precisely because they promise new ways to explore phenomena that are difficult to examine by other means (Barberousse and Vorms, 2014; Boschetti et al., 2012). As Symons and Boschetti (2013) note, computational models are currently allowing research into topics where cognitive, ethical, political, or practical barriers would otherwise loom large. Whether in nuclear weapons testing, climate science, studies of the behavior of epidemics, or studies of the internal dynamics of stars, to take just a handful of cases, computational models are often the *only* viable research tool for scientists (2012: 809). Similarly, applications of Big Data science to epidemics, energy usage, social movements, and the like all have the property of generating results that are otherwise inaccessible (at least within any practical timescales and resource constraints) without the use of software.

Another way of thinking about the intrinsic reliance of Big Data on software is to focus not only on its methods but also on the nature of its results. These results mainly involve pattern discovery. We analyze a set of granular data points in order to detect relational structures. Consider twitter trends. Millions of short texts are mined to find concurrent terms or combinations thereof. These are in turn correlated to other factors related to the authors, i.e. gender, geographic location, etc. Patterns emerge. But the way we arrive at such patterns is through the statistical analysis of correlated data points. Whether these results are conveyed via visualizations or mathematical formulas they are the result of very large numbers of computations. As discussed above, even just considering the number of available data points, these methods are computational and they are so as a matter of practical necessity.

Consider attempting to understand what is going on inside a star like our Sun. We can know facts about the center of the Sun. We have indirect means of learning about chemical composition through spectral analysis and the like, but other than that, the only ways to draw inferences about the processes taking place under the surface of the sun are those made available to us via computational models. This applies almost by definition (Gantz and Reinsel, 2011) to phenomena considered paradigmatic in the Big Data literature. This is because many of the insights brought about with Big Data techniques would otherwise be unavailable, or simply neglected by other analytical methods. Thus Big Data science is unavoidably software dependent.

In addition to being an intrinsically computational method, the value of Big Data derives from the patterns it extracts and the correlations revealed thereby. However, this means two things. First, tied to the notion of pattern recognition and correlating millions of bits of data *comes* the need to visualize them. Such patterns and their insights would be of no use if they were presented to us solely via a spreadsheet and a mathematical function for example. As we discussed above the term Big Data was coined because of the challenging constraints of memory and processing power but more particularly as they relate to visualization.[26] Beyond the challenge of static visualization, many problems in Big Data involve real time inputs and processing and as such we can say that Big Data does not just create a static representations but rather creates artifacts that are more akin to scientific simulations. This is the case for example in the case of Numerical Weather Prediction systems which not only process past data to predict future occurrences but also compare the model's output to real time sensors tracking the weather (Bauer et al., 2015). It is in this sense that the model ceases to be merely an explanatory representation and becomes a simulation (Weisberg, 2013) whose key insights derive from the dynamic nature of the visualization (Bollier and Firestone, 2010).[27]

Epistemic opacity

Among the most challenging philosophical problems facing Big Data as a SIS is assessing its role in the process of creating, gathering, and uncovering new insights and knowledge. The scientific status of Big Data is a topic of ongoing debate. Lazer et al. (2014) have argued that most prominent applications of Big Data are not properly scientific insofar as the sources of data are unreliable. Specifically, they argue, the data that serve as the basis for Big Data projects are not derived from scientific instruments (Lazer et al., 2014).

By contrast, philosophers of science have debated whether computer-based methods generate models that are closer to theoretical abstractions or to empirical experiments (Barberousse and Vorms, 2014; Morrison, 2015).[28] Addressing the epistemic challenges of computational methods in science Paul Humphreys (2009) argues that the central problem is the mediation of our epistemic access to the phenomena of interest. This is because computational methods can involve an ineliminable "epistemic opacity" (Barberousse and Vorms, 2014; Humphreys, 2009), which Humphreys defines in the following way:

> "A process is epistemically opaque relative to a cognitive agent X at time t just in case X does not know at *t* all of the epistemically relevant elements of the process." (Humphreys, 2009)

Epistemic opacity, understood in this sense is not a new feature of scientific inquiry, nor is it unique to computational methods. Humphreys recognizes that a parallel issue arose with the emergence of Big Science, i.e. when scientific inquiry became an ineliminably social endeavor in which no individual was in control of the complete process of inquiry (Humphreys, 2009; Longino, 1990). However, Humphreys regards the computational turn in science as generating a qualitatively different form of epistemic opacity. Some of the problems stem from lower level operational issues such as the semantics of computational processes. In a relatively obvious sense, human-level computer languages are already highly mediated with respect to machine-level implementation. This results simply from being compiled through several syntactic layers in order for code to be accessible to human programmers. Another example at a higher level are unavoidable numerical discretization choices that enable higher-order representational features such as visualizations (Humphreys, 2009). According to Humphreys, both features of

computational techniques represent novel instances of epistemic opacity.

One concrete example of how the social nature of software contributes to epistemic opacity in a novel way is the effect of so-called "legacy code". This is programming code that has been built by engineers either using programming languages that have fallen out of favor or that for some other reason may be difficult for later programmers to understand. Coding is a highly creative engineering task and although the code may do its job appropriately there may, occasionally be no way for contemporary users to know exactly how it achieves its function (2009). As a matter of fact, legacy code is common in computer science. One could argue that certain analogous legacy methods or processes are part of traditional big-science projects. However, unlike say a scientific instrument whose inner workings are well-understood, it may not be evident how some piece of legacy software contributes to the functional role of the whole piece of software. One could easily imagine being able to reverse-engineer the functionality of non-software aspects of a scientific project if one knew its function. However, it is not always the case that one can understand the function of legacy code in some large system.

When dealing with legacy code it may prove easier and more viable to merely work around the already functioning code, even if no one actually understands it.[29] In big, ongoing projects it is often economically unfeasible to discard the legacy code and begin from scratch (Holzmann, 2015). This is particularly the case with critically important systems whose operation cannot be interrupted like flight control software. In such cases the system must be kept running as it is being patched or updated.

There are other distinctive sources of epistemic opacity resulting from the use of computational methods that have no parallel in other aspects of conventional scientific inquiry. Consider weak emergence. Weak emergence is characterized by the emergence of unintended/non-programmed/unexpected behavioral patterns in running simulations (Humphreys, 2009). Patterns that were not known before the simulation was turned on and ran (for more on this see Symons, 2002, 2008). Weakly emergent phenomena are characterized, among other things, by their dependence on the actual running of a simulation. That is to say, there would be no way of having found those patterns apart from running the simulation itself. They are the product of the actual dynamics of the simulation and cannot be deduced from nor reduced to any of the elements that conform it (Bedau, 1997).

Reliance on computational methods involves a distinct kind of epistemic opacity from the social epistemology aspects of human-centered inquiry where the central issue is the bias and subjectivity inherent in interpretation. The consequences of this epistemic opacity are not easily solved through some simple fix or revision of appropriate methods to deal with them. Computational methods, as Humphreys argues are "*essentially* epistemically opaque" (Humphreys, 2009). A process is *essentially* opaque in this way to an agent at "if it is *impossible*, given the nature of X, for X to know all of the epistemically relevant elements of the process" (Humphreys, 2009).

This last formulation of epistemic opacity serves to elucidate the kinds of epistemic challenges at play in our discussion, namely those that are features of the systems in questions and not merely contingent limitations of individual researchers or of teams of researchers. As such, it also serves to distinguish the general concept of epistemic opacity from a related issue concerning the concept of black boxes in systems analysis.[30] Black box theory is in principle a mathematical approach that allows for schematization of non-linear functions between an input and a result without the need to know exactly what the internal structure of the function is or without particular regard to the nature of the input or results (Bunge, 1963). It was later adopted by emerging fields in the study of complex systems (Ethiraj and Levinthal, 2004) and business related issues concerning organizational structures and product design (Brusoni and Prencipe, 2001). Although black boxes and epistemic opacity are related in that both are issues concerning gaps in knowledge of a given system, they are very different concepts. In particular, black box theory is more of a pragmatic approach to an information system that can function in a need-to-know basis. That is, it is an attempt to schematize in a formal manner an information system with the minimum amount of information possible being transmitted from one state to the next and to do so despite possible limitations. Epistemic opacity on the other hand is concerned with more than just the pragmatic constraints associated with specific methods or technologies. It is about the nature of knowledge per se and in particular about the ways in which knowledge can be conveyed or can fail to be so. Black boxes are just one of the many instances of epistemic opacity. In other words, all black box problems are instances of epistemic opacity but not every instance of epistemic opacity is a black box. But more importantly, not all black boxes are instances of *essentially* opaque processes.

Error

Humphreys' argument that computational methods suffer from epistemic opacity is strengthened when we consider the role of software error (see also

Barberousse and Vorms, 2014; Floridi et al., 2015; Newman, 2015).[31] In this section, we examine the role of error in software intensive systems and explain why traditional approaches to handling error in a scientific context fall short. As briefly stated above, by error we simply mean the many ways in which a software system may fail. This may include erroneous calculations, implementations, results, etc. The important point here is not error per se but our epistemic relation to it in the context of inquiry.

Scientific claims are often—if not always—of a statistical nature (Mayo and Spanos, 2010). Increasingly sophisticated manipulation, interpretation, and accumulation of data have made the probabilistic aspect of scientific claims become more pressing (see Keller, 2003; Metropolis and Ulman, 1949). In light of the statistical nature of contemporary science Deborah Mayo has called for a new philosophy of statistical science in order to account for error and probability inherent in modern scientific inquiry (Mayo and Spanos, 2010). Mayo proposes what she calls 'severe testing'. A method by which a given hypothesis is said to have various degrees of reliability depending on how likely it is to have been falsified by a test. Unlike traditional accounts of confirmation, error-based statistical assessments such as Mayo's measure the ability to choose from one hypothesis over another by virtue of the extent of error-detecting testing methods applied to it. The degree to which these tests are able to detect error determines their severity. A hypothesis that is tested with methods that have a high likelihood of finding errors in it is said to pass a severe test. Severity is formally defined as follows:

A hypothesis H *passes a severe test* T *with data* x_0 *if*
1. *x_0 agrees with H, and*
2. *with very high probability, test T would have produced a result that agrees less well with H than does x_0, if H were false or incorrect.*

Informally, the severity principle suggests that a high degree of trust is warranted in cases where a hypothesis is not shown to be wrong in the face of tests that have a high probability of finding it wrong if the hypotheses were indeed false (Parker, 2008). Further, Mayo suggests that concentrating on choosing among highly probed hypotheses is crucially distinct from those approaches that rely on highly probable ones. In the former case we have a stronger positive account for falsification.

Wendy Parker (2008) argues that Mayo's error-statistical approach, and in particular her severity principle can help make the case for the epistemic import of computer-based methodology in science. This is because, according to her, Mayo explicitly accepts simulations as a method that helps scientist assess whether some source of error is absent in an experiment by estimating "what they would be more or less likely to observe if [any] source of error were present in the experiment." (Parker, 2008) Thus, we can have severe testing of hypotheses concerning possible sources for error in a particular experiment. For now, this first step allows Parker to make the case that computer-based methods are a reliable source of evidence at least with respect to sources of error in experiments given Mayo's account. When computer-based methods, such as simulations, are about a system that is not a conventional experiment and for which we have no real-world access the same approach can be applied according to Parker. Parker appeals to Mayo's account in the following way.

Simulation results are good evidence for H to the degree that:

(i) results fit the hypothesis, and
(ii) the simulation wouldn't have delivered results that fit the hypothesis if the hypothesis had been false (Parker, 2008).

For Parker one task is to ensure that (ii) holds. If (ii) holds then we can apply Mayo's notion of evidence to simulation experiments. This is even if such simulations are of the kind that cannot be immediately compared to actual data from a system, like those simulations that have to do with future states of a system. An example of these simulations could be computer experiments seeking to predict future weather patterns (Parker, 2008). According to Parker, appeal to lower level severity tests, as explained above, can ensure that (ii) is the case. That is, by making sure that errors that could have been part of the simulation are absent from the simulation we can then say that simulations are good sources of evidence and thus we can rely on them. Parker offers a taxonomy of error to help supplement her point. Although this taxonomy in itself may have its limitations and problems (i.e. see Floridi et al., 2015)[32] Parker thinks that while it is unclear that there are in fact procedures that allow us to assess the magnitude of some error's impact,[33] the list nevertheless provides evidence that "we do have some understanding of the different sources of error that can impact computer simulation results." (Parker, 2008)[34]

Path complexity and Big Data

As discussed above, Big Data is a software-intensive science. Given this dependence on software, as we will see below, testing applications of Big Data using conventional statistical inference theory (CSIT) is not an option. The reason for this is primarily due to the role

of conditionality in software (Horner and Symons, 2014; Symons and Horner, 2014).

The challenge is that for every conditional statement in piece of code the number of possible paths that must be tested grows. Pieces of code frequently contain conditional statements or their equivalents, that is, they take the form of "if…then/or else" statements. Thus, if a 10 line-long program has a conditional of this kind the lines to be tested would doubled to 20. Each of these conditionals augments the lines of code to be tested exponentially. Each conditional line of code alters the number of paths available to a given program. This increases the program's path complexity. Assessment of error distribution directly relates to degrees of reliability when testing software. Standard statistical techniques demand some degree of random distribution in the sample of interest. This element of random distribution is not available in the context of software testing. While random distributions are a reasonable assumption in natural systems, this is not the case in software systems since it is not feasible ahead of time to exclude the possibility that the distribution of error is caused by a non-random element in its constitution. Thus there is simply no way, other than by assumption or by exhaustive testing, to know whether or not a particular error distribution in software is the product of a random element or not (Symons and Horner, forthcoming). Thus, there is no way, other than by mere (unwarranted) assumption, to legitimately deploy statistical techniques that demand that the error distribution in a system have some degree of randomness to it. As exemplified by the discussion on path complexity, brute force attempts at exhaustive testing, as Symons and Horner argue, for any conventional program is an impractical task given meaningful time constraints.

Even the simplest computer programs have 1000+ lines of code and an average of one conditional statement per every 10 lines. Thus, for example, the time resources required for testing a program with 1000 lines of code with this average of conditionals would exceed many-fold the age of the universe.[35] A program consisting of 1000 lines of code would be a very small program for anything in the Big Data context. Most computer programs used in these context are large and in scientific applications more generally are commonly in the hundreds of thousands of lines of code (Horner and Symons, 2014; Symons and Horner, 2014).

The most important consequence of the path-complexity catastrophe is the fact that statistical methods no longer apply in a straightforward manner to the detection of error in software system.

It may be countered that modularity in software systems may be a way to diminish the impact of path complexity and thus reduce the epistemic opacity related to it.[36] Perhaps, it can be argued, by breaking a system into epistemically manageable modules we may indeed be able to carefully test each and every one of them independently and thus have a reliable error assessment of the system as a whole. If this is the case then, we can independently rely on each of them and by extension on all of them together. At first sight this sounds like a plausible approach to the problem of path complexity in particular and epistemic opacity in general. However, path complexity grows at catastrophic rates even given relatively small numbers of lines of code. The interplay between modules will introduce untested paths even in cases where the modules themselves are reliable. The discussion above about the obstacles to the deployment of conventional statistical methods shows that even at a smaller scale the only truly available testing technique for assessment of error distribution would be an exhaustive brute force one. Even if we were to grant that massive modularity and exhaustive testing was a viable method for software design and testing, integrating modules will result in epistemic opacity.

Although modularity may indeed make black boxes a bit more manageable, the dynamics among the modules would quickly evolve into a particularly complex system with its own problems. One immediate concern is the assumption that software (and indeed any other modular system) develops as a cohesive, all-encompassing unifying endeavor rather than as a patchwork (Winsberg, 2010).

While unification and modularity can be part of a protocol for future software development, it is not currently in place and the question remains as to whether it can be implemented in scientific inquiry and the large software systems that already underlie it. Take climate modeling for example. When considering climate models, Winsberg (2010) cites at least three kinds of uncertainty that have to be taken into consideration: structural uncertainty, parameter uncertainty, and data uncertainty. The most important source of uncertainty for our current discussion is structural uncertainty of the model itself, which includes considerations regarding "a plethora of auxiliary assumptions, approximations, and parameterizations all of which contribute to a degree of uncertainty about the predictions of these models." (2010) Each of these assumptions, approximations, and parameterizations is based upon segments, or modules of software code that implement them. Let us for a second, in a very simplistic and rough way, think of each of the many modeling layers that go into the software that predicts climate as modules. Even if we exhaustively specify/test each module, the interactions among modules, their epistemic transparency, and therefore their reliability as a functioning system them won't be as straightforward. Consider, for example, that after 70+ years of climate modeling the complexity

surrounding the integration of so many different (one may argue modular) systems/models has only allowed scientist to claim a degree of accuracy that averages merely a day per decade (Bauer et al., 2015). That is, after seven decades, and the use of the most sophisticated and powerful software, the integration of the multiple modules of climate modeling is anything but done. If anything this example elucidates the difficulty of managing integration of large complex simulations systems. Furthermore, it exemplifies how modularity may not even be an option in scientific practice.

Example

There has been a recent trend in the past decade or so to use the vast amount of data generated by internet searches in attempts to create predictive models. These models range from prediction of American Idol winners (Ciulla et al. 2012), political election outcomes, unemployment rates, box-office receipts for movies, and song positions in charts (Goel et al., 2010). But perhaps the best known among these attempts has been the flu tracker function: GFT. Researchers at Google expected the data from accumulated queries to yield correlational patterns that, all by themselves, would tell a story about the presence and spread of the disease (Lazer et al., 2014; Lazer and Kennedy, 2015). GFT exemplifies the spirit of so-called *empiricist* interpretation of Big Data, discussed above. However, the researchers' hopes did not materialize. Although some correlations were discovered, Google's flu tracker continued to consistently generate spurious correlations and, more seriously, reporting false flu numbers (Lazer et al., 2014; Lazer and Kennedy 2015; Olson et al., 2013).

GFT was designed to predict, in real time, the advent of a flu epidemic. The innovative aspect of this tracker was its reliance on the relatively loose search queries typed into Google's search engine. These data, they hoped, could serve as the basis for predictions concerning the behavior of the epidemic (Cukier and Mayer-Schoenberger, 2013). The core idea behind this project was to provide an alternative to conventional epidemiological surveillance and prediction systems which relied on medical reports of Influenza-like illnesses (ILI's) from regional clinics to the Centers for Disease Control and Prevention (CDC's). In particular it hoped to foresee an epidemic outbreak from search queries that would indicate a strong presence of flu-like symptoms based on specific flu-related words and combinations of these words typed into the search engine. This, they argued, could be done if not in real time, at least faster than reports from patients seeking care at local clinics, which could take a number of days. However, after its launch as an open tool for flu surveillance in 2008 there were two seriously embarrassing moments for GTF. One of them was the fact that it failed to predict the A/H1N1 flu pandemic in 2009. This led Google to actually modify its original algorithm in an attempt to get more accurate results. However, the second problem was that GTF suffered from general gross overestimation. In particular, it overestimated by a large margin 100 times out of 108 during the flu season between 2011 and 2012 (Lazer et al., 2014) and it greatly overreported flu cases during the 2012 and 2013 A/H3N2 pandemic (Olson et al., 2013).

It is by now well known that Google's flu tracker failed to achieve what it was designed to do, namely predict and report ILI's better and faster than the conventional surveillance tools available. It simply didn't predict at all or predicted erroneously. Because of this, the project is often taken to exemplify "Big Data hubris" (Lazer et al., 2014), the often underlying assumption that large amounts of data and the patterns that are discovered through its analysis can yield results independently from or without the aid of principled theoretical underpinnings.

Although the disappointing errors of GFT have been rigorously documented and measured (Olson et al., 2013; Salzberg, 2014; see the supplemental material in Lazer et al., 2014) what is most interesting to our discussion is the ambiguity regarding their nature and source. Many of these studies focus particularly on the margin of error but are not clear about what caused the errors. Some researchers (Cook et al., 2011) for example, ascribe the errors to issues like seasonality, the fact that outbreaks happened outside of what is commonly thought to be flu-season. This meant that common flu-related terms were less likely to have been used in queries to the search engine.[37] Others ascribe the errors to differences of age distribution and geographical heterogeneity occurring during model fitting periods of GFT (Olson et al., 2013).

Lazer et al. (2014) offer two possible culprits. The first is due to neglect of traditional statistical techniques. Some of the error here can be fixed once GFT incorporates conventional statistical methods that can provide correlational filters. These methods inform modern pattern finding techniques in traditional research beyond Big Data (Lazer et al., 2014). If conventional statistical methods were deployed along with the GFT a reflective equilibrium between data from ILI's surveillance and search terms could better calibrate GFT. Given the results presented by Horner and Symons the suggestion to take statistics seriously, while generally sensible, might have additional complications that are beyond the scope of this paper.

Lazer et al. (2014) suggest that another possible cause for the errors in GFT is what they call algorithm dynamics undergone by Google's search algorithm. Algorithm dynamics, according to them, are

modifications to Google's search algorithms that are introduced in order to enhance the functionality of the search engine. They are of two kinds: blue team dynamics, those that the service provider deploys for greater efficiency and usefulness of search results; and red team dynamics, those done by users of the service for personal benefit such as prominence and visibility. According to them, blue team dynamics are what is most likely behind GFT's errors. The evidence that they cite is a correlation between reported changes to the algorithm and the surge of predictive errors in GFT. According to the authors, what makes the system yield errors is the way in which search results skew the queries themselves, queries which are in turn used to extract the terms for the GFT to analyze. That is, the search results of Google's search engine influence the prominence of search terms that users input and these skewed inputs then are used by GFT to predict the presence or absence of the flu (Lazer et al., 2014).

That results generated by the search engine can modify queries and input from the very source that is supposed to be furnishing the data for prediction is troublesome enough. However, there is something deeper going on. Although Lazer et al. (2014) define the blue team algorithm dynamics undergone by Google's search algorithms in the context of Google's particular business model, one can extend the term beyond Google or GFT. Other social media platforms engage in algorithm dynamics too, in particular blue team dynamics (Lazer et al., 2014). In fact, algorithm dynamics, insofar as they are defined as the changes made to any software product by those designing it, affect all aspects of software production and development. These modifications include model fitness processes and functional additions to the underlying software that are necessary to its proper functioning. Thus, algorithm dynamics are an essential feature of the kind of artifact that software is (Holzmann, 2015) and not merely a product of arbitrary human intervention.

Given the extent and scope of these dynamics in Big Data more generally, we have a bigger issue on our hands than merely biased data gathering. In particular the issues discussed in ''The epistemic status of Big Data'', ''Error'', and ''Path complexity and Big Data'' sections: epistemic opacity due to sheer volume of people, number of processes, legacy code, and path complexity catastrophe given the number of lines of code involved in projects of such magnitude are at play. But the challenge is not merely related to product development and modification. The epistemically relevant feature of this cycle of updating software results from our inability to test for error and our dependence on systems that are susceptible to it. In other words, this is an issue of knowledge acquisition, reliability, and, therefore, trust.

The software behind GTF and similar Big Data projects falls prey to the path complexity catastrophe as described by Symons and Horner. Whatever efforts we introduce to mitigate error in these systems will be undermined by the fact that they incorporate a vast number of individual machines and computational methods to yield even the simplest of results. And as discussed above, even if we characterize the problem in terms of modules, the process is highly unlikely to become less opaque.

Discussion

Issues of path complexity and epistemic opacity are more than merely abstract theoretical preoccupations. As stated in the introduction to this article, some of the limitations and risks involved in the use of computational methods in public policy, commercial, and scientific contexts only become evident once we understand the ways in which these methods are susceptible to error. In the broader social and political context, a precondition for understanding the potential abuses that can result from the deployment of Big Data techniques by powerful institutions is a careful account of the epistemic limits of computational methods. A clear sense for the nature of error in these systems is essential before we can decide how powerful they should become and how much trust we should grant them (see for example Paparrizos et al., 2016). By way of illustration, we have focused our attention on the limitations of GFT as a predictive tool that can be a supplement to ILI's surveillance. The consequences of overestimation in this context are not as immediately troubling as the consequences for other systems that are in use in governmental and military contexts. For example, if we relate our discussion to Software Studies research, such as that of Louise Amoore for example, we can see the immediately troublesome implications that a conventional account of epistemic trust on Big Data systems could have. In her research (Amoore, 2011), Big Data systems are in charge of calculating and assessing the security risks posed by individuals flying from one part of the world to another. Without having a proper understanding of the nature of error inherent to these systems, assessing whether they are flagging the right people, or the right number of people becomes ever more challenging.

Debates concerning the epistemic status of Big Data in the Critical Data Studies literature must take account of the nature of error in software intensive contexts. We have shown that an account of error management and reliability can be profitably introduced into the agenda of Critical Data Studies. Symons and Horner's concept of path complexity for example, highlights the limitations of testing given intrinsic features of software. The problem of reliability and the changing character of trust in the context of Big Data projects pose an ongoing challenge for Critical Data Studies.

Acknowledgments

We are very grateful to the anonymous referees of this journal as well as to the guest editors of the issue Andrew Iliadis and Federica Russo for their helpful feedback.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

Notes

1. By taking the epistemic status of Big Data as the starting point, we should not be understood as claiming that the users of these technologies are motivated primarily by the pursuit of truth.
2. Chris Anderson's provocative (2008) article "The end of theory: the data deluge makes the scientific method obsolete" is widely cited in this context although as Martin Frické (2015) notes, "apparently Anderson never believed or advocated the theses of his own paper but wrote it to provoke response" (see also Norvig, 2008).
3. More recently the pitfalls associated with atheoretical uses of Big Data have become clear even to large corporate interests like IBM (Marr, 2015). In 2015, Marr considered a Data Guru by the industry participated in an interview for IBM's community podcast. In it, he discusses the main point of his forthcoming book. He emphasized that the full resolution, 'more means more' approach to Big Data is misguided. The dynamic nature of Big Data problems require it to collect, analyze, and solve issues in real time. This means that old data may not be as helpful to solve current problems. More interestingly, he thinks that strategic inquiry before collection is the new way to go. This position stands in sharp contrast to the "death of theory" thesis espoused by advocates of the unrestricted correlation camp. See also Jacobs (2009) for a discussion of the importance of analysis in Big Data. What he calls "the pathologies of Big Data" are due to an uncritical attitude towards accumulated data.
4. By "error" we simply mean to encompass the wide range of ways in which a software system may fail. This may include erroneous calculations, implementations, results, etc. The important point here is not errors in coding per se but the epistemic implications of those errors in the context of inquiry. Thus, error detection and correction are the focus of this paper. For a more detailed account of error in software see Parker (2008) and Floridi et al. (2015).
5. In 2009, Adam Jacobs describes the development of increasingly powerful computing technology and argues that the challenges associated with Big Data are due to analysis rather than size "The pathologies of Big Data are primarily those of analysis. This may be a slightly controversial assertion, but I would argue that transaction processing and data storage are largely solved problems." (39)
6. Most publications discussing Big Data over the past five years have praised the predictive power of the new methods and the seemingly unprecedented insights brought by the visualization techniques that it enables. Some have adopted a skeptical stance towards the commercial hype surrounding Big Data (The most sophisticated of these include Bollier and Firestone, 2010; Boyd and Crawford, 2012; Kitchin, 2014).
7. The use of the term 'empiricist' is only marginally related to views that philosophers would recognize as empiricist. A better label for this cluster of views might be "atheoretical" or "anti-theoretical".
8. He quickly dismisses the Kuhnian approach by saying that paradigmatic accounts are overly "sanitized" and "linear" accounts of scientific inquiry (2014). Nevertheless, he endorses the Kuhnian notion of "paradigm" as useful in the Big Data debate.
9. Cukier and Mayer-Schoenberger argue the volume size in Big Data brings about three drastic changes to data analysis: unrestricted sampling (in accordance with (i) above), tolerance of inaccuracy as a tradeoff of the vastness of possible correlations, and lastly giving up on "our quest to discover the cause of things" in exchange for predictive prowess. Although some of these correlate with Kitchin's list, we think that the question about whether scientific inquiry can do away with causal insight is a very important one that is not discussed carefully enough in the literature and that is worthy of a paper in and of itself.
10. Thus, we can say that (1) is implausible given that in a dynamic real world/real time problem $N = \text{All}$ as a sample is very difficult to obtain/define for problems of practical interest.
11. Although it is widely acknowledged that the "end of theory" claims are hyperbolic at best (Boyd and Crawford) (Bollier and Firestone, 2010), most of the criticism is anthropocentric and social in nature. That is to say, as discussed above, it often makes reference to some aspect of social epistemology such as individual and collective biases.
12. Longino (1990) points out that these are features that have a long history in the development of science as it grew from highly individual projects in the 1800s to multidisciplinary institutional ventures. However, Contrary to Kitchin, Boyd and Crawford, Longino redefines objectivity as a product of the social character of science. For her, an addition of subjectivities tends to neutralize particular biases and sift out highly individualized values and preferences. So, for her, objectivity is not absent in the scientific process, but rather stems from a different source than conventionally thought.
13. We are grateful to an anonymous referee for encouraging us to respond to Amoore's work on security.
14. For a thorough overview of the many dimensions of interest in philosophy of computation see Rapaport (2015); for an interesting analysis on function and malfunction in software see Floridi et al. (2015).

15. One of the biggest questions arising from the use of computer simulations in science is whether they are part of the scientist's empirical toolkit (Barberousse et al., 2009; Floridi, 2012; Winsberg, 2010).
16. She cites the study of shock wave behavior and neutron diffusion as topics to which Ulam, von Neumann, Fermi and others applied novel computational techniques.
17. Independently of the computations themselves though, the method was a novel statistical approach to serial processes that could not be made faster using the classical—non statistical—approach even by using multiple computers (Metropolis and Ulam, 1949). Thus the method wasn't only adding speed but also a conceptual shift towards probability theory that brought with it novel epistemic challenges (Winsberg, 2010).
18. See also Symons (2008) for further discussion of the relationship between computational modeling and explanation.
19. For a very different perspective on the relationship between simulation and theory see Morrison (2015). On Morrison's view simulations can play a role equivalent to experimental evidence in relation to scientific theories.
20. What she has in mind here are models, like cellular automata that are not attempts to capture some specific physical phenomenon. This third stage is targeted towards "phenomena for which no equations, either exact or approximate, exists (as, e.g., in biological development), or for which the equations that do exists simply fall short (as, e.g., turbulence)." (2003: 210; see also Symons, 2008).
21. It is important to note that for Fox-Keller, this third sense of simulations is particularly important because its aim is no longer to simulate neither differential equations nor fundamental (albeit idealized) particles of a given system but rather the phenomenon itself. That is, cellular automata, for example, were simulations that described and elucidated patterns about the systems carrying out the simulations themselves. Although cellular automata are more famously considered to be a simulation similar to those discussed on the second stage, this was only a consequence of the visualization similarities with real life cell formation. Originally however, they were constructed to simulate themselves. Fox-Keller describes this confusion by stating that "despite its explicitly biological allusion, [cellular automata] was developed by—and for the most part has remained in the province of—physical scientists." (2003) The resemblance to biological process of self reproduction was only noted later.
22. The question of whether Big Data is indeed a suitable scientific instrument is still an open question (Lazer et al., 2014), for example have the following to say about Google's flu tracker reliance on Big Data methodology: "The core challenge is that most big data that have received popular attention are not the output of instruments designed to produce valid and reliable data amenable for scientific analysis." (Lazer, 2014)
23. Whether any computational simulation does involve either the testing and/or the expanding of any given theory and to what extent it may do so is the subject of a vast and open philosophical debate. This is particularly the case when computational simulations are taken to be part of the empirical tools available to scientists (see for example Barberousse et al., 2009; Barberousse and Vorms, 2014; Winsberg, 2010).
24. As discussed below, Symons and Horner sought to distinguish SIS from non-SIS science by virtue of the degree of conditionality present in both (2014; forthcoming).
25. So, for example the problem of understanding the Stokes flow over a finite cylinder is analytically intractable as are a range of other problems from fluid dynamics (see e.g. Ferziger and Peric, 2012).
26. We must remember that one of the most important aspects in the development of computational methods for analysis of dynamic systems was their visualization (Keller, 2003). In fact, some simulations, like cellular automata, later came to be regarded as powerful epistemic tools, somewhat analogous to natural systems, because of the fact that their macroscopic properties, that is their visual evolutions, resembled real patterns visible in cell formation. Fox-Keller ascribed this key insight to joint work by Von Neumann and Ulam and further cites (Toffoli and Margolus, 1987).
27. Bollier, for example, argues that visualization in the data industry is a sense-making tool. He ties this to his criticism of the "raw data" advocates and argues that many of the insights drawn from Big Data can only emerge when seen by an expert and seldom arise solely as a product of numerical calculations. One example of this is how Google research found that two out of three cows align their bodies to the north pole just by observing images from Google Maps. No machine, he argues, could have done this alone. This is important in our context, not because of the epistemic limitations on machine recognition, but rather because it shows the intrinsically visual nature of so much of Big Data analysis and the correlation this visualizations have with philosophical debates about simulation.
28. This distinction matters because of the epistemic import of the methods themselves. If simulation is closer to theory, some say, then no novel knowledge can be generated from them. All we can reasonably expect are coherence assessments of internal theoretical principles. If simulations are like experiments, on the other hand, then we have reasons to include them in our empiricist toolkit (Barberousse and Vorms, 2014).
29. Furthermore, we must consider the possibility that in adding new code to legacy code one may even be exacerbating its opacity.
30. We thank an anonymous referee for bringing to our attention the similarity between black box theory and epistemic opacity as well as mentioning modularity as a possible response to the problem of epistemic opacity.
31. Some efforts have been made to provide a taxonomy of error in software, however they focus on external sources such as inaccurate design. For a thorough review of malfunction in software see Floridi et al. (2015).
32. For a detailed account of the degree to which this account of error may figure in the software that underlies simulations see Floridi et al. (2015). In it they argue that certain kinds of error are only possible to a limited degree (a type/token distinction) in software. Further they argue

that such error is always from an external source. That is, all of the errors listed above are external to the software itself, since software will always do what it was designed to do (whether the design fits the task intended for the software is an external problem).

33. Parker makes some remarks concerning the limitations of this approach: first she thinks that severe testing on simulations is rare; second, she acknowledges that formal statistical analysis of the kind used by Mayo to support simulation processes has much work to do and whether it ends up playing "as large a role (or the same role)" in simulations remains to be seen; third, error directly related to the model used to build the simulation is a very hard problem, particularly considering that many traditional/observational assumption go into such models. She suggests an extra statistical approach to deal with this last problem, however, for details that we will explain below, this may not work either.
34. In so far as application goes, the severity principle, as formulated by Mayo and as adopted by Parker's in her discussion of computer simulations, is still a philosophical principle at its core and as such it does its job mainly as a background assumption at work when science is conducted. That is, the principle is mainly a meta consideration about what constitutes appropriate epistemic support for a scientific hypothesis and how this may ultimately be granted legitimization in the realm of scientific explanation. We thank an anonymous reviewer for the opportunity to clarify this.
35. For details of the argument, please see Symons and Horner (2014).
36. Once again we thank an anonymous reviewer for bringing this to our attention.
37. Researchers (Cook et al., 2011) believe that user's search terms change depending on the season even if the symptoms are the same. Given the common seasonal patterns of flu pandemics, even if users were exhibiting flu-like symptoms they would search for different terms in the winter from those in the spring.

References

Amoore L (2011) Data derivatives on the emergence of a security risk calculus for our times. *Theory, Culture & Society* 28(6): 24–43.

Amoore L (2014) Security and the incalculable. *Security Dialogue* 0967010614539719.

Anderson C (2008) The end of theory: The data deluge makes the scientific method obsolete. Available at: http://www.wired.com/science/discoveries/magazine/16-07/pb_theory (accessed 24 January 2016).

Arbesman S (2013, August) Five myths about Big Data. *The Washington Post*. Available at: www.washingtonpost.com/opinions/five-myths-about-big-data/2013/08/15/64a0dd0a-e044-11e2-963a-72d740e88c12_story.html (accessed 24 January 2016).

Barberousse A, Franceschelli S and Imbert C (2009) Computer simulations as experiments. *Synthese* 169(3): 557–574.

Barberousse A and Vorms M (2014) About the warrants of computer-based empirical knowledge. *Synthese* 191(15): 3595–3620.

Bauer P, Thorpe A and Brunet G (2015) The quiet revolution of numerical weather prediction. *Nature* 525(7567): 47–55.

Bedau MA (1997) Weak emergence. *Noûs* 31(11): 375–399.

Berry DM (2011) The philosophy of software. In: *Code and Mediation in the Digital Age*. London: Pal-grave..

Bollier D and Firestone CM (2010) *The Promise and Peril of Big Data*. Washington, DC: Aspen Institute, Communications and Society Program, p. 56.

Boschetti F, Fulton EA, Bradbury R, et al. (2012) What is a model, why people don't trust them, and why they should. In: Raupach MR, McMichael T, Finnigan JJ, et al. (eds) Negotiating Our Future: Living Scenarios for Australia to *2050*. Canberra, ACT: Australian Academy of Science.

Boyd D and Crawford K (2012) Critical questions for big data: Provocations for a cultural, technological, and scholarly phenomenon. *Information, Communication & Society* 15(5): 662–679.

Brusoni S and Prencipe A (2001) Unpacking the black box of modularity: Technologies, products and organizations. *Industrial and Corporate Change* 10(1): 179–205.

Bryant R, Katz RH and Lazowska ED Big-data computing: Creating revolutionary breakthroughs in commerce, science and society, December 2008, pp. 1–15. Available at: http://www.cra.org/ccc/docs/init/Big_Data.pdf.

Bunge M (1963) A general black box theory. *Philosophy of Science* ▮▮: 346–358.

Chen M, Mao S and Liu Y (2014) Big Data: A survey. *Mobile Networks and Applications* 19(2): 171–209.

Ciulla F, Mocanu D, Baronchelli A, et al. (2012) Beating the news using social media: The case study of American Idol. *EPJ Data Science* 1(1): 1–11.

Cook S, Conrad C, Fowlkes AL, et al. (2011) Assessing Google flu trends performance in the United States during the 2009 influenza virus A (H1N1) pandemic. *PloS One* 6(8): e23610.

Cox M and Ellsworth D (1997) Application-controlled demand paging for out-of-core visualization. In: *Proceedings of the 8th conference on visualization'97,* Phoenix, AZ, USA, 18–24 October 1997, Los Alamitos, CA, IEEE Computer Society Press.

Cukier K and Mayer-Schoenberger V (2013) Rise of Big Data: How it's changing the way we think about the world, The. *Foreign Affairs* 92: 28.

Ferziger JH and Peric M (2012) *Computational Methods for Fluid Dynamics*. Springer Science & Business Media.

Ethiraj SK and Levinthal D (2004) Modularity and innovation in complex systems. *Management Science* 50(2): 159–173.

Floridi L (2012) Big Data and their epistemological challenge. *Philosophy & Technology* 25(4): 435–437.

Floridi L, Fresco N and Primiero G (2015) On malfunctioning software. *Synthese* 192(4): 1199–1220.

Frické M (2015) Big Data and its epistemology. *Journal of the Association for Information Science and Technology* 66(4): 651–661.

Frigg R and Hartmann S (2012, Fall) Models in science. In: Zalta EN (ed) *The Stanford Encyclopedia of Philosophy*.

Available at: http://plato.stanford.edu/archives/fall2012/entries/models-science (accessed 24 January 2016).

Frigg R and Reiss J (2009) The philosophy of simulation: Hot new issues or same old stew? *Synthese* 169(3): 593–613.

Gantz J and Reinsel D (2011) Extracting value from chaos. *IDC iview* 1142: 9–10.

Goel S, Hofman JM, Lahaie S, et al. (2010) Predicting consumer behavior with Web search. *Proc Natl Acad Sci* 107(41): 17486–17490.

Holzmann GJ (2015) Code inflation. *IEEE Software* ▮▮(2): 10–13.

Horner J and Symons J (2014) Reply to Angius and Primiero on software intensive science. *Philosophy & Technology* 3(27): 491–494.

Hilliard R (2000) Ieee-std-1471-2000 recommended practice for architectural description of software-intensive systems. *IEEE* 12: 16–20. Available at: http://standards.ieee.org (accessed 13 June 2016).

Humphreys P (2009) The philosophical novelty of computer simulation methods. *Synthese* 169(3): 615–626.

Jacobs A (2009) The pathologies of big data. *Communications of the ACM* 52(8): 36–44.

Keller EF (2003) *Models, simulation, and "computer experiments"*. Available at: http://www.informatics.indiana.edu/jbollen/I501F11/readings/week8/Fox-Keller_2002_MODELS_SIMULATION_AND_COMPUTER_EXPERIMENTS.pdf (accessed 24 January 2016).

Kelling S, Hochachka WM, Fink D, et al. (2009) Data-intensive science: A new paradigm for biodiversity studies. *BioScience* 59(7): 613–620.

Kitchin R (2014) Big Data, new epistemologies and paradigm shifts. *Big Data & Society* 1(1): 1–12 DOI: 10.1177/2053951714528481.

Kuhn TS (1962) *The Structure of Scientific Revolutions*. Chicago: Chicago University Press.

Lazer D, Kennedy R, King G, et al. (2014) The parable of Google Flu: Traps in big data analysis. *Science* 434: 343.

Lazer D and Kennedy R (2015) What we can learn from the epic failure of Google Flu Trends. *Wired*.

Longino HE (1990) *Science as Social Knowledge: Values and Objectivity in Scientific Inquiry*. Princeton, NJ: Princeton University Press.

Marr B (2015) *Big Data: Using SMART Big Data, Analytics And Metrics To Make Better Decisions And Improve Performance*. Chichester, UK: John Wiley & Sons.

Mayo DG and Spanos A (2010) *Error and Inference: Recent Exchanges on Experimental Reasoning, Reliability, and the Objectivity and Rationality of Science*. New York, NY: Cambridge University Press.

Metropolis N and Ulam S (1949) The Monte Carlo method. *Journal of the American Statistical Association* 44(247): 335–341.

Morrison M (2015) *Reconstructing Reality: Models, Mathematics, and Simulation*. New York, NY: Oxford University Press.

Newman J (2015) Epistemic opacity, confirmation holism and technical debt: Computer simulation in the light of empirical software engineering. Available at: http://hapoc2015.sciencesconf.org/conference/hapoc2015/pages/Newman.pdf (accessed 24 June 2016).

Norvig P (2008) All we want are the facts, ma'am. Available at: http://norvig.com/fact-check.html (accessed 22 January 2016).

Oberkampf WL, Trucano TG and Hirsch C (2004) Verification, validation, and predictive capability in computational engineering and physics. *Applied Mechanics Reviews* 57(5): 345–384.

Olson DR, Konty KJ, Paladini M, et al. (2013) Reassessing Google Flu trends data for detection of seasonal and pandemic influenza: A comparative epidemiological study at three geographic scales. *PLoS Computational Biology* 9(10): e1003256.

Paparrizos J, White RW and Horvitz E (2016) Screening for pancreatic adenocarcinoma using signals from web search logs: Feasibility study and results. *Journal of Oncology Practice*, p.JOPR010504.

Parker WS (2008) Computer simulation through an error-statistical lens. *Synthese* 163(3): 371–384.

Rapaport WJ (2015) Philosophy of computer science. Available at: http://www.cse.buffalo.edu/∼rapaport/Papers/phics.pdf

Salzberg S (2014) Why Google flu is a failure. *Forbes. com [online]*. pp. 03–24. Available at: http://www.forbes.com/sites/stevensalzberg/2014/03/23/why-google-flu-is-a-failure/#33bbe7d4344a (accessed 24 January 2016).

Steadman I (2013)Big Data and the death of the theorist. *wired. co. uk*, Vol. 25, pp. 2013-01.

Symons J (2002) Emergence and reflexive downward causation. *Principia: An International Journal of Epistemology* 6(1): 183–201.

Symons J (2008) Computational models of emergent properties. *Minds and Machines* 18(4): 475–491.

Symons J and Boschetti F (2013) How computational models predict the behavior of complex systems. *Foundations of Science* 18(4): 809–821.

Symons J and Horner J (2014) Software intensive science. *Philosophy & Technology* 27(3): 461–477.

Symons J and Horner J (forthcoming) Software error as a limit to inquiry for finite agents: Challenges for the post-human scientist. Available at: http://www.johnsymons.net/wp-content/uploads/2016/06/Final-JKHJFS_20160320_0635_IACAP1.pdf (accessed 24 June 2016).

Toffoli T and Margolus N (1987) *Cellular Automata Machines: A New Environment For Modeling*. MIT press.

Weisberg M (2013) *Simulation and Similarity: Using Models to Understand the World*. New York, NY: Oxford University Press.

Winsberg E (2010) *Science in the Age of Computer Simulation*. Chicago: University of Chicago Press.

This article is a part of special theme on Critical Data Studies. To see a full list of all articles in this special theme, please click here: http://bds.sagepub.com/content/critical-data-studies.