In defense of reliabilist epistemology of algorithms

Durán, Juan M.

DOI
10.1007/s13194-025-00664-2

Publication date
2025

Document Version
Final published version

Published in
European Journal for Philosophy of Science

European Journal for Philosophy of Science (2025) 15:37
https://doi.org/10.1007/s13194-025-00664-2

PAPER IN GENERAL PHILOSOPHY OF SCIENCE

In defense of reliabilist epistemology of algorithms

Juan M. Durán[1]

Received: 19 August 2024 / Accepted: 3 June 2025 / Published online: 11 June 2025

Abstract
In a reliabilist epistemology of algorithms, a high frequency of accurate output representations is indicative of the algorithm's reliability. Recently, Humphreys challenged this assumption, arguing that reliability depends not only on frequency but also on the quality of outputs. Specifically, he contends that radical and egregious misrepresentations have a distinct epistemic impact on our assessment of an algorithm's reliability, regardless of the frequency of their occurrence. He terms these *statistically insignificant but serious errors* (SIS-Errors) and maintains that their occurrence warrants revoking our epistemic attitude towards the algorithm's reliability. This article seeks to defend reliabilist epistemologies of algorithms against the challenge posed by SIS-Errors. To this end, I draw upon *computational reliabilism* as a foundational framework and articulate epistemological conditions designed to prevent SIS-Errors and thus preserve algorithmic reliability.

Keywords Reliabilist epistemologies of algorithms · Computational reliabilism · SIS-Errors · Paul Humphreys

1 Introduction

In 2009, there was a short-lived debate about the alleged novelty of computer simulations in the scientific domain. Frigg and Reiss argued that, although technologically novel, computer simulations did not constitute a philosophical novelty nor "[a] revolutionary departure from everything that philosophers were worried about in the past" (Frigg & Reiss, 2009, 601). In response, Humphreys highlighted four specific issues in the context of computer science that bear philosophical novelty (Humphreys, 2009). One of these issues is *epistemic opacity*.

✉ Juan M. Durán
j.m.duran@tudelft.nl

[1] Technology, Policy and Management, Delft University of Technology, Jaffalaan 5, 2628 BX Delft, The Netherlands

Humphreys provided the following formal definition:

> [A] process is epistemically opaque relative to a cognitive agent *S* at time *t* just in case *S* does not know at *t* all of the epistemically relevant elements of the process (Humphreys, 2009, 618)[1]

To grasp this definition, Humphreys offers an analogy with mathematical proofs. According to this analogy, a mathematician might consider at some point in the proof that a particular step is epistemically relevant for the justification of the theorem—or, conversely, that the step is sufficiently trivial to be eliminable. The key to the analogy is that mathematicians can, and often do, survey mathematical proofs. It is through this form of surveyability that they confer justification to their results.

With algorithms, Humphreys tells us, the situation is rather different. For starters, "no human can examine and justify every element of the computational processes that produce the output of a computer simulation or other artifacts of computational science" (Humphreys, 2009, 618). The surveyability of a mathematical proof, i.e., that *S* knows at *t* all the epistemically relevant elements in the proof, is, in principle, unrealizable in the context of algorithms. This, roughly, forms the basis of epistemic opacity, and it has been the main route that many have taken (Durán & Formanek, 2018; Beisbart, 2021; Boge, 2022), including Humphreys himself (2020; 2021).

A central motivation for seeking justification is that algorithms are often epistemically opaque. How, then, do we currently justify their outputs? What is the prevailing epistemology of algorithms? Transparency emerges as an initial response, precisely because it is framed as the opposite of opacity (Creel, 2020). Broadly speaking, transparency justifies the belief that the algorithm's output represents by offering reasons or supporting evidence for that belief (Kroll et al., 2017; Guidotti et al., 2019; Zerilli, 2022). Reasons and supporting evidence is typically achieved by "convey[ing] the internal state or logic of an algorithm" (Wachter et al., 2018, 845), that is, by revealing or explaining the functions, values, and properties that produce the output in question (Durán, 2021). Understood in this way, transparency treats justification as *internal* to the algorithm.

Humphreys initially considered transparency as a potential candidate for the epistemology of algorithms but quickly dismissed it as unsuitable.[2]

> [i]f we think in terms of such a [computer] process and imagine that its stepwise computation was slowed down to the point where, in principle, a human could examine each step in the process, the computationally irreducible process would become epistemically transparent. What this indicates is that the practical constraints we have previously stressed, primarily the need for computational speed, are the root cause of all epistemic opacity in this area. Because those constraints cannot be circumvented by humans, *we must abandon the insistence on epistemic transparency for computational science.* What replaces it would

[1] There is a further specification on the nature of *S* in Humphrey's definition of *essential epistemic opacity*. For the purposes of this article, this specification is unproblematic.

[2] This is, of course, not to say that transparency is not intrinsically valuable. There are several noteworthy attempts to promote transparency in various forms, such as efforts aimed at understanding (e.g., Sullivan, 2022; Páez, 2023).

require an extended work in itself, but the prospects for success are not hopeless. (Humphreys, 2004, 150. Emphasis mine.)

The suggested alternative is to construe justification as *external* to the algorithm, that is, as a form of reliabilist epistemology. Humphreys' position is, in fact, one that adopts a frequentist version of process reliabilism (Goldman, 1979), where a high ratio of beliefs that represent confers reliability to the process. There are, however, two key modifications to the original process reliabilism that are central to his argument. First, Humphreys specifies that the notion of "process" refers specifically to computer processes-namely, algorithms executed by a computer. This contrasts with perceptual, introspective, and testimonial processes also discussed in the context of reliabilist epistemologies. Second, the beliefs in question concern algorithmic outputs that, to a degree acceptable to the relevant epistemic community, accurately represent a fact (F) or a fact that entails F (Humphreys, 2020, 13:50). Under this heading, Humphreys argues that a single output misrepresenting a fact F may epistemically compel us to revoke our assessment toward the algorithm's reliability, even if the overall frequency of representations remains unaltered. In other words, it is not only the quantity of output representations that matters for the reliability of the algorithm; their quality is also crucial.

To illustrate this problem, consider the following toy example: say that an algorithm is used in forensics for identifying, via facial detection, the suspect of a crime. Suppose that the algorithm is highly successful in identifying the right suspects and, therefore, is reliable. Now, according to Humphreys, if the algorithm were to misidentify a single individual, given the egregious epistemic nature of this misidentification, we have grounds for revoking our assessment of the algorithm from reliable to unreliable. Humphreys referred to this problem as *statistically insignificant but serious errors* (henceforth, SIS-Errors), and considers it a core problem for any reliabilist epistemology of algorithms.

This article defends reliabilist epistemologies against SIS-Errors. To this end, I adopt *computational reliabilism* (henceforth, CR) for this task (Durán, forthcoming). The reasons for this choice are fully presented in Section 4. Now, it is worth mentioning that my defense takes two routes that depart in novel ways from those standardly found in the literature. First, SIS-Errors are considered an epistemological problem. As I briefly cover in Section 2, the literature on algorithmic reliability typically addresses concerns about (mis)representations in terms of design decisions, coding strategies, and implementation practices. That is, reliability is primarily a matter of the methodology of algorithms. Admittedly, CR does offer its own version of this methodological matter, and the argument presented here does rely, to some extent, on how CR builds on practices and methodologies. But SIS-Errors are primarily framed and discussed as a problem of justification, and therefore as a matter of belief formation. As such, the main issue here is our epistemic attitude towards the reliability of the algorithm. The second route consists of creating epistemic conditions such that we *suspend*, *revise*, and *override* our assessment of the reliability of the algorithm, beyond the occurrence of SIS-Errors. Again, this is primarily an epistemic concern rather than a question about which methodologies could prevent the occurrence of SIS-Errors. Only very

tangentially will this article discuss how the epistemic conditions put forward here can be designed and implemented in an algorithm.

The strategy for this article is the following. First, I follow the literature in identifying sources of algorithmic errors. This is at the basis of SIS-Errors understood as misrepresentations that warrant a revoke of our epistemic attitude towards the reliability of the algorithm. I also discuss in detail SIS-Errors, what they are, how they occur, and why addressing them is crucial for reliabilist epistemologies of algorithms. All of this happens in Sections 2 and 3. I then map algorithmic errors with the reliability indicators advanced by CR. This is a key move, as it allows me to argue that SIS-Errors occur when a reliability indicator is *inadequate*, *incorrect*, or *missing*. This tripartite distinction is explained in Section 4. I then use these results to discuss a variety of epistemic conditions that, I submit, protect the reliability of an algorithm against the need to revoke our epistemic attitude in the presence of SIS-Errors. This happens in Sections 5.1 and 5.2. Finally, in Section 6, I present some further thoughts on the epistemological implications of SIS-Errors for any epistemology of algorithms.

2 When things go wrong

Algorithms sometimes fail. They miscompute, misrepresent, and deviate from their intended goals. Such algorithmic errors (henceforth 'errors') generate considerable epistemic anxiety. Pearl succinctly captures the challenge of dealing with these errors in the following way:

> Once you unleash it on large data, deep learning has its own dynamics, it does its own repair and its own optimization, and it gives you the right results most of the time. But when it doesn't, you don't have a clue about what went wrong and what should be fixed. In particular, you do not know if the fault is in the program, in the method, or because things have changed in the environment. (Pearl, 2019, 15)

Following Pearl, we can typify three classes of errors in algorithms. These are, $class_1$ errors, which occur when the algorithm produces incorrect results due to miscalculations. This could be due to issues like rounding errors, overflow, or underflow in numerical computations.[3] There are also $class_2$ errors, understood as errors that arise when the methods or techniques implemented in the algorithm are inherently flawed, or inappropriate for specific tasks. This includes issues with the models, metrics, or algorithmic techniques used (e.g., using an inappropriate sorting algorithm for a specific task), as well as the interpretation and implementation of core concepts (e.g., 'criminal/non-criminal' in the case of forensic AI discussed earlier). Finally, there are $class_3$ errors. These errors occur when changes in the context or environment render the algorithm and its outputs unsuitable, regardless of how well they performed in the

[3] It is unclear whether Pearl attributes these errors to the algorithm itself or to the computational process executing it. For example, a division-by-zero error may produce incorrect calculations, but its cause could stem from poor programming practices or bit flips induced by electromagnetic interference. The nature of these errors and the methods for addressing them differ (see, for instance, Primiero (2020); Pfleeger and Atlee (2009)). For simplicity, I will treat them as indistinguishable.

past. For example, a facial detection algorithm designed to identify individuals based on the curvature of their mouth will probably fail in the case of facial occlusion, such as when wearing masks.

It goes without much arguing that these errors do not necessarily occur independently of one another, thus making them difficult to isolate. The crucial point is, as suggested by Pearl, that either individually or collectively, the occurrence of these errors compromises the reliability of the algorithm. This is an intuitive and reasonable claim that conflicts with reliabilist commitments. Consider again forensic science, which increasingly relies on algorithms for facial and voice detection (Ruifrok et al., 2022). The Federal Rule of Evidence 702 (FRE 702) governs the admissibility of expert witness testimony, such as forensic experts in U.S. federal courts. Among other stipulations, FRE 702 mandates that the methods used by the expert "are based on a reliable, refutable scientific basis, that it has been verified and error rates are known, and that it is available for peer-review and publication" (Jacquet & Champod, 2022, 3). Imagine now an algorithm for facial detection that can place suspects at a crime scene. Suppose that this algorithm has a flawless history of accurately identifying suspects, with a high predictive accuracy. Under this heading, it is reasonable to assume that most forensic experts would, perhaps unknowingly, adopt a reliabilist epistemology. They are reliabilists precisely because the algorithm has a strong history of success in identifying the right suspect. When called to testify, these experts are confident that their algorithm meets FRE 702 requirements to an acceptable extent. As reliabilists, they must also accept that the algorithm might occasionally misidentify a suspect. This happens very rarely, and insofar as it does not shift the frequency of accurate identifications, there are no real reasons to revoke their assessment of the reliability of the algorithm. There is, after all, an acceptable degree of limited competence in their testimony: the algorithm is reliable to the best of their epistemic efforts and within their field's acceptable degree of certainty. Additionally, these experts count with an epistemic safety net: in forensic science, different sources determine the final ruling on a suspect—with forensic DNA being the holy grail. So, even in the presence of an egregious error, such as misidentifying a suspect, neither the algorithm's reliability nor the epistemic assessment of the expert needs to be revoked.

Problems arise when algorithms gain greater influence in determining the culpability of a suspect, particularly in cases where DNA evidence is unavailable (Carriquiry et al., 2019; Delgado et al., 2021). In such cases, the epistemic role of algorithms is somewhat shifted: they are no longer merely tools that assist experts in producing and communicating knowledge but are increasingly regarded as independent sources of knowledge themselves. We have seen this happen multiple times, as algorithms increasingly take over decision-making processes. And while some of these decisions may be harmless, egregious errors often lead to serious epistemic and moral consequences. This poses a challenge for reliabilism, as even a small number of errors—despite a high frequency of accurate output representations—can undermine claims about knowledge and justification. This is, in essence, Humphreys' critique of any reliabilist epistemology of algorithms. As he argues: "The quantitative success of an epistemic agent needs to be balanced with the severity of the errors that agent makes. If the errors are serious enough, they can undermine the belief that the agent truly knows what they are talking about" (Humphreys, 2020, 8:40). In this view, it

is no longer the frequency of accurate output representations that matters most, but also their quality. Humphreys encapsulates this idea succinctly by noting that statistically insignificant but serious errors can and do undermine otherwise high algorithmic reliability (Humphreys, 2020).

Does Humphreys present a compelling challenge to reliabilist epistemologies? I believe so, and this article is an attempt to defend reliabilism against SIS-errors. As noted in the introduction, the arguments put forward here are primarily epistemic. That is, they are a matter of how we maintain justification under the occurrence of SIS-Errors. We must note, however, a growing tradition in the philosophical literature that treats reliability as contingent upon design and implementation strategies that generate accurate representations. When approached this way, reliability is a methodological concern, and SIS-Errors can be resolved through various design practices. In a recent survey, Grote, Genin, and Sullivan (Grote et al., 2024) largely framed the ongoing debate on the reliability of algorithms in this way, fostering "an accessible introduction to key concepts in statistics and machine learning—as far as they are concerned with reliability" (Grote et al., 2024, 2). This framing allows the authors to unify what they claim to be technical issues of statistical learning theory with methodological concerns about the robustness of models and what they call 'socio-technical accounts of reliability'—the latter two being framed as distinct philosophical views on reliability. Buijsman has also put forward a similar argument: "[w]hen we have a belief-forming process that produces a certain output, how reliable is the process that produced it? That is the central question I am posing here, and where I've argued that the answer requires a method of determining the range of evaluation over which this reliability is determined" (Buijsman, 2024, 2655). A final example of this methodological orientation is Duede, who evaluates three distinct conceptions of reliability. Upon close inspection, one can discern a methodological emphasis underlying Duede's treatment of reliability. For instance, so-called 'instrumental reliability' depends on how abstract processes such as algorithms are designed and implemented (Duede, 2022, 491). In this article, however, I do not aim to examine the methodology for designing and implementing reliable algorithms. If there is any methodological foundation in what follows, it is the one inherited through the adoption of computational reliabilism (CR) as the primary epistemological framework.

3 Misrepresentations and SIS-Errors

Let me now discuss what SIS-Errors are, how they occur, and why they constitute a central problem for any reliabilist epistemology. I begin by using Humphrey's definition of accurate representations. Let us note that the use of 'instruments' in the definition below entails algorithms in the sense given above.

> Let F be a fact. Then an instrument I provides a basis for knowledge that F if and only if I contains an output representation R, R is an accurate representation of F or of a fact that entails F, and a reliable process forms the representation R, where a reliable representation-producing process is one that produces a high proportion of accurate output representations (Humphreys, 2020, 13:50)

Following this definition, a *misrepresentation* occurs when the output of an algorithm (R) does not match, to the degree permissible by the relevant community, an accurate representation of a fact (F) or a fact that entails F.[4] Now, an output R does not match an accurate representation of F when a reliable representation-producing process has failed to form said representation. In Humphreys' terms, this happens when an algorithmic error of some kind has occurred. To give a well-known example of this, consider the case of a misclassification of the husky, the wolf, and the snow (Ribeiro et al., 2016). The goal of the algorithm is to accurately classify whether a given picture is of a husky or of a wolf. As it is known, the algorithm creates and relies on irrelevant features for the classification (e.g., snow), instead of inherent features of the animals (e.g., fur, shape, eyes, etc.). As a result, it creates spurious correlations that lead to misrepresent a husky as a wolf with a snow background.

Now, according to Humphreys, if the misrepresentation is radical and egregious and the number of observed outputs is not large enough to shift the reliability of the algorithm, then that output misrepresentation is a statistically insignificant but serious error (SIS-Error) (Humphreys, 2020, 12:00).[5] When an SIS-Error occurs, and always, according to Humphreys, the algorithm can no longer be considered the provider of a basis for knowledge that F.

This reconstruction shows that there is a correspondence between algorithmic output misrepresentations (R), classes of errors in term of Pearl's categories, and representation-producing processes relevant for the formation of beliefs that R accurately represents F—or a fact that entails F. And all this seems right. To illustrate this three-fold correspondence, consider the Ariane 5 Flight 501 Failure. Approximately 37 seconds after launch, the vehicle's structural integrity failed due to extreme stresses caused by the deviation in its intended trajectory (i.e., R misrepresents F), triggering its own self-destructive mechanism to prevent harm to people or property on the ground. The investigation later showed that the Inertial Reference System (IRS) attempted to convert a 64-bit floating point number to a 16-bit signed integer. The floating-point value, representing the rocket's horizontal velocity relative to the launch pad, exceeded the maximum value that the 16-bit integer could store. This caused an arithmetic overflow (i.e., class_1 of Pearl's categories). The error arose from the reuse of modules from the Ariane 4 rocket without adequately verifying its compatibility with the Ariane 5's different flight dynamics (Agency, 1996). Formal verification of the algorithmic module would have detected the semantic mispresentation (i.e., formal verification is the representation-producing procedure (Fetzer, 1998)).

Why do SIS-Errors, such as the Ariane 5 case, constitute a central problem for any reliabilist epistemology? Because the IRS system worked properly in a number of past occasions—all those related to Ariane 4 as well as all testing on Ariane 5.

[4] Let me quickly note that the notion of "accuracy" should not be merely taken as gauging how well an algorithm prediction correspond to actual outcomes, but also in terms of the correctness, robustness, and overall reliability of the procedures employed that lead to that output. This is a common misconception about reliabilism.

[5] Humphreys provides no indications on how to delimit a non-SIS-Error from an actual SIS-Error. For simplicity, I assume that the relevant community can recognize these instances, and the primary purpose of this recognition is to evaluate whether they should revoke their assessment of an otherwise reliable algorithm. No moral, political, or other characteristics necessarily define or derive from these errors.

As the frequency-based reliabilist that Humphrey is, the system operating Ariane 5 was reliable. Statistically speaking, its disintegration during takeoff is an insignificant error. Epistemically speaking, we can no longer consider the algorithm controlling the onboard IRS to be reliable.

4 SIS-Errors and computational reliabilism

Computational reliabilism (CR) follows process reliabilism in that an algorithm's output is justified if it is produced by a reliable process[6] (Durán & Formanek, 2018; Durán, forthcoming). The focus is then on the consistency of producing, most of the time, accurate output representations.[7] As a reliabilist epistemology, CR contrasts with transparency-based approaches in that it does not justify belief in an algorithm's output by appeal to internal reasons or supporting evidence. It also departs from process reliabilism in the ways by which beliefs are formed. To CR, it is central to identify practices, metrics, methodologies, and research cultures that convey our best epistemic efforts to justify the belief that an algorithm's outputs represent, to the degree permissible by the relevant community, a fact in the world—or a fact entailed by it. These serve as indicators of good methodological, scientific, and social practices and are divided into three types of reliability indicators (RI): *type*$_1$*-RI: Technical performance of algorithms*, which "focuses on the specification, coding, execution, maintenance, and other technical features that contribute to the performance of the algorithm (e.g., high accuracy and low rate of errors, but also tolerance to domain change, repurposability, reusability, modularity, etc.)" (Durán, forthcoming); *type*$_2$*-RI: Computer-based scientific practice*, which "focuses on securing algorithmic-based scientific research [resulting from] the operationalization and implementation of scientific concepts, causal structures, models and theories, laws and law-like principles, taxonomies, but also scientific metaphors and intuitions, values (epistemic and otherwise), idealizations, abstractions, and representations" (Durán, forthcoming); and *type*$_3$*-RI: Social construction of reliability*, which "focuses on broader goals related to accepting—or rejecting—algorithms and their outputs by diverse communities (e.g., scientific, academic, the general public), the realization of intended values and goals, and the overall assessment of the algorithm's scientific merits [through] debates, experimentation and testing, replicability of results, and other forms of intellectual exchange." (Durán, forthcoming).

Each type-RI subsumes diverse token-RI, the latter understood as specific instances of type-RI. Examples of *token*$_1$-RI include verification and validation metrics, robustness analysis, and other practices that enhance precision and accuracy of the algorithm and its outputs. Examples of *token*$_2$-RI include the various ways in which scientific concepts and theories are interpreted and implemented in the algorithm. Examples

[6] It must be noted that CR holds that a process is broader than the algorithm *qua* logico-mathematical entity. It also encompasses a wider socio-techno-scientific context in which the algorithm is designed, used, and maintained.

[7] Strictly speaking, computational reliabilism refers to the algorithm's output as 'scientifically valid outputs', which includes but is not limited to accurate output representations. The reason is that CR constitutes a valid epistemology beyond the representationalist view.

of $token_3$-RI range from clinical trials to Participatory Technology Assessments.[8] Let us finally note that CR acknowledges that errors—such as rounding mistakes, division-by-zero, incomplete databases, or unskilled programming—can and do occur, revealing the inherent fragility of reliabilist epistemologies discussed in this article.

Now, the value of analyzing SIS-Errors with CR is that we can draw a seamless correspondence between misrepresentations, Pearl's classes of errors, and reliability indicators. Briefly, *program errors* fall under $class_1$ errors and are associated with $type_1$-RI, since they pertain to the performance of the algorithm. Misrepresentations due to miscomputations or other computer-based failures are of this kind. *Implementing a flawed method*, a $class_2$ error, is an instance of $type_2$-RI, as it involves scientific practices and domain knowledge embedded in the algorithm. Thus, misrepresentations stemming from wrongly implementing a scientific concept, for instance, are of this kind. Finally, *changes in the environment*, a $class_3$ error, are either instances of $type_1$-RI—such as changes in methods for data analysis—or of $type_3$-RI—such as shifts in scientific expectations regarding differences between training and testing contexts. Misrepresentations arising from changes in the domain of applicability, for example, fall into this category.

What advantages does this mapping offer? Two are of particular interest for this article. First, it connects algorithmic misrepresentations to types of reliability indicators. If SIS-Errors warrant revoking our initial assessment of an algorithm's reliability, then our best strategy is to trace the relationship between the error that led to the egregious misrepresentation and the RI that failed to confer reliability. As I will frame it here, SIS-Errors arise when one or more reliability indicators are *inadequate*, *incorrect*, or *missing*.[9] I will elaborate more on this point in the next section. Second, and at the risk of some repetition, SIS-Errors should be understood as an epistemological matter rather than a methodological one. The objective is to safeguard algorithmic reliability by drawing on epistemic conditions that enable us to suspend, revise, or override our beliefs in response to changing circumstances, new evidence, and emerging constraints. The focus, therefore, is not on prescribing optimal design or coding practices to prevent SIS-Errors—the standard approach in the literature—but on establishing conditions that significantly reduce the likelihood of having to revoke our assessment of the reliability of the algorithm when SIS-Errors occur.

[8] Participatory Technology Assessments engage diverse stakeholders—such as citizens, policymakers, scientists, industry representatives, and others—in the evaluation and decision-making processes surrounding the development, deployment, and regulation of new technologies.

[9] An anonymous reviewer correctly pointed out that non-SIS-Errors may also result from inadequate, incorrect, or missing reliability indicators (see next section). This suggests that non-SIS-Errors, like SIS-Errors, signal a failure to meet the criteria for reliability, which ultimately implies the unreliability of the algorithm in question. In other words, any form of error implies some degree of unreliability of the algorithm. While this is a point that warrants careful debate, I agree in principle with the reviewer's concern. Intuitively, it seems uncontroversial to hold that the reliability of an algorithm is, so to speak, readjusted in light of the number of accurate representations and misrepresentations it produces. Let us recall that CR allows for a level of tolerance toward non-SIS-Errors, provided that their frequency does not shift their reliability. Under CR, then, it is legitimate to hold that non-SIS-Errors are either absorbed by the algorithm's prior reliability or, following the logic developed in the case of SIS-Errors, addressed through one or more of the epistemic conditions proposed here.

4.1 Inadequate, incorrect, and missing reliability indicators

Let me now further motivate the value of the mapping introduced earlier. To this end, consider an otherwise reliable facial detection algorithm. Its reliability derives, *inter alia*, from the implementation of industry standards for bio-indicators in facial recognition, the maintenance of a well-curated database (e.g., biometric passport photos), and the exclusion of cases involving underexposed or overexposed photographs. The algorithm remains reliable as long as the conditions of its use remain consistent with its design specifications. Now, suppose we introduce facial occlusions, such as surgical masks during the COVID-19 pandemic outbreak. While the algorithm may continue to function adequately in many cases, there is a significant likelihood that it will misrepresent some individuals (Ekenel & Stiefelhagen, 2009). After all, facial occlusion falls outside its original design parameters. Suppose further that some of these misrepresentations qualify as SIS-Errors. As Humphreys argues, under such conditions, we can no longer maintain that the algorithm forms beliefs appropriately, regardless of its prior accuracy in facial detection. The changing conditions warrant an epistemic revocation of the algorithm's reliability. Under these conditions, one or more RI is *inadequate* for belief formation—inadequate in the sense that it confers reliability only under specific, circumscribed circumstances.

If these changing conditions persist over time or become permanent, the algorithm can no longer be considered reliable. In such cases, we must seriously entertain the possibility that *incorrect* RI are being used to confer reliability to the algorithm. To illustrate what I mean by an incorrect RI, consider a somewhat extreme example of facial detection: the use of a convolutional neural network (CNN) to identify suspected criminals based on facial traits (Wu & Zhang, 2016, 2017). This CNN was claimed to be highly reliable, as it demonstrated a predictive accuracy of approximately 95%, even after undergoing retraining on every layer and modifications to its architecture (Wu & Zhang, 2017, 3). The epistemic confidence in the algorithm was set so high that the authors claimed to have discovered the "law of normality for faces of non-criminals" (Wu & Zhang, 2016, 8). However, the accuracy of the algorithm was almost exclusively derived from matching facial traits between the training and testing databases. No additional factors contributed to it. No model or concept of criminality was implemented in the algorithm, nor did scientific debates follow these findings. Lacking any connection to a body of scientific knowledge or practices, it is difficult to sustain claims about the algorithm's reliability. Yet, its predictive accuracy remained surprisingly high. Under this heading, it should not be difficult to see that algorithms of this kind are highly prone to SIS-Errors. The reason is that their reliability is conferred by an incorrect reliability indicator—namely, type$_1$-RI— which only supports high accuracy and a low margin of error, rather than type$_2$-RI, which involves the interpretation and implementation of concepts, models, theories that track the sources of criminality (e.g., social, economic, political, psychological).

Finally, SIS-Errors might also occur when key RI are *missing*. This was the case of NarxCare (Bamboo Health, 2023), where a patient (Kathryn) was incorrectly flagged as a drug user and shopper (Pozzi, 2023). The SIS-Error occurred because, in assessing the reliability of the algorithm, no consideration was given to the fact that prescription

drugs for the patient's pets were also listed under the owner's name. To restate the point counterfactually, and thus clarify the sense in which a reliability indicator was missing: had the algorithm's reliability been conferred by an indicator sensitive to the nuances of drug assignment across individuals,[10] no such SIS-Error would have arisen.

5 Epistemic scaffolding

With these ideas in mind, let me now demonstrate how CR, as a reliabilist epistemology, can maintain its status as an adequate epistemology of algorithms even in the presence of SIS-Errors. To this end, it is useful to distinguish between *random* SIS-Errors and *systematic* SIS-Errors. While this distinction does not alter anything discussed thus far, it allows me to exclude arbitrary cases of SIS-Errors and better safeguard CR. The real challenge, as we shall see, arises when the algorithm produces systematic errors.

Before I start, let me point out the obvious: the epistemic conditions outlined here are neither exclusive nor exhaustive, as additional epistemic constraints may be identified, and those proposed here may also apply to different cases. It is the urgency of this issue that requires an approach of comparable complexity—and inevitable flaws—as the one presented here.

5.1 SIS-Errors and random errors

Random errors are arbitrary, unpredictable, unreproducible, and typically rare faulty executions of the algorithm, its instantiations (input variables, parameters, data choices, procedures, metaparameters, etc.), or its data processing. These errors are neither systematic nor consistent; they occur sporadically and are often difficult to reproduce or find its source. Random error could result from selecting a row of data with missing entries, encountering race conditions in a multi-threaded algorithm, or experiencing floating-point miscalculations that yield different outputs across runs.

Random SIS-Errors are arbitrary, egregious misrepresentations generated by an otherwise reliable algorithm. They simply occur. Yet whether such errors undermine the reliability of the algorithm remains an open epistemic question. Does the mere randomness of an error warrant revoking our assessment of the system's reliability? To address this, I introduce the notion of *epistemic bad luck*, a novel member of the epistemic luck family. The central idea is that we have been unfortunate in encountering an SIS-Error when, in fact, there is nothing wrong with the reliability of the algorithm. Thus, we can maintain the reliability assessment of the algorithm if a given instance of SIS-Error can be reasonably identified as the product of a random, anomalous execution—ultimately, an instance of epistemic misfortune, rather than epistemic failure.

[10] For instance, design provenance (i.e., information about how, why, and for whom the algorithm was constructed), epistemic stratification (e.g., monitoring whether the algorithm preserves relevant epistemic distinctions, such as different individuals taking different drugs), or domain-specific validity checks.

As we know, cases of epistemic luck occur when one's beliefs are true not because they were formed through a reliable or justified process, but rather due to mere chance (Zagzebski, 1994; Pritchard, 2005). Gettier cases are examples of this: they involve belief-forming processes that are ultimately unreliable, resulting in beliefs that are true by sheer luck. A simple example of epistemic luck is a student who guesses the correct answer to a math problem without understanding its underlying principles or operations.

To frame epistemic luck in the context of algorithms, it corresponds to an output that accurately represents *F* despite being generated by a randomly faulty computation (e.g., a one-time integer overflow). Understood this way, cases of epistemic luck are not problematic for assessing the reliability of an algorithm, since they still yield outputs that represent. Whether or not the output was produced by a faulty process is, from the standpoint of reliabilism, largely irrelevant. An output that represents a fact in the world accurately—even if by sheer luck—is, by definition, not an error, and therefore does not qualify as an SIS-Error.

In contrast to traditional epistemic luck, which concerns accidentally true beliefs, *epistemic bad luck* refers to cases where false beliefs are formed by processes that temporarily lack justificatory force. To reinterpret the earlier example in this new context: the student guesses the wrong answer on a math exam, despite generally understanding the underlying principles and operations. For that particular instance, however, their guess fails to track the truth. This is not a case of knowledge gone wrong—it is simply a temporary justificatory disruption. Understandably, epistemic bad luck holds little interest for traditional analytic epistemology, in which no claims to knowledge arise from unjustified false beliefs. But it becomes a central issue in computational reliabilism, where the reliability of an algorithm must account for randomly occurring errors that generate (temporary) misrepresentations.

Epistemic bad luck, then, involves a misrepresentation generated by a randomly faulty algorithm. If the misrepresentation is particularly egregious, it qualifies as an SIS-Error and thus warrants revoking our assessment of the algorithm's reliability. However, the reliability of the algorithm crucially depends on whether such an error is the result of a random failure or a systematic flaw—for it could very well be a case of epistemic bad luck.

To address such cases, I propose an *anti-epistemic bad luck* conditioning to supplement computational reliabilism and maintain the reliability of the algorithm. This condition holds that a belief formed on the basis of an algorithmic output is epistemically warranted only if the probability that an SIS-Error recurs independently is negligible relative to the prior probability that the output is correct due to non-random, calculation-preserving processes. By discounting reliability assessments grounded in coincidental error agreement, this condition helps filter out outputs that misrepresent due to bad luck rather than genuine epistemic reliability. Consider the following formulation:

> **Anti-epistemic bad luck condition:** Let an algorithm be executed at least twice under similar conditions (i.e., similar input variables, parameters, data configurations, metaparameters, etc.). If the algorithm produces identical SIS-Errors across independent executions, then the probability that these outputs are

instances of epistemic bad luck is negligibly small. Therefore, such recurrence provides strong evidence that the error is systematic rather than random.[11]

The anti-epistemic bad luck clause intends to shift the focus from SIS-Errors to evaluating whether an algorithm's output has been legitimately generated within a reliabilist framework. If the output qualifies as an SIS-Error in two or more identical yet independent executions of the algorithm, we can reasonably conjecture that the same egregious misrepresentation has been reproduced. Such a result strongly suggests that the likelihood of encountering cases of epistemic bad luck twice under identical conditions is no longer astronomically small but instead points to a more systematic issue. Consequently, the anti-epistemic bad luck clause indicates that the SIS-Error in question may not be a mere anomaly but rather a symptom of a faulty algorithm, thus necessitating a distinct approach to diagnosis and resolution.

5.2 SIS-Errors and systematic errors

Systematic errors are non-arbitrary and potentially reproducible faulty executions of the algorithm, its instantiations (input variables, parameters, data choices, procedures, metaparameters, etc.), or its data processing. These errors are not random and, with the right tools, they can be identified and quantified. In this context, the occurrence of an SIS-Error resulting from a systematic failure could stem from a number of sources: invalid assumptions (e.g., misconfigured parameters), flawed logic, or biases inherent in the algorithm or data, just to mention a few. For instance, if an algorithm utilizes a programming language that truncates numbers larger than 256 bits, the total number of possible representations is, therefore, 2^{256}. Any representation beyond this value will be truncated or rounded-off. Occasionally, these truncation and rounding-off errors might be harmless. However, if high accuracy is required, these errors will consistently affect the outputs. Another example stems from assumptions made during an algorithm's specification and coding. If the algorithm misses that sex distinction is relevant for medical diagnosis, for instance, female-specific cancers and autoimmune diseases will not be represented properly.

Systematic errors are pervasive in algorithmic applications, and a great deal of effort is geared towards minimizing them. We briefly saw in Section 2 that this is the preferred tactic of both advocates and critics of reliabilism. Recall also from Pearl the range of errors that can occur: from the computation of the algorithm ($class_1$ *errors*) to poor programming skills ($class_2$ *errors*) to changes in the context of applicability of the algorithm ($class_3$ *errors*). Individually or jointly, systematic errors of these kinds can lead to SIS-Errors in Humphrey's sense. Indeed, as I have presented it thus far,

[11] An anonymous reviewer correctly observes that it is not always possible—or rational—to rerun an algorithm under similar conditions. The example is the Ariane 5 failure on page 7, where the reliability of the system was compromised by a single instance of misrepresentation. As the reviewer notes: "Even if this were to be considered an instance of epistemic bad luck, it seems inappropriate to dismiss the case solely on the basis that the error was random." I take it that such situations are better understood as cases of *incorrect reliability indicators* and should therefore invoke *the anti-defeat clause*, which requires the identification of a defeater RI that accounts for an Inertial Reference System capable of handling 64-bit floating-point numbers. I thank the reviewer for this insightful observation.

when an egregious misrepresentation of an otherwise reliable algorithm occurs, we have an SIS-Error. According to Humphreys, the occurrence of SIS-Errors warrants revoking our assessment: where we once had a reliable algorithm, we now no longer have justification. How can a reliabilist epistemology of algorithms, such as CR, deal with systematic SIS-Errors? Earlier, I argued that our epistemic attitude regarding SIS-Errors is contingent on having *inadequate*, *incorrect*, or *missing* reliability indicators. In what follows, I advance epistemic conditions *vis-à-vis* these failing reliability indicators.

As presented earlier, an *inadequate reliability indicator* is one that fails to accurately represent a fact in the world under changing conditions. The example used was a reliable facial detection algorithm that implements standard bio-facial markers. However, when used in contexts where facial occlusion occurs, the algorithm might misidentify individuals.

A counteracting measure to avoid shifts in the reliability of an algorithm is to set up a *conditionally reliable RI clause* that tracks its reliability across varying contexts (Goldman, 1986; Alston, 1995). In this way, it is possible to protect CR against failures due to changing conditions.[12] To present this idea more formally,

> **Conditionally reliable token-RI** is one that confers reliability [to the new context] if the methods, standards, and breath of application is based on are also reliable [for the new context]

Conditional reliability hinges on the dependability of a token-RI within a specific context of operation of the algorithm. It tracks representation by assessing how likely it is that the token-RI maintains reliability across contexts. Suppose the algorithm implements method $\mathcal{M}$, a biometric procedure that extracts facial features to create a unique facial signature. This process includes measuring distances between key facial landmarks, such as the eyes, nose, mouth, and jawline. Say further that $\mathcal{M}$ is based on the Viola-Jones algorithm for object detection, a widely accepted approach in the relevant communities (Viola & Jones, 2001). Thus, the token$_2$-RI = {the detection of faces occurs using method $\mathcal{M}$, and no facial occlusion is permissible} is conditionally reliable with respect to the metrics, standards, and breadth of application recognized by the relevant community.

Now, suppose the algorithm is employed in scenarios involving facial occlusion. In such cases, misrepresentations are to be expected. If the algorithm is applied in sensitive contexts—such as border security—this increases the likelihood of SIS-Errors. Conditional reliability, then, permits us to uphold our beliefs insofar as the initial conditions that conferred reliability to the algorithm still apply in the new context. That is, method $\mathcal{M}$ that tracks output representation in the absence of facial occlusion must remain operational. When these conditions are not met—such as in changing circumstances where individuals wear face masks while attempting to cross border security—we are advised to revoke our assessment toward the algorithm's reliability.

[12] Changing circumstances cover a broad range of cases. The example above covers Pearl's class$_3$ errors. Additionally, we can consider class$_2$ errors—for instance, when a concept implemented in the algorithm is later reinterpreted, potentially expanding its scope. In such cases, the initial interpretation becomes invalid.

What conditional reliability allows us to do is to suspend, rather than revoke, our assessment until the initial conditions under which the algorithm is reliable are reestablished, or until the algorithm is modified in a way that restores justification. In other words, we do not simply discard a reliable algorithm because it performs inadequately under altered conditions.

Let us note that conditional reliability aligns well with our intuitions about what it means to be reliable: our visual perception of a barn might be reliable on a clear day, but not necessarily so on a foggy one. We are, indeed, entitled to suspend our assessment until the initial conditions for the reliability of our visual perception are restored—that is, the day clears. Interestingly, CR also acknowledges that an algorithm's reliability is context-dependent (Durán, forthcoming).

As suggested, conditional reliability offers a way forward for the methodologist, too. When the RI adapts to new conditions, the reliability of the algorithm can, so to speak, be resumed. This occurs when an extended version, say token$_2$-RI* = {the detection of faces occurs through method $\mathcal{N}$, which covers all cases addressed by method $\mathcal{M}$ plus facial occlusion}, is implemented. Token$_2$-RI* is resistant to contexts of facial occlusion, and is therefore a net positive contributor to the reliability of the algorithm.

An inadequate RI typically emerges within a temporary and limited context. But as Humphreys rightly observed, algorithms produce SIS-errors irrespective of contextual variation, and these errors tend to persist over extended periods. Pearl typified these errors as those where a flawed method is implemented (*class$_2$ error*); under CR, these are interpreted as instances of an *incorrect token-RI*. To addressed such cases, I propose an *anti-defeat clause*, where the justificational status of a belief is preserved as long as the RI remains resilient against new, potentially undermining alternatives. This translates into testing whether CR is conferring reliability through an incorrect reliability indicator, and if that were the case, enforcing an epistemic obligation to accept the alternative (i.e., the defeater). More formally,

> **Anti-defeat clause:** *S* is epistemically warranted in maintaining token-RI as a suitable reliability indicator unless there is a defeater token-RI* epistemically available to *S* such that, if *S* were to use token-RI*, *S* would no longer hold the belief that the output represents a fact *F*.

In epistemology, an anti-defeat clause plays various roles, such as preserving coherence within a justified belief system and guiding the integration of new knowledge (e.g., Lehrer and Paxson, 1969; Dretske, 1981; Sosa, 2007). In the context of computational reliabilism, I use it to preserve the epistemic status of token-RIs against potential defeaters. This clause ensures that a belief based on an algorithmic output remains epistemically warranted as long as the relevant token-RI remains resilient in the face of conflicting evidence, reasons, or challenges. If a defeater token-RI* becomes epistemically available to *S*, and if *S*'s evaluation shows that token-RI* better accounts for reliability than the original token-RI, then *S* is no longer epistemically warranted in maintaining the belief. In such cases, *S* is epistemically obliged to revise—and possibly accept—the defeater reliability indicator. In this sense, the anti-defeat clause thus functions to regulate when beliefs formed on the basis of computational reliabilism

retain their epistemic standing, conditional on the undefeated status of their underlying reliability indicators.

To grasp the anti-defeat clause, consider an agent S who relies on a token$_2$-RI = {The metrics utilized for identifying a photo as {criminal, non-criminal} are: the distance between the eyes is α, the curvature of the mouth is β, and the curvature of the nose is δ} as a basis for conferring reliability on a facial-detection algorithm (e.g., Wu and Zhang, 2016; Wu and Zhang, 2017). Suppose further that the algorithm misclassifies the fifth photo of a suspect as belonging to a criminal (i.e., SIS-Error). The anti-defeat clause stipulates that S is epistemically warranted in maintaining token$_2$-RI unless an epistemically available defeater—token$_2$-RI*—undermines that warrant by offering a more robust reliability indicator. For instance, such a defeater might take the form: token$_2$-RI* = {The metrics utilized for identifying a photo as {criminal, non-criminal} must exclude facial traits and include metrics that align with social, psychological, economic, or other theories of criminality} Akers and Sellers 2012. As per the hypothesis, token$_2$-RI* provides a more epistemically robust basis for maintaining the reliability of the algorithm by aligning better with independently justified theories of criminal behavior and avoiding biases inherent in facial trait analysis. S is then epistemically obligated to revise their epistemic attitude towards maintaining token$_2$-RI and, if token$_2$-RI is defeated, adopt token$_2$-RI* as the new reliability indicator for the algorithm. The anti-defeat clause thus ensures, to the extent possible, that epistemic warrant for beliefs formed on the basis of algorithmic outputs is preserved, conditional on the undefeated status of the token-RIs used to justify them.

It is crucial to recognize that replacing token$_2$-RI with the defeater token$_2$-RI* is likely to have cascading effects on the overall reliability assessment of the algorithm. That is, accepting the defeater may influence previously justified beliefs, potentially shifting our assessment of the algorithm's reliability again. For example, adopting the defeater token$_2$-RI* to avoid revoking reliability in response to SIS-Error$_1$ might inadvertently give rise to a new SIS-Error$_2$. While such a scenario is plausible, the anti-defeat clause does not require that the adoption of token$_2$-RI* extend beyond addressing the original SIS-Error$_1$. Nor does it mandate revision of past beliefs justified under the original reliability indicator. Potential issues arising from cascading defeaters can be addressed by introducing a further *defeater-defeater clause*, which stipulates that token$_2$-RI* must not itself be undermined by a further reliability indicator (say, token$_2$-RI**). This latter condition also helps prevent an infinite regress of defeaters and stabilizes the structure of reliability assessments.

The superstructure of CR consists of three types-RI that fairly cover the performance of the algorithm, the implementation of scientific methods and concepts, and the social debates that follow the generation of outputs. As discussed thus far, inadequate and incorrect reliability indicators are largely addressed by type$_1$-RI and type$_2$-RI. This means that SIS-errors are treated as technical errors: errors in performance or errors in implementation, as typified by Pearl's *class*$_1$ and *class*$_2$ errors, respectively.

However, it is conceivable that SIS-Errors also occur when there are *missing* RI altogether. To illustrate what I mean by this, recall the case of NarxCare (Bamboo Health, 2023), where Kathryn, the patient, was wrongly flagged as a drug user and shopper (Pozzi, 2023). The reason was that the algorithm did not cover cases where

the pet's drug prescriptions were under the owner's name. Surely, it was reliable for all other cases, but not for Kathryn. This is the reason why I treat such situations as cases of missing reliability indicators. To be clear, it was the absence of an RI that enabled the occurrence of an SIS-Error, not the presence of an inadequate or incorrect one.

Missing reliability indicators are difficult to address from a technical perspective, mainly because they entail some form of epistemic ignorance. This is why the social construction of reliabilism, as advanced by type$_3$-RI, is crucial. As I have argued elsewhere, this indicator "focuses on broader goals related to accepting—or rejecting—algorithms and their outputs by diverse communities (e.g., scientific, academic, the general public), the realization of intended values and goals, and the overall assessment of the algorithm's scientific merits. This occurs through debate, experimenting and testing, replicability of results, and other forms of intellectual exchange" (Durán, forthcoming). The suggestion on how to deal with cases of missing RI, then, is to 'supercharge' this indicator with the epistemic responsibility of detecting and dealing with SIS-Errors at a social level. A somewhat formal description would be:

> **Supercharging type$_3$-RI**: Maintaining algorithmic reliability depends on subjecting their outputs to debate and other forms of scientific engagement. In this sense, the social construction of belief plays a crucial role in determining reliability and can, at times, take precedence over other indicators

At its core, supercharging type$_3$-RI functions as an 'epistemic precautionary principle,' where the relevant community evaluates the merits of an algorithm's output and may override any prior epistemic stance on its reliability. In the case of NarxCare, the SIS-Error also stemmed from physicians failing to cross-check a patient's medical history, and possibly debate on the algorithm scientific merits.[13] Supercharging type$_3$-RI enables the retention of the reliability of NarxCare when social and scientific debates challenge SIS-Errors—again, by cross-checking Kathryn's medical history, for instance. In other words, revoking our epistemic assessment of NarxCare can be overridden if the relevant community confirms the algorithm's reliability.

While we may have identified a viable approach to addressing SIS-Errors arising from missing RI, we inevitably inherit the complexities inherent in any social and scientific debate—ranging from conflicts to partial solutions. For this reason, enhancing type$_3$-RI represents a partial return to a human-centric epistemology. This approach remains consistent with Humphreys' "hybrid scenario," which envisions humans and machines interacting in a way that challenges reliabilist epistemologies (Humphreys, 2009, 2021).

[13] Indeed, according to Szalavitz (2020), this was the primary reason for the algorithm's failure. Of course, there is a legitimate concern that cross-checking *every* output would undermine the very purpose of using algorithms. Supercharging type$_3$-RI should not be interpreted as advocating for a similar approach, but rather as a strategy applicable to critical cases. For instance, if Kathryn is flagged as a drug shopper, that specific output warrants further scrutiny.

6 Final thoughts

Humphreys argues that SIS-errors pose a fundamental challenge to any reliabilist epistemology of algorithms. I believe this to be correct. But it also strikes me as a general challenge to any epistemology of algorithms, not just reliabilism. In this article, I defend computational reliabilism (CR) by outlining conditions designed to preserve an algorithm's reliability. Will these conditions cover every real or hypothetical instance of SIS-errors? Probably not. Additional refinements to CR—or alternative epistemological frameworks—may be necessary. No epistemology is flawless, and much of this article navigates uncharted territory.

Let me offer a final thought. Humphreys appears to suggest that the occurrence of a single SIS-Error is sufficient to justify revoking our assessment of an algorithm's reliability. I worry that this line of thinking may do more harm than good. Algorithmic errors are inevitable, and some will qualify as SIS-Errors. Does Humphreys seek complete epistemic warrants? Is he demanding absolute certainty in the algorithm's reliability? In principle, this does not appear to be the case. Yet there is an expectation that extends beyond what any epistemology can reasonably provide. Moreover, in some well-defined contexts, the occurrence of one or more SIS-Errors does not necessarily warrant revoking our assessments of the reliability of the algorithm. This is a point I did not address in the article, but it is worth considering briefly. Take the case of BenevolentAI, which identified baricitinib—a rheumatoid arthritis drug—as a potential treatment for COVID-19 by inhibiting JAK-STAT signaling pathways, thereby reducing interferon-mediated antiviral responses (Favalli et al., 2020, 1013). This discovery proved highly effective in alleviating COVID-19 symptoms, making it a timely breakthrough during the pandemic. However, inhibiting interferon also increases susceptibility to other viruses, such as herpes zoster and herpes simplex, which can be particularly harmful to immunodeficient patients (Favalli et al., 2020). Should we revoke our epistemic assessment of BenevolentAI's reliability in light of these adverse effects? I do not believe so. If an algorithm's reliability depends on both the quality and quantity of its outputs, then an SIS-Error in one context may not qualify as one in another. Indeed, not all SIS-Errors are qualitatively equivalent. The reliability of BenevolentAI appears to remain intact for all cases excluding immunodeficient patients. In fact, it remains reliable across the full range of baricitinib's known side effects, including for patients with cardiovascular disease, infections, or a history of clotting disorders (Taylor et al., 2017; Agency, 2021). How do we preserve the reliability of algorithms? As suggested at the end of the previous section, this requires a return to an algorithm/human-centric epistemology—one capable of discerning complex cases such as these. This article defends computational reliabilism as exactly such an epistemology.

Acknowledgements This article is deeply indebted to Emanuele Ratti, who read and supported the views expressed here and tirelessly encouraged me to pursue them. I am also grateful to Emma-Jane Spencer for her sustained—and at times unjustified—support. Your friendships cut through so much noise. Thank you both.

I would also like to thank the two anonymous reviewers, as well as the editor of the special issue, for their incisive comments and their insistence on key issues that significantly improved the article.

References

Agency, E. M. (2021). Olumiant (baricitinib): Summary of product characteristics (smpc). Available at: https://www.ema.europa.eu.

Agency, E. S. (1996). Ariane 501 - presentation of inquiry board report. Accessed: 2024-08-03.

Akers, R. L., & Sellers, C. S. (2012). *Criminological Theories: Introduction, Evaluation, and Application* (6 ed.). Oxford University Press.

Alston, W. P. (1995). How to think about reliability. *Philosophical Topics, 23*(1), 1–29. https://doi.org/10.5840/philtopics19952311

Bamboo Health (2023). Narxcare and patients. Retrieved December 9, 2022, https://bamboohealth.com/narxcare-and-patients/. Technical report.

Beisbart, C. (2021). Opacity thought through: on the intransparency of computer simulations. *Synthese, 199*, 11643–11666.

Boge, F. J. (2022). Two dimensions of opacity and the deep learning predicament. *Minds and Machines, 32*(1), 43–75.

Buijsman, S. (2024). Over what range should reliabilists measure reliability? *Erkenntnis, 89*(7), 2641–2661. https://doi.org/10.1007/s10670-022-00645-4

Carriquiry, A., Hofmann, H., Tai, X. H., & VanderPlas, S. (2019). Machine learning in forensic applications. *Significance, 16*(2), 29–35. https://doi.org/10.1111/j.1740-9713.2019.01252.x

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

Delgado, Y., Price, B. S., Speaker, P. J., & Stoiloff, S. L. (2021). Forensic intelligence: Data analytics as the bridge between forensic science and investigation. *Forensic Science International: Synergy, 3*, Article 100162. https://doi.org/10.1016/j.fsisyn.2021.100162

Dretske, F. (1981). *Knowledge and the Flow of Information*. MIT Press.

Duede, E. (2022). Instruments, agents, and artificial intelligence: novel epistemic categories of reliability. *Synthese, 200*(6), 491. https://doi.org/10.1007/s11229-022-03975-6

Durán, J. M., & Formanek, N. (2018). Grounds for trust: Essential epistemic opacity and computational reliabilism. *Minds and Machines, 28*(4), 645–666.

Durán, J. M. (2021). Dissecting scientific explanation in AI (sXAI): A case for medicine and healthcare. *Artificial Intelligence, 297*, 103498.

Durán, J. M. (forthcoming). Beyond transparency: computational reliabilism as an externalist epistemology of algorithms, In Philosophy of Science for Machine Learning: Core Issues and New Perspectives, Durán, J.M., & Pozzi, G. (eds.). Synthese Library.

Ekenel, H. K., & Stiefelhagen, R. (2009). Why is facial occlusion a challenging problem? In M. Tistarelli & M. S. Nixon (Eds.), *Advances in Biometrics, Berlin, Heidelberg* (pp. 299–308). Berlin Heidelberg: Springer.

Favalli, E. G., Biggioggero, M., Maioli, G., & Caporali, R. (2020). Baricitinib for covid-19: a suitable treatment? *The Lancet, 20*, 1012–1013.

Fetzer, J. H. (1998). Program verification: The very idea. *Communications of the ACM, 37*(9), 1048–1063.

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

Goldman, A. (1979). What is justified belief? *The Justification of Belief*, 1–23.

Goldman, A. I. (1986). *Epistemology and Cognition*. Cambridge, MA: Harvard University Press.

Grote, T., Genin, K., & Sullivan, E. (2024). Reliability in machine learning. Philosophy. *Compass, 19*(5), e12974. https://doi.org/10.1111/phc3.12974

Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2019). A survey of methods for explaining black box models. *ACM Computing Surverys, 51*(5), 1–42.
Humphreys, P. (2020). Neural nets: Why reliabilism is an inappropriate epistemology for them. YouTube video. Accessed: 2024-08-15.
Humphreys, P. (2021). Epistemic opacity and epistemic inaccessibility. *Pre-Print*.
Humphreys, P. W. (2004). *Extending Ourselves: Computational Science, Empiricism, and Scientific Method*. Oxford University Press.
Humphreys, P. W. (2009). The philosophical novelty of computer simulation methods. *Synthese, 169*(3), 615–626.
Jacquet, M., & Champod, C. (2022). Automated face recognition in forensic science: Review and perspectives. *Forensic Science International, 325*, 110851.
Kroll, J. A., Huey, J., Barocas, S., Felten, E. W., Reidenberg, J. R., Robinson, D. G., & Yu, H. (2017). Accountable algorithms. *University of Pennsylvania Law Review, 165*(3), 633–705.
Lehrer, K., & Paxson, T. (1969). Knowledge: Undefeated justified true belief. *The Journal of Philosophy, 66*(8), 225–237.
Pearl, J. (2019). The limitations of opaque learning machines, In Possible Minds: 25 Ways of Looking at AI, Brockman, J. (ed.) Penguin Books. Chapter 2.
Páez, A. (2023). Algorithmic bias, algorithmic discrimination, and the ethical role of computer scientists. *Minds and Machines, 33*(1), 37–58. https://doi.org/10.1007/s11023-023-09616-5
Pfleeger, S. L., & Atlee, J. M. (2009). Software Engineering: Theory and Practice (4th ed.). Pearson.
Pozzi, G. (2023). Automated opioid risk scores: a case for machine learning-induced epistemic injustice in healthcare. *Ethics and Information Technology, 25*(1), 3. https://doi.org/10.1007/s10676-023-09676-z
Primiero, G. (2020). *On the Foundations of Computing*. Oxford University Press.
Pritchard, D. (2005). *Epistemic Luck*. Oxford: Oxford University Press.
Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "why should i trust you?": Explaining the predictions of any classifier. KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining: 1135–1144.
Ruifrok, A., Vergeer, P., & Rodrigues, A. M. (2022). From facial images of different quality to score based lr. *Forensic Science International, 332*, 111201. https://doi.org/10.1016/j.forsciint.2022.111201
Sosa, E. (2007). A Virtue Epistemology: Apt Belief and Reflective Knowledge, Volume I. Oxford University Press.
Sullivan, E. (2022). Understanding from machine learning models. *British Journal for the Philosophy of Science, 73*(1), 109–133. https://doi.org/10.1093/bjps/axz035
Szalavitz, M. (2020). The pain was unbearable. so why did doctors turn her away? *Wired*.
Taylor, P. C., Takeuchi, T., & Burmester, G. R. (2017). Safety of baricitinib in patients with active rheumatoid arthritis: an integrated analysis of clinical trial data. *Annals of the Rheumatic Diseases, 76*(5), 899–907. https://doi.org/10.1136/annrheumdis-2016-210457
Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Volume 1, pp. I–511.
Wachter, S., Mittelstadt, B., & Russell, C. (2018). Counterfactual explanations without opening the black box: automated decisions and the gdpr. *Harvard Journal of Law and Technology, 31*(2), 841–887.
Wu, X., & Zhang, X. (2016). Automated inference on criminality using face images.
Wu, X., & Zhang, X. (2017). Responses to critiques on machine learning of criminality perceptions (addendum of arxiv:1611.04135).
Zagzebski, L. (1994). The inescapability of gettier problems. *The Philosophical Quarterly, 44*(174), 65–73. https://doi.org/10.2307/2220147
Zerilli, J. (2022). Explaining machine learning decisions. *Philosophy of Science, 89*(1), 1–19. https://doi.org/10.1017/psa.2021.13