When algorithms go wrong, who is liable?

As automated decisions affect more and different areas of our lives, we are faced with ethical and legal questions that are likely to change the way we think about algorithms and the law. By Sofia Olhede and Patrick Wolfe

If you find yourself suspected of a crime in Durham, a city in northeast England, there is every chance that the decision on whether police keep you in custody will be informed by artificial intelligence (AI). The system has been trained using five years of data on offending histories, followed by two years of live tests, in which suspects were categorised as at low, medium or high risk of offending. These predictions were then compared to actual outcomes. "Forecasts that a suspect was low risk turned out to be accurate 98% of the time, while forecasts that they were high risk were accurate 88% of the time," reported the BBC (bit.ly/2yUathA). Not bad, you might think. But there are occasions where the algorithm made the wrong call – and that might mean holding someone in custody unnecessarily.

Who is liable for these "wrong" decisions? Should algorithms be held to different legal standards than human decision-makers? And should an algorithm be able to explain itself when it makes a mistake? These are some of the questions to be asked about the increasing involvement of algorithms in the decision-making process, in whatever setting – be it legal, financial, medical or otherwise.

Decision factors

Many factors impact an algorithm's decision-making ability: from the specific algorithm that was chosen, to the data set that "trained" the algorithm, to the decision-making criteria that were used. Moreover, there are many ways to define a good decision, ranging from a focus on "average performance" to "minimax" – in which the worst possible algorithmic performance (a miscarriage of justice, say) is limited, rather than most decisions being fair on average. The variability in choice of algorithm, training data and criteria makes the notion of liability (criminal or otherwise) hard to tackle. What constitutes negligent behaviour, and what is an "algorithmic act of God"? To explore this issue, first let us talk data – specifically the fact that any analysis algorithm, and its subsequent decisions, are dependent on the data set that was used to train it. The oft repeated example of Google Flu Trends shows the vulnerability of automated analysis when data variation is overlooked, and how this can lead (in Google's case) to a catastrophic overestimate of the prevalence of flu.[1] Thus, even if an algorithm seems well justified and correct, if key data is withheld by design or chance its performance might instead become very poor. But who is responsible if the wrong data is used? The data provider? The analyst? Answers are unclear and may well be driven by case law.

Safety tests

[1] Lazer, D., Kennedy, R., King, G. and Vespignani, A. (2014) The parable of Google Flu: Traps in big data analysis. Science, 343, 1203–1205.

Turning to algorithms themselves, these come in many shapes and forms, and are usually assessed on four aspects: "computational complexity" describes the storage and computing resources an algorithm requires to run; "typical performance" describes how good recommended decisions are on average, but says nothing about how bad the worst performance could be; "stability" refers to an algorithm's performance over time, measuring how likely it is to degrade as more and more decisions are taken; and "robustness" asserts that if a few observations are outliers that do not conform to the pattern of the rest, the algorithm will still make good decisions.

Trading these concepts against one another is a matter of engineering design. An algorithm that performs best on average may not be the most robust, while a robust algorithm may have higher computational complexity. And for an algorithm to remain stable, we may need to restrict the data it ingests. What would constitute a "reckless" design choice in such complex settings? The burden of proof will likely depend on the context of the automated decision, and the consequences that might follow. Over time we must look to develop standards and requirements to govern the trade-offs between these design choices. Algorithms for important decisions may well be subject to greater scrutiny: in the same way that aircraft components and software systems are tested exhaustively before being cleared for use, we may make similar requirements of fully automated decision-making systems using simulated data sets for which we have empirical evidence (or "ground truth") to compare against. In many cases, if we are willing to make assumptions about our data then algorithms can be proved to behave in certain ways. But how do we judge whether to trust simulation studies in lieu of experiments, or the assumptions we have made about our data? When will it be reckless to do so? And is it fair to expect the average consumer of an algorithm's output to understand these fine distinctions, or their possible consequences? Irrespective of which algorithm we choose, another area of potential liability is the actual code that implements it. Code is usually modular, and an automated decision-making system typically brings many different pieces of code together. Who is liable when modules appear to perform well in isolation, but fail to do so when they are part of a larger system? And what if code is repurposed, or if its intended purpose was never clearly stipulated?

A question of fairness

Perhaps the most philosophical of questions is: What does it mean for an algorithm to be "fair"? This is a veritable minefield, not least because fairness can often have a technical meaning linked to the statistical property of "unbiasedness". But fairness means different things in different settings, and in the legal sense it could be argued to be aligned with the principle of equality: "All are equal before the law and are entitled without any discrimination to equal protection of the law" (bit.ly/2yUzRJv).

Algorithms are already being challenged on this basis. In the USA, there are concerns of racial bias in parole decisions linked to the use of software to assess the risk that criminals will reoffend (bit.ly/2yTLLU1).

Significant further debate has followed.[2] Discrimination in parole decisions might arguably be due to the use of automated risk assessments. But does this mean the underlying algorithm is unfair? It very much depends on the training data that was used. An algorithm that sees a biased selection of alleged offenders may not be capable of producing a fair result in a legal sense. More technically: it may be impossible to constrain an automated decision-making system to deliver equal false positive and false negative rates if recidivism prevalence is not the same across gender, ethnicity or other demographic groupings.[3]

Fairness can also be related to the notion of transparency – the question of how much we are entitled to know about any automated system that is used to make or inform a decision that affects us. Hiding the inner workings of an algorithm from public view might seem preferable, to avoid anyone gaming the system. But without transparency, how can decisions be probed and challenged?

Thus, we arrive at the central conundrum at the heart of algorithms and the law. Automation in decision-making seems attractive in its potential to remove the bias and idiosyncrasy of human decision-making. Yet anything automated must be designed, and this design requires input data generated by real human interactions, and so is liable to reflect any existing inequities. Moreover, algorithmic design requires that humans make choices about costs and rewards, and how we balance these. For now, public opinion is quiescent: we vacillate between a wish to reap the benefits of new automated decision-making technology and a fear of being subjected to these selfsame decisions without recourse. It remains to be seen which of these two outcomes – our best wishes or our worst fears – will eventually be realised. But somewhere between them lies the fascinating intersection of algorithms and the law.

---

[2] Kleinberg, J., Mullainathan, S., and Raghavan, M. (2016) Inherent trade-offs in the fair determination of risk scores. arXiv: 1609.05807.

[3] Chouldechova, A. (2017) Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. Big Data, 5(2), 153–163.