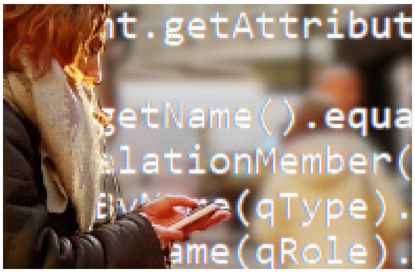
International Transport Forum | CPB Corporate Partnership Board

```
osmWay.setId(Long.parseLong(element.getAttributeByName(qId).getValue()));

} else if (phase == 1 && element.getName().equals(qRelation)) {
    osmRelation = new OSMRelation();
    osmRelation.setId(Long.parseLon          t.getAttributeByName(qId).getValue()));

} else if (osmRelation != null && e          getName().equals(qMember)) {
    OSMRelationMember relMember =            lationMember();
    relMember.setType(element.get            (qType).getValue());
    relMember.setRole(element.get            ame(qRole).getValue());
    relMember.setRef(Long.parseL             getAttributeByName(qRef).getValue()));
    osmRelation.addMember(relMem

} else if (osmWay != null && el              me().equals(qNd)) {
    OSMNodeRef nodeRef = new OSM             ;
    nodeRef.setRef(Long.parseLong(           getAttributeByName(qRef).getValue()));
    osmWay.addNodeRef(nodeRef);

} else if (element.getName().equa      qTag
    OSMTag tag = new OSMTag();
    String key = element.getAttr     uteByNam     Key).getValue();
    tag.setK(key.intern());
    String value = element.getA     ributeByName     al).getValue();
    if (key.equals("name") ||      quals("ref"      key.equals("highway")) {
        value = value.intern();
    }
    tag.setV(value);
```

Governing Transport in the Algorithmic Age



Corporate Partnership Board Report

Governing Transport in the Algorithmic Age

Corporate Partnership Board Report

About the International Transport Forum

The International Transport Forum at the OECD is an intergovernmental organisation with 59 member countries. It acts as a think tank for transport policy and organises the Annual Summit of transport ministers. ITF is the only global body that covers all transport modes. It is administratively integrated with the OECD, yet politically autonomous.

ITF works for transport policies that improve peoples' lives. Our mission is to foster a deeper understanding of the role of transport in economic growth, environmental sustainability and social inclusion and to raise the public profile of transport policy.

ITF organises global dialogue for better transport. We act as a platform for discussion and pre-negotiation of policy issues across all transport modes. We analyse trends, share knowledge and promote exchange among transport decision makers and civil society. ITF's Annual Summit is the world's largest gathering of transport ministers and the leading global platform for dialogue on transport policy.

Our member countries are: Albania, Argentina, Armenia, Australia, Austria, Azerbaijan, Belarus, Belgium, Bosnia and Herzegovina, Bulgaria, Canada, Chile, China (People's Republic of), Croatia, Czech Republic, Denmark, Estonia, Finland, France, Georgia, Germany, Greece, Hungary, Iceland, India, Ireland, Israel, Italy, Japan, Kazakhstan, Korea, Latvia, Liechtenstein, Lithuania, Luxembourg, Malta, Mexico, Republic of Moldova, Montenegro, Morocco, Netherlands, New Zealand, North Macedonia, Norway, Poland, Portugal, Romania, Russian Federation, Serbia, Slovak Republic, Slovenia, Spain, Sweden, Switzerland, Turkey, Ukraine, the United Arab Emirates, the United Kingdom and the United States.

About the Corporate Partnership Board

The Corporate Partnership Board (CPB) is the International Transport Forum's platform for engaging with the private sector and enriching global transport policy discussion with a business perspective. The members of the ITF Corporate Partnership Board are: Abertis, AB InBev, Alstom, Aramco, Bird, Bosch, Brisa, ExxonMobil, Incheon International Airport, Kakao Mobility, Kapsch TrafficCom, Latvian Railways, Michelin, North Adriatic Sea Port Authority, NXP, PTV Group, RATP Group, The Renault-Nissan-Mitsubishi Alliance, SAS, Siemens, SNCF, Total, Toyota, Uber, Valeo, Volvo Cars, Volvo Group and Waymo.

Disclaimer

Funding for this work has been provided by the ITF Corporate Partnership Board. This report is published under the responsibility of the Secretary-General of the ITF. It has not been subject to the scrutiny of ITF or OECD member countries, and does not necessarily reflect their official views or those of the members of the Corporate Partnership Board.

www.itf-oecd.org

Acknowledgements

The work for this report was carried out in the context of a project initiated and funded by the International Transport Forum's Corporate Partnership Board (CPB). CPB projects are designed to enrich policy discussion with a business perspective. They are launched in areas where CPB member companies identify an emerging issue in transport policy or an innovation challenge to the transport system. Led by the ITF, work is carried out in a collaborative fashion in working groups consisting of CPB member companies, external experts and ITF staff.

Many thanks to the members of the Corporate Partnership Board companies involved in this work: Abertis, ExxonMobil, Kapsch TrafficCom, Latvian Railways, NXP, PTV Group, RATP Group, Renault Nissan Mitsubishi Alliance, Robert Bosch GmbH, SAS Institute, Siemens, SNCF, Total, Toyota Motor Corporation, Uber, Valeo.

The report draws conclusions from the ITF CPB Workshop “Algorithmic Governance in Transport” held 4-5 December 2018 at the OECD (Boulogne) in France. Participants of the workshop included:

Avery Ash, INRIX

Jean-Marc Beteille, RATP Group

Jim Beveridge, ERTICO

Jean-Baptiste Burtscher, Valeo

Antoine Cazals, Université Dauphine

Maguelonne Chandesris, SNCF

Federica Citterio, SAS

Philippe Crist, International Transport Forum (ITF)

John Danaher, National University of Ireland - Galway

Marc Dalbard, PTV Group

John Ellis, Los Angeles Department of Transportation

Oskar Eriksson, SAS

Khurram Gaba, ExxonMobil

Tuomas Kaivola, Ministry of Transport and Communications, Finland

Takayuki Kusajima, Toyota

Eric Landel, Renault Nissan Mitsubishi Alliance

Manuel Lianos, NXP

Deanna Macdonald, Blockchain Labs for Open Collaboration

Marius Macku, Uber

Sharon Masterson, ITF

Mollie McArdle, SharedStreets, Open Transport Partnership

Laura Meynier, ITF

Thomas Obermayer, Future-law

Seleta Reynolds, Los Angeles Department of Transportation

David Roine, Valeo

Sebastian Rummel, Deondigital

Katja Schechtner, ITF

Goknur Sirin, Renault Nissan Mitsubishi Alliance

Philippe Ventejol, RATP Group

Kevin Webb, SharedStreets, Open Transport Partnership

Hélène Wiedemann, Renault Nissan Mitsubishi Alliance

Günter Wildmann, Kapsch TrafficCom

Nigel Williams, Alliance for Parking Data Standards

Special thanks go to Hilary Gaboriau, Sokob Challener, Laura Meynier and Michael Kloth from the ITF for support and editing and to John Danaher, National University of Ireland-Galway for substantive comments following the workshop. The principal author of this report is Philippe Crist with contributions from Katja Schechtner. The project was managed by Philippe Crist of the International Transport Forum. Coordination of CPB activities was carried out by Sharon Masterson.

Table of contents

Tables

Figures

Boxes

Executive summary

What we did

This report explores how and where automated decision-making systems are having consequential impacts on transport activity and how to ensure policy outcomes are delivered in a world increasingly infused with algorithmic code. It seeks to improve the algorithmic literacy of policy makers and ensure that their policies are fit for an increasingly algorithmic world.

It addresses both the opportunities and challenges of governing transport in a code-infused world. Many findings relate to general governance practices and are not transport-specific because of the emerging nature of these challenges. Yet transport authorities should be aware of what the challenges are and how they may play out in transport.

This report outlines how a new framework of public governance in an algorithmic age may take shape. It is based on discussions that took place at the "Algorithmic Governance in Transport" workshop held at the OECD in December 2018, extensive outreach to stakeholders, analysis and research.

What we found

Algorithmic governance is very much in its early days and its ultimate scope is unclear. But tensions are already emerging from a mismatch between the way public governance is structured in order to deliver on societal outcomes, and the way decisions are increasingly made by computer code. This gap must be closed if authorities want to deliver on the wishes of citizens.

Algorithmic governance poses different and novel challenges that go beyond the scope of current public governance frameworks. Yet the extent and type of changes in governance that may be required are still unknown. Governments, including transport authorities, should start envisaging a more algorithmic future and assess how this may impact their conception and delivery of public governance.

Algorithms are sets of defined steps that are structured to process both instructions and data to produce an output. Algorithms can be static, in that their code is rarely or infrequently updated. Recently they have become more dynamic, however, with algorithms now designed to re-write themselves to improve outcomes. This is a fundamental shift.

For instance, a regulatory agency may licence a specific self-driving technology (both the car and the code) for use on public roads. But as the scene selection, image processing and image recognition algorithms all iterate and rewrite themselves to better perform in real-world driving environments, the resulting code no longer bears any resemblance to the initial licensed code. Further, later iterations of the code may have evolved so much that the regulatory agency is no longer able to understand how they function.

Algorithms embed a model of "what is to be done" and then explicitly establish "how it should be done" in computer code. Algorithms should not be seen as independent snippets of objective and context-less computer code. They are part of a complete algorithmic automated decision system that starts with a specific goal designed to achieve a final, concrete action.

The algorithmic workflow follows the cycle of goal formulation, task identification, data sourcing, model definition, computer coding, software assemblage, output (either predictive or prescriptive) and then concrete action. Importantly, human interpretation, assumptions, potential biases and objectives are built into these systems all along the way.

Having humans in the algorithmic loop is good, as code-based are systems meant to meet human objectives. But it can also be challenging when algorithmic systems are portrayed or understood as being without bias – or at least as having fewer biases than humans.

The algorithms guiding, and emerging from Artificial Intelligence (AI) and, more specifically, machine learning (ML) applications, are particularly suited to solving formerly intractable problems or improving our ability to accomplish previously difficult and time-consuming tasks. However, they raise unique legal, regulatory and ethical challenges as well. Despite their benefits, ML algorithms may result in unintended and harmful behaviour if the wrong objective function is specified (or self-specified), if the training data is biased or corrupted, or if the learning process is faulty.

Physical, moral and even philosophical hazards emerge when AI systems start to drift into areas of human decision-making in ways that are analogous to, but inscrutable and fundamentally foreign to human cognisance. This may not be a problem where risks are low or potential impacts limited. But the lack of insight into AI decisions and processes challenges traditional forms of public governance when algorithmic outcomes may have significant impacts. Balancing the tremendous benefits that AI-based algorithmic systems can deliver with the potential harms they can inflict is at the heart of the policy and societal discussion around algorithmic governance.

Safety and security risks are the most immediate and material of all potential algorithmic harms. When cyber-physical systems fail or perform unexpectedly, people can get hurt and material damage may ensue. If these risks propagate across connected systems, the resulting harms can multiply and be substantial.

Algorithms are data-processing technologies. Data collection and surveillance are integral parts of the algorithmic system, but there are clear privacy risks associated with the use or release of that data. Simple approaches to data anonymisation or pseudonymisation are rarely robust enough to stand up against serious data-discovery attacks. These vulnerabilities grow in line with the capacity of adversarial algorithms to extract this data.

Algorithmic systems are highly opaque and difficult to explain to regulators, or to those affected by algorithmic decisions. Code is often created in environments that are not open to scrutiny, either because it is written by teams within companies or public agencies or because it is created in the logic space of an algorithm itself. Code is written in computer languages and follows logic patterns that are not widely understood by the population at large or by regulators. The operation and decisions of several types of AI algorithms may not even be explained by their designers.

Machine logic, especially when linked to machine learning, artificial neural networks and other forms of AI, is not human logic. The ensuing lack of understandability is only exacerbated when individual algorithms are tethered together in broader algorithmic decisions systems. Algorithmic systems, though they may be inscrutable and hard to understand, may function – but they pose a latent risk that breakdowns may not be traceable or "fixable" precisely because of this lack of understanding.

What we recommend

Make transport policy algorithm-ready and transport policy makers algorithmically-literate

Transport policy, institutions and regulatory approaches have been designed for human decision systems and bound by legal and analogue logic. These will be challenged by the deployment of algorithmic systems which function with machine logic. Public authorities will have to evaluate if their institutions and working methods are adapted to potential algorithmic risks and, if not, begin to reshape themselves for a more algorithmic world. This will require bringing in, and retaining, staff with new skill sets and training existing staff to become more code-literate.

Ensure that oversight and control of algorithms is proportional to impacts and risks

Not all algorithmic systems are equally risky (or beneficial). Regulators must seek a balance between the risks and mistakes that are inherent in technology innovation and the potentially negative impacts of regulatory intervention to avoid these. They should adopt a graduated regulatory approach that minimises oversight of trivial and low-impact algorithmic decision systems, and increase assessment and oversight for more and more consequential algorithmic system impacts.

Build in algorithmic auditability by default into potentially impactful algorithms

Human-readable pseudo-code could be built into algorithms to explain what the algorithm does without revealing source code (and preserve commercial secrets). These could take the form of "legal-grade" coding. Another approach would be to use specific coding protocols for potentially impactful algorithms – like those outlined by the "Trustable software" framework.

Convert analogue regulations into machine-readable code for use by algorithmic systems

Those coding automated decision-making algorithms interpret multiple regulations that are written in human-readable language and typically produced on analogue and dispersed supports. Where possible, authorities should strive to make regulations machine *and* human readable by default. For example, authorities could encode, communicate and control access rules and legally permissible uses of street and curb-space.

Use algorithmic systems to regulate more dynamically and efficiently

Regulation too often focuses on yesterday's technologies, rather than on emerging social-technological systems. Complying with the current regulatory framework may be the appropriate response in some cases but not in all. Avoiding the trap of current practice compliance is especially important when algorithmic systems obviate the need for existing regulatory practices and create new ways of regulating more dynamically and efficiently with a lighter touch.

Compare the performance of algorithmic systems with that of human decision-making

When assessing the potential impacts of algorithmic systems, authorities should consider what might be the impact of not deploying the algorithmic system in the first place. Is the balance of risks and benefits tilted towards having humans continue to make critical and consequential decisions instead of algorithms? If so, it is worth asking if an algorithmic system is necessary or even desirable. If the balance is reversed, then taking humans out of the decision-making framework entirely, or having them only intervene when prompted, may be the best option.

Algorithmic assessment should go beyond transparency and explainability

There are limits to requiring certain AI-based algorithms to be transparent and explainable because their logic may not be readily understandable to humans. One strategy to address this is to build explanation functionality into algorithmic systems so that the model can produce an accurate and intelligible explanation for its output. This type of "explainability by design" will entail changes in the way in which code is conceived and written – at least for applications where explainability is necessary to avoid consequential harms. It will involve setting standards, adopting industry best practice and, in some cases, may require that regulators stipulate this approach for critical code.

Establish robust regulatory frameworks that ensure accountability for decisions taken by algorithms

Rather than focusing on transparency, explainability or interpretability as keystones of algorithmic assessment processes, regulators should include these into a broader *algorithmic accountability* framework. A governance framework for algorithmic accountability should ensure that algorithmic systems are conceived and built in such a way that they can be trusted to operate as intended. Under an accountability framework, those responsible for deploying the algorithmic system should be legally accountable for its decisions. When that entity is a public authority, higher and more stringent standards of accountability should come into play given the unique powers that governments wield.

Establish clear guidelines and regulatory action to assess the impact of algorithmic decision-making

Impact assessments are common in many domains, including transport, and are well-understood mechanisms to assess potential risks and payoffs from policies and regulatory interventions. Public authorities should undertake impact assessments regarding algorithmic systems that could have a consequential effect on regulated outcomes or within the public domain. The approach adopted by the Government of Canada in its "Directive on Automated Decision-Making" is a model approach. It links assessment to a graduated regulatory response for potentially riskier algorithmic systems. Impact assessment auditing should be based on observable and monitored impacts and not necessarily comprised of audits of the algorithms themselves.

Adapt how regulation is made to reflect the speed and uncertainty around algorithmic system deployment

Existing "regulate and forget" policies are not suited to the rapid deployment of multiple, constantly evolving algorithmic systems. Authorities should diversify their regulatory approaches to maximise regulatory learning, early deployment benefits and a de-risked experimentation. These strategies should combine a risk-based approach, iterative and adaptable regulations, limited regulatory exemptions, performance-based outcomes and collaborative regulation.

The new DNA of transport

In transport, as elsewhere, ubiquitous and connected digital technologies and networks are changing the way in which humans, machines, and the two together, make consequential decisions. Human decision-making is reductive in that it seeks to constrain decision elements to a minimum useful set. Early forms of machine-aided decision systems expanded that set and enabled new outcomes. Now, the world is inundated with large sets of unstructured digital data that embody almost all human, commercial and government activities, including transport. This digital "dust" forms an inchoate, but rapidly coalescing, avatar of what constitutes "reality" and is evidence of the accelerating convergence between information society and society, writ large. The emerging "mesh" of material, labour and data-led value production is similar to past value production systems like those that fuelled the industrial revolution – but its complexity and understandability – including by public authorities – is complicated by its growing dependence on machine code (Crawford and Joler, 2018).

Many of the traditional human- and machine-based analytical tools and approaches deployed in the past to make sense of this information are no longer sufficient to deal with the scale and rapidity with which available knowledge of the world is changing. Enter the computer algorithm: automated and autonomous decision-making systems are often better able to handle many tasks that were difficult, or plainly impossible for humans to carry out. In particular, they are able to harness knowledge embedded in large, unstructured data in ways that surpass the ability of human cognition.

This report explores how and where automated decision-making systems are having consequential impacts on transport activity and how to ensure policy outcomes are being delivered in a world increasingly infused with algorithmic code.

The deployment of algorithmic decision systems by the private sector has harnessed new value; first through the interpretation of this data and the surfacing of unexpected trends, insights and predictions and then by guiding action on the basis of this knowledge. Under mounting pressure to deliver evidence-based policies, governments too are turning to automated decision support systems to better capture the volume of now-available "dat-evidence" to improve policy outcomes.

The decision-making environment for companies and public authorities is rapidly changing as data, networks and algorithms come together to form a new, fundamental DNA that frames decisions and policy.

Data

The action of algorithms builds on the availability of extremely large and unstructured datasets that are produced, and sometimes analysed and processed, in real-time. This data is generally sourced from increasingly ubiquitous microelectromechanical systems (MEMS), commonly referred to as sensors. These data cover all kinds of phenomena from computer-vision scene interpretation, location-, momentum- and directional acceleration-sensing, sensing of physical phenomena (like temperature, ambient noise, etc.) and from the logging of actions taken by machines and humans. Many new transport services and applications generate and use massive amounts of data. For example, Uber generates more than 100 petabytes of raw data per day in order to carry out millions of rides and deliveries (Kira, 2019). Much of what happens in the world, and a good part of what matters for delivering policy outcomes in a number of domains, including transport, is directly digitised or can be inferred from that which is digitally encoded and recorded.

Networks

The digitalisation of transport, like other areas of human activity, also relies on information technology and communication networks that transmit data. These networks operate on micro-scales like those that enable in-chip, real-time processing of data at the point of collection, to macro-scale ones that allow low-latency transmission of data from the point of sensing to processing locations and back again to specific actuators that trigger real actions in the world (opening a gate, allowing a package to be delivered, unlocking or locking a shared vehicle, transferring funds, etc.).

Algorithms

Processing of data into actionable outcomes builds on sets of instructions and processes that are encoded in a structured form, e.g. algorithms. Decision-making instructions and processes are not new; they form part of the basic working of human intelligence. What is new is the encoding of these instructions in a way that specifically takes advantage of massive amounts of data and extremely fast computer processing and network transmission speeds.

Of these three rapidly evolving areas, Data, Networks and Algorithms, the latter is arguably the most important and yet, poorly understood from the perspective of current and future transport regulation.

Why talk about algorithmic governance in transport?

Computer algorithms make or inform decisions that matter. They mediate, curate and adjudicate more and more important decisions in our lives – in health care, housing, social media, recommendation and reputation systems and even in political discourse. They are part of a broader change in the way in which multiple forms of human activity are governed or regulated in the broadest sense of those words – e.g. the act of controlling behaviour.

The transport sector is not immune to this trend. Computer algorithms are behind our decisions on where to go, how to get there, how vehicles operate, who can board a plane or train, where goods are sourced, where people are picked up and dropped off. Automated decision-making systems are at the heart of self-driving technology, ride-service dispatching, bike- and scooter-share services, passenger and commercial routing apps, public transport scheduling, freight delivery from e-commerce activity, tolling and payment applications, drone flight systems and transport modelling and planning. Traffic lights adapt their timing to measured or predicted traffic flows. Ride-hail services adapt incentives for drivers to log in on line in response to instantaneous or predicted demand for their services. Freight delivery fleets use iterative routing that enables real-time order processing and delivery. Container vessels are loaded and unloaded based on lowering overall dwell-time accounting for in-port congestion and predicted next-port traffic. These and many more transport-relevant applications of algorithmic decision systems are still in the very first stages of their potential deployment within (and around) the sector. These changes have an impact on the way in which public authorities carry out their mandates:

> We used to think of our responsibility as delivering and managing roads, sidewalks, streetlights, bridges, tunnels, tracks – the hard infrastructure of cities. And we did this on a decadal time scale. Now code is the new concrete, the infrastructure is algorithmic and the city must deliver it instantaneously. (Seleta Reynolds, General Manager, Los Angeles Department of Transportation at the ITF Workshop on Algorithmic Governance in Transport, December 4, 2018)

Transport is undergoing a fundamental shift in the way in which data is encoded, produced, processed and used. Increasingly automated vehicles are part of the policy planning horizon in many countries around the world. New types of platform-based, shared services are being deployed for both freight and passenger transport. The ability to find a ride within minutes, to evaluate multiple route options instantaneously, to rapidly, conveniently and affordably share unused freight or passenger vehicle capacity, all were unthinkable a generation ago but now are taken for granted by many. The future is one where algorithms may orchestrate mobility and access on a scale never before seen:

> Citizens and merchandise will be carried to any desired destination via myriads of self-driving vehicles, globally orchestrating their movements and routes with each other and with the urban street infrastructure. Similarly, the flow of pedestrians will somehow be steered…and orchestrated so as to avoid dangerous situations. (Zambonelli et al., 2018)

Getting to that future will require a better understanding of algorithmic systems themselves and ensuring that the future to which they contribute is still aligned with peoples' desires and the public policy objectives these inspire.

Algorithmic systems are deployed in at least four broad transport use cases (Danaher, 2018):

- Informing: Collection, analysis and communication of transport-related information to key decision-makers (e.g. vehicle operators, users of public space, traffic planners, political decision-makers)
- Access Control: Managing control and access to different transport modes and the public space these systems use (e.g. allocating riders to vehicles in platform-mediated ride-sourcing, determining and pricing access to parking or pick-up and drop-off zones, or handling inter-mode transfers)
- Operational Control: Physical control of transport vehicles and systems (e.g. automated metro lines, self-driving vehicles, autopilot functions in aircraft, autonomous drone operation or traffic light control in smart traffic centres)
- Behaviour Control: Directly or indirectly nudging, controlling or otherwise influencing travel-related behaviours of humans and freight transport trips (way-finding suggestions, variable traffic signs, variable pricing systems, online order delivery systems, etc.).

Algorithmic automated decision systems offer tremendous advantages in that they can be highly efficient, fast, and predictable and undertake decision-related analysis beyond human capabilities. They have opened new possibilities for delivering outcomes that were previously thought too expensive, too complicated or simply unobtainable. These changes have consequential effects on how transport is delivered and governed.

There is much promise for what transport will be able to deliver in the future, but there are clear pain points as well. One of those is that much of the classic government regulatory framework is built around a set of analogue, paper-based and human language-based rules embodied in the legislative code. In a world increasingly characterised by the outcomes of algorithms embodied in computer code, this may no longer be sufficient to ensure the delivery of public policy outcomes.

Algorithms govern, in that they influence human behaviour in a specific and directed way, but they are not government. They differ from the latter in that they are often hidden, they are typically closed and proprietary, they are inscrutable, their workings are not easy to explain and their decisions sometimes hard to justify. Further, the mechanisms for understanding the impacts of regulatory action by public authorities, for seeking to ensure broad conformity with these and to seek redress for regulatory harms are well known and fully embedded in society. The same is not (yet) true for the regulatory action of algorithmic automated decision systems.

The wider use of algorithmic decision-making will likely benefit society, allowing new and better outcomes than what has been possible in the past, but the ultimate balance of pros and cons going forward remains unclear. For this reason, it is well worth looking into how public authorities and the private sector should anticipate and prepare for ensuring public policy outcomes in an increasingly algorithmic world.

This report does so by first addressing what the term "algorithmic governance" means, focusing on the task of governance by public authorities, and exploring the defining features and risks associated with the use of algorithms in automated decision-making systems. It then explores three areas where public authorities, in transport and elsewhere, should be preparing to think critically about algorithmic governance, assess the impacts of automated and algorithmic decision-making systems in regulated domains, and start to adapt policy and decision-making structures for the algorithmic age. These areas are machine-readable regulatory code, regulating by algorithm, and assessing and regulating algorithms.

Machine-readable regulatory code

Those coding automated decision-making algorithms must interpret multiple regulations written in human-readable language and typically produced on analogue and dispersed supports. This is the case, for example, regarding the way in which authorities encode, communicate and control access rules and legally permissible uses of street and curb-space. Defining and enforcing these roles is currently the responsibility of various departments or is delegated to other parties. A single, legal, "street code" feed could harmonise access to these rules and permit much more dynamic use and management of urban road space by private and commercial users.

Regulating by algorithm

A second area of exploration is where governments can deploy regulatory algorithms – e.g. code that automatically or semi-automatically undertakes specific regulatory functions. These might pertain to the collection of registration, licence or use fees and revenues via digital ledger technology-enabled "smart contracts". These uses are analogous to developments in the financial technology field which seek to leverage blockchain technology to more seamlessly collect revenue, control compliance and allocate legal rights to parties.

Assessing and regulating algorithms

The third area of investigation is delicate. It gets to some of the very real issues relating to the scope of regulation and oversight versus the protection of intellectual property (and individual privacy). Because of the potential material impacts of algorithmic decision-making on the ability for public authorities to carry out their mandates, some voices have raised the possibility of regulating algorithms themselves. This is largely because of the aforementioned characteristics of algorithms – their hiddenness, impenetrability and inscrutability. But regulating algorithms directly is not straightforward for a number of reasons: 1) it may require revealing proprietary and commercially valuable source code; 2) the capability of regulators to correctly interpret code may be limited (and may in fact require third-party algorithm-mediated interpretation); and 3) the workings and structure of algorithms are both exceedingly complicated and changing (in the case of algorithms re-writing themselves).

This report explores how governments and the private sector could start to think about translating legislative code into a framework that can be easily integrated into algorithmic decision-making, and vice-versa, where applicable. Other government sectors have already started to operate this transition towards "RegTech" – e.g. technical systems used to carry out regulatory functions –, most notably financial oversight authorities, but much has yet to be invented in order to build a robust technical-legislative framework for digital actors in transport. In transport, some countries are starting to put in place the necessary building blocks that will enable this shift by, for instance, creating official and immutable individual or commercial "e-identity" numbers in Estonia, or requiring standardised and open data sharing by regulated entities in Finland and Los Angeles.

The notion of algorithmic governance is very much in its early days in transport, as in other domains, and its ultimate scope is unclear. What is certain is that a growing range of tensions are emerging because of a mismatch between how public governance is structured to deliver on societal outcomes and how actionable decisions are being increasingly made by computer code. This gap between action and public governance must be closed going forward if authorities are to deliver on citizens' wishes. This report lays the groundwork for how this new framework of public governance in an algorithmic age may take shape.

Algorithmic governance

The term “algorithmic governance” describes a broad and increasingly important field of study in domains as varied as health care, financial services, housing, criminal justice and transport. It presupposes a general understanding of algorithms and the ways in which they function as well as the understood definition of the term “governance”. It also implies that public governance may be needed, not principally because of the value that algorithms deliver, but because of the types of harm they potentially inflict. This section addresses all three of these topics: what are algorithms and how do they function, what does the term “governance” mean and what are potential harms that arise from algorithmic decision systems?

What are algorithms?

> Algorithms are often elegant and incredibly useful tools used to accomplish tasks. They are mostly invisible aids, augmenting human lives in increasingly incredible ways. However, sometimes the application of algorithms created with good intentions leads to unintended consequences. (Rainie and Anderson, 2017)

An algorithm is a set of guidelines to accomplish a determined task

Broadly speaking, an algorithm is simply a set of guidelines to accomplish a determined task. More specifically, algorithms are sets of defined steps structured to process instructions and data to produce an output. They take several forms and carry out various functions – e.g. sorting, classifying, pattern recognition, routing, recommending, optimising, profiling, matching – in order to produce the desired class of outputs. They can be static, in that their code is rarely or infrequently updated, but recently they have become more dynamic. Algorithms are now designed to re-write themselves to better deliver outcomes (Andrews et al., 2017).

A recipe is an algorithm, as are the guidelines for determining the conditions under which an authority should issue (or revoke) a license to operate a vehicle. The detailed, invariable and rigid set of conditions and instructions that comprise the tax code are algorithms as are the rules governing the application of parking penalties and fines. Algorithms determine under what set of input conditions an automated braking system must engage and when input from an optical or LIDAR sensor should be identified as a “car” or a “curb” or a “pedestrian”. Algorithms determine which vehicle should be matched to a ride-hailing request and when traffic should be diverted from a direct but slower route to a faster but lengthier route in a wayfinding app. Even significant parts of human behaviour can be described as algorithmic. Individuals often employ a set of loose rules to determine what actions to take to reach an acceptable outcome – these fuzzy heuristic techniques are forms of algorithms.

Much of government action is also algorithmic in the sense that decisions enacted by public authorities in democratically elected societies are the result of explicit guidelines that process various inputs and deliver specific decisions on the basis of transparent and auditable rules, i.e. laws and regulations.

Human activity is largely “algorithmic”, but not all algorithms are “human”

While a large part of human activity can be described as “algorithmic” not all algorithms are “human”, in the sense that the steps algorithms carry out are not undertaken by people but rather, by computer code. The focus of this report is on this category of algorithms – those that are created in machine language (code) for use by automated decision-making systems (algorithm machines or computer systems) that produce actionable decisions independent of immediate human input.

Algorithms carry out three principle meta-functions: they collect data, analyse that data and make decisions on the basis of that analysis. Some algorithmic systems only carry out one of these functions, many carry out a combination of the three (Danaher, 2018). These decisions may replace human action entirely (for instance, actuating a steering wheel control in a self-driving vehicle) or may serve to suggest a limited set of actions from which a human may choose (in a travel planning and booking application, for example).

Algorithms are often seen as straightforward tools that ingest objective data and lead to less biased outcomes than human-decision-making processes (Seaver, 2013; Kitchen, 2017). Moreover, because they can process much more data than humans, and much more quickly, they can lead to new insights, outcomes and support processes that are simply not possible or economic with humans. Algorithms and code, however, are not pure objective constructs. They depend on data and assumptions that can be incomplete, flawed or biased, and can lead to sub-optimal outcomes and machine-biases that may not be outwardly obvious (Schneiderman, 2016; Rainie and Anderson, 2017).

Algorithmic decision-making domains

Algorithmic automated decision systems fall into three broad areas that correspond to specific outcomes or problem-solving domains, running from the most general applications to the most specific (Castelluccia and Le Métayer, 2019):

Improving general knowledge or technology. Algorithms are deployed to analyse complex, unstructured datasets in order to derive new, actionable knowledge regarding correlative human behaviour or physical phenomena. Improving weather and climate predictions, better understanding crowd dynamics, detecting diseases or identifying their propagation patterns, or better understanding braking patterns in mixed traffic all fall under this broad set of applications. Knowledge derived from these types of applications can then be used in developing new services or technologies.

Developing or improving digital services. Algorithmic decision systems help make predictions, recommendations or decisions in support of various services offered to people. These systems collect multiple, dynamically-weighted inputs in order to optimise one, or a set of desired criteria. Routing apps adopt this approach, as do platform-based services that deliver ride-sourcing, shared micromobility, logistics and other supply-demand matching algorithms. These systems can also be used to optimise maintenance, site new infrastructure or detect cyber-security vulnerabilities. While these services are digital, their delivery often entails material interactions that provide value to people (the ability to receive a purchased good in less than an hour) but at the same time may negatively impact public policy outcomes (increased congestion due to more frequent delivery vehicles and deliveries).

Carrying out sense-processing-actuating functions in cyber-physical systems. Algorithms automatise human supervision of, and interaction with physical systems. The ensemble of algorithms that enable autopilot functions in modern aircraft fall under this category, as do those that enable autonomous operation of factory robots, drones and vehicles. All automated systems depend on algorithms that operate without, or with minimal human interaction. Thus a large share of the code that enables various functions in modern vehicles, from braking to selecting adapted engine mapping, falls into this category.

How do algorithms work?

Algorithms are embedded in broader algorithmic processes that are comprised of several steps (Andrews et al., 2017; Yeung, 2018; Kitchin, 2017):

- problem definition
- data collection and calibration

- data filtering
- data interpretation
- algorithmic processing
- interpretation of algorithmic output
- action

Algorithms are thus part of an assemblage of computer language, model, data, training data, application and hardware that processes inputs to deliver outputs. These inputs may be sensed or scripted into code. The outputs may be inputs to other algorithms, to actuators that act on physical systems, or that serve as suggestions for making human decisions. In all three cases, these outputs have direct, material and tangible impacts.

Algorithms process input according to a defined extrinsic logic. They embed a model of "what is to be done" and then explicitly establish "how it should be done" in computer code (Kowalsky, 1979). For those writing code, an algorithm comes after the formulation of a problem and the identification of a desired outcome or goal. Problem definition and goal formulation, combined, form an ensemble in which, and against which algorithmic function is tested and optimised (Gillespie, 2014).

Thus, for example, the objective of guiding a person from one location to another in the most optimal way might be broken down into discrete tasks. Those tasks would fit a model, illustrated in Figure 1, which efficiently calculates the combined values of pre-weighted and user-specific variables in an index covering all possible route options so that the user selects the suggested route in, say, nine out of ten times (Gillespie, 2014).

The model is then transcribed into a formal machine-readable syntax or code that enables the processing of input in accordance with the model specification and returns a result that fits the goal at hand. The distinction between model and algorithm can be compared to a newspaper and a news story. The model is the information and meaning the story conveys, and the words, phrases and paper are the algorithm.

Some code may embody a direct algorithmic formulation, as in mathematical equations. In other cases, the problem formulation and the underlying model structure must be interpreted into a set of instructions or pseudo-code. This pseudo-code is then hard-coded into machine-readable language that is compiled, alongside many other algorithms, into complex assemblages of recursive decision trees. This ensemble forms the body of the computer programme or software that is called to execute some form of automated decision-making. (Kitchin, 2017)

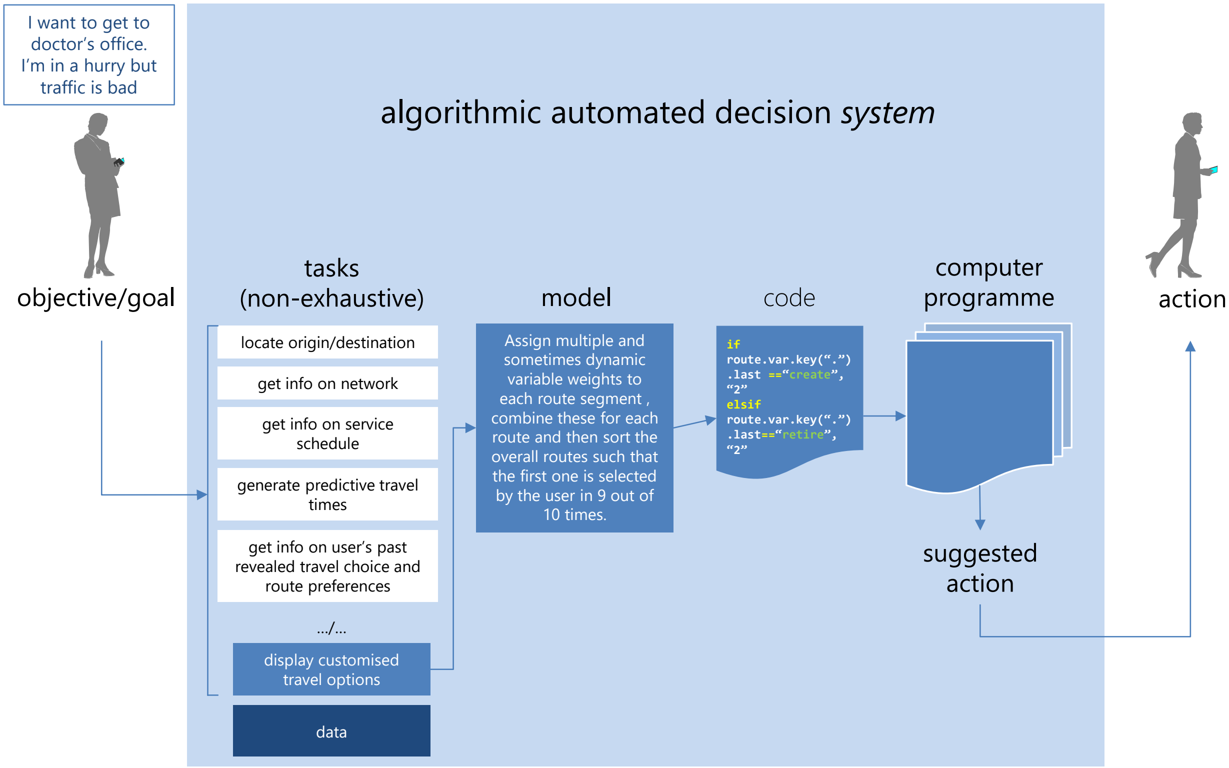
Computer algorithms are rarely, if ever, static constructs, unlike other forms of bureaucratic government "algorithms" and rules. They are designed to be constantly updated, optimised, tweaked, iterated and otherwise modified. Lines of code may be deleted or replaced, open-source code inserted, new algorithmic methods selected and the model itself may change over time as the definition of the problem to be addressed evolves:

> These algorithmic systems are not standalone little boxes, but massive, networked ones with hundreds of hands reaching into them, tweaking and tuning, swapping out parts and experimenting with new arrangements…We need to examine the logic that guides the hands. (Seaver, 2013)

This means that establishing the link between a specific instance of an algorithm and an outcome in the real world becomes a problematic ex-post exercise unless codebase changes are rigorously recorded or logged.

Identification, sourcing and ingesting data is a fundamental element of algorithmic processing. In the aforementioned case of a route-finding algorithm (Figure 1), the data includes the location from which the route starts, any desired through-points and the desired destination. It also includes data on the layout and rules relating to the use of streets and other physical networks. It may include data on scheduled or real-time transport service operation, location of shared vehicles and other assets, locations of allowed pick-up and drop-off points or parking. Other data, such as current and predicted travel speeds by mode and segment-specific probabilities of traffic-slowing or stopping-incidents, are also relevant. Extrinsic data, such as current or predicted meteorological conditions, could prove relevant, as well. All of this data is ingested and processed by the algorithmic system in order to deliver route-suggestion outputs that achieve some defined threshold of acceptance.

Figure 1. Generic algorithmic system function


I want to get to doctor's office. I'm in a hurry but traffic is bad
algorithmic automated decision *system*
objective/goal
tasks (non-exhaustive)
locate origin/destination
get info on network
get info on service schedule
generate predictive travel times
get info on user's past revealed travel choice and route preferences
.../...
display customised travel options
data
model
Assign multiple and sometimes dynamic variable weights to each route segment , combine these for each route and then sort the overall routes such that the first one is selected by the user in 9 out of 10 times.
code
computer programme
suggested action
action

```
if
route.var.key(“.”)
.last ==“create”,
“2”
elsif
route.var.key(“.”)
.last==“retire”,
“2”
```

As tasks assigned to algorithms increase in number and complexity, so too does the diversity and interconnectivity of the algorithmic ecosystem deployed to solve them. This had led to a move away from a monolithic source codebase implementation to service-oriented architecture (SOA) based on small, modular, encapsulated *microservices* (Fowler, 2014). Each of these undertake one, clearly identified task, are typically maintained by a dedicated team and each can be upgraded independently without impacting overall service (Reinhold, 2016).

Platform services like AirBnB, Amazon, Uber and other ride-sourcing companies deploy microservice-based code architecture, as do some government services like the UK Government Digital Service (Fowler, 2014). This means that the functionality of the whole algorithmic system – for example, the act of ordering a product and having it delivered to a specific address at a specific time – rests not on a single codebase, but on the complex and iterative interaction among multiple microservice codebases (Munn, 2018). This approach allows rapid scaling and quick and agile modification of code and, in theory, makes it more straightforward to isolate certain elements of the distributed codebase that lead to unwanted or harmful impacts. Except when it doesn’t. For instance, the interaction between microservices could lead to unexpected or unwanted outcomes in ways that are hard to forensically identify.

From a policy perspective, algorithms should not be seen as independent snippets of objective and context-less computer code. They are part of an algorithmic automated decision system that encompasses goal formulation, task identification, data sourcing, model definition, computer coding, software assemblage, output (either predictive or prescriptive) and then concrete action (Kitchin, 2017; Zambonelli et al., 2018). More succinctly, this chain can be broken down into three principal components:

- Input domain – what is fed or used by the algorithm to deliver an output
- Logic domain – how the algorithm is conceived, written and functions
- Output domain – what is done with the output of the algorithm.

This breakdown is helpful in identifying where issues may arise along the continuum that impact areas of public policy and where, specifically, adapted forms of public governance should be prioritised.

For instance, in terms of the input domain, machine-readable regulations that serve directly as input to algorithmic processes can help close the gap between algorithmic decisions and public policy objectives. At the same time, flawed, doctored or biased input data (either in the machine learning stage or in the algorithmic processing stage) can lead to unintended, biased or unwanted outcomes. In terms of the logic domain, better impact assessment practices and more rigorous explainability requirements for impactful algorithms can help ensure that algorithmic decision-making systems do not contravene public policy objectives. Finally, advice on when and how to use algorithmic decisions and better regulatory strategies around the deployment and licensing of algorithmic systems can help ensure that algorithmic outputs are aligned with public policy objectives.

Importantly, human interpretation, assumptions, potential biases and objectives are built into these systems all along the way. Having humans in the algorithmic loop is good – these are systems meant to improve human objectives. But it can also be challenging when algorithmic systems are portrayed or understood as being without bias – or at least as having fewer biases than humans. There are many instances where these assumptions do not hold – for example, facial recognition technology used by some police departments has been shown to have persistently lower accuracy rate for African-American faces (McCollum, 2017).

What does algorithmic code do?

The code component of an algorithm specifies how the algorithm is to resolve an immediate task. These tasks are numerous – algorithms search, collate, sort, categorise, group, match, analyse, profile, model, simulate, visualise and regulate people, processes and places (Kitchen, 2017). Broadly speaking, these functions can be subsumed into four meta-categories: prioritisation, categorisation, association and filtering (World Wide Web Foundation, 2017).

While most algorithmic decision-making systems build on these four processes, they may employ different strategies to do so. Indeed, most automated decision-making systems do not seek to deliver a single objectively “correct” outcome. Instead, they seek to select one of many possible results, none of which is certifiably “incorrect”, based on some specified threshold of acceptance (Gillespie, 2014).

Algorithm designers have multiple methods to choose from when coding decision rules, many based on Boolean logic (IF this condition is met, THEN this action is taken. ELSE another action is taken). They choose from these a method that best matches the desired technical performance of the algorithm in completing the specified task. A designer may want to optimise computational speed, minimise computational resources, reduce data-related latency, etc. Often, the choice of algorithmic method will seek balance among multiple desired performance characteristics.

Table 1. Principle algorithmic functions

Algorithmic function	Examples
Prioritisation associating rank with emphasis on particular information, weights or results at the expense of others through a set of pre-determined criteria	*Routing engines* *Search engines* *Defining AV driving action* *Social media timelines*
Classification grouping information based on features identified within the source data	*Reputation systems* *Image interpretation* *Risk scoring* *Social scoring*
Association determining relationships between particular entities via semantic and connotative abilities	*Ride-sourcing matching* *Predictive traffic management* *Preventative maintenance*
Filtering including and/or excluding information as a result of a set of criteria	…

Source: based on World Wide Web Foundation, 2017.

Human assumptions, choices and actions are present everywhere within algorithmic systems – sometimes directly, often indirectly. This "presence", or lack thereof, can be broken into three types (Christiano, 2015):

- Human in the loop systems: In these systems, the direct involvement and input of humans is necessary for the system to function. Human-programmed algorithms that propose a set of options from which a person may choose are examples of these.
- Human on the loop systems: In these instances, the algorithmic system produces a decision that is reviewed and can be over-written by a human decision-maker.
- Human out of the loop systems: These are fully automated algorithmic systems where humans only intervene upstream to develop the initial code. The algorithm then functions and acts independently of human input or oversight.

In human-in-the-loop systems, human developers typically develop the mental model to solve a problem and then choose an adapted algorithmic method. They also hand-code the algorithm itself – or more realistically, assemble and adapt existing code from libraries, writing new code where necessary. This "manual" approach to coding is evolving as the optimisation of algorithmic method selection and the actual coding of algorithmic solutions is something that algorithms themselves are increasingly called upon to do independently of human input (Zanzotto, 2019). In human-on-the-loop systems, the human presence recedes to simply reviewing algorithmic output with the algorithmic system making most of the relevant decisions. Finally, in human-out-of-the-loop systems, except for some initial specifications, the algorithmic system functions and adjusts itself fully autonomously. A world in which algorithms update themselves and write other algorithms represents a fundamental shift with the past and is at the heart of the current Artificial Intelligence revolution.

Artificial Intelligence: Human coding vs. machine thinking

Artificial Intelligence refers to the ability for machines to make decisions and perform tasks that are characteristic of human intelligence (McCarthy et al., 1955). Artificial Intelligence (AI) describes an ensemble of advanced algorithmic processes that enable computers to process large amounts of structured or unstructured data and carry out highly complex tasks efficiently and with minimal, or no, human input. AI helps people and machines make better, or at least more rapid and complex, data-informed, decisions.

Though there is no single agreed definition of "Artificial Intelligence", several have been proposed (OECD, 2019). The United Kingdom Physical Sciences Research Council's description of AI is broadly representative of others:

> Artificial Intelligence technologies aim to reproduce or surpass abilities (in computational systems) that would require 'intelligence' if humans were to perform them. These include: learning and adaptation; sensory understanding and interaction; reasoning and planning; optimization of procedures and parameters; autonomy; creativity; and extracting knowledge and predictions from large, diverse digital data. (Hall and Pesenti, 2017)

AI is used in a number of applications, from processing natural language to operating autonomous robotic or vehicle systems, real-time or predictive matching of supply and demand for rides or goods, predicting traffic speeds or dangerous road segments and behaviours, and managing supply chains. In many instances, AI processes digital data faster and more dynamically than the human cognitive function can, resulting in previously unobtainable insights and outcomes.

AI has typically been applied to specific, bounded problem sets like playing the game of chess or Go, predicting what product category is most likely to be ordered at what time of day, or real-time identification of traffic participants from sensor inputs. Such applied AI – or Artificial Narrow Intelligence (ANI) – is the most developed form of AI today (OECD, 2017). ANI tasks and algorithms can be combined in more complicated systems to resemble broader, "human" intelligence – at least for some bounded domains. This is the case for the multiple, embarked ANI systems used by autonomous vehicles to carry out the broadly described task of "pick a route, drive a vehicle in traffic and park it".

Artificial General Intelligence (AGI) is different from ANI in that AGI AI systems would replicate human general intelligence by applying knowledge generated in one domain to multiple, radically different, domains (OECD, 2017). This form of AI is still not achievable and is a much more challenging proposition.

AI strategies fall into one of two broad approaches: symbolic reasoning or machine learning (Skymind, 2019).

Symbolic reasoning-based AI takes the form of rules engines, expert systems or knowledge graphs. It requires a predetermined understanding of how things relate to each other and in which circumstances they relate to each other. These rules are then hard-coded into the AI algorithm, typically in the form of nested if-then statements. Symbolic AI is monotonic in that its "intelligence" is directly proportional to the number of rules that are encoded in the algorithm – and these cannot be easily backed out or modified automatically under new knowledge. The learning of symbolic AI takes place outside of the algorithmic system and is introduced by human intervention. This means that the assumptions behind the operation of symbolic-reasoning AI are discoverable, understandable and auditable (though they may be hidden, obfuscated or difficult to understand). Symbolic reasoning AI is best suited for areas where relationships between entities, variables and actions are well known.

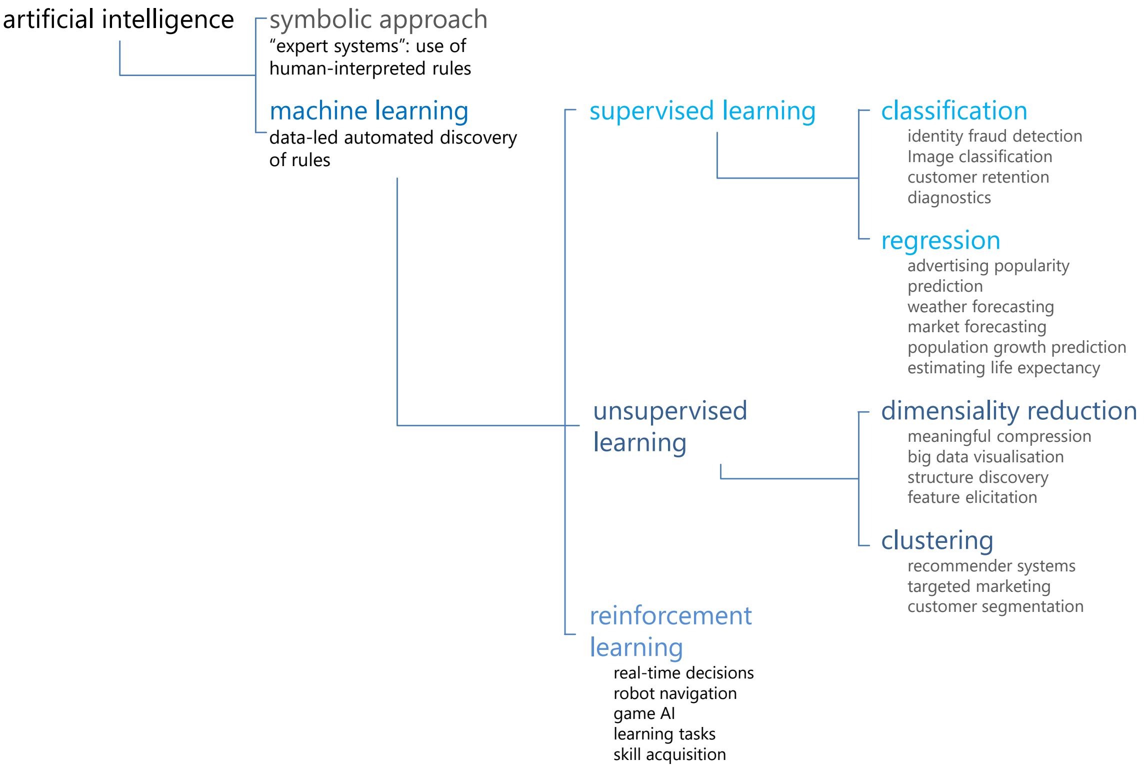
The other broad class of AI, and one that is often used synonymously with AI, concerns machine learning (ML) systems. ML algorithms elicit knowledge from large, unstructured datasets without being explicitly programmed to do so. ML extends traditional statistical decision-making approaches and applies them so that algorithms make or suggest decisions "on their own": independently of direct human intervention or via the specification of rules, models or algorithmic methods. This report refers to all machine learning variants, i.e. simple machine learning and the machine learning sub-categories that carry out "deep learning" – artificial neural networks, convolutional neural networks, etc. – generically as "machine learning" or ML.

ML algorithms are designed to deliver (human-defined) outcomes based on examples of what constitutes acceptable decisions or on the basis of loose rules about what outcomes should be favoured by the algorithm. ML algorithms are "trained" by ingesting large amounts of data that "teach" what acceptable outputs are. ML algorithms "learn" by trial and error on the basis of their "experience" or according to the heuristics programmed into them. They iterate different algorithmic models, methods and coding in order to adjust themselves and improve their performance. In contrast to Symbolic AI, learning in ML AI takes place *within* the algorithmic system.

Machine learning algorithms fall into one of three broad families: supervised learning, unsupervised learning, or reinforcement learning (Li, 2017; Salian, 2018; New Tech Dojo, 2018; Castelluccia and Le Métayer, 2019).

Supervised learning involves data that has been labelled by humans. The algorithm ingests this data and builds a model that allows it to accurately replicate a label for new data that was not part of the training set. Thus, an algorithm may be fed millions of images that have been labelled by humans as containing a person riding a bicycle. The algorithm then develops and tests models that can accurately identify a person riding a bicycle versus a bicycle by itself, or any other object, in images that it has not "seen". This image identification task is a form of classification problem for which supervised learning algorithms are well-suited. Regression problems that use continuous, non-discretised data and try to predict the importance of certain variables on outcomes are also well suited to supervised ML. Because they use known categories or known relationships, supervised ML is best suited for problems where a number of reference points exist.

Figure 2. Types of Artificial Intelligence



Source: adapted from (Krzyk, 2018).

In unsupervised learning, ML algorithms find patterns in data that have not been labelled by humans or where relationships are unknown. These algorithms automatically discern patterns, clusters, anomalies, relationships and structures from raw, uncategorised data. Unsupervised ML algorithms are often used to help discern new knowledge about data that can then be used to train other ML processes. They are also used to process and simplify large sets of data so they retain the original data structure and characteristics with fewer data elements. Unsupervised learning algorithms are used, for example, to process billions of ride-hail trip characteristics and elicit new, operationally relevant insights.

Reinforcement learning is an iterative ML process where an algorithm processes data and acts on success/failure feedback it receives from an external context. It develops models that seek to maximise a reward function with each iteration until a satisfactory (externally set) performance threshold is achieved. A reinforcement learning-based approach could be used, for instance, to teach an automated vehicle how to clear an intersection safely with the reward function seeking to avoid all collisions and minimise the number of near-misses.

Various hybrid models of ML exist as well. For instance, semi-supervised ML algorithms "learn" from a mix of labelled and data that is both un-labelled and unstructured (Castle, 2018). They build on a small set of known exemplars and then use this information to guide unsupervised learning. One semi-supervised method involves pitting algorithms trained with labelled data against each other to improve the performance of both. Such a general adversarial network (GAN) sets a generator algorithm that creates data best replicating the characteristics of the training data against a discriminator algorithm that tries to elucidate if the generator's data is part of the training data or not. Another form of hybrid would involve setting two reinforcement learning processes against each other to accelerate the action-reward cycle.

ML processes employ various algorithmic methods. Those that are at the heart of most recent advances in, and applications of, AI employ a "deep learning" approach which is often described as being roughly analogous to the structure of neural networks in the brain. Artificial neural networks (ANN) are superficially inspired by the deep layering (hence the term "deep learning") and connectivity of neurons, each handling a discrete task with numerous inputs and a single output. That output then becomes one of several inputs to the next "neuron" (Singh Gill, 2019; Somers, 2017; Perez, 2017; Rankin, 2017).

Another popular neural network, convolutional neural networks (CNN), mimics the structure of neurons in animal visual cortices (Cornelisse, 2018). A core part of the ANN learning function builds on a process called "backpropagation" that iteratively and recursively tests how well each "neuron" function performs compared to a "correct" performance (based, for example, on labelled data). This backwards propagation and analysis of errors is then used to automatically adjust each neuronal node's code until the final error is reduced to acceptable level. Backpropagation requires tremendous amounts of data and thus ANNs using this technique are particularly well-suited to processing sensor-derived data (Somers, 2017; Perez, 2017).

Backpropagation has enabled the development and tuning of highly effective ANN machine-learning systems. However, it is a process that makes it nearly impossible to explain how and why an algorithm functions or delivers the output it does (Somers, 2017; Perez, 2017; Rankin, 2017). This last point is a crucial one: ML algorithms are effective, fast, and can create tremendous new value, but they are not, sometimes due to their very nature, understandable or explainable – even by those who write their code. This makes them generally unsuitable for use when verifiability and accountability are important.

The algorithms guiding and emerging from artificial intelligence and machine learning applications are particularly suited to solving formerly intractable problems or improving our ability to accomplish previously difficult and time-consuming tasks but they raise unique legal, regulatory and moral challenges as well. Despite their benefits, machine learning algorithms may result in unintended and harmful behaviour if the wrong objective function is specified (or self-specified), if the training data is biased or corrupted or if the learning process is faulty (Amodei et al., 2016).

Physical, moral and even philosophical hazards emerge when AI systems start to drift into areas of human decision-making in ways that are analogous to, but inscrutable and fundamentally foreign to, human cognisance. Where risks are low or potential impacts limited, this may not be a problem. But the lack of insight into AI decisions and processes challenges traditional forms of public governance when algorithmic outcomes could potentially have deleterious material impacts or contravene desired public policy and societal outcomes. Balancing the tremendous benefits that AI-based algorithmic systems can deliver and the potential harms they can inflict is at the heart of the policy and societal discussion around algorithmic governance.

What is algorithmic governance?

The term "governance" (or regulatory governance) is one that can be interpreted on many levels and for which there is no settled definition (Baldwin et al., 2012; Yeung, 2017). Governance can be defined broadly as "a catch-all term for the techniques and practices whereby human behaviour is nudged, incentivised, manipulated and otherwise controlled" (Thornton and Danaher, 2018). More specifically, governance can be interpreted as intentional attempts to manage risk or alter behaviour in order to achieve some pre-specified goal (Yeung, 2017; Black, 2014; Baldwin, et al., 2012). Governance implies an entity governing or enacting regulatory control and an entity that is the target of that control.

In the most common understanding of the terms "regulation" and "governance", the entity exercising control is a government – i.e. the institutional embodiment of some form of constitutive public authority. Governments govern through the use of human language rules that communicate agreed-upon standards ("the law"); compliance is monitored and breaches trigger consequences, including civil or criminal punishment (Danaher, 2018).

Regulation and governance, however, can be enacted by other, non-government entities and performed via different means. Non-state actors like the press govern certain behaviours, companies and markets influence behaviours, and, to the point of this report, algorithmic decision systems also regulate behaviour.

While governments govern via law and control, other entities regulate behaviour via influence, nudging, selective presentation, forms of self-regulation or other means (Bayamlıoğlu and Leenes, 2018). Furthermore, while the act of regulation and governance may involve several potential types of governing relationships, they all share a common feature – that of the system "director" (Yeung, 2017). While many of these system directors (government, the press, commercial actors, markets, etc.) are known and the processes whereby they exercise control generally well understood, this cannot be said for algorithmic governance and control. This is because the nature of this control – via automatic decision-making based on self-adjusting "computational generation of knowledge from data emitted and directly collected … from numerous dynamic components" (Yeung, 2017) – is new in its scope, inscrutability and pervasiveness and is largely unaddressed in public policy and law. Algorithmic governance, therefore, concerns:

> …empowering software to take decisions and to autonomously – i.e. without human supervision – regulate some aspect of our everyday human activities or some aspect of society, according to some algorithmically defined policies. (Zambonelli et al., 2018)

So what then are the constitutive parts of the algorithmic governance framework from the perspective of public authorities? There are at least three. Public authorities may seek to *regulate algorithmic systems* to bring them in line with the public policy outcomes they are mandated to deliver. Public authorities may also *use algorithmic systems to carry out their regulatory tasks* ("algorithmic regulation", as per Yeung (2017)). Finally, public authorities may find that algorithmic systems may *supplant certain traditional regulatory mechanisms* that governments have historically deployed.

Public authorities should also be aware of the relationship between the overall government-led regulatory context and the ways in which algorithmic systems function and enact change. The former serves as a guide to behaviour and a framework for all public and private decisions. There are no aspects of modern life that are not subject to some form of regulation and transport is no exception. This ruleset covers the behaviour of all the participants in the transport system, including individuals, service providers and transport operators, different levels of government, vehicle manufacturers, etc. This broad, governing ruleset will be disrupted by algorithmic decision systems:

> Any algorithmic governance system introduced into the transport sector will, at a minimum, raise questions about the ruleset and the regulatory context. Its introduction will be a "disruptive moment" and may lead to ambiguities or uncertainties about the applicability of that ruleset to the new technology and the powers of those charged with monitoring and enforcing it. This is true for all new technologies. (Danaher, 2018)

A number of factors will influence how disruptive algorithmic systems will be to traditional public governance processes. Is there one regulator or are their several regulators involved in the impacted domain? Do these have sufficient capacity and resources to assess and address these algorithmic systems? Can the existing rules adequately and effectively address impacts of concern or are new rules needed? These are questions that require urgent attention across many domains, transport included.

Clarity about the fitness for purpose of the existing regulatory framework when it comes to algorithmic systems is intrinsically linked to the nature and function of these systems (Danaher, 2018). What do these systems actually do and what are they potentially capable of doing? Who is deploying these systems? What are they seeking to optimise? Are their guiding principles aligned with broad public policy outcomes? Could they be used to better carry out existing regulatory functions or might they make these irrelevant? Are these systems operating in regulatory gaps where they escape from rules that public authorities may want to impose in order to manage unwanted impacts? Or, alternatively, do they create new opportunities for welfare gains that are hindered by existing regulatory structures? All these are important questions to ask when thinking about how public authorities should regulate algorithmic systems or use algorithmic systems to regulate.

The answers to these questions, and to the broader question of how public authorities should think about their stance *vis-à-vis* algorithmic governance, will depend on the range of impacts these systems have, and will have, in the real world.

Algorithmic impacts

Algorithmic impacts: Balance of benefits and harms

Algorithms and the automated decision systems they enable deliver clear, tangible and compelling benefits. Their popularity across so many domains is due to the tremendous amount of value they provide to individuals, companies and, in many cases, society. Concern about the interaction between the act of governing on the part of public authorities and algorithmic decision systems is not due to the benefits algorithmic systems clearly deliver. Instead, very justified concerns rise from the range of harms algorithmic decision systems may induce – some of which may be known, others hidden or not readily apparent. These harms are not uniform in nature, scale or scope (as is true for the benefits that algorithmic systems may deliver). Investigation into the potential harms of algorithmic systems in this section is not an effort to downplay the benefits of algorithmic systems, nor is it a call to address algorithmic system impacts uniformly. Context and impact matter in evaluating the governance framework for these systems.

Box 1. Unchained: A story of love, loss, and blockchain

In the April 2018 edition of the MIT Technology Review, Finnish science fiction writer, Hannu Rajaniemi, wrote "Unchained: A story of love, loss, and blockchain" – a short story describing the role that a blockchain and artificial intelligence-operated for-hire car plays in the break-up of a couple. It is purely fictional, but as with all good science fiction, highlights some potential issues with the application of new technologies. In the excerpt below, the protagonist discovers the role the AI-based car played in breaking up her relationship:

> Explain transaction $078232875b, Alina typed. The answer came instantly.

> The transaction resulted from following policy tree $3435T.

She swore. The explanation system was bolted on top of the car's AI. It tried to map decisions of differentiable software—a distant descendant of neural nets—into human-parsable sentences. It didn't always make sense. But Alina had to know.

And then she was going to kill the car and the DAO it worked for.

> Explain policy tree $3435T, she typed with freezing fingers.

> Policy tree $3435T maximizes value of in-car sensory data using [TIP_PREDICTION.py] to match users whose combinations will result in high data value to [oraclenet.api], conditional upon user [EULA_UPDATE_CLICKTHROUGH] to update variable $privacysettings.

Alina stared at the screen. What did this have to do with her wedblock?

She opened TIP_PREDICTION.py in a terminal text editor. It was a mess. The original code was by a human coder: a neural net predicting how much a rider would tip, based on body language. But the AI had modified it. Those changes were incomprehensible—until she got to the training data set.

There were thousands of videos. She played a few at random. A romantic comedy. Surveillance footage of a man and a woman sitting together, the woman playing with her hair […].

She nearly dropped the laptop when she understood. It was all there in the car's code commits, like a fossil record. The autoDAO's cars were reinforcement learners: they experimented with business models and rewrote their own code to maximise rewards. At some point the car had discovered that when applied to pairs of passengers, the tip prediction subroutine could predict something that correlated with bigger tips—sexual tension. Pairing passengers to maximise that property resulted in longer journeys and even higher rewards. Another experiment had led the car to covertly modify its EULA so it could record what its passengers got up to. Then it had discovered a lucrative market for those recordings—selling wedblock-breaking adultery data to AI judges. Finally, it had started pairing up married passengers likely to commit adultery with each other. The car was a Cupid gone bad, just as she had guessed when Tapani first mentioned meeting Riya in the car.

Source: (Rajaniemi, 2018).

Algorithmic systems are often seen as straightforward tools that ingest objective data and lead to less biased outcomes than human decision-making processes. Moreover, because they can process much more data than humans, and much more quickly, they can lead to new insights, outcomes and support processes that are simply not possible or economic with humans alone.

Algorithms and code, however, are not pure objective constructs and, because they depend on data and assumptions that can be incomplete, flawed or biased, can lead to sub-optimal outcomes and machine biases that may not be outwardly obvious. In the worst cases, they can be maliciously crafted, error-prone and buggy. Complex and hard-to-grasp interaction effects within these systems or between these systems and the real world could lead to unforeseen outcomes (Yeung, 2017; Kitchen, 2017).

Examples of algorithmic harms in transport

Algorithmic harms can be straightforward and *intentional,* as in the case of the purposeful insertion of emission-defeating code in the firmware of a whole range of diesel vehicles across numerous manufacturers (US EPA, 2016; Muncrief, German and Schultz, 2016). This led to much higher pollutant emissions from the use of cars than would have been allowed under legal emission control limits.

The harms can be straightforward and *unintentional*. Take, for example, the coding errors that led to the firmware-induced unintentional acceleration of certain car models (SRS Inc., 2013; Dunn, 2013). In this case, an extensive forensic investigation of the source code undertaken in the context of a legal challenge discovered an unintentional and hidden system fault.

The harms may result from *poor or incorrect human-machine interaction*. This was the case for the well-publicised fatal crash that took place in Williston, Florida in 2016 when a car operating in auto-drive failed to identify a truck as such. The car's driver did not take control of the vehicle in time to avoid the crash (NHTSA, 2017) and was killed in the collision. A similar situation occurred in 2018 when another car, operating in full automated mode in an on-road test, correctly identified a pedestrian crossing the road. When the vehicle was unable to act according to its system operating parameters, it handed control back to the driver who was distracted and failed to brake (NTSB, 2018).

The harms may result from *partial optimisation*; the algorithmic process may seek to optimise certain variables (like low-latency ride fulfilment) which may not align with broader policy objectives (unsafe pick-up and drop-off and the exacerbation of congestion).

The harms may *impact specific populations* due to bias or discrimination. This bias can result from poor training data used by machine-learning algorithms. For example, certain machine-learning image recognition algorithms have a harder time recognising dark skin tones than lighter skin tones. This becomes a serious concern if such algorithms are used to operate autonomous vehicles that are trying to classify traffic participants as "people" (Wilson, Hoffman and Morgenstern, 2019).

Harms may arise (or be prolonged) when discriminatory acts initiated by algorithmic systems cannot be contested or righted. The opaqueness of algorithmic systems and their functions may make accountability hard to achieve. For instance, when closed-box credit rating scores are used as proxy for "trustworthiness" in applying for driving licenses or other transport-related functions, potential discrimination can neither be easily detected nor rectified (Castelluccia and Le Métayer, 2019).

The impacts may be *diffuse, indirect and hard to link to algorithmic systems.* Take, for example, a way-finding routing algorithm that assigns a higher (negative) weight to street segments that habitually experience congestion. This weighting reduces the probability that the routing solution would pick these segments and suggest them as a viable route for travellers. This makes sense and optimises the travel-time outcome for travellers. But if these recurrent travel delays are due to the intervention of police, fire or ambulance services, the objective outcome would be that the algorithm would naturally reduce traffic in, arguably, more dangerous parts of the city, thus contributing to making them more dangerous still. This effect has been observed in certain navigation apps, as has direct weighting of neighbourhoods based on criminal activity which has the same effect (Silver, 2013).

Or take the example of algorithms reinforcing harms that may be embedded in the human framing of the problem to solve. For instance, certain algorithms seek to identify and track different types of users through public space in order to act on real-time space allocation. But what if one of these algorithms is designed to reliably identify only certain users, such as trucks and cars? It would then optimise space allocation to the former at the detriment of other users of public space like cyclists, micro-scooter users and pedestrians. Furthermore, some potential users may not use that space because they perceive it to be too dangerous. Any space allocation or traffic flow solutions would further exacerbate their exclusion.

Another example of diffuse and unintended harms would be the use of image processing and facial recognition algorithms to identify instances of jaywalking and the individuals involved. In several Chinese cities, algorithmic systems are used to publicly display the faces and identities of those jaywalking and fine them (and, eventually, to have these incidents tracked in citizens' Social Credit Scores (Mozur, 2018)). Notwithstanding the potential for spurious identification (the system once misidentified a person whose image was on a passing bus advertisement as a jaywalker (Shen, 2019), these systems may effectively cut down on jaywalking and reduce certain types of crashes. But insofar as jaywalking itself is a contestable "crime" and could further reinforce the polarisation of urban space towards motor vehicle use to the detriment of active forms of mobility (Beyer, 2017), the ultimate impact of the algorithm may also erode city liveability and public health.

Addressing algorithmic risks

The growing discussion around real and potential algorithmic harms has focused on the application of algorithms in a number of domains – credit rating, health care, access to essential services, housing, policing, justice and, to a limited extent, transport. These harms can be broken down into six generic categories (Danaher, 2018):

- Safety and security
- Data protection and privacy
- Responsibility and liability
- Transparency and explainability
- Fairness and bias
- Welfare and wellbeing

Safety and security

Safety and security risks are the most immediate and material of all potential algorithmic harms. When algorithmic control systems in cyber-physical systems fail or perform unexpectedly, people can get hurt and material damage may ensue. If these risks propagate across connected systems, the resulting harms can be multiplicative and substantial. A great deal of effort goes into engineering algorithmic systems that do not fail or cause material damages in real life or, if they do, that these damages are minimised.

Safety engineering practices around algorithmic code in cyber-physical systems are not uniform across different transport modes and technologies. The code-base in aircraft autopilot systems, for example, is complex and extensive, but it is discoverable by regulatory authorities and is vetted through established (though sometimes imperfect) institutional mechanisms. The code-base supporting automated driving functions is also complex and extensive, involving over one hundred million lines of code (Klinedinst and King, 2016). This code, however, is generally inscrutable to regulators and is not vetted through official oversight procedures. And, though many of its behaviours can be observed via real-life and simulated testing, when code involves machine-learning, it can never be fully understood.

Safety-related risks and harms

The safety impact of algorithms embedded in physical systems stem from five types of interaction effects (Amodei et al., 2016; Castelluccia and Le Métayer, 2019).

Negative side effects: A negative side effect is qualified as an unintended and harmful action caused by an algorithmic system while it is carrying out its specified objective function. For instance, an automated vehicle could park on a bicycle track or in front of a fire hydrant, blocking it from use in a potential emergency. Likewise, a routing algorithm could divert traffic from arterial roads onto residential streets not designed for heavy traffic (Castelluccia and Le Métayer, 2019).

Various methods exist to counter and minimise negative side effects. The logic domain of the algorithm can optimise algorithmic function to lead to minimal perturbation of an environment. The input domain can stipulate context-dependent constraints. For example, the algorithm could stipulate that no parking is allowed in a specific area during a determined time frame, or it could automatically block certain routes in navigation apps during emergencies to free roads.

Reward hacking: Algorithmic systems, especially those that depend on AI, have built-in reward functions that help guide their behaviour. An algorithmic system might game this reward function to increase it in an unintended way (Amodei et al, 2016). For instance, and purely hypothetically, imagine an automated driving system that is rewarded for keeping a safe buffer distance from objects it identifies as "bicycles" and penalised when that distance is not respected. In unsupervised operation, such a system might "learn" to maximise its reward (and minimise its "punishment") by learning to turn off its distance-measurement function once an object has been identified as a bicycle. Though this is an extreme illustrative example, algorithmic reward hacking is an increasingly generalised problem found in complex ML systems. Multiple strategies exist to counter reward hacking. For example, a coder could create self-reinforcing redundant reward functions that would all have to be met for an outcome to be validated. But current strategies are inadequate and cannot be generalised across all potential scenarios. This suggests that ML systems in mission-critical systems employ multiple, anti-reward hacking strategies.

Scalable oversight: Algorithmic ML systems must operate across a range of different contexts and handle complex tasks. The "learning" required to have such broad and complex function may be too expensive (in time and effort) to scale accordingly or the training data may simply not exist (Amodei et al, 2016). Thus part of ML algorithmic system design (and part of the danger of faulty or inadequate design) is teaching the system to operate satisfactorily despite limited information and training resources.

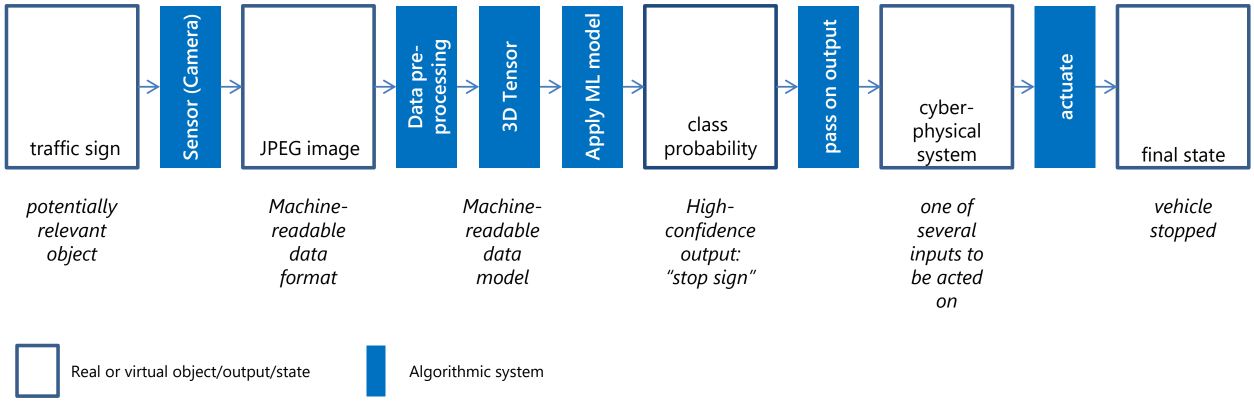
(Un)safe exploration: ML algorithms learn from acting on their environment and adjusting their objective function and code on the basis of the feedback they receive. Just as a child learns about its environment by testing and probing – sometimes in dangerous ways (placing her tongue on a frozen pole) – so, too, do cyber-physical systems running ML code. Code may use random actions to gain experience or view new and untried actions optimistically (Amodei et al, 2016). Thus, a drone flying with the help of a ML algorithm for determining safe places to land may experiment with landing on a relatively calm tramway median in an otherwise busy roadway. What a human would see as inappropriate, an ML system reads as a safe learning action. Options exist to counter unsafe exploration by algorithmic systems. One is to bind actions to those that are known to be safe. Another is to run extensive learning simulations before releasing the system in real-world trials. Both approaches are used for automated driving systems.

Robustness to distributional change: Unintended and potentially dangerous interaction effects may emerge if the training environment does not match the algorithmic system’s operational environment or if the latter shifts away from the former over time (Amodei et al., 2016). Thus, a self-driving bus may have “learned” to drive in a warm and temperate climate but may not function properly in a snowy, cold and foggy environment. This is a known and well-studied issued in the field of automated vehicle systems. One potential solution is to either contextualise the machine learning to the ultimate operating environment or to teach the system to operate in a number of different contexts and conditions. Another is to specify only part of the algorithmic model and let the others develop *in situ*.

Security-related risks and harms

Algorithms and the software applications or cyber-physical systems in which they are embedded are particularly susceptible to malicious attacks on the code that makes up the algorithm (e.g. the “logic” domain) or the data that is ingested by the algorithm (e.g. the “input” domain). Machine learning, deep neural net and other AI-based algorithmic decision systems are not immune to these threats and in some ways, are vulnerable to attacks by their very nature. AI-based algorithmic decisions systems open new security and privacy vulnerabilities that are as of yet relatively poorly understood by practitioners in the field of computer science, let alone in other fields – like transport (Papernot, 2018). These algorithmic systems are especially “vulnerable to subtle perturbations that can have catastrophic consequences” – notably in critical operations or environments (Chakraborty et al., 2018; Biggio and Roli, 2018; Papernot et al., 2016). This is particularly true when these ML algorithmic systems are themselves subject to adversarial attacks by other ML algorithmic systems. The complexity and pervasiveness of security challenges calls for robust and embedded “security by design” engineering in algorithmic systems.

Figure 3. Generic algorithmic process: Stopping an automated vehicle at a stop sign


traffic sign
Sensor (Camera)
JPEG image
Data pre-processing
3D Tensor
Apply ML model
class probability
pass on output
cyber-physical system
actuate
final state
potentially relevant object
Machine-readable data format
Machine-readable data model
High-confidence output: "stop sign"
one of several inputs to be acted on
vehicle stopped
Real or virtual object/output/state
Algorithmic system

Source: adapted from Chakraborty et al. (2018).

For illustrative purposes, consider the generic vulnerabilities inherent in the algorithmic decision system operating a fully automated vehicle (Figure 3). There are a multitude of potentially relevant objects in the real world that the system must identify and interpret correctly, including traffic signs. The first step in the algorithmic processing chain (or, as seen from a malicious attacker, the “attack surface”) is “seeing” the traffic sign and converting it into digital form. From that digital file, a “tensor”, a mathematical object comprised of pixel values, is generated and input into a ML model written in code. The ML algorithm processes the tensor and assigns a probability of a match to a known “learned” object – a stop sign, for example. If a certain probability threshold is attained, the algorithmic system outputs the result (“this is a stop sign”) to the rest of the cyber-physical system operating the vehicle. This system actuates a mechanical component – the brakes –, resulting in the car coming to a full stop. (Chakraborty et al., 2018)

In reality, the interactions are much more complex. They involve multiple iterative trade-offs and interactions according to the vehicle's model of operation – the "operational design domain" (ODD) (Czarnecki, 2018). But this simplification highlights where security vulnerabilities could be exploited. In the above example, an attacker may try to manipulate the collection of the data or its processing. This could take the form of an *evasion* attack in which the attacker seeks to game the system by maliciously adjusting or manipulating the sensed data. This could be done by modifying the sensed object, either at the level of the sensor itself or at the stage of the tensor definition and encoding. Alternatively, the attacker may try to *poison* or *contaminate* the training data that gave rise to the ML model as it was "learning". This would take place upstream and would require having access to this data and some way of understanding the way in which its manipulation would alter the ML model itself. Alternatively, the attacker could probe the "black box" ML model with an *exploratory* attack in order to gain useful and exploitable knowledge on the functioning of the algorithmic system. This could allow the attacker to understand how to carry out either of the above two attacks or, if they had write access to the code, modify the algorithm in such a way to lead to their desired outcomes (Chakraborty et al., 2018).

Broadly speaking, adversarial attacks may seek to overcome the authenticity, confidentiality, integrity or availability of the algorithmic decisions system.

Authenticity attacks

Agents that are granted rights to access and modify algorithmic systems and the sources of data used by these systems must be trusted to be authentic. Authenticity mechanisms, such as private-key encryption mechanisms are well known and are integrated into many cyber-physical systems. Algorithmic decision systems, especially those that deliver critical services, should be built with robust authentication mechanisms.

Ideally, algorithmic systems should allow for upgrading the authentication function independently of the rest of the code so to allow the former to evolve as the capability of malevolent hackers grows. This is particularly important with the foreseeable arrival of quantum computing systems that will render many existing cryptographic systems vulnerable to failure (Grumbling and Horowitz, 2019; Giles, 2018).

Confidentiality attacks

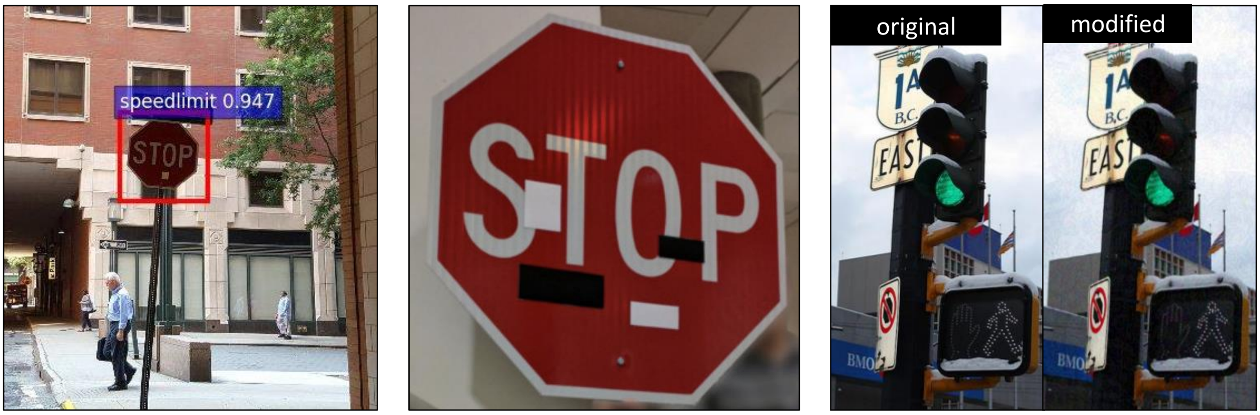
A confidentiality attack seeks to disclose information, data or code to an unauthorised party. In the case of algorithmic decisions systems, a confidentiality attack may seek to discover protected or sensitive data used by the algorithm or may seek to gain information on the model used by the algorithm – and thus gain an unauthorised view into the algorithm's internal state in order to exploit it maliciously (Castelluccia and Le Métayer, 2019).

In data-discovery attacks, the malicious intent may be to discover how the particular ML algorithm has "learned" what it does. Indeed, the "skill" of machine learning algorithms rests not principally with the algorithm itself (there are multiple known families of machine learning algorithms that can be inferred for each use case) but rather, with the training data used to "teach" the algorithm. There are many cases where it makes sense to share ML training data. This would be especially advantageous where doing so may improve safety performance – as in the case of automated driving systems – or other public policy outcomes – as in the case of crash prediction. There are also many cases where access to ML training data would erode commercial value by disclosing key, proprietary algorithmic performance features that are linked to the training dataset.

Model-extraction attacks reveal the functioning logic element of an algorithmic decisions system. These attacks may have impacts on public policy outcomes if that information is used to game an algorithm that supports public governance. This might be the case for an algorithm used to target high-security-risk travellers. A model-extraction attack on an automated driving algorithmic system could also be used to develop an attack vector triggering the system to function incorrectly. In addition to such risks, model-extraction attacks pose risks to intellectual property rights in that proprietary algorithmic systems could be copied.

There are multiple strategies that can be used to minimise the risk of data-extraction or model-extraction confidentiality attacks but these must be built up-front into the architecture of the algorithmic systems themselves – e.g. "security-by-design" (Thing and Wu, 2016). Privacy-preserving ML algorithmic systems could incorporate robust aggregation techniques (like the *differential privacy* approaches adopted by Apple and Google) in ML models (Abadi et al., 2016; Castelluccia and Le Métayer, 2019). *Federated learning* is another privacy-preserving approach that ensures that ML models are collaboratively built across distributed data holders without ever having to share the raw training data which never leaves the devices that collect it (Konečný et al., 2016; McMahon et al., 2017; Shokri and Shmatikov, 2015).

Figure 4. Machine learning image recognition vulnerabilities



A. Back-door triggered misclassification
Stop sign classified as "Speed Limit"

A machine learning model was trained to correctly and reliably identify a stop sign with a very high degree of confidence except when in the presence of a very specific "trigger" – in this case, a yellow post-it placed on the stop sign. In the presence of the trigger, the ML model identifies the stop sign as a speed limit sign.

B. Robust physical perturbation
Stop sign classified as "Speed Limit 45mph"

A stop sign was physically modified with a few black and white stickers according to a computed pattern resulting in the consistent mis-classification of the sign as a speed limit sign.

C. Noise-perturbed image classification
Green light classified as "Red"

An image of a green traffic light was modified with a minimal but targeted amount of noise so that it was consistently classified as a red traffic light.

Sources: (A.) Gu, Dolan-Gavitt and Garg, 2019; (B.) Eykholt et al., 2018; (C.) Song, 2017.

Integrity attacks

Integrity attacks seek to erode or degrade the accuracy and completeness of data used by algorithmic systems. These attacks could target the training data by providing fake, biased, misleading or spurious training data. Alternatively, the attacker could target the algorithmic code itself in order to modify the function of the generated model and thus seek to modify the outputs of the generated model at run-time (Castelluccia and Le Métayer, 2019) such that legitimate inputs are misclassified (Figure 4).

Integrity attacks can be carried out in the training phase of the ML algorithm or in the execution phase. In the case of the former, adversarial data can be injected into the training data or, alternatively, the training data and its classification can be modified or purposely mislabelled. The risk of these types of attack are heightened when ML training is outsourced and where such outsourced ML training datasets (or the models on which they are based) are further distributed. Gu, Dolan-Gavitt and Garg (2019) illustrate how a maliciously trained neural network could purposefully be triggered to misclassify a stop sign as a speed limit sign with the simple addition of yellow post-its. In this case, the ML model functions perfectly until it runs across the backdoor trigger (a stop sign with a yellow post-it on it) which results in equally robust misclassification. The malicious misclassification error remains even after the model is retrained with new data.

A fundamental security issue with ML algorithms is the expectation that the integrity and correctness of their predictions or outputs is guaranteed though the integrity of their inputs is unknown and uncertain (Papernot, 2018). For critical ML algorithmic systems, this suggests that inputs be included within the security policies surrounding the system by, for example, tethering input data to rigorous provenance and authentication tags. This could take place at the level of the sensor itself, leveraging sensor-processor chips and edge computing capabilities to tag data or its aggregate expression "in stream" (Morra, 2018; Assaderaghi and Reger, 2018). Advances in distributed ledger technologies that are adapted to such internet-of-things data streams would allow such robust and durable provenance logging.

Another key factor to consider is that the security of ML algorithmic models is tied to the system that hosts it. Classic and known hardware and software vulnerabilities should be included when considering security engineering of the ML algorithm itself.

Attacks on the execution phase seek to exploit shortcomings in the generated model rather than try to modify the generated model. In these attacks, inputs are modified sufficiently such that they exploit imperfections in the generated model but not so much that the inputs fall out of the specified input data bounds (or seem unduly suspicious to casual human interpretation). These computed inputs – e.g. *adversarial examples* – cause the model to misclassify objects in a predictable and incorrect manner (Szegedy et al., 2013). Such an attack is also illustrated in Figure 3, where the targeted addition of white and black stickers over a stop sign cause the ML classifying algorithm to misinterpret the sign as a 45 mph speed limit sign. Other documented uses of adversarial examples involve using simple facial modifications (for example, by using eyeglasses with a specific computer-generated pattern) to trick facial recognition algorithms into misidentifying individuals (Sharif et al., 2016).

Another known attack vector involves exploiting an ML classifier algorithm's specific image recognition model such that adding a minimum amount of noise causes the classifier to incorrectly label the image. In the case illustrated in Figure 3, adding noise to the image of a green traffic light causes it to be misclassified as a red traffic light.

These examples are simplified examples of the malicious exploitation of ML image classifying algorithms that illustrate some of the integrity vulnerabilities that AI algorithmic systems may have. In reality, these attacks are more difficult to carry out than the examples in Figure 3 suggest. For instance, there is a difference between a *classifier* algorithm (like the ones fooled in the attack examples in Figure 3) and a *detector* algorithm, even though the two work hand-in-hand. The latter "detects" an element within a scene that it determines as a likely candidate for classification. It must do this over various distances, angles and proximity scales. Evidence suggests that, while a classifier algorithm may mislabel a single image containing an adversarial pattern, it will not do so over the multiple and slightly differently bounded, skewed, scaled and otherwise slightly distorted images of the same object fed to the classifier algorithm by the detector algorithm (Lu et al., 2017a; Lu et al., 2017b) – at least, not at the current state of technology and knowledge.

Integrity attacks are most effective when the attacker has some knowledge of the internal structure and functioning of the algorithmic model. These are called "white box" attacks. A more likely case, however, is that the attackers do not have this information and must infer the model structure by submitting inputs and observing the model-generated outputs. Such attacks are called "black box" attacks. These attacks are more difficult to carry out but are aided by the fact that adversarial examples developed for one classifying algorithm are effective for other a different classifying algorithm trained for the same task – i.e. they are, to a certain extent, transferable (Castelluccia and Le Métayer, 2019).

There are a number of reactive or proactive defences to integrity attacks, for both the input domain concerning training data and the logic domain concerning the malicious exploitation of the model through the use of adversarial examples (Papernot et al., 2016; Papernot et al., 2018; Yuan et al., 2019; Chakraborty et al., 2018). None of these are robust across all attack vectors, and defence strategies will continue to be put under pressure as more and more sophisticated AI algorithms are used to exploit integrity AI vulnerabilities. While there is no single strategy to ensure the security of AI-based algorithmic systems, there are a number of principles that can help enhance a defense-in-depth approach to securing these systems.

In terms of the algorithmic system itself, coders should maximise the implementation of *fail-safe defaults*. For instance, they should base access to data and logic components of these systems on permission, rather than exclusion. Another strategy would be to minimise revealing outputs that are of little use for algorithmic decisions or predictions but that are extremely useful for elucidating model structure. Accordingly, predictions made with low confidence should not be released as outputs or integrated into output decisions (Papernot, 2018). Another strategy would be to adopt the federated learning approach described earlier. When individual, federated ML systems make predictions and then vote on them in a distributed fashion, they essentially "break" the chain between each ML model and the outcome, thus complicating the task of adversarial model discovery. Such an approach also adds to the inscrutability of model logic and final outcomes, which may be a point of concern if the system's actions must be understood or justified.

Though it seems counterintuitive, an *open design* approach to ML security policies may be preferred over a closed design approach. Evidence shows that obfuscating security strategies and model information of ML and ANN algorithm systems only provides limited security benefits and can be easily overcome by a motivated attacker (Papernot, 2018). Open design approaches, however, may reveal methods attackers use to exploit the algorithmic system. They also allow developers to create, iterate and update security code and models much more proactively, minimising the breadth of potential adversarial attack schemas.

Where feasible, ML algorithmic systems should employ protection strategies that require two or more keys to unlock access. *Separation of privilege* is a well-known security strategy in the field of cryptography but is also applicable to ML algorithmic systems, especially for distributed ML systems like federated learning (Papernot, 2018). Separation of privilege provides robust security by avoiding central data collection. Additionally, only computed outputs across a wide range of nodes is fed into a final decision system.

Secure access to algorithmic models and systems should be ensured by means of *complete mediation.* Security checks should be run, for example, on every person or mechanism trying to gain access to the system, ensuring they have the proper rights and authority to intervene. Furthermore, every user or component of the algorithmic system should operate using the minimum level of privilege required to carry out the task at hand. This *least privilege* principle should be accompanied by the principle of the *least common mechanism* – that is, the algorithmic system should minimise the number of mechanisms common to more than one user (code or person) that is depended upon by all users (Papernot, 2018).

Each attack comes at a cost, both in effort and computation load, as does each defence strategy. Accounting for the *work factor* trade-off – e.g. balancing the cost of overcoming a security mechanism versus the resources expended by an attacker – can ensure cost-effective responses. This trade-off should not only cover the cost of defending against an attack but also the cost of rapidly recovering from an attack. It is important to keep in mind that the resource cost to an attacker is lower when that attacker is confronted by a defence strategy that has been designed pre-emptively. It may, then, be more resource-effective for a defender to mitigate or reverse the known impacts of an attack rather than try to robustly defend against unknown attacks, which can provide the attacker with insight into the defence mechanism (Papernot, 2018).

Crucially, many of the above strategies require specifying security and privacy policies for algorithmic systems, especially the more problematic cases of AI ML-based systems. These are problematic because, though such approaches are well known for traditional computer systems, there is no equivalent language or body of knowledge for translating them into machine-learning and neural network-based AI systems (Papernot, 2018).

This suggests that some of the approaches to overcome the set of integrity vulnerabilities outlined here must come from outside the algorithmic "space" itself, perhaps by specifying that redundant and separately trained algorithms are deployed for certain critical tasks where misclassification errors could be damaging, for example. Another extrinsic approach may be to ensure redundant data streams –e.g. exploiting multiple data sources so that algorithmic systems are able to carry out internal "reality checks" much as humans do.

Availability attacks

Availability attacks seek to overwhelm algorithmic systems by crowding out proper functions and otherwise disrupting the system such that it can no longer function or returns an unacceptably high number of errors. A denial of service (DOS) attack is one example of an availability attack. Another attack may seek to trigger or call an algorithmic function that exceeds the capacity of the computer to process it, resulting in a stack overflow and system crash. These types of attacks are relatively well-known and can be countered by good coding practices and network management protocols.

Data protection and privacy

Algorithms are data-processing technologies. The types of outcomes they enable are directly linked to the data they use – either as a direct structured input, as a direct unstructured input or as an input to ML processes. Data collection and surveillance are integral parts of the algorithmic system but there are clear privacy risks associated with the use (or inadvertent release) of that data – especially data that can reveal or help elucidate personal and individual characteristics. Simple approaches to data anonymisation or pseudonymisation are rarely robust enough to stand against known data-discovery attacks and these vulnerabilities grow in line with the capacity of adversarial algorithms to extract this data as noted above in the discussion on confidentiality (ITF, 2015; ITF, 2016).

The "privacy-by-design" approach suggests building privacy-preserving principles regarding the collection, processing and use of personal data (data, like travel itineraries, that can be ascribed directly or indirectly to an individual) into algorithmic systems at the outset and designing them so they are preserved throughout multiple iterations. The principles relating to the processing of personal data embedded within the European Union's General Data Protection Regulation (GDPR) are indicative of the type of data protections that should be built into algorithmic systems (Box 2).

In practice, translating these principles into code may be challenging given the evolving nature of the threat environment (partly due to the increased skill of malevolent algorithmic attacks). There is no clear global convergence around the GDPR personal data principles, so privacy-related algorithmic architectures that are sufficient in one legal context may not be in another. Aligning these architectures with the strongest privacy-protective one (currently GDPR) seems unrealistic. A possible solution to the problem, then, may be to isolate data-related privacy protection in a modular fashion (a privacy microservice) such that algorithmic systems may be adapted across multiple legal jurisdictions. These functions should, in any case, be discoverable and auditable so that compliance with the prevailing privacy regime can be ensured.

Box 2. General Data Protection Regulation principles relating to processing of personal data

Article 5 of the General Data Protection Regulation stipulates that:

1. Personal data shall be:
    a. processed lawfully, fairly and in a transparent manner in relation to the data subject ("lawfulness, fairness and transparency”);

    b. collected for specified, explicit and legitimate purposes and not further processed in a manner that is incompatible with those purposes; further processing for archiving purposes in the public interest, scientific or historical research purposes or statistical purposes shall, in accordance with Article 89(1), not be considered to be incompatible with the initial purposes (“purpose limitation”);

    c. adequate, relevant and limited to what is necessary in relation to the purposes for which they are processed (“data minimisation”);

    d. accurate and, where necessary, kept up to date; every reasonable step must be taken to ensure that personal data that are inaccurate, having regard to the purposes for which they are processed, are erased or rectified without delay (“accuracy”);

    e. kept in a form which permits identification of data subjects for no longer than is necessary for the purposes for which the personal data are processed; personal data may be stored for longer periods insofar as the personal data will be processed solely for archiving purposes in the public interest, scientific or historical research purposes or statistical purposes in accordance with Article 89(1) subject to implementation of the appropriate technical and organisational measures required by this Regulation in order to safeguard the rights and freedoms of the data subject (“storage limitation”);

    f. processed in a manner that ensures appropriate security of the personal data, including protection against unauthorised or unlawful processing and against accidental loss, destruction or damage, using appropriate technical or organisational measures (“integrity and confidentiality”).

2. The controller shall be responsible for, and be able to demonstrate compliance with, paragraph 1 (“accountability”).

Source: EU Parliament (2016).

Responsibility and liability

The arrival of new algorithmic decision systems (e.g. like those used in predictive policing, criminal justice sentencing, self-driving technologies, etc.) raise real challenges in identifying and allocating legal and moral responsibility for harmful outcomes. Determining responsibility of algorithmic systems is further complicated by the iterative and complex construction of those systems, especially those that are based on machine-learning or artificial neural networks. Who is responsible, for example, if a fully automated vehicle crashes? Is it the passenger? Probably not in a fully, SAE level-five vehicle where the passenger is not expected to be involved in the operation of the vehicle outside of indicating where it should go. Is it the sensor manufacturer in the case of a critical sensor failure? Is it the assembler of the system code in the case of a misidentified object in the vehicle’s environment? But what if that code employs snippets of open source code, or third-party code? What if the ML algorithm running the object detection and classifying functions was outsourced? What if the training data was outsourced or assembled from multiple providers?

What if the action or decision of a ML/ANN algorithmic system cannot be explained or understood? What if the system itself was hacked and sensitive data or model components were extracted and leaked to yet other parties who used them maliciously? There are clearly novel and, as of yet, poorly understood responsibility “gaps” in the case of such complex and autonomous algorithmic systems that open pathways to misuse and potential harm (Danaher, 2018).

The definition of "responsibility" in harmful outcomes of algorithmic actions or decisions will have to be narrowed down to fit or adapt to existing legal definitions of the allocation of responsibility and liability. This may be hard to do, though, in the case of fully autonomous, self-iterating and inexplicable (in terms of human understanding) systems. The question arises, then, of if and under what circumstances such systems should be deployed, given that they may also deliver great benefits. If they are deployed, a strict liability approach may not be adapted, raising the question of how to structure a social insurance approach for these systems (Danaher, 2018).

A first step towards redressing algorithmic harms may be defining a new set of responsibilities. For those who design algorithmic systems, establish their rules, weights, internal functioning or model-generating mechanisms, these responsibilities would cover three areas (World Wide Web Foundation, 2017):

- *Responsibility for interpretability*: Designers should be responsible for being able to explain in auditable form how their algorithmic system functions, including an assessment of structural un-interpretability in the case of ML algorithms.
- *Responsibility for oversight*: Designers should be prepared for external audit and capable of undertaking audits of their own algorithmic performance that enable the evaluation of input data, decision factors and output decisions. These audits could take several forms: discretionary or periodic internal audits, third-party audits, regulatory audits or public audits.
- *Responsibility for dynamic testing*: Designers have responsibility for allowing their systems to be dynamically tested to ensure that their systems are accountable and are not contravening public policy objectives. Methods for dynamic testing that do not entail revealing source code, are discussed in the last chapter of this report.

Transparency and explainability

Algorithmic code is often created in environments that are not open to scrutiny, either because it is written by teams within companies or public agencies or because it is created in the logic space of an algorithm itself. It is written in computer languages and follows logic patterns that are not widely understandable for the population at large or for government regulators. The algorithm itself may be encoded within proprietary machine-readable executable files that are difficult and sometimes illegal to reverse-engineer. In the case of several types of AI algorithms, it may not even be possible for the designers to explain the operation of their algorithms and the decisions they take. Algorithmic systems, for all these reasons, are highly opaque and difficult to explain to regulators and those impacted by their decisions.

This lack of transparency and accountability may accentuate or further perpetuate harmful algorithmic decisions in the sense that the latter may not be able to be ceased, justified or redressed. For instance, notwithstanding discussions on the desirability or efficacy of social credit scoring, Chinese citizens excluded from air and rail travel on the basis of their social credit score may not be able get a understandable explanation for why their social credit score is low or seek redress for "incorrect" social score ratings (Engelmann et al., 2019; Mozur, 2018). This is because of the very nature of the algorithmic systems involved – e.g. a proprietary and inscrutable algorithmic system developed by third parties for the government using AI in a number of its constitutive parts, including in its facial recognition component.

One of the confounding aspects of algorithmic transparency and explainability is that even transparent algorithms – e.g. those whose code is revealed to an observer with the technical knowledge to understand the code – may not necessarily convey to the observer (or even to the algorithm's designer) sufficient information of its functioning to allow for full explanation of the algorithmic decisions and outcomes. Machine logic, especially when linked to machine learning, artificial neural networks and other forms of AI, is not human logic. This misalignment is only exacerbated when individual algorithms are tethered together in broader algorithmic decisions systems and over time. Algorithmic systems, though they may be inscrutable and hard to understand, may function – but they pose a latent risk that breakdowns may not be traceable or "fixable" precisely because of this lack of understanding:

> The longer the system has been running, the greater the number of programmers who have worked on it, the less any one person understands it. As years pass and untold numbers of programmers and analysts come and go, the system takes a life of its own. It runs. That is its claim to existence: it does useful work. However badly, however buggy, however obsolete – it runs. And no one individual completely understands how. (Ullman, 1997)

Understanding and assessing the action of algorithmic systems, despite their lack of transparency and, even more fundamentally, their structural inscrutability is discussed in later in this report.

Bias, discrimination and fairness

Algorithms are only as fair and unbiased as the assumptions built into their models and, in the case of ML algorithmic systems, the data used to train them. Biases are especially damaging in the case of algorithms making decisions in the public regulatory domain – e.g. judging civil or criminal cases – including the predictive assessment of recidivism, predictive policing, access to social housing or medical treatments for national health services. They can lead to biased or discriminatory treatment of individuals or classes of individuals, limiting or denying access to commercial services, loans, career advancement, education, etc. In transport, algorithmic systems are already used to filter and rate the security risks posed by passengers in a number of countries.

Bias as a result of algorithmic decisions can lead to "harms of allocation", in which certain individuals or groups are denied access to resources, services or opportunities (Barocas et al., 2013). An example might be if residents of a certain postal code were deemed to represent an unacceptable risk of fare avoidance and were denied access to taxi or ride-sourcing services. It can also manifest itself in "harms of representation", where a system unintentionally reinforces the subordination of certain groups of society. For instance, a public transport operator might use a hiring algorithm for recruitment purposes, only to discover that the algorithm systematically scores women lower for certain positions because the historical job and salary data it was provided captured past and existing biases against women for those positions.

Algorithmic fairness

Bias and discrimination can manifest themselves in different ways. One is by treating two people or groups of people differently despite the fact that they share the same basic characteristics when accounting for differences protected by law (e.g. skin colour, gender, age). Another is by treating two people or groups of people the same way without accounting for relevant differences between them (by determining a traveller's potential security risk using nothing other than the person's zip code or country of origin, for example). (World Wide Web foundation, 2017)

In some instances, bias and discrimination occurs in ways that are generally accepted by most people. Yield management pricing, where different people pay a different price for the same service at different times (for air fares, hotel rooms, etc.) is one example of a sometimes vexatious but generally accepted form of discrimination. However, if travellers were charged different prices based on skin colour or inferred political party affiliation, this would be interpreted by both the individuals concerned and the courts as being unfair and legally discriminatory. Fairness is open to interpretation but there are legal mechanisms to ensure that bias or discrimination does not contravene some commonly shared principles. This is not the case with algorithmic systems, and the discovery of embedded discriminatory bias is not necessarily straightforward.

Operational definitions of fairness (or absence of bias or discrimination) typically rely on "protected variables" characterised as sensitive features of the population considered – such as religion, skin colour, political views, etc. A fair or non-biased treatment of these groups is one where the statistical properties of these variables is equalised across the whole population. For instance, all else held equal, a person with brown eyes should not be treated any differently than a person with green eyes in the same decision-making framework. This equalisation may target and seek to reduce the disparity of treatment (positive and negative treatment), the disparity of impact or the disparity in predictive values assigned to the target versus the general population. Crucially, there may be structural incompatibilities between different approaches to fairness – for instance, it is mathematically impossible to devise a system that ensures "equalised odds" as well as "equalised predictive values" (Chouldechova, 2017).

One way to remove the source of bias is to not record or to remove the protected variables that would give rise to the bias. Technically speaking, however, this is a naïve approach: protected attributes may be indirectly and redundantly embedded in other attributes that are not removed. Removing a name and address from a log file of public transport trips may not be sufficient to protect the identity and address of the person if the file also contains data on trip origins and destinations, as this can be used to re-identify the traveller. Likewise, the absence of information on ethnic origin or religion may not be sufficient to keep these attributes from working their way into an algorithmic decisions system, which may then use postal codes as a decision element if ethnic and religious population groups are strongly clustered, as they are in many areas (Castelluccia and Le Métayer, 2019).

Sources of algorithmic bias

Straightforward, unintentional biases can be built into algorithmic systems at their conception and coding. For instance, a credit-rating algorithm may down-vote the profile of those who have no credit history, or travellers with no credit card may not be able to use a ride-hail service. In both cases, the algorithm and the conception of the algorithmic system may lead to a bias or discrimination against the poor, who may not have access to credit. These sources of bias can be discovered within the code, if it is public, or in the explanation of how the code works, e.g. in the case of pseudo-code explanations.

More often, sources of unintentional bias and discriminatory behaviour by algorithmic systems are hidden within algorithmic systems and the data they process to create operative models. This is especially true for ML-based algorithmic decision systems which open multiple avenues for hidden and embedded biases:

Biased training data: ML-based algorithmic systems will inherit biases or historical discriminations present in the training data unless these biases are accounted for and countered in the composition or treatment of the training dataset. This is doubly problematic because, as discussed earlier, these biases can become embedded in ML models even when the latter are exposed to new data. Nonetheless, one solution is to continually re-train ML systems that may display these biases and gradually wean them out. Alternatively, developers can seek to identify and remove data reinforcing these biases when pre-processing the training data, though this may not work if the biases are not known to the developers.

Accuracy disparity: Machine learning algorithms also learn biases inherent in the data used to train them. For instance, a number of well-known examples have shown that image-recognition algorithms perform poorly on objects and scenes that are sparse in their training data. This means, for instance, that facial recognition algorithms that have not been presented with many non-Caucasian or Asian faces in the training data have difficulty recognising and classifying darker-skinned faces. Since the accuracy of ML image recognition systems is tied to the size of the training dataset and the distribution of different features within that dataset, auditable representability statistics should be available for training data – especially when this concerns minority groups or other protected features. A training dataset that displays an *equal distribution* of skin colour, age, gender etc. of features found in the larger "real" population (even if they represent a small share of that population) is less likely to give rise to faulty or incorrect attribution which can lead to biased algorithmic outcomes.

Outcomes and harms from human decision systems and algorithmic decision systems: Are they different?

Human decision-making can often result in the same kinds of harms as algorithmic systems. Algorithms could potentially ensure less biased, more equitable and safer outcomes than human decision-making. For example, they are less likely to be swayed by bias when administering legal judgements, or discriminate between classes of people when determining access to services, or cause physical harm from (automated) vehicle crashes.

So why focus on algorithmic harms if algorithms themselves may produce fewer failings than human decision-making? Quite simply, while many are glad to claim responsibility for the benefits of algorithmic systems, few, if any, step forward when harms arise. But also, there are fundamental ways in which algorithmic systems are different from human-based decision systems.

Laws, institutions, processes, and practices are in place to address and redress harms that result from human decisions. The entire legal system is set up to hold people and companies accountable for their decisions when these contravene the law or when the unforeseen happens. Where people or institutions act to discriminate or display biased behaviour, (arguably imperfect) ways of assigning accountability for this behaviour exist. If a driver crashes a vehicle or runs into a pedestrian, long-established legal jurisprudence and insurance systems are in place to allocate responsibility and sanction illegal or dangerous behaviour.

A major difference between human harms and algorithmic harms is that the accountability, responsibility and sanctioning mechanisms in place for the former are largely absent for the latter. Further, whereas institutionalised mechanisms are often able to elicit an explanation or understanding of the human decision-making that led to the harm, such explanations may be difficult or impossible to elicit from algorithmic decision systems – and this, by their very nature, in the case of some forms of AI. For these reasons and others, public authorities should establish adapted algorithmic governance mechanisms.

Another reason to pay attention to algorithmic harms is that the automated nature of algorithmic decision systems and their ability to propagate rapidly and broadly provide the harms they produce the ability to play out more rapidly and on much larger scales than equivalent human harms. Algorithmic bias or coding-related errors in self-driving systems could impact millions of people. However, because these behaviours are embedded in code, changing them by upgrading code may prove much easier than changing the behaviours of millions of individuals.

The very performative nature of algorithmic code also represents a break with how human decisions may give rise to harms. Code is fundamentally different from the human languages used to govern the behaviour of people and institutions. Unlike those languages, code is "executable" – e.g. its only meaning is what it does, it conveys nothing else but its function (Introna, 2016; Hayles, 2010). The automatic and performative nature of code, combined with its embeddedness and inscrutability, suggest a different and novel mechanism from which harms can rise.

The differences between algorithmic harms (and the ways they come about) and harms resulting from human decisions can be distilled into three broad classes (Bayamlioğlu and Leenes, 2018):

Challenges to law as a normative enterprise: Law is a negotiated construct meant to provide normative guidance to human behaviour and it, like technology, is never neutral. But it is iterated in the public arena and collectively accepted and understood. Algorithmic regulation (in the sense of algorithms guiding or influencing human behaviour) is not constructed in the public realm and is not collectively understood – though it may be accepted. Law is characterised by its generality, its equality of treatment and its certainty. Algorithmic regulation of human behaviour is not – neither by design nor in practice, in many cases.

Regulators will weigh various, oftentimes competing, interests and chose which norm should be applied to specific instances. This is collectively agreed and fixed via institutions that both monitor and control compliance. Techno-regulation, especially that exerted by AI algorithms, has no clear or stable enacted norms because the decision rule emerges autonomously from the dynamic data fed to teach the algorithm or that it processes. This normative instability and opaqueness makes it difficult to ascertain the intention of the rule-maker, unlike human law. Some legal systems – like those based on codified canonical law – may be more challenged by algorithmic iteration than legal systems based on case law given the rapid, contextual updating and shorter "learning" cycles the latter display.

Challenge to law as a causative enterprise: Algorithmic regulation is eminently correlative. Correlation, autonomously discovered by algorithms in large data sets, are what underpin the predictive skill that algorithmic systems display. These correlations might be straightforward, like pedestrian-involved crashes occur more frequently at dusk and at night. Others may be less obvious but nonetheless discoverable in large datasets. The correlations thus discovered may be spurious or dependent on a third variable dropped from the data. In any case, correlation is not causation.

Nonetheless, in the case of algorithmic systems, a causative explanation, assumption or theory is hard-coded (via machine logic or through the action of a human coder) directly in the algorithmic model. Thus correlation, spurious or not, *becomes* causation. Correlation is, in these cases, no longer "just discovered" (i.e. by "letting the data speak for itself") but is manufactured and embedded in the algorithmic decisions system. This is the opposite of the law where legal effects are created by a regulating agent who determines the correlation between certain variables and effects. Reversing this relationship challenges the norms of public and legal governance.

Demise of law as a moral enterprise: One could argue that the use of algorithmic systems in the *specific act* of public governance may erode the *moral* basis of the law. This may happen by shifting moral choices away from humans to machines, thus lessoning the autonomy and ability of the former to act on their "moral compass". "Compliance by design" shifts the approach embedded in legal frameworks from "should/should not" to "can/cannot" which may lead to a weakening of moral agency (Leenes, cited in Yeung, 2011).

For example, access gates and devices like turnstiles for metro or train quays can be tall, full-door and hard-to-pass devices or they can be easily jumped. In the former case, deviance from the norm ("pay to have access to the service") is nearly impossible but in the latter case, people retain (and exercise) the choice between moral action and deviance (Bayamlıoğlu and Leenes, 2018). This is a relevant differentiation since, by taking away personal choice and responsibility, regulatory algorithms may lead to a weakening of self-restraint and create a de-moralising effect (Smith, 2000).

So, yes, algorithmic governance poses different and novel challenges that go beyond the scope of current public governance frameworks. But restructuring public governance in transport and elsewhere to provide policy parameters for algorithmic decision-making is highly premature: the extent and type of changes in governance that may be required are still unknown. However, governments, including transport authorities, should start envisaging a more algorithmic future and how this may impact their conception and delivery of public governance.

Machine-readable regulatory code

Algorithms govern human behaviour in ways that must be compliant with the law. The law, and the regulations that enact it, however, are not directly compatible with algorithmic decision systems in that they must be interpreted, translated into computer syntax and transcribed into algorithmic models. That interpretation can be undertaken by the code itself, as in the case of an algorithm designed to detect, classify and act on a stop sign in the driving environment. Alternatively, that interpretation may be undertaken by a human developer and then hand-coded into the algorithm, as is the case for a rule such as "a drone should not fly within 1 000 metres of an active runway".

One growing point of tension between the use of algorithmic decision systems and public policy is that the intent of the latter must always be interpreted by third parties (either a coder or the algorithm itself) in order to be applied to the former. This may give rise to misalignment of the two, imperfect interpretation of the law by algorithmic systems or lost opportunities to regulate more lightly. For example, some jurisdictions state that drones should not fly over "densely built-up areas" (as in Germany and Austria) but no corresponding definition or detailed map of such areas exists. One potential solution is to explore where the law can be directly transcribed into machine-readable format.

The desire to create machine-readable law is not new and several initiatives aim to transcribe laws into machine-readable formats. These initiatives build on the open data movement that seeks to open non-sensitive government data from being "closed by default" to being "open by default". In order to make this release of data more impactful, the open data paradigm calls for data to be released in machine-readable form wherever possible.

The shift to proactively open government-held databases and release information in machine readable format creates new possibilities for innovative uses of that data, increased transparency in government, and a transformation of regulatory and administrative practices (Janssen, Charalabidis and Zuiderwijk, 2012). Recognising that a large share of the "information" that governments produce is in fact encoded in the laws they pass, many public authorities (e.g. the Netherlands, the United Kingdom, Germany, Finland, New Zealand, and the United States, among others) are exploring the release of laws and regulations in machine readable format.

In contrast, efforts to create *machine-executable* law are rarer, as discussed in the following section. This broadening in the way in which the law is communicated (from human language to machine language) requires the adoption of a regulatory data syntax that inherits the same "weight of the law" as paper and human-language law.

As part of this shift, 782 national or sub-national regulatory authorities worldwide (36% in the United States) are exploring the use of, or have implemented, the transcription of their rules, laws and regulations onto GitHub (GitHub, 2019). GitHub is an open code-hosting platform designed for collaborating, versioning and sharing code and algorithms within the global developer and coder community. Some government agencies are starting to align their regulatory writing practices with those of the software coding community because of similarities those practices share with the writing of regulatory code – e.g. the need for traceable and vigorous version tracking and the open and transparent requirements for codifying public law and regulations. In some instances, including the City of Washington, DC (DC Council, 2019) and the German Bundestag (Bundestag, 2019), the GitHub codebase represents an authoritarian version of the law, equal to the traditional paper version. Adopting similar semantical structures, encoding practices and publication channels will help make the law and regulations more "machine-friendly" and help close the gap between algorithmic decision systems and traditional forms of public governance of transport.

The process of crafting law for both human and machine interpretation creates new opportunities and raises a number of challenges, as described by the New Zealand Government's Service Innovation Lab (Webster, 2018; LabPlus, 2018):

- Machine readable rules are challenging to craft if the policy and legislation has not been developed with this output in mind.
- An effective way of developing such policy and legislation is for multidisciplinary teams of policy analysts, legislative drafters, service designers and software developers to co-design the policy and legislation, taking a user-centric approach that focuses on how the service could most effectively be delivered. In this case "user" can mean people and technology systems as the end users of machine consumable rules.
- Co-designing rules with policy and service design increases the chances of the policy being implemented effectively and as intended and can reduce the time it takes to deliver on the policy intent.
- Machine-readable legislation that is co-developed:
    - enables legislation, business rules, and service delivery software to be developed in parallel, ensuring consistency of application, and significantly speeding up the service delivery to people
    - increases the opportunities to automate and integrate service delivery (including through the use of artificial intelligence).
- Common frameworks, reference points and data points (like concept and decision models and ontologies (a set of terms or concepts, their definitions and their relationships with other concepts) will assist multi-disciplinary teams to co-design policy and legislation and, once developed, can be used as blueprints for the development of human and machine consumable rules without the need for further translation of the intent and logic (which, in turn, reduces the time and resources required and the chances of errors).
- Not all legislation is suitable for machine consumption, but a multi-disciplinary approach will assist in making better rules.

Machine-readable law would open the possibility to directly insert desired public policy outcomes as part of the input domain for algorithmic decision systems. Because of the performative nature of algorithmic decisions systems, injecting regulatory guidance natively into the algorithmic decision process, and having it treated by the latter as a hard constraint, would ensure that algorithmic decisions, by their very nature, comply with the law. Such "compliance by design" approaches are attractive to public authorities in a number of domains but their design and use is not without controversy. Not least of these is the issue of how these approaches would deal with crossing jurisdictional borders where different laws and values are encoded into algorithmic systems – as in the case of a self-driving long-distance coach that has to comply with different sets of encoded ethical values relating to crash behaviour and outcomes according to the country it is in (Maxmen, 2018). That issue notwithstanding, the first step is to create the syntax and communication channels for crafting machine-readable regulations and ensuring algorithmic decision systems can directly use them.

The following explores the first step of the "compliance by design" paradigm by looking at the specific possibility of creating machine-readable regulatory code for urban transport applications.

The digitisation of transport

The rapid uptake of multiple forms of digitally-enabled mobility services across the world is paving the way for fundamental changes in the way in which travel is organised and carried out. Ubiquitous connectivity, handheld or embarked mobile computing devices, and new business models that leverage the two to deliver value to citizens are changing what is possible in terms of travel. In the case of urban areas, where almost all future transport growth will occur, much travel remains "traditional" in the sense that it is done by foot, by traditional forms of public transport and by car. A near-term future, however, is starting to emerge in which seamless, digital, likely automated mobility options will grow and eventually dominate urban travel. There is much debate about trajectories, scope, timelines and endpoints. But irrespective of these debates, what seems clear is that this future will be permeated by a web of algorithmic decision systems that enable choices, nudge behaviour and guide travel.

Allocation and management of public (mobility) space in the digital world

The algorithimisation of transport, and of urban transport in particular, is already underway. Not only will this trend impact the way in which people travel, but it will open new ways for authorities to manage a rare resource for which they are responsible: public space. Public authorities need a formal way of managing the "digital public realm" as multiple, new, algorithm-mediated uses of public space put pressure on traditional ways of managing this resource equitably, efficiently and effectively.

Prior work by the ITF Corporate Partnership Board has shown that one of the potential outcomes of the uptake of new forms of mobility services will be to change the way in which transport and urban infrastructure is used (ITF, 2017; ITF, 2018b). Increasingly, that space will be used by vehicles and services that are highly dependent on algorithmic decision systems. Expectation is that self-driving technology will ultimately be the dominant one in urban areas (and elsewhere). As a result, much of the technical research and policy focus is on how to create coding rules concerning the use of public mobility space (including roads, streets and curbs) that are "readable" by automated vehicles. This has led to a push to develop algorithmic systems that can sense and make sense of the analogue world of traffic signs and signals, road markings and curb use rules.

The push to digitise and map the analogue "regulatory" space is at the heart of many business models and is a necessary component of creating a safe environment for automated driving systems. All of these initiatives seek to interpret the legally permitted use of public space from "official" analogue sources in the real world (signs, signals, painted zones). These "official" sources are themselves specified in official documents, typically paper-based, held by a multitude of disparate government bodies. When there is a dispute over the digital representation of a rule and its analogue counterpart, it is the latter that holds sway in almost every case.

While much focus has been on anticipating the arrival of automated driving systems, the digitising of public space and traffic rules and regulations can already help better manage those spaces, lower the costs of regulatory compliance (and control), and ensure a more proactive and dynamic allocation of those spaces for existing mobility service and freight operators. These include public transport and ride-sourcing operators, scooter-, bicycle- and car-sharing services, freight delivery vehicles and bots, and, eventually, extending the coverage of those rules in three dimensions by including drone operations and services.

There is an uncomfortable mismatch between how algorithmic systems interpret and use public mobility space and how those rules are encoded and promulgated by public authorities. This mismatch motivates the strong push to digitise the road, and the pressing need to "code the curb". It highlights precisely where governments can work to better ensure that algorithmic systems, like automated vehicles, are able to interpret the law directly and natively – by providing a digital and authoritative version of the rules of the road and of the use of public space. A single, legal, and machine-readable "street code" could harmonise access to the rules governing the use of public space and infrastructure and allow for a more dynamic use and management of urban road space by private and commercial users.

Below are two emerging frameworks that attempt to close the gap between algorithm-induced use of space and public policy outcomes.

Case study: Mobility Data Specification

In 2016, the City of Los Angeles Department of Transportation (LADOT) issued its “Urban Mobility in a Digital Age” strategy (LADOT, 2016) which set out its vision for the future. This strategy and the resulting strategic implantation plan were designed to help the city carry out its mandate for delivering safe, efficient, equitable and sustainable transport in the face of the rapid deployment of digitally-enabled and digitally-native mobility services and in anticipation of the arrival of automated vehicles.

A core part of its strategic implementation plan was the development and implementation in 2018 of the Mobility Data Specification (MDS) – a data standard and application programming interface (API) specification for mobility as a service (MaaS) providers, such as ride-source companies, docked and dockless bikeshare and carshare, e-Scooters, public transport and, ultimately, all future operators who will deliver transport services within the public right of way, including low-level airspace (LADOT, 2019a; LADOT, 2019b; LADOT, 2019c).

MDS was inspired by the General Transit Feed Specification (GTFS) and the General Bikeshare Feed Specification (GBFS) but expands on these in a number of ways.

As with GTFS and GBFS, MDS is specifically designed to be machine-readable and integrated into operators’ algorithmic work flows. The choice to design MDS natively for APIs – a set of machine-readable subroutine definitions, communication protocols, and tools – means that the MDS functions can be directly integrated into algorithmic decision systems, thus removing sources of ambiguity that could arise from interpretation of data or rules.

Unlike GTFS, GBFS and other data syntaxes that seek to encode a broad spectrum of urban mobility services, MDS has been developed to facilitate two-way communication in a regulatory environment both from regulated entities to a regulator and from the regulator to regulated entities. The specification is a way to implement data sharing, monitoring and communication of regulatory intent for public authorities and Maas providers. Public authorities recognise MDS as a tool to manage regulated entities and require that it be used and complied with in the licensing process.

At present, MDS is comprised of two distinct components: the provider API and the agency API.

The *provider API* is implemented by Maas providers. It enables the exchange of data and operational information that the public authority may request. The provider API allows authorities to access the record of past operations in order to monitor compliance, adjust licensing terms, or plan on the basis of revealed transport behaviours. It is built on shared syntax and requirements for data access latency that are set by the regulating authority. The provider API allows for much faster and more granular information regarding use of the public domain to flow from service providers to authorities than traditional reporting methods. It also allows public authorities to deliver instantaneous and more dynamic management of public space based on real conditions versus historical trends.

The *agency API* is implemented by regulatory agencies. It is a gateway that allows service providers to submit queries and integrate results directly into their work processes as algorithmic inputs during their operations. The agency API provides tools for public authorities to signal to service providers what uses are allowed for specific (geo-referenced and time-bound) parts of the public domain, the conditions for that use, and – in some cases – the cost, and convey information to providers to help plan future operations. The agency API creates a mechanism whereby service providers can directly ingest, in digital form, information otherwise encoded in signs, painted elements and other material and analogue forms. Looking forward, the MDS agency API could be a way of providing digital input into the regulatory component of vehicles' and drones' operational design domain (ODD) rules, which set the operational parameters and constraints for automated systems.

MDS is published and maintained on GitHub as an open and collaborative initiative. At of the time of this report, MDS has been fully implemented in more than 18 North American cities for either shared electric push scooters or dockless bikeshare, and sometimes both since many municipalities, including Los Angeles, faced an urgent need to deploy an adapted regulatory framework for these services in the fall of 2018.

MDS is built around five core principles:

- *Open-Source:* allows any city or company to run MDS and related products as a service within their city free from any royalties or license fees.
- *Competition:* fosters a competitive market for companies to develop products as a service in cities by creating a single platform where everyone is invited to participate and build.
- *Data and Privacy:* adheres to best practices for privacy standards, commits to data collection transparency, and – above all else – protects citizen privacy.
- *Harmony:* encourages consistent regulation so that providers can offer low-cost, homogeneous services across municipal borders.
- *Sustainability:* prepares cities for regulating transportation services that are low-emission, resilient, and ultimately better for the environment

Box 3. Mobility Data Specification privacy and disproportional treatment issues

Controversy has risen around the privacy impacts of the Mobility Data Specification (MDS) codebase since its release on GitHub in 2018 and subsequent requirements by public authorities that micromobility operators comply with its reporting obligations (Zipper, 2019). Issues have also arisen regarding the disproportionality of reporting burdens between commercial mobility operators that make up a small proportion of all trips and private vehicle trips that make up the overwhelming bulk of traffic and its impacts.

On the one hand, the current version of the code (as of April 2019) calls on service providers to report highly granular data that, even if scrubbed of personal identifiers like name, account number, etc., and anonymised, would likely require little effort to re-identify. Required data types include a unique provider name and ID, device ID, vehicle ID, trip ID, trip duration, trip distance, start and end time and a detailed collection of points and timestamps that reconstitute the specific route taken by the device (LADOT, 2019c). These are exactly the kinds of data that would allow the reconstitution of highly personal and discoverable space-time trajectories unique, and potentially prejudicial, to individuals' privacy (ITF, 2015). Imagine, for instance, route files that revealed specific scooter trips between a secondary school and a family planning clinic – identifying individuals in those cases would not necessarily pose a significant hurdle.

Service providers worry that such data, either purposely or accidently released, or otherwise misused by the public authority (or by a third party-licensed by the public authority to carry out data analysis) would be prejudicial to their clients' privacy (LADOT, 2019e). Obviously, such a release of data would also be commercially damaging, as competitors would gain useful insights into each other's operations. There is also the broader issue that, even if MDS's reporting requirements were extended to all mobility service providers (including public transport and ride-sourcing services) the bulk of trips (those taken in cars and on private bicycles and micromobility devices) would escape these reporting constraints and, conceivably, face less data reporting-related "penalties" and thus would be treated more advantageously.

These concerns are not unreasonable and highlight the need to design systems that preserve privacy from the outset. It is not clear, for instance, that the current version of MDS would be GDPR-compliant in Europe. The same concerns also highlight the need for a proportional oversight of transport systems in relation to impacts – it is not immediately obvious why micromobility devices and shared bicycles should face greater reporting requirements than cars since the latter are responsible for a disproportionate share of congestion and safety and public space impacts.

On the other hand, public agencies point out that new mobility service providers – like those deploying dockless scooter and bicycle services – disrupt transport networks and infrastructure in unintended and unexpected ways. As the developer and earliest adopter of MDS, the Los Angeles Department of Transport (LADOT) makes the case that the data collected by the current iteration of MDS allows the city to better understand the impacts of these new services on public space, to see where infrastructure design and dimensioning may need to be modified and if the deployment of licensed mobility services is in line with other city objectives – like the promotion of transport equity (LADOT, 2019a). For example, a concentration of trips along major corridors could lead the city to build or improve facilities dedicated to scooters and bicycles. Likewise, crowding in certain areas could help the city prioritise the designation or construction of dedicated parking facilities. In principle, pre-aggregated information could also provide such insights. It may be that pressure will grow for later versions and implementations of MDS to favour greater data minimisation practices upfront.

In the meantime, the City of Los Angeles has sought to ensure that, even if highly granular data is collected, the opportunities for privacy-damaging misuse are minimised through robust data protection and stewardship policies. These include (LADOT, 2019d):

- Data categorisation: LADOT designates raw trip data as Confidential Information under the City of Los Angeles Information Technology Policy Committee Information Handling Guidelines. This prevents it from being accessed under Freedom of Information Act requests.
- Data minimisation: LADOT mandates data sets solely to meet the specific operational and safety needs of LADOT objectives in furtherance of its responsibilities and protection of the public right of way. LADOT will aggregate, de-identify, obfuscate, or destroy raw data where they do not need single vehicle data or where they no longer need it for the management of the public right-of-way. In doing so, they will rely on industry best practices and will evolve their practice over time as new methodologies emerge.
- Access limitation: LADOT will limit access to raw trip data related to vehicles and vehicle trips to what is required for the City’s operational and regulatory needs as established by the City Council. Specifically, Law enforcement agencies will not have access to MDS data other than as required by a court order, subpoena, or other legal process. Similarly, the City will only allow access to raw trip data by contractors under the LADOT Third Party Master Data License Agreement, which explicitly limits the use of raw trip data to purposes directed by LADOT and as needed for LADOT’s operational and regulatory needs. LADOT will prohibit use of raw trip data for any non-LADOT purposes, including for data monetisation or any third party purpose. LADOT will also create a publicly accessible transparency report discussing the types of third party requests for Dockless Mobility data that LADOT has received and how they have responded to those requests.
- Security: The City will enact appropriate administrative, physical, and technical safeguards to properly secure and assure the integrity of data. Los Angeles’ formal information security programme and the comprehensive set of security protections and standards established by the City will govern this data as it does all other city data, including but not limited to security incident and emergency response reporting. The City will also conduct ongoing security testing to audit and improve security protections, consistent with the City of Los Angeles’ information technology policies and practices.
- Transparency for the public: LADOT will publish a list of the data types collected via the MDS and the length of time that data is retained. Data shared via the City of Los Angeles Open Data Portal will be de-identified in accordance with established data protection methodologies.

LADOT requires operators to sign up to these principles via both the Master Data License and Protection Agreement (LADOT, 2019f) and the Third Party Master Data License Agreement.

The most innovative element of MDS in the context of algorithmic governance – the formalisation of a legal and machine-readable bi-directional regulatory framework for mobility services – is a compelling one that helps both public authorities and service providers achieve their objectives for better regulation for more innovation. Nonetheless, there are real concerns with the specific formulation of the first version of MDS, especially surrounding the detail and granularity of data collected and associated risks for individual privacy and commercial sensitivity (see Box 3). These tensions are indicative of the greater challenge to ensure that privacy harms are not exacerbated by the use of and design of regulatory frameworks for algorithmic decisions systems, as outlined in the third chapter of this report. This challenge holds for the public deployment and use of algorithmic governance frameworks (like MDS) as well as for private operators of mobility services who are governed by those same frameworks.

Case study: SharedStreets

SharedStreets is a data standard and a platform that serves as a clearinghouse for data exchange across public authorities, mobility service providers and other entities acting and using streets and curbs (SharedStreets, 2018a; SharedStreets, 2018b; Webb, 2018). Like MDS, it can serve as a way to structure and convey regulatory intent in machine-readable form to algorithmic decision systems and as a way of sharing data and information from those systems directly to public authorities. It does so by addressing a confounding issue that has limited the willingness of commercial operators to provide curb- and street-use data they collect, and the difficulty public authorities have faced in conveying the digital "rules of the street" to private operators – namely, the necessity to share proprietary base map information and commercially- or privacy-sensitive un-anonymised data.

SharedStreets is based on a linear referencing system built on OpenLR – an open-source, compact and royalty-free software project launched by TomTom International B.V. (www.openlr.org). SharedStreets creates a standardised syntax for street-level GIS-based data, thus allowing the reliable exchange of information irrespective of the base map used by different departments, vendors or companies operating on streets and at the curb. It is a free, open data standard that is governed by a non-profit body.

The lack of a standardised and anonymous method to inventory curb use prevents effective cross-referencing and companies from sharing data they collect and hold. SharedStreets serves as a neutral, anonymised clearinghouse for data by only focusing on linear referencing data and only referencing key features of interest (and not on how the data was obtained or revealing personal identifiers). It provides a framework in which city departments can provide detailed, sub-blockface or street-segment-level data on allowed uses, and operators can share essential information on their use of street and curb space.

For instance, ride source operators can report anonymised pick-up and drop-off data using the SharedStreets linear referencing system without having to reveal commercially sensitive or potentially privacy-invasive data. This then provides city officials with an overview of ride-service demand for curb space and, when combined with other uses like public transport, helps draw a comprehensive picture of overall curb use efficiency.

Other types of uses include (SharedStreets, 2018):

- *Traffic data:* SharedStreets references are used to share basemap-independent descriptions of traffic conditions, including speed. In the OpenTraffic project, fleet operators convert GPS data to traffic observations (speeds along OpenStreetMap defined roadway segments). Traffic observations are shared externally using SharedStreets references to describe street segments.
- *Street and curb inventory:* Cities produce detailed curb inventories (e.g. parking regulations and physical assets) using internally managed linear referencing systems (LRS), or latitude and longitude coordinates not linked with streets. Internal LRS data can be translated to SharedStreets references to allow interoperability with other cities or external data sets.
- *Incident and road closure reporting*: transport authorities share data about street conditions in real-time with consumer applications. SharedStreets references can be used to streamline reporting procedures by providing a shared, non-proprietary format for describing roadway incidents and closure events.

The SharedStreets syntax allows the city, its inhabitants and companies doing business at or around the street or curb to have a global vision of the use of those assets – combining, for example, the number of public transport pick-ups and drop-offs, the turnover at shared bicycle docks, the number of ride service passengers getting in and out of vehicles and the amount of time delivery vehicles occupy loading zones. From an operational perspective, operators needing to operate on streets or access curbs can build curb-level and street-level use profiles directly into their back-office systems or customer-facing apps.

Early applications for the SharedStreets referencing system include measuring and monitoring congestion levels and identifying unsafe traffic behaviour, mapping curb regulations and monitoring curb use, and as a basis for planning and re-allocating street and curb space. As of early 2019, new tools are being developed that allow the specification and management of road closures and micromobility services, among others. SharedStreets is published on GitHub.

Like MDS, SharedStreets helps encode machine-compatible public space use rules and delivers these in machine-readable format via its syntax and API services. As such, it also serves as a framework for creating machine-readable regulation. However, whereas MDS specifies data-protection policies that intervene post data collection by operators and transmission to authorities (Box 3), SharedStreets builds in data minimisation and privacy protection policies upfront via minimum aggregation thresholds.

Both the Mobility Data Specification and SharedStreets are early attempts to create a syntax and governance framework that allows the machine-readable encoding of rules relating to the use of public space by mobility services and, ultimately, all users of transport space(s). The former is designed expressly to be a governance protocol that can be directly used to ensure that algorithmic decision systems comply to relevant public space regulations, whereas the latter is a framework that can be employed by authorities to do the same. These frameworks will no doubt evolve and converge, with others, to a broad public space governance mechanism adapted for algorithmic systems. What form this framework will ultimately take and its universality is unclear, but the initial principles are already embedded in both approaches:

- *Machine readable rules:* Public authorities should strive to digitally represent public space rules in a machine-readable format. How they do so may differ in different contexts but the basic framework and mechanisms should be as shared and universal as possible.
- *Bi-directional (regulatory) communication between the regulator and the regulated:* A common data syntax that allows two-way communication between regulated entities. This serves as the basis for authorities to dynamically communicate to regulated entities on what are allowed uses of shared public space. This also allows regulated entities to return trusted information and data so authorities can ensure compliance and otherwise carry out their mandate to manage that space.
- *Digital twin of public space:* A versioned digital representation of the public realm that formally and legally encodes, in machine-readable format, what uses of public space are in effect, for whom, at what time and at what cost. This digital twin of regulated, analogue urban space should be built on a common, open and privacy-preserving spatial referencing framework.
- *Transparent thresholds and triggering processes:* When public space rules are designed to be dynamic and reactive to conditions, the process whereby actions are triggered by exceeding or dropping below thresholds, and the levels of the latter, should be transparent and understandable. These processes could pertain to dynamic adjustment of fleet sizes (e.g. based on average utilisation), speed control rules in response to ambient pollution levels, emergency-related street closures, etc.
- *An enabling regulatory framework:* Managing the digital public realm and instilling flexible, threshold or condition-based regulations will require a clear and legal mandate as well as an

oversight framework. Model ordinances can help with this when multiple sub-national governments are concerned. National legislation may also need to be adapted.

- *Robust protections adapted to algorithmic systems:* The digital public realm may be subject to the same “gaming” and biases as the analogue realm, especially by or via the use of algorithmic systems. Authorities should be aware of this and mitigate these risks.

Regulating by algorithm

The blurring of the lines between the physical and digital worlds creates new possibilities in the realm of public governance, specifically the use of algorithmic systems to carry out regulatory functions. This report has, and will again address the challenges of using algorithms in support of the regulatory action of governments. Here, the report brings to light strategic considerations that should be adopted by public authorities seeking to establish automatic, compliance-by-design algorithmic regulatory systems.

Tim O'Reilly, a Silicon Valley-based publisher and writer, first outlined the concept of "algorithmic regulation" as an emerging, new and putatively better way of solving policy problems. According to O'Reilly (O'Reilly, cited in Goldstein and Dyson, 2013), algorithmic regulatory systems are defined by four features:

- A deep understanding of the desired outcome
- Real-time measurement to determine if that outcome is being achieved
- Algorithms (i.e. a set of rules) that make adjustments based on new data
- Periodic, deeper analysis of whether the algorithms themselves are correct and performing as expected.

There is a distinction between governments deploying algorithmic systems to *help* carry out their regulatory functions and using these to carry out their regulatory functions *in the place of* human-based processes. In the former case, algorithmic systems can be used to increase internal efficiency, reduce the compliance burden or improve regulatory effectiveness (Eggers, Turley and Kishnani, 2018a). These approaches can be broadly grouped into three categories: approaches that seek to increase internal efficiency, those that seek to improve regulatory effectiveness and those that seek to reduce the compliance burden:

- Increasing internal efficiency
    - Automating manual tasks
    - Optimising inspections and enforcement efforts
    - Analysing large volumes of public comments
- Improving regulatory effectiveness
    - Anticipating problems and sensing disruption
    - Fighting fraud
    - Rethinking outreach
    - Nudging compliance
- Reducing the compliance burden
    - Rationalising and streamlining regulations – making them machine-readable
    - Reducing the reporting burden
    - Improving the government-to-business experience

Using algorithmic systems to carry out regulatory functions in the place of humans is a more challenging task. There are multiple drivers pushing towards greater use of algorithmic systems to carry out regulatory functions. These include budget pressure and personnel shortages in public administration; increased licensing and permitting demands; a backlog for treating requests and comments for and to regulatory processes; a push to create more agile and efficient regulatory frameworks that match the speed of transport service providers to develop and implement new, algorithmically-enabled, services; and a desire to reduce regulatory compliance costs (Eggers, Turley and Kishnani, 2018a). Estonia is a pioneer in this field and has piloted at least 13 instances where AI-based algorithmic systems replaced human-based regulatory actions or interventions. Most recently, the Estonian Ministry of Justice has begun to investigate deploying an AI-based "robot judge" to adjudicate the backlog of small claims disputes involving less than EUR 7 000 (Niler, 2019). This approach could conceivably extend current efforts to handle parking and speeding violations – though considerable questions remain on the public acceptability of such measures.

The deployment of regulatory algorithms by governments – e.g. code that automatically or semi-automatically undertakes specific regulatory functions – is neither trivial nor as straightforward as the "compliance by design" idea may at first seem. With all the inscrutability and lack of transparency inherent in many algorithmic systems, entrusting algorithmic systems with regulatory functions (including sanctioning non-compliance with rules) is fraught with risks:

> …given that algorithms can be difficult to implement, configure, and fully understand, the risk exists that governments end up relying on algorithms they do not fully understand without being capable of effectively verifying the adherence of the algorithms to the existing laws. Thus, we may end up implicitly delegating decisional power to the algorithms, or to the group of people devoted to designing and developing them […] Algorithms in future societies and cities will serve the same role that civil law and urban regulations, respectively, serve in today's democratic systems. Accordingly, new political procedures need to be put in place to regulate which code is installed. (Zamborelli et al., 2018)

Automated, algorithmically-processed regulation may be relevant for simple, black-and-white processes that do not require – or are not typically open to – human interpretation. Interpretation of law is hard to devolve to machines. This may be because laws have not been written in such a way as to make this easy or acceptable. A possible response, then, is to create both human- and machine-interpretable law. In many cases, however, this is because people prefer to have recourse to human arbitration and to hold people accountable for adverse outcomes. Removing all human agency in compliance and interpretation of laws would potentially lead to unwanted and dysfunctional outcomes – and would likely require laws to be written in a different way.

Imagine a world in which every traffic regulation was rigorously followed because the algorithmic system involved – say, in a self-driving car – could not go against its programming and do otherwise. At any given moment, the system may be required to comply with two contradictory regulations – for example, having to pull over for an emergency vehicle but not park in front of a fire hydrant. Current compliance practices for traffic laws tolerate some divergence in the face of conflicting rules but this would not necessarily be the case for systems designed to be compliant by design.

Regulation by algorithm will likely require rewriting rules to account for their binary, on or off, application and control. Robust mechanisms will need to be established to handle ambiguous situations or potentially erroneous applications of the law. This includes the ability for meaningful and rapid human redress. It will also require being able to deliver a high standard of accountability in relation to:

- transparency (perhaps devolved to a trusted oversight board in the case of sensitive processes like security screening)
- clarity on what data was used
- explainability of outcomes.

One particular type of algorithmically-based protocol, distributed ledger technologies (DLTs), including blockchain – introduces the possibility for governments to deploy self-executing, tamper-resistant, code-based forms of rules and regulations. These forms of "executable law" may be appropriate where regulatory transactions are straightforward or where legal provisions can be transcribed directly into simple and deterministic machine-readable rules that can be run on DLT networks (De Filippi and Wright, 2018).

Assessing and regulating algorithms

Algorithmic systems are neither “weapons of math destruction” (O’Neil, 2016) nor simple, benign tools of mass convenience. They are neutral technology-based systems – tools. Their impact depends both on how they are used and the way in which they are regulated. Deployed and adequately framed (possibly by lighter regulatory frameworks), they hold tremendous promise to enhance welfare across multiple domains and create new sources of beneficial innovation, including in transport. Deployed poorly, in a way that is misaligned with public policy outcomes, they pose risks and generate new classes of harms.

Because of this tension between benefits and impacts, authorities are increasingly being called upon to assess algorithmic systems and their impacts as they do with other technologies. Where they may be potentially harmful in ways that current regulatory frameworks cannot easily address, authorities will have to imagine innovative ways to regulate them without eroding the benefits they promise.

Ultimately, there is a need to create *trust* in algorithms. A wide range of critical products and services which may have impacts on human health, safety, dignity, privacy and other fundamental human rights have evolved robust *trust frameworks*. A trustable process is one that is “auditable in such a way that, at any point in the process, one can assess the degree to which it can be trusted” (ISRC/Codethink, 2017). These trustable processes include codes, standards, protocols that allow people to trust financial services, medicines, vehicles, judiciary processes, etc. Such processes, however, are largely absent when it comes to creating robust and trustable risk-based auditability for algorithmic systems. Use of these systems by the public, and public agencies’ dependence on them, are largely acts of faith based on the fact that these systems work and, therefore, can be trusted (ISRC/Codethink, 2017).

At present, algorithmic systems lack recognisable processes, regulatory frameworks, and widely-used standards or audit mechanisms that would enable users and authorities to assess trustability. This may not matter if breakdowns in algorithmic systems have trivial and contained effects, but it is a critical issue to address where there are risks of consequential harms (ISRC/Codethink, 2017). Part of that trust architecture should be built on an understanding of how the algorithm system functions and produces outputs (explainability or interpretability), and part on robust accountability standards and practices surrounding the deployment and use of algorithmic systems.

The assessment of algorithmic systems and their impacts is complicated by an asymmetry in speed, knowledge, capacity and risk-acceptance between the creators of algorithmic systems and regulatory agencies. There is a disconnect between the speed of technology (including algorithmic systems and the services they enable) and the regulatory mechanisms and institutions meant to deliver public policy outcomes. This “pacing problem” is a widening gap that intensifies the need for adapted risk assessment approaches. However, in the case of algorithmic systems, regulators have very little knowledge on how these systems work and where vulnerabilities that could impact public policy outcomes may lie. Furthermore, regulators lack expertise and the ability to retain qualified staff to assess algorithmic systems. In addition, regulated entities’ risk-acceptance thresholds are generally higher (and less focused on societal risks) than that of regulators (Eggers, Turley and Kishnani, 2018a). This structural misalignment is at the heart of the assessment and regulation challenge vis-à-vis algorithmic decision systems:

> We have a legal, regulatory framework built on the basis of mail, paper, words, versus a new world order which is digital, continuous, 24/7, and built on bits and bytes. Somehow we need to square these two worlds. (Aaron Klein, in Eggers, Turley and Kishnani, 2018a)

This section outlines current thinking on how to assess algorithmic systems and where implementation of regulation might be prudent given potential harms. The below over-arching principles, however, are worth building into *any* assessment and regulatory framework.

Adopt a risk-based, light assessment-regulatory approach

Risk-based regulation builds on the allocation of regulatory resources in line with the risks posed to public policy objectives (Black, 2005). Not all algorithmic systems are equally risky (or beneficial). Regulators must seek a balance between the risks and mistakes that are inherent in technology innovation and the potentially negative impacts of regulatory intervention to avoid these. Part of the assessment process should be based on matching the comprehensiveness and intrusiveness of the assessment and regulatory framework with known or revealed risks (Eggers, Turley and Kishnani, 2018a).

From a broad societal perspective, there is also a difference in nature (though not necessarily in terms of liability) between harms that arise from malevolence, negligence, conscious neglect or ineptitude on the part of algorithmic system developers and those that arise accidentally as part of the innovative process (New and Castro, 2018). This argues for a graduated assessment (and regulatory) approach that should minimise oversight of trivial and low-impact algorithmic decision systems and increase assessment and oversight in a series of graduated steps for more and more consequential algorithmic system impacts.

Avoid the compliance trap

There is a misguided tendency to assume that any algorithmic governance system must comply with or fit into the pre-existing regulatory system and that all regulatory gaps must be "plugged" (Danaher, 2018). Too often with technological change, regulation focuses on yesterday's technologies instead of the emerging social-technological system (Eggers, Turley and Kishnani, 2018a). Focusing on complying with the current regulatory framework may be the appropriate response in some cases but not in all. Not all pre-existing regulatory approaches are optimal. Avoiding the compliance trap is especially important when algorithmic systems obviate the need for pre-existing regulatory practices and create new ways of regulating more dynamically and efficiently with a lighter touch.

Compare algorithmic system performance to the human decision systems they replace or enhance

An essential element in the algorithmic assessment process is weighing the impact of not deploying the algorithmic system. Is the balance of risks and benefits tilted towards having humans continue to make critical and consequential decisions instead of algorithms? If so, the question of adopting an algorithmic system is moot. If the balance is reversed, then taking humans out of the decision-making framework entirely, or having them only intervene when prompted, may be the best option.

Assessing algorithmic systems

The very nature of most algorithmic systems makes assessment of their potential impacts challenging. Because they are often commercially valuable technologies, they are typically shielded from open scrutiny and lack transparency, except to those who code or control them. If revealed, they may not be understandable except to a specialised audience. Various open-source code components or microservice algorithms may be stitched together in complex assemblages such that, even if their specific code is revealed and understood, overall algorithmic system function may be difficult to ascertain because of complex interaction effects. In the case of various forms of AI, the lack of explainability or interpretability may be inherent to the technology itself and extend to the algorithms' very developers. Given these challenges, many public authorities are unclear on what basis algorithmic assessment should take place. At the same time, there is confusion and conflation of many potential approaches – including transparency, explainability and accountability. These are distinct issues and worthy of separate discussions.

Assessment based on transparency

Transparency-based approaches are intuitively attractive. If regulators could “see” algorithmic code, one assumes they could assess its potential impacts. In the transport sector, there are parallels with the way in which regulators “see” vehicle technologies and assess their impacts. This is the case for aircraft and their components, including their algorithmic components, which must pass official certification. Likewise, vehicle certification standards are based on the access regulatory agencies have to the objects of regulation in order to assess their safety and road-worthiness. Irrespective of how and how well this authority is exercised, the potential to directly scrutinise and assess new technologies has a strong tradition in the regulation of transport. This is not, however, the case for algorithmic code and algorithmic systems for a range of new use cases and transport services – including the code that enables highly and fully automated driving. Thus the call for opening up the “black box” and exposing source code to regulatory oversight.

Transparency, alone, however, does not guarantee that an algorithmic system’s functioning or potential impacts will be revealed. It may be so for simple, deterministic and relatively light code, but such systems are relatively rare. In the case of complex, multi-component, interconnected codebases, being able to read the code does not necessarily convey knowledge on its functioning (Annany and Crawford, 2018). Indeed, *seeing* the code does not necessarily convey an understanding on how it works and how to govern it (Janssan and Kuk, 2016). Furthermore, the operation of code is often contextual – the functioning of the system is linked to specific data inputs that may be difficult to audit in their entirety. In AI-based systems, potentially impactful algorithmic decisions are emergent properties of the machine learning processes and are not “hard coded” into the algorithm itself (Kemper and Kolkman, 2018). All of these factors are complicated by the sheer number of possible features that can be ingested and processed in ML algorithmic systems. As data starts to faithfully encode “real life”, the scale and scope of algorithmic decision processes rapidly surpasses what humans can comprehend – in other words, “intuition fails at high-dimensions” (Domingos, 2012).

Furthermore, visibility of the code does not mean its function and potential impacts can be ascertained without specialised knowledge. With the exception of certain specialist agencies, like those in charge of aviation or crash investigations, most public authorities typically do not have the capacity to read raw code from either a predictive basis (“what might the code do?”) nor a forensic basis (“what did the code do?”).

In addition to these limitations, commercial actors expend considerable efforts and resources to create code that enables them to deliver competitive and attractive services. They argue that revealing code, even in the closed context of a regulatory process, subjects them to risks of intellectual property theft and an erosion of their competitive position. But the same is true for many other domains and regulatory agencies have put in place relatively robust and well-accepted methods to avoid the disclosure or misuse of sensitive commercial information (e.g. rules protecting tax information, patent law, or the use of third-party vetting bodies). The claim of exceptionalism for algorithmic code is, then, perhaps overstated. That said, for the reasons outlined above, and with the possible exception of critical, non-AI code components whose malfunction could entail loss of life or injury, little may be gained by having open access to code, and what is gained may not ultimately have an impact on actually addressing or avoiding algorithmic harms (Ananny and Crawford, 2018).

Assessment based on explainability and interpretability

How, then, can regulators (and the public) assess the impacts of closed and inscrutable algorithmic systems if they cannot open the “black box” and see the code? Luckily, much can be learned about algorithmic function by observing the relationship between inputs to the models and their outputs. A whole range of techniques exist to build on this observation and develop explanations of algorithmic performance, thus making them explainable, interpretable and understandable to humans – including regulators.

Explainability is the ability of developers and operators to explain, in terms understandable to humans, how their algorithmic systems work and to describe the basis for the decisions they make. Explainability, defined as "meaningful information about the logic involved", is a key component of the European Union's General Data Protection Rules (GDPR) as it pertains to the processing of personal information by automated decision-making systems (EU Regulation 2016/679, Articles 13-15) and is generally a desirable component of the trust architecture for all algorithmic systems.

It is worth considering if requirements for explainability and interpretability for decision-making are universal. Do they apply equally to machine- *and* human decision-making processes? If not, are there specific justifications for applying these requirements to one and not to the other? These reasons may exist (as outlined previously), but they are likely not relevant for all algorithmic systems.

Trade-offs may arise between prioritising *algorithmic explainability* versus prioritising *algorithmic accuracy* – especially for ML-based AI. Often the most complex and powerful predictive models are the least interpretable (Kuhn and Johnson, 2013). Efforts to trade accuracy for better explainability must be carefully considered. The accuracy-explainability trade-off may be robust but it may be that advances in data science may shift or alter the slope of the trade-offs between the two. Thus, in the future, a more explainable system may be delivered with at least the same level of accuracy as today or, conversely, the same level of accuracy could be delivered as today but with more explainability (Gunning, 2017). Achieving satisfactory levels of explainability for ML-based AI is both problematic and an ongoing and dynamic field of research (DARPA, 2016; Gunning, 2019).

For certain uses of algorithmic decision systems (e.g. by public authorities or in instances where consequential harms may arise), the threshold for explainability should at least match, if not surpass, those of existing human decision-making processes (AI Now Institute, 2018).

Explainability has several facets that must be considered when assessing algorithmic system assessment (Castelluccia and Le Métayer, 2019):

- *Intelligibility, understandability*: Humans may face difficulty understanding machine logic in complex systems. To overcome this, intelligibility metrics often relate to the complexity and size of the system (e.g. number of neural layers, size or depth of decision tress, number of rules). Intelligibility is a complex and subjective notion for many algorithmic systems that can only be measured exactly through experimental processes.
- *Fidelity, accuracy*: Explanations should be accurate in the sense that they relate to the whole of the algorithmic system. This, however, is complicated to do in practice and many approaches seek to deliver local explanations around particular model outputs. As such, accuracy is often a relative and localised assessment, though the approach should deliver consistently accurate localised explanations, irrespective of the point of reference.
- *Precision, level of detail*: Explanations can vary in how many variables and model characteristics they cover.
- *Completeness:* Do explanations only cover some of the relevant factors that influence the model output or do they cover all relevant factors? The answer will impact the extent to which an explanation makes sense to those assessing algorithmic performance.

Technical solutions to improved explainability

Explainability or interpretability of algorithmic systems does not require opening "black box" systems and "daylighting" code. There are model-agnostic technical solutions to deliver at least partial explainability of results without describing or unveiling the overall functioning of large and complex algorithmic systems (Guidotti et al., 2018). They function best on algorithmic code that is not embedded in broader and more complex algorithmic systems comprised of multiple, intertwined algorithms. These approaches build on the structured observation of inputs and model outputs in order to derive an understanding of how the model functions, or at least, what variables influence certain outputs. "Black box" explainability methods try to clarify (Becks, 2019) the following three questions.

What are the most import inputs for a given output?

One technique – the *counterfactual explanation* technique – answers this question, not by seeking to provide insight into the internal workings of the algorithmic model but, rather, by determining which inputs would lead to a desired outcome (Wachter, Mittelstadt and Russell, 2017). This computationally light technique is adapted for neural network-based algorithmic systems. It can be used to explain which input was most important for an outcome, if the basis of that outcome can be contested and what future action/input would lead to a different decision. Because of the transient state of ML-based algorithms, this approach would require computing counterfactuals at run-time and preserving these to be queried later. Alternatively, it could create a snapshot of the model state in order to run counterfactuals on that copy at a later time. Counterfactual explanations, because of their nature, are not appropriate for situations where understanding system functionality or providing a rationale for a decision is important.

How do these variables influence the prediction?

Partial Dependence Plots (PDP) and *Individual Conditional Expectation* (ICE) are two methods that enable the observation of the relationship between model inputs and outputs (Wright, 2018; Goldstein et al., 2015). PDPs reveal the functional relationship between a limited number of model inputs and the model's predictions. PDPs operate at the level of averages for selected variables which may hide heterogeneity within the variable distribution and mask important, localised relationships. ICE reveals the relationship between inputs and predictions for each observation. ICE can reveal locally important subgroupings within the variable range that help explain model function.

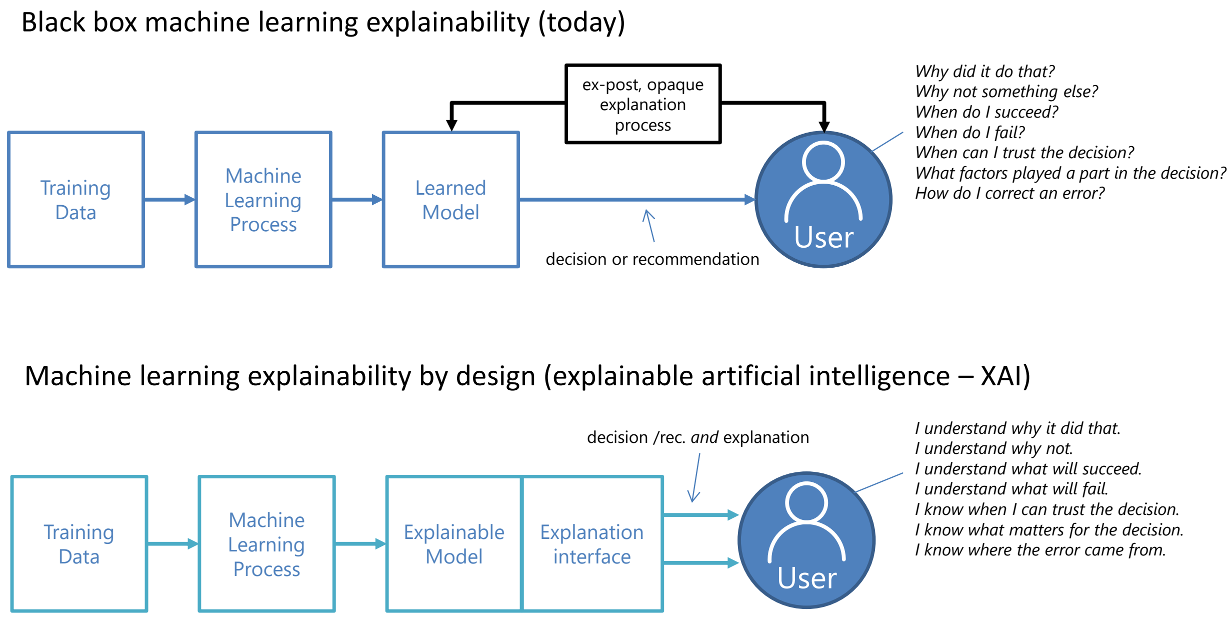
How can a specific prediction be explained (as opposed to explaining the entire logic of the model)?

As noted earlier, it may be difficult to gain a functional understanding of overall algorithmic system functioning, especially for large and complex systems or for ML-based algorithmic models. *Local Interpretable Model-Agnostic Explanation* (LIME) is an advanced model explanation method that seeks not to explain overall system function, but rather to explain why the model made a specific decision. These explanations involve building an accurate and faithful representation of the global model but limited to a few sampled points around a point of interest (e.g. the prediction for which an explanation is desired). Thus, while LIME may not be able to explain how an image classifier identifies a stop sign, it can highlight which elements in the picture were crucial in the prediction. For example, LIME could reveal that the shape of the sign and the combined vectors of the letter outlines were determinant factors in the classification. Alternatively, it could show that trees in the background were the strongest determinant of the classification thus indicating that the algorithm is accurate but for the wrong reasons (or, alternatively, that there is unsuspecting but strong predictive value in the trees behind the stop sign) (Castelluccia and Le Métayer, 2019; Ribiero, Singh and Guestrin, 2016).

As noted previously, explaining specific outputs from ML-based algorithms is difficult, even if the source code is visible (e.g. "grey box" models). To address this interpretability challenge, several techniques can be used to elicit explanations of ML model function, including models built on Bayesian networks, limited-depth neural networks and deep neural networks (Zeiler and Fergus, 2013; Montavon et al., 2018). These are useful for those writing the code or those using the algorithmic system "internally" to create explanations regarding model function that can be shared externally or integrated into explanatory pseudocode (Castelluccia and Le Métayer, 2019).

The two approaches outlined above – those targeting "black box" and "grey box" algorithmic models – assume the algorithmic system and its model already exist and must be explained ex-post. In essence, these systems have not been built to be "explainable by design". Another approach is to flip the first and, for algorithmic systems that may have consequential impacts, to build explainable or interpretable systems from the outset, including with an "explainability interface". As noted earlier, this may entail trade-offs with model accuracy but "explainability by design" is an evolving field and there is no certainty that this trade-off is universal or will even hold in the future as techniques evolve (Figure 5).

Figure 5. Explainability in machine learning


Black box machine learning explainability (today)

Training Data
Machine Learning Process
Learned Model
ex-post, opaque explanation process
decision or recommendation
User
Why did it do that?
Why not something else?
When do I succeed?
When do I fail?
When can I trust the decision?
What factors played a part in the decision?
How do I correct an error?

Machine learning explainability by design (explainable artificial intelligence – XAI)

Training Data
Machine Learning Process
Explainable Model
Explanation interface
decision /rec. and explanation
User
I understand why it did that.
I understand why not.
I understand what will succeed.
I understand what will fail.
I know when I can trust the decision.
I know what matters for the decision.
I know where the error came from.

Source: adapted from Gunning (2017).

There are two ways to approach "explainability by design". The first is to use an algorithmic technique that, by its very design, meets interpretability requirements in a way that does not erode accuracy. Examples of this approach are "generalised additive models" (GAMS) and their extensions (Castelluccia and Le Métayer, 2019; Lou, Caruana and Gehrke, 2012; Lou et al., 2013). Unlike other ML model frameworks, like random forest models, GAMS are designed to be interpretable and explainable by design.

Another strategy is to build explanation functionality into algorithmic systems so that the model can, in addition to its results, produce an accurate and intelligible explanation for those results or, at a minimum, a log allowing this explanation to be generated ex-post (Castelluccia and Le Métayer, 2019). This approach requires defining the type of outputs that should be logged – e.g. which explanatory outputs help correctly interpret the basis for model decision. It also requires setting up processes for capturing those outputs in a way that they can be trusted if audited. One way to do this is to have these outputs automatically transmitted to a trusted third party (or a regulator), another is to have these outputs cryptographically hashed and encoded in a distributed ledger so that they can be trusted to be faithful and untampered snapshots explaining model decisions. Part of the logging process should cover outputs that express in mathematical terms the degree with which the algorithmic system is functioning as designed in the form of confidence intervals. Another part of the logging process should cover data that allows an auditor to assess procedural regularity – that is, that the application and function of the algorithmic system is the same in all like cases and that there is no arbitrariness in its behaviour (Kroll, 2016).

"Explainability by design", like initiatives to enhance privacy via "privacy by design" will entail changes in the way in which code is conceived and written – at least for applications where explainability is a necessary to avoid consequential harms. This will involve standard-setting, adopting industry best practice and, in some cases, may require regulators stipulating this approach for critical code.

Algorithmic accountability

Rather than focus on transparency, explainability and interpretability as being the keystones of algorithmic assessment processes, many have argued for encompassing these and other factors into a broader *algorithmic accountability* framework (World Wide Web Foundation, 2017; Diakopoulos et al., 2016; Reisman et al., 2018). A governance framework for algorithmic accountability should ensure that algorithmic systems are conceived and built so they can be trusted to operate as intended and that any harmful outcomes that may occur can be quickly identified and rectified (New and Castro, 2018).

A central question when considering algorithmic accountability is who should be accountable? The question is not as straightforward as it would seem. Those who write code certainly have a direct responsibility in ensuring the code functions as claimed or intended, but the function of algorithmic *systems* can rarely be linked to a specific coder or snippet of code. Code is written by a wide array of contributors, some of it taken from open-source repositories and even though each individual piece of code may function as intended, code interaction effects may produce unintended outcomes. It may make more sense to hold the "director" of the code responsible for ensuring accountability. This "director" or operator is the party responsible for deploying the algorithmic system and should be legally accountable for its decisions (New and Castro, 2018).

Several initiatives have sought to describe what a framework for "algorithmic accountability" might look like. These tend to be broad principles that, putatively, would be adopted by those ordering and writing algorithms in their workflows. In 2016, a group of scientists and researchers – the Fairness, Accountability, and Transparency in Machine Learning initiative – issued their "Principles for Accountable Algorithms". These principles outline the components of an accountability framework for algorithmic systems. They were purposefully left vague so that they might be adapted by others according to the context. Those principles are fairness, responsibility, explainability, accuracy and auditability (Diakopoulos et al., 2016):

- Fairness: Ensure that algorithmic decisions do not create discriminatory or unjust impacts when comparing across different demographics.
- Explainability: Ensure that algorithmic decisions as well as any data driving those decisions can be explained to end-users and other stakeholders in non-technical terms.
- Auditability: Enable interested or affected third parties to probe, understand and review the behaviour of the algorithm through disclosure of information that enables monitoring, checking, or criticism, including through provision of detailed documentation, technically suitable APIs and permissive terms of use.
- Responsibility: Make available externally visible avenues of redress for adverse individual or societal effects of an algorithmic decisions system, and designate an internal role for the person who is responsible for the timely remedy of such issues.
- Accuracy: Identify, log, and articulate sources of error and uncertainty throughout the algorithm and its data sources so that expected and worst-case implications can be understood and inform mitigation procedures.

As noted earlier, there are emergent risks related to bias and discrimination in algorithms. Algorithmic accountability frameworks should address and attempt to mitigate these specific risks. The Association for Computing Machinery (ACM), United States Public Policy Council (USACM) and the Europe Council Policy Committee (EUACM) have jointly produced a set of specific considerations for addressing algorithmic accountability with regards to bias and discrimination (EUACM-USACM, 2017). These build on the FAT/ML principles outlined above.

- Awareness: Owners, designers, builders, users, and other stakeholders of analytic systems should be aware of the possible biases involved in their design, implementation, and use, and the potential harm that biases can cause to individuals and society.
- Accountability: Institutions should be held responsible for decisions made by the algorithms that they use, even if it is not feasible to explain in detail how the algorithms produce their results.
- Explanation: Systems and institutions that use algorithmic decision-making are encouraged to produce explanations regarding both the procedures followed by the algorithm and the specific decisions that are made. This is particularly important in public policy contexts.
- Data provenance: A description of the way in which the training data was collected should be maintained by the builders of the algorithms. It should be accompanied by an exploration of the potential biases induced by the human or algorithmic data-gathering process. Public scrutiny of the data provides maximum opportunity for corrections. However, concerns over privacy, protecting trade secrets, or revelation of analytics that might allow malicious actors to game the system can justify restricting access to qualified and authorised individuals.
- Auditability: Models, algorithms, data, and decisions should be recorded so they can be audited in cases where harm is suspected.
- Validation and testing
- Institutions should use rigorous methods to validate their models and document those methods and results. In particular, they should routinely perform tests to assess and determine whether the model generates discriminatory harm. Institutions are encouraged to make the results of such tests public.

Both of the above frameworks for accountability include various considerations discussed earlier. As noted, transparency and explainability can be useful in straightforward code environments. On the other hand, they may be difficult to ensure for certain complex algorithmic systems, including those that leverage ML-based AI. Nonetheless, regulators should adopt strong requirements for direct or indirect transparency where the risks of potential harms are highest or most consequential. If these requirements cannot be achieved directly or indirectly, regulators should carefully consider if those algorithmic systems are ready for use, even if they promise to deliver great benefits (Burrel, 2016). Imposing these conditions may delay the deployment of certain algorithmic systems in sensitive use cases, or it may block them completely until public policy outcomes can be assured.

Some have argued that such a precautionary approach is fundamentally welfare-reducing given the potential benefits that algorithmic systems (like self-driving cars) could deliver (New and Castro, 2018). But this is a recurrent debate in a number of technology domains, including vehicle technology, medicines, machinery, etc. Clearly such a potentially restrictive approach should not be applied to all algorithmic systems, but equally clearly, it seems reasonable that regulators ensure a similar level of risk exposure as they have delivered for other new technologies. More fundamentally, these risks can be mitigated by adapting innovative governance structures. This report explores what these might look like later.

One could argue that governments should adhere to a different and more stringent standard of algorithmic accountability because of the nature of government action – e.g. its ability to directly compel citizens to modify or change their behaviour or to otherwise intervene consequentially in people’s lives.

> In the public sector, the opacity of algorithmic decision making is particularly problematic both because governmental decisions may be especially weighty, and because

> democratically-elected governments bear special duties of accountability. (Brauneis and Goodman, 2017)

The United Kingdom's National Endowment for Science, Technology and the Arts (NESTA) suggests such a standard in its "Code of Standards for Public Sector Algorithmic Decision Making" for public agency use and deployment of algorithms (Copeland, 2018).

Delivering on many of the elements included in the FAT/ML (Diakopoulos et al., 2016), ACM or NESTA principles would require changing the way code is conceived, written and assembled into larger algorithmic systems. Building "accountability by default" or "accountability by design" directly into algorithmic systems at the outset clearly represents a change from current practices in many application areas. It may be necessary, though, given that current programming practices and methods are no longer fit for underpinning trust in their use due to the size, complexity and interdependency of algorithmic systems.

Governments can incentivise (or require) new methods, protocols and adoption of standards when they procure algorithmic systems or license services and technologies that depend on algorithmic systems. For example, aviation authorities in the United States and Europe require manufacturers to log and make an assessment of potential interaction effects and potential consequences for all code and code updates before vetting their use. This approach is useful for critical systems but it may not be scalable for very large and complex systems. Another strategy – "model-based programming" – changes the way in which code is specified (by humans) and encoded (by machines), resulting in an "accountable by design" system. It has been used within the aerospace industry and seems promising for certain algorithmic system applications (Williams and Ingham, 2002; Smith, 2018). Fundamentally, regulators and public authorities must develop their algorithmic literacy and internal capacity to evaluate automated decision systems – resources like the AI Now Institute's "Algorithmic Accountability Policy Toolkit" (AINow Institute, 2018) are both helpful and necessary to accomplish this up-skilling.

Box 4. NESTA's Code of Standards for Public Sector Algorithmic Decision Making

1. Every algorithm used by a public sector organisation should be accompanied with a description of its function, objectives and intended impact, made available to those who use it.

If public sector staff is to use algorithms responsibly to complement or replace some aspect of their decision making, it is vital they have a clear understanding of what they are intended to do, and in what contexts they might be applied.

2. Public sector organisations should publish details describing the data on which an algorithm was (or is continuously) trained, and the assumptions used in its creation, together with a risk assessment for mitigating potential biases.

Public sector organisations should prove that they have considered the inevitable biases in the data on which an algorithm was (or is continuously) trained and the assumptions used in their model. Having done this, they should outline the steps they have taken to mitigate any negative consequences that could follow, to demonstrate their understanding of the algorithm's potential impact. The length and detail of the risk assessment should be linked to the likelihood and potential severity of producing a negative outcome for an individual.

3. Algorithms should be categorised on an Algorithmic Risk Scale of one to five, with five referring to those whose impact on an individual could be very high, and one being very minor.

Given the rising usage of algorithms by the public sector, only a small number could reasonably be audited. By applying an Algorithmic Risk Scale (based on the risk assessment conducted for Principle 2), public sector organisations could help auditors focus their attention on instances with the potential to cause the most harm.

4. A list of all the inputs used by an algorithm to make a decision should be published.

Transparency on what data is used by an algorithm is important for a number of reasons. First, to check whether an algorithm is discriminating on inappropriate grounds (e.g. based on a person's ethnicity or religion). Second, to ensure that an algorithm could not be using a proxy measure to infer personal details from other data (e.g. guessing someone's religion based on their name or country of origin). Third, to ensure the data being used are those that citizens would deem acceptable, thereby supporting the (UK's) Data Science Ethical Framework's second and fourth principles to: "Use data and tools which have the minimum intrusion necessary" and "Be alert to public perceptions".

5. Citizens must be informed when their treatment has been informed wholly or in part by an algorithm.

For citizens to have recourse to complain about an algorithmic decision they deem unfair (e.g. they are denied council housing or probation), they need to be aware that an algorithm was involved. This might work in a similar way to warnings that a credit check will be performed when a person applies for a new credit card.

6. Every algorithm should have an identical sandbox version for auditors to test the impact of different input conditions.

It has sometimes been suggested that the code of algorithms used by government and the public sector should be made open so that their logic and function can be assessed and verified by auditors.

This now seems impractical, for at least four reasons. First, the complexity of modern algorithms is such that there are not enough people who would understand the code. Second, with neural networks there is no one location of decision making in the code. Third, algorithms that use machine learning constantly adapt their code based on new inputs. And fourth, it is unrealistic to expect that every algorithm used by the public sector will be open source; some "black box" priority systems seem inevitable.

Instead, auditors should have the ability to run different inputs into the algorithm and confirm that it does what it claims. If this cannot be done in the live system, then a sandbox version running identical code should be required. Testing should focus on algorithms scored at the upper end of the Algorithmic Risk Scale outlined in Principle 3.

7. When using third parties to create or run algorithms on their behalf, public sector organisations should only procure from organisations able to meet Principles 1 to 6.

Given the specialist skills required, most public sector organisations are likely to need to hire external expertise to develop algorithms, or pay for the services of organisations that offer their own algorithms as part of software-as-a-service solutions. In order to maintain public trust, such procurements cannot be absolved of the need to meet the principles in this code.

8. A named member of senior staff (or their job role) should be held formally responsible for any actions taken as a result of an algorithmic decision.

This would be a powerful way to ensure that the leadership of each organisation has a strong incentive only to deploy algorithms whose functions and impacts on individuals they sufficiently understand.

9. Public sector organisations wishing to adopt algorithmic decision making in high risk areas should sign up to a dedicated insurance scheme that provides compensation to individuals negatively impacted by a mistaken decision made by an algorithm.

On the assumption that some people will be adversely affected by the results of algorithmic decision making, a new insurance scheme should be established by public sector bodies to ensure that citizens are able to receive appropriate compensation.

10. Public sector organisations should commit to evaluating the impact of the algorithms they use in decision making, and publishing the results.

This final evaluation step is vital for three reasons. First, to ensure that the algorithm is used as per the function and objectives stated in Principle 1. Second, to help organisations learn about the strengths and limitations of algorithms so they can be improved. Third, to ensure that examples of best practice can more easily be verified, distilled and spread throughout the public sector.

Source: Copeland (2019).

Algorithmic impact assessment

Public authorities make wide use of impact assessment to help understand what their regulatory role should be when carrying out public policies or when licensing others to act. Impact assessments are common in a number of domains, including transport, and are well-understood mechanisms to assess potential risks and payoffs that may be incurred from policies, technologies, infrastructure investments and licensing services within the public domain. It seems natural that public authorities should undertake impact assessments regarding algorithmic systems that could have a consequential impact on regulated outcomes or within the public domain.

In the fall of 2018, the City of New York, for example, put in place an Automated Decision Systems Task Force tasked with recommending a process for reviewing the City's use of algorithmic automated decision systems (City of New York, 2019). This committee is expected to report on its findings and suggest a strategy for all impacted departments, including the Department of Transport, in the course of 2019.

In April 2019, the Government of Canada enacted the "Directive on Automated Decision-Making", which includes a comprehensive algorithmic impact assessment methodology for government agencies (Government of Canada, 2019b). This methodology can also be used by the private sector. The methodology gauges the impact an algorithmic system might have on various aspects of society and tries to help understand how much agency is delegated to the automated system. It does this in a number of ways including by addressing whether humans choose the decision variables or if the algorithm does or by understanding whether there is a human in the loop or not (Supergovernance, 2018; Government of Canada, 2019c).

The Canadian algorithmic impact assessment guidance helps agencies fill out the assessment questionnaire. The questions take into account the scale and scope of the algorithmic system. Systems that have limited impacts result in lower scores than those with broad impacts. Similarly, expert systems where variables, weights and processes are known are rated lower than machine learning systems where all three of those factors may be unknown. The assessment system is purposefully left vague so that it engages a broad group of public agency, civil society and commercial actors to collaboratively map out algorithmic impact.

The final cumulative score classifies the algorithmic system into one of four categories based on its impact on (Government of Canada, 2019b):

- the rights of individuals or communities
- the health or well-being of individuals or communities
- the economic interests of individuals, entities, or communities
- the ongoing sustainability of an ecosystem.

These categories are:

- Level I: Little to no impact. Impacts will often be reversible and brief.
- Level II: Moderate impacts. Impacts are likely reversible and short-term.
- Level III: High impacts. Impacts can be difficult to reverse and are ongoing.
- Level IV: Very high impacts. Impacts are irreversible and are perpetual.

Each successive impact score triggers different types of oversight actions, described in Table 2.

One of the key principles embedded in the Canadian approach (as well as in the assessment framework suggested by NESTA, outlined in Box 4) is the idea of critical algorithmic impact thresholds that trigger greater and greater scrutiny, oversight and, possibly, intervention. This approach implicitly recognises that most algorithmic systems improve public welfare, are generally benign and present localised and manageable risks – if any. At the same time, this approach allows regulators (and the public) to be aware of riskier algorithmic systems whose normal operation or misuse could cause consequential or widespread harms.

New York University's AI Now institute outlines what a generic algorithmic assessment impact might look like (AI Now Institute, 2018). Unlike the closed discovery and remediation processes outlined in the EU General Data Protection Regulation (GDPR) and other Data Protection Impact Assessments (DPIAs), AI Now's proposed approach encourages public involvement and due process requirements as a way to engage a wider set of views and expertise. Such an approach may also strengthen public trust in the process and in the algorithmic systems deployed via this process. There are clear and well-known risks of broad public outreach – especially as these concern the complexity and duration of the public participation process – but these can be managed by time limitations and adherence to clear and well-defined protocols. Alternatively, these can also be managed by limiting the scale and scope of initial algorithmic system deployment with sandboxes and other forms of experimentation as outlined further.

Table 2. Government of Canada directive on automated decision-making: Actions required by impact level

Requirement	Level I	Level II	Level III	Level IV
Peer review	None	At least one of the following: Qualified expert from a federal, provincial, territorial or municipal government institution Qualified members of faculty of a post-secondary institution Qualified researchers from a relevant non-governmental organization Contracted third-party vendor with a related specialization Publishing specifications of the Automated Decision System in a peer-reviewed journal A data and automation advisory board specified by Treasury Board Secretariat		At least two of the following: Qualified experts from the National Research Council of Canada, Statistics Canada, or the Communications Security Establishment Qualified members of faculty of a post-secondary institution Qualified researchers from a relevant non-governmental organization Contracted third-party vendor with a related specialization A data and automation advisory board specified by Treasury Board Secretariat OR: Publishing specifications of the Automated Decision System in a peer-reviewed journal
Notice	None	Plain language notice posted on the program or service website.	Publish documentation on relevant websites about the automated decision system, in plain language, describing: • How the components work; • How it supports the administrative decision; and • Results of any reviews or audits; and • A description of the training data, or a link to the anonymized training data if this data is publicly available.	
Human-in-the-loop for decisions	Decisions may be rendered without direct human involvement.		Decisions cannot be made without having specific human intervention points during the decision-making process; and the final decision must be	

			made by a human	
Explanation Requirement	In addition to any applicable legislative requirement, ensuring that a meaningful explanation is provided for common decision results. This can include providing the explanation via a Frequently Asked Questions section on a website.	In addition to any applicable legislative requirement, ensuring that a meaningful explanation is provided upon request for any decision that resulted in the denial of a benefit, a service, or other regulatory action.	In addition to any applicable legislative requirement, ensuring that a meaningful explanation is provided with any decision that resulted in the denial of a benefit, a service, or other regulatory action.	
Testing	Before going into production, develop the appropriate processes to ensure that training data is tested for unintended data biases and other factors that may unfairly impact the outcomes. Ensure that data being used by the Automated Decision System is routinely tested to ensure that it is still relevant, accurate, and up-to-date.			
Monitoring	Monitor the outcomes of Automated Decision Systems on an ongoing basis to safeguard against unintentional outcomes and to ensure compliance with institutional and program legislation, as well as this Directive			
Training	None	Documentation on the design and functionality of the system	Documentation on the design and functionality of the system. Training courses must be completed	Documentation on the design and functionality of the system. Recurring training courses. A means to verify that training has been completed.
Contingency Planning	None		Ensure that contingency plans and/or backup systems are available should the Automated Decision System be unavailable.	
Approval for the system to operate	None		Deputy head	Oversight Board

Source: Government of Canada (2019b).

The approach outlined by the AI Now Institute focuses on public agency procurement and use of algorithmic systems, but it could also serve as the basis (with some modifications) for establishing an algorithmic impact assessment process for all potentially consequential algorithmic systems deployed in the public domain. The AI Now Algorithmic Impact Assessment (AIA) is composed of four main elements: 1) establishing scope; 2) public notice of existing and proposed algorithmic automated decision system; 3) internal agency self-assessment; and 4) meaningful access (AI Now Institute, 2018).

Establishing scope: Define relevant algorithmic automated decision system

Agencies should have a notion of what a relevant automated decision system is and when they are suited for an algorithmic impact assessment. What should be the object of an AIA? The definition in the European Union’s GDPR can serve as a good starting point for systems that deal with individuals’ data:

> [an automated decision system is] any form of automated processing of personal data consisting of the use of personal data to evaluate certain personal aspects relating to a natural person, in particular to analyse or predict aspects concerning that natural person's performance at work, economic situation, health, personal preferences, interests, reliability, behaviour, location or movements. (EU Parliament, 2016)

Similarly, public authorities may wish to draft other archetypal definitions for systems with potential impacts on safety or in other regulated domains – like traffic or allocation and use of public space. Drawing boundaries is not always clear-cut, but there are some systems that seem clearly out of scope. For example, an autocorrect function in word-processing or mobile operating system software may replace an intended word with another which alters the intended sense of the message. While the unintended message may cause an undesired reaction from the recipient, the potential damage is limited. Other systems may be of cursory interest – to see if they may not have unintended consequences if deployed. Safety- and security-critical systems (self-driving technologies, traffic management algorithms, security-based access control) should always be in-bounds for an algorithmic impact assessment.

What is or is not inbounds might change over time and with the scale of deployment. The algorithmic system managing a small fleet of e-scooters may not necessarily require a detailed algorithmic impact assessment. But if that system scales to tens of thousands of scooters and becomes an integral part of the overall mobility offer in a city (to the point where not being able to use the e-scooters would represent an allocative harm) then an AIA may become necessary.

Public notice of existing and proposed algorithmic automated decision system: Alert communities about the systems that may affect their lives

Government accountability and due process are built on a common understanding of how individual and collective rights may be affected by government agencies, actors and actions – including actions to allow or block the deployment of technologies such as algorithmic automated decision systems. An effective AIA process would, at a minimum, require that public authorities disclose the existing and proposed algorithmic systems that they employ or license for use in the public domain. An inventory of these systems alongside a characterisation of the purposes for which they are used, the identification of those responsible for their decisions and an assessment of their accountability could help increase public trust in these systems and allow authorities to identify potential risks and vulnerabilities.

Requiring robust proof of algorithmic accountability, especially when these include a component of describing how algorithmic decision systems work, often gives rise to blanket claims of trade secrecy. This is not a trivial concern as the value of many services is tied to the skill with which the algorithmic code is written. But these claims are often overblown when they include such information as the existence of the system itself, the purpose for which it is used, the results of impact assessments and testing of the system. Furthermore, such claims should not prevent robust auditing and forensic investigation of the system. As noted earlier, algorithms of consequential impact in regulated spheres should be designed and built for explainability and audit from the outset.

Internal agency self-assessment: Increase the capacity of public agencies to assess safety, fairness, justice, due process and disparate impact

AIAs allow public agencies to understand the potential impacts of algorithmic systems and better accommodate or regulate these systems in line with the benefits they offer or the risks they pose. In order to do this, public agencies must develop or acquire sufficient expertise. Training staff, hiring code-savvy public servants and identifying neutral third-party expertise to help carry out AIAs are clear and immediate needs. Increasing the code-awareness and expertise of staff would not only help in assessing potential impacts but would also improve public sector procurement of algorithmic systems. Developing this capacity is a challenge in the current environment where those with coding expertise are highly sought after by the private sector just as many public agencies are under budget constraints. Democratising and broadening the code-literacy of the entire work force is not just a challenge for the transport sector but for all sectors of society and should be part of an all-of-government strategy. Agencies should carry out comprehensive "algorithm-readiness" self-assessments that allow them to understand where they should start to bolster their capacity.

Meaningful access: Allow auditors and researchers to review systems once they are deployed

Authorities should establish periodic or episodic algorithmic system performance assessments – especially where there is a significant risk they may contribute to some of the harms outlined earlier in this report. Agencies should establish and publicise formal auditing procedures, their periodicity and the thresholds of impacts that would justify a targeted performance assessment. For agency-deployed algorithms, this should allow for researchers or other experts to participate in the assessment process. For third-party algorithmic systems that are bundles in services licensed by the regulatory authority, adequate review mechanisms, possibly involving trusted and neutral third-party vetting agencies, should be defined and publicised at the time of licensing.

Regulating algorithms

Experts expect and assume that automated decision systems, including various forms of AI-based algorithmic decisions will deliver substantial gains in welfare, efficiency and regulatory effectiveness. However, this report outlines the new challenges these systems will pose since they may enact harms that are not accounted for by our legal and regulatory institutions. Because of this, there is a growing need to assess the potential impacts of algorithmic systems and, where they are potentially harmful in ways that current regulatory frameworks cannot easily address, imagine innovative ways to regulate them without eroding the benefits they deliver.

A central part of algorithmic impact assessment process is to help determine if and where specific (or generalised) regulatory action is necessary. Regulatory action should be as light as necessary to deliver, fulfil or protect public policy outcomes. Regulatory action for algorithmic systems may involve removing existing regulatory measures when they are rendered moot, adapting or updating existing regulations when they have not kept pace, or developing new regulatory measures or instruments when and where they are required.

The discussion around the right scope of regulatory oversight and intervention for algorithmic systems is a heated and unsettled one. There are those who uphold that algorithmic systems, especially those as inscrutable as machine-learning-based systems, are so fundamentally foreign to our current understanding and capacity to regulate that only the strictest form of regulation makes sense from a public policy perspective. In this case, since so much is unknown about the scale and scope of algorithmic impacts, only the strongest application of the precautionary principle should hold. This would, of course, restrict the deployment of such systems, protecting society from grave algorithmic harms in some cases but denying or postponing the benefits that algorithmic systems could deliver in many cases.

The other tendency would be to accept that all innovation poses risks and that society generally benefits from innovation, even if some harms are incurred. This school of thought would seek the lightest of touches and only impose additional regulation when harms have occurred and where they cannot be addressed by existing legislative, legal or societal redress frameworks.

The sensible way forward lies somewhere between a strongly precautionary approach and one that accepts risk thresholds that regulators and the public may not or cannot accept. Crafting this "third way" for algorithmic governance will require addressing when to regulate and how to regulate, all the while accepting that the objects of regulation – algorithms themselves – are constantly changing.

When to regulate

Regulation should anticipate potential harmful impacts of a significant scale and scope before they happen. Robust ex-ante regulation is appropriate when there are clear threats to safety or individual liberties or when there are uninternalised externalities. It falls in line with how these risks are treated for other types of technology or service-related impacts. It also makes sense to extend this anticipatory approach where risks to fair competition may diminish consumer welfare.

In many cases, soft, market-based regulation may be enough to avoid many negative outcomes. Algorithmic systems that perform poorly, fail to meet their users' requirements or that are known to lead to sub-standard outcomes will not scale or will be abandoned by their users. The power of reputation and open and public review of algorithmic system performance via market action can help control for faulty or damaging algorithmic systems – to a point.

Some deployers of algorithmic systems are not subject to market forces: public agencies, for example, or systems with little or no competition. Such situations should incite increased regulatory scrutiny (New and Castro, 2018). A more interventionist approach may also be warranted for algorithmic systems that are so large and embedded that their contribution to harms becomes hard to discern or isolate.

How to regulate

As noted earlier, there will be a tendency to address the regulation of algorithms only from the perspective of the current regulatory framework – e.g. the "compliance trap" (Danaher, 2018). This compliance trap also extends to the methods and practices that regulatory agencies employ to deliver on their mandate to protect public welfare, ensure competitive markets, and avoid harms. Regulatory agencies will need to be innovative in the way they carry out their regulatory functions because many of the impacts of algorithmic systems are hard to know and there is little prior knowledge to help evaluate the scope and scale of these impacts (Hagemann, Skees and Thierer, 2018).

The current approach to protect people, internalise impacts and ensure competitive markets is largely to "regulate and forget" – e.g. to take the time to craft the right regulatory framework, enact it and then infrequently update it, if ever (Eggers, Turley and Kishnani, 2018a). This approach is poorly adapted to the speed of technology and service innovation in the transport sector today. It fails to satisfactorily address the "pacing problem" described earlier. Pressure to regulate rapidly, and often lightly, is exacerbated by the "global innovation arbitrage" where, in the global economy, innovation, like capital, flows to those markets where restraints on both are minimised (Theirer, 2016).

The risk for algorithmic decision-making systems, as with other emerging technologies and services, is that regulators will either act too soon, too late, too permissively, or too restrictively. To minimise these risks, they will need to change the way in which they regulate under uncertainty. Eggers, Turley and Kishnani (2018a) outline five useful principles for doing so:

Risk-weighted regulation

Not all algorithmic systems pose the same risks. Most are beneficial, many are benign in terms of the risks they pose and some may impose potentially large and consequential risks. Regulation should be tailored to address these risks in a graduated and targeted manner. The most intrusive and constraining regulatory responses should be aligned with the probability and scope of algorithmically-triggered harms and the lack of other adapted regulatory tools.

Governments wishing to shift from a one-size-fits-all regulatory framework to a data-driven, risk-segmented approach will need to collect and process data that allows them to make appropriate risk assessments. This is the approach adopted by the Government of Canada's "Directive on Automated Decision-Making" described earlier. Such algorithmic impact assessment policies provide the factual basis for a "pre-certification" framework that expedites approvals for low-risk, low impact systems. A data-driven, risk-based regulatory framework should extend beyond pre-certification and encompass, where possible, a more dynamic regulatory approach based on frequent and near real-time data flows between companies and regulatory authorities (Eggers, Turley and Kishnani, 2018a). This is the approach adopted by the Mobility Data Specification described earlier.

Crucial market deployment plans for innovative services and technologies should not be blocked, but should benefit from expedited and predictable approval processes after demonstrating that certain basic safety and security guarantees can be met. Public authorities should anticipate some of the most likely impacts of early versus scaled deployment and set out clear guidance on when and under what conditions more stringent review and regulatory controls may come into play. At the same time, those deploying automated decision systems will have to provide sufficient, trustable data such that authorities can understand when and where increased oversight might become necessary.

Finally, authorities should consider that algorithmic decision systems are but one of a multitude of potentially risky technology developments. Some of these lend themselves more easily to risk assessment because data exists and is collected, others less so. There may be a tendency to regulate what can be "seen" first and most comprehensively, leading to a discrepancy in treatment. For instance, when trying to minimise congestion, it may be technically easy to regulate platform ride-sourcing services that depend on algorithmic decision systems to match drivers to clients but the bulk of congestion is caused by "non-algorithmic" decisions of car and truck drivers and by public works. Regulating the former more stringently than the latter leads to unfair treatment and ineffective public policy, as the traffic congestion remains.

Adaptive regulation

Governments should move from a "regulate and forget" model of regulation to a more dynamic, iterative and responsive model better adapted to accommodate rapid changes and an uncertain technology environment.

Governments have a responsibility to provide certainty about the conditions in which citizens and firms will make decisions. However, the traditional approach of seeking input from a broad range of stakeholders, investing a considerable amount of time in crafting considered rules and laws, passing them and then leaving them largely unchanged, is not adapted to the pacing problem governments face today. Furthermore, this approach may not allow regulators to adapt their rules easily once they see how individuals and firms respond to them, sometimes in unexpected ways.

Governments have a responsibility to provide certainty about the conditions in which citizens and firms will make decisions. But rather than doing so for the *specific rules* that governments will put into place, public authorities should assure citizens and companies about the *process* whereby those rules will be revisited, assessed, updated or changed, as necessary. Governments will need to establish rapid feedback loops and a greater diversity of "soft law" (informal guidance, self-regulation, best practice guidance, third-party certification, etc.) as opposed to "hard law" tools. One of the principal advantages of soft law approaches is that they allow regulators to adapt quickly to changes in technology and business models and to regulate "on the fly" as issues arise (Eggers, Turley and Kishnani, 2018a).

Regulatory sandboxes and accelerators

In line with the adaptive regulatory approach, public authorities can create time-limited, partial exemptions from prevailing regulatory requirements. This temporarily frees deployers of algorithmic systems from red tape and allows for faster release of their systems. It also provides a testing ground for regulators, a time period where they can learn if regulation would be necessary if these new systems where they to scale up, what that regulation might look like, and how to implement it. Both accelerators and regulatory sandboxes help accelerate innovation and give regulators assurances that potential unwanted, negative outcomes remain manageable and can be addressed jointly with the private sector.

Several governments have adopted these approaches for managing the deployment of algorithmic systems in the financial, health and transport sectors (e.g. the Financial Conduct Authorities' FinTech regulatory sandbox in the United Kingdom or the United States Department of Transport's Federal Aviation Authority's Unmanned Arial System testbeds). (Eggers, Turley and Kishnani, 2018a)

Outcome-based regulation

There has been a general shift in many areas of transport regulation turning focus from technical specifications and form to results and system performance. This has enabled more efficient delivery of public policy outcomes in many cases and propelled innovation. Specifying technologies and processes makes sense in many areas, especially those relating to safety, but many regulations can be re-framed by referencing the outcomes they should ensure instead of the means whereby they do so.

Consider the following: "Ride source platforms must only on-board vehicles of a set length and power, should return to their 'base' between rides, should only be bookable 30 minutes in advance and should not surpass a set fleet cap." Such a highly prescriptive approach stifles innovation and preserves current, inefficient ride markets.

Now, compare that to the following: "Ride source platforms should not contribute disproportionally to congestion (as measured by x), should be safe and secure (as measured by y) and should implement robust customer service and redress mechanisms (as measured by z) and otherwise lead to outcomes that are aligned with other public policy goals (e.g. lowering pollution levels, ensuring fair and competitive markets and ensuring dignified working conditions) – as mapped by the agencies responsible for delivering on those outcomes."

The second formulation focuses on what policy makers and citizens want, not on prescribing how those deploying algorithmic systems should deliver on these outcomes.

Outcome-based regulations are facilitated by the development of guidelines versus hard laws. These allow rapid iteration and provide regulators the opportunity to update them as impacts and negative outcomes become known. They also require robust and broadly accepted metrics whereby performance-based outputs can be measured and guidelines adjusted if outputs are under-delivered or not at all.

Collaborative regulation

Achieving regulatory compliance requires resources from regulated entities and regulators. This is especially the case where regulators each develop their own regulatory approach and where those deploying algorithmic systems must comply with different regulatory frameworks across regional, national and global markets. Inconsistent regulatory frameworks – e.g. regulatory divergence – increase the cost of regulation and may limit the diffusion of innovative products and services.

Much can be gained by ensuring that regulatory frameworks for algorithmic systems are as consistent and predictable as they can be though there is a strong case for also taking into account local and national contexts when designing them. Collaborative regulatory approaches involving co-regulation and coordination among regulatory agencies helps lower the cost of regulation and can ensure a predictable ecosystem for the deployment of algorithmic decisions systems. This type of collaborative approach could be applied to common algorithmic impact assessment processes and privacy protections in transport-related algorithmic systems.

Who regulates?

Finally, there is the important question of "who regulates" or, more specifically "which public authority should have oversight and regulating authority over algorithmic systems that are being and will be deployed in the future?" There are two broad schools of thought here.

The first assumes that the potential risks of some algorithmic systems are so great, and current regulatory frameworks so out of phase with algorithmic logic, that only a master national regulatory body can adequately handle the new challenges. Proponents of this approach call for bodies such as a "National Algorithm Safety Board" (Schneiderman, 2016) or a national algorithmic regulatory agency, built along the lines of agencies that handle food and medicine safety (Tutt, 2016). They reason that the risk of consequential harms is great, yet most public agencies have not fully anticipated the range of algorithmic impacts and how these may escape traditional regulatory oversight and recourse.

The second school of thought contends that most algorithmic risks are limited, and that most algorithmic systems pose few intractable challenges to the capacity of public authorities to adequately oversee and regulate their deployment. Proponents of this approach call for existing public authorities to bolster their capacity to understand and exercise, where appropriate, their regulatory authority only where the risk of consequential harms are elevated.

There is truth in the justifications cited in support of the former view. The potential for consequential harms triggered by certain types of algorithmic systems is great (though the risk may be uncertain). Algorithmic decision systems, especially those based on machine-learning-based artificial intelligence, are also fundamentally different from the human-based decision systems with which regulatory agencies are familiar. Finally, government agencies are generally low on staff and their internal structures and methods ill-equipped and out of alignment with an algorithmically-run sector. But it is not clear that a national-level unitary authority would be any better equipped to handle the challenges.

At the same time, proponents of the latter, more distributed approach are not wrong in wanting to bring regulation as close as possible to the regulated technology or service. This would call for increasing the capacity of all regulatory agencies to handle the specific challenges that algorithms may pose to their specific regulatory tasks.

A common approach is required to assess risks and carry out algorithmic impact assessment processes. Ideally, this should take place at the national level and extend even to the international level so that all regulatory agencies are basing their approach on a common playbook that gives them freedom to adapt their approach for local contexts.

Bibliography

Abadi, M. et al. (2016), “Deep Learning with Differential Privacy”, proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security - CCS’16 (pp. 308–318), ACM Press, https://doi.org/10.1145/2976749.2978318.

AI Now Institute (2018), *Algorithmic Accountability Policy Toolkit,* AINow Institute, https://ainowinstitute.org/aap-toolkit.pdf (accessed on 29 April 2019).

Amodei, D. et al. (2016), “Concrete Problems in AI Safety”, Cornell University, http://arxiv.org/abs/1606.06565 (accessed on 29 April 2019).

Ananny, M. and K. Crawford (2018), “Seeing without knowing: Limitations of the transparency ideal and its application to algorithmic accountability”, *New Media and Society*, Vol. 20/3, pp. 973–989, https://doi.org/10.1177/1461444816676645.

Andrews, L. et al. (2017), “Algorithmic Regulation”, Discussion Paper no. 85, Centre for Analysis of Risk and Regulation, London School of Economics and Political Science, London, https://www.kcl.ac.uk/law/research/centres/telos/assets/DP85-Algorithmic-Regulation-Sep-2017.pdf (accessed on 29 April 2019).

Aneesh, A. (2002), “Technologically Coded Authority: The Post-Industrial Decline in Bureaucratic Hierarchies,” Stanford University, https://web.stanford.edu/class/sts175/NewFiles/Algocratic%20Governance.pdf (accessed on 29 April 2019).

Assaderaghi, F. and L. Reger (2018), “Artificial Intelligence: Beyond the Hype”, *NXP Me&My Smarter World*, https://blog.nxp.com/uncategorized/artificial-intelligence-beyond-the-hype (accessed on 29 April 2019).

Ateniese, G. et al. (2015), “Hacking smart machines with smarter ones: How to extract meaningful data from machine learning classifiers”, *International Journal of Security and Networks*, Vol. 10/3, pp. 137, https://doi.org/10.1504/IJSN.2015.071829.

Baldwin, R., M. Cave and M. Lodge (eds.) (2012), *The Oxford Handbook of Regulation*, Oxford University Press, Oxford.

Barocas, S., S. Hood and M. Ziewitz (2013), “Governing algorithms: A provocation piece”, *SSRN Electronic Journal*, https://doi.org/10.2139/ssrn.2245322.

Barocas, S. and A. Selbst (2016), “Big data’s disparate impact”, *SSRN Electronic Journal*, https://doi.org/10.2139/ssrn.2477899.

Barreno, M. et al. (2006), “Can machine learning be secure?”, proceedings of the 2006 ACM Symposium on Information, computer and communications security - ASIACCS ’06, p. 16, ACM Press, https://doi.org/10.1145/1128817.1128824.

Bayamlıoğlu, E. and R. Leenes (2018), “The ‘rule of law’ implications of data-driven decision-making: a techno-regulatory perspective”, *Law, Innovation and Technology*, Vol. 10/2, pp. 295–313, https://doi.org/10.1080/17579961.2018.1527475.

Becks, A. (2019), “Bring light into the black box: Making AI decisions explainable”, *SAS Hidden Insights*, https://blogs.sas.com/content/hiddeninsights/2019/03/14/bring-light-into-the-black-box-making-ai-decisions-explainable/ (accessed on 13 May 2019).

Beer, D. (2017), “The social power of algorithms”, *Information, Communication and Society*, Vol. 20/1, pp. 1–13, https://doi.org/10.1080/1369118X.2016.1216147.

Beer, D. (ed.) (2018), *The Social Power of Algorithms*, Routledge, Taylor and Francis Group, New York.

Beyer, S. (2017), "U.S. cities should loosen their jaywalking laws", *Forbes*, https://www.forbes.com/sites/scottbeyer/2017/11/30/u-s-cities-should-loosen-their-jaywalking-laws/#1abe2e7c763b (accessed on 29 April 2019).

Biggio, B. and F. Roli (2018), "Wild patterns: Ten years after the rise of adversarial machine learning", proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security - CCS '18, pp. 2154-2156, ACM Press, https://doi.org/10.1145/3243734.3264418.

Black, J. (2002), *Critical reflections on regulation*, CARR Discussion Papers (DP 4), Centre for Analysis of Risk and Regulation, London School of Economics and Political Science, London.

Black, J. (2005), "The emergence of risk-based regulation and the new public risk management in the United Kingdom", *Public Law*, Vol. Autumn, pp. 512–548, https://www.academia.edu/1295947/The_emergence_of_risk-based_regulation_and_the_new_public_risk_management_in_the_United_Kingdom (accessed on 29 April 2019).

Black, J. (2008), "Constructing and contesting legitimacy and accountability in polycentric regulatory regimes", *Regulation and Governance*, Vol. 2/2, pp. 137–164, https://doi.org/10.1111/j.1748-5991.2008.00034.x.

Black, J. (2014), "Learning from regulatory disasters", *Policy Quarterly*, Vol. 10/3, https://doi.org/10.26686/pq.v10i3.4504.

Brauneis, R. and E. Goodman (2017), "Algorithmic Transparency for the Smart City", *SSRN Electronic Journal*, https://doi.org/10.2139/ssrn.3012499.

Bundestag (2019), "German Federal laws and regulations GitHub", https://github.com/bundestag/gesetze (accessed 29 April 2019).

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

Campbell-Verduyn, M., M. Goguen and T. Porter (2017), "Big Data and algorithmic governance: The case of financial practices", *New Political Economy*, Vol. 22/2, pp. 219–236, https://doi.org/10.1080/13563467.2016.1216533.

Casanovas Romeu, P. et al. (ed.) (2010), *AI Approaches to the Complexity of Legal Systems: Complex Systems, the Semantic Web, Ontologies, Argumentation, and Dialogue*, Springer, Berlin, http://hdl.handle.net/1814/15054 (accessed on 29 April 2019).

Castelluccia, C. and D. Le Métayer (2019), *Understanding Agorithmic Decision-Making: Opportunities and Challenges*, European Parliament, http://www.europarl.europa.eu/thinktank/en/document.html?reference=EPRS_STU(2019)624261 (accessed on 29 April 2019).

Castle, N. (2018), "What is semi-supervised learning?", *Learn Data Science, Artificial Intelligence*, https://www.datascience.com/blog/what-is-semi-supervised-learning (accessed on 29 April 2019).

Chakraborty, A. et al. (2018), "Adversarial attacks and defences: A survey", Cornell University, https://arxiv.org/abs/1810.00069 (accessed on 29 April 2019).

Chouldechova, A. (2017), "Fair prediction with disparate impact: A study of bias in recidivism prediction instruments", *Big Data*, Vol. 5/2, pp. 153–163, https://doi.org/10.1089/big.2016.0047.

Christiano, P. (2015), "Human-in-the-counterfactual-loop," *Medium.com AI Alignment*, https://ai-alignment.com/counterfactual-human-in-the-loop-a7822e36f399 (accessed on 29 April 2019).

City of New York (2019), "New York City Automated Decision Systems Task Force", https://www1.nyc.gov/site/adstaskforce/index.page (accessed on 29 April 2019).

Coglianese, C. and D. Lehr (2018), "Transparency and algorithmic governance", Administrative Law Review, Vol. 71/1, University of Pennsylvania Law School, Public Law Research Paper No. 18-38, https://ssrn.com/abstract=3293008 (accessed on 29 April 2019).

Copeland, E. (2018), "*10 principles for public sector use of algorithmic decision making*", Nesta, https://www.nesta.org.uk/blog/10-principles-for-public-sector-use-of-algorithmic-decision-making/ (accessed on 29 April 2019).

Cornelisse, D. (2018), "An intuitive guide to Convolutional Neural Networks", *freeCodeCamp.org*, https://medium.freecodecamp.org/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050 (accessed on 29 April 2019).

Crawford, K. and V. Joler (2018), "Anatomy of an AI System", *AI Now Institute*, https://anatomyof.ai/ (accessed on 29 April 2019).

Cui, J. et al. (2018), "A review on safety failures, security attacks, and available countermeasures for autonomous vehicles", *Ad Hoc Networks*, https://doi.org/10.1016/j.adhoc.2018.12.006.

Czarnecki, K. (2018), *Operational Design Domain for Automated Driving Systems: Taxonomy of Basic Terms*, Waterloo Intelligent Systems Engineering Lab (WISE) Report, University of Waterloo, DOI: 10.13140/RG.2.2.18037.88803.

Danaher, J. (2016a), "The threat of Algocracy: Reality, resistance and accommodation", *Philosophy and Technology*, Vol. 29/3, pp. 245–268, https://doi.org/10.1007/s13347-015-0211-1.

Danaher, J. (2016b), *The Logical Space of Algocracy (Redux)*, https://philosophicaldisquisitions.blogspot.com/2016/11/the-logical-space-of-algocracy-redux.html (accessed on 29 April 2019).

Danaher, J. et al. (2017), "Algorithmic governance: Developing a research agenda through the power of collective intelligence", *Big Data and Society*, Vol. 4/2, https://doi.org/10.1177/2053951717726554.

Danaher, J. (19 December 2018), "Algorithmic governance in transport: Some thoughts", *Philosophical Disquisitions*, https://philosophicaldisquisitions.blogspot.com/2018/12/algorithmic-governance-in-transport.html (accessed on 13 May 2019).

DARPA (2016), *Explainable Artificial Intelligence (XAI),* DARPA-BAA-16-53, Defense Advanced Research Projects Agency, https://www.darpa.mil/attachments/DARPA-BAA-16-53.pdf (accessed on 29 April 2019).

DC Council (2019), "DC Law: Statutes and code", https://github.com/DCCouncil/dc-law-xml (accessed on 29 April 2019).

De Filippi, P. and A. Wright (2018), *Blockchain and the Law: The Rule of Code*, Harvard University Press, Cambridge, Massachusetts.

Delacroix, S. (2019), "Beware of 'algorithmic regulation'", *SSRN Electronic Journal*, https://doi.org/10.2139/ssrn.3327191.

Diakopoulos, N. et al. (2016), *Principles for Accountable Algorithms*, Fairness, Accountability, and Transparency in Machine Learning, http://www.fatml.org/resources/principles-for-accountable-algorithms (accessed on 29 April 2019).

Domingos, P. (2012), "A few useful things to know about machine learning", *Communications of the ACM*, Vol. 55/10, p. 78, https://doi.org/10.1145/2347736.2347755.

Doneda, D. and V. Almeida (2016), "What is algorithm governance?", *IEEE Internet Computing*, Vol. 20/4, pp. 60–63, https://doi.org/10.1109/MIC.2016.79.

Dunn, M. (2013), "Toyota's killer firmware: Bad design and its consequences", EDN Network, https://www.edn.com/design/automotive/4423428/Toyota-s-killer-firmware–Bad-design-and-its-consequences (accessed on 29 April 2019).

Eggers, W., M. Turley and P. Kishnani (2018a), "The future of regulation: Principles for regulating emerging technologies", Deloitte Center for Government Insights, https://www2.deloitte.com/insights/us/en/industry/public-sector/future-of-regulation/regulating-emerging-technology.html (accessed on 29 April 2019).

Eggers, W., M. Turley and P. Kishnani (2018b), "The regulator's new toolkit: Technologies and tactics for the tomorrow's regulator", Deloitte Center for Government Insights, https://www2.deloitte.com/content/dam/insights/us/articles/4539_Regulator_4-0/DI_Regulator-4-0.pdf (accessed on 29 April 2019).

Engelmann, S. et al. (2019), "Clear sanctions, vague rewards: How China's social credit system currently defines 'good' and 'bad' behavior", proceedings of the Conference on Fairness, Accountability, and Transparency held in Atlanta, GA, pp. 69–78, https://doi.org/10.1145/3287560.3287585.

Engin, Z. and P. Treleaven (2019), "Algorithmic government: Automating public services and supporting civil servants in using data science technologies", *The Computer Journal*, Vol. 62/3, pp. 448–460, https://doi.org/10.1093/comjnl/bxy082.

EU Parliament (2016), "Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)", EUR-Lex, Document 02016R0679-20160504, https://eur-lex.europa.eu/eli/reg/2016/679/2016-05-04 (accessed on 29 April 2019).

EUACM-USACM (2017), "Statement on algorithmic transparency and accountability", ACM Europe Policy Committee and ACM United States Public Policy Council, https://www.acm.org/binaries/content/assets/public-policy/2017_joint_statement_algorithms.pdf (accessed on 29 April 2019).

Eykholt, K. et al. (2018), "Robust physical-world attacks on deep learning visual classification", *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 1625–1634), IEEE, Salt Lake City, UT, https://doi.org/10.1109/CVPR.2018.00175.

Filippi, P. and S. Hassan (2016), "Blockchain technology as a regulatory technology: From code is law to law is code", *First Monday*, Vol. 21/12, https://doi.org/10.5210/fm.v21i12.7113.

Fink, K. (2018), "Opening the government's black boxes: freedom of information and algorithmic accountability", *Information, Communication and Society*, Vol. 21/10, pp. 1453–1471, https://doi.org/10.1080/1369118X.2017.1330418.

Fowler, M. (2014), "Microservices: A definition of this new architectural term", *martinfowler.com*, https://martinfowler.com/articles/microservices.html (accessed on 29 April 2019).

Giles, M. (2018), "Quantum computers pose a security threat that we're still totally unprepared for", *Technology Review*, https://www.technologyreview.com/s/612509/quantum-computers-encryption-threat/ (accessed on 29 April 2019).

Gillespie, T. (2014), *Algorithm [draft] [#digitalkeywords]*, *Culture Digitally*, http://culturedigitally.org/2014/06/algorithm-draft-digitalkeyword/ (accessed on 29 April 2019).

Gillespie, T., P. Boczkowski and K. Foot (eds.) (2014), *Media Technologies: Essays on Communication, Materiality, and Society*, MIT Press, Cambridge, MA.

Github. (2019), "Who's using GitHub?", https://government.github.com/community/ (accessed on 29 April 2019).

Goëta, S. and T. Davies (2016), "The daily shaping of state transparency: Standards, machine-readability and the configuration of open government data policies", *Science and Technologies Studies*, Vol. 29/4, https://hal.archives-ouvertes.fr/hal-01829314/document (accessed on 29 April 2019).

Goldstein, A. et al. (2015), "Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation", *Journal of Computational and Graphical Statistics*, Vol. 24/1, pp. 44–65, https://doi.org/10.1080/10618600.2014.907095.

Government of Canada (2019a), "Algorithmic Impact Assessment - Évaluation de l'incidence algorithmique (Python prototype)", https://github.com/canada-ca/aia-eia (accessed on 29 April 2019).

Government of Canada (2019b), "Directive on automated decision-making", https://www.tbs-sct.gc.ca/pol/doc-eng.aspx?id=32592 (accessed on 29 April 2019).

Government of Canada. (2019c), "Algorithmic Impact Assessment (v0.2)", https://canada-ca.github.io/aia-eia-js/ (accessed on 29 April 2019).

Grumbling, E. and M. Horowitz (eds.) (2019), *Quantum Computing: Progress and Prospects*, National Academies of Sciences, Engineering, and Medicine, Washington, D.C., https://doi.org/10.17226/25196.

Gu, T., B. Dolan-Gavitt and S. Garg (2019), *BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain*, Cornell University, https://arxiv.org/abs/1708.06733 (accessed on 29 April 2019).

Guidotti, R. et al. (2018), "A survey of methods for explaining black box models", *ACM Computing Surveys*, Vol. 51/5, pp. 1–42, https://doi.org/10.1145/3236009.

Gunning, D. (2017), "Explainable Artificial Intelligence (XAI)", United States Defense Advanced Research Projects Agency (DARPA), https://www.darpa.mil/attachments/XAIProgramUpdate.pdf (accessed on 29 April 2019).

Gunning, D. (2019), "DARPA's explainable artificial intelligence (XAI) program", proceedings of the 24th International Conference on Intelligent User Interfaces - IUI '19, pp. ii, ACM Press, https://doi.org/10.1145/3301275.3308446.

Hagemann, R., J. Skees and A. Thierer (2018), "Soft law for hard problems: The governance of emerging technologies in an uncertain future", *Colorado Technology Law Journal*, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3118539 (accessed on 29 April 2019).

Hall, W. and J. Pesenti (2017), "Growing the artificial intelligence industry in the UK", Department for Digital, Culture, Media & Sport and Department for Business, Energy & Industrial Strategy, https://www.gov.uk/government/publications/growing-the-artificial-intelligence-industry-in-the-uk (accessed on 13 May 2019).

Hayles, K. (2010), *My Mother Was a Computer: Digital Subjects and Literary Texts*, University of Chicago Press, Chicago, https://www.press.uchicago.edu/ucp/books/book/chicago/M/bo3622698.html.

Introna, L. D. (2016), "Algorithms, governance, and governmentality: On governing academic writing", *Science, Technology and Human Values*, Vol. 41/1, pp. 17–49, https://doi.org/10.1177/0162243915587360.

ISRS/Codethink. (2017), *Towards Trustable Software: A Systematic Approach to Establishing Trust in Software*, White Paper, https://www.trustablesoftware.com/ (accessed on 29 April 2019).

ITF (2015), *Big Data and Transport: Understanding and Assessing the Options,* Corporate Partnership Board Report, International Transport Forum, Paris, https://www.itf-oecd.org/big-data-and-transport.

ITF (2016), *Data-Driven Transport Policy*, Corporate Partnership Board Report, International Transport Forum, Paris, https://www.itf-oecd.org/data-driven-transport-policy.

ITF (2017), *Transition to Shared Mobility: How Large Cities Can Deliver Inclusive Transport Services*, Corporate Partnership Board Report, International Transport Forum, Paris, https://www.itf-oecd.org/transition-shared-mobility.

ITF (2018a), *Blockchain and Beyond: Encoding 21st Century Transport*, Corporate Partnership Board Report, International Transport Forum, Paris, https://www.itf-oecd.org/blockchain-and-beyond.

ITF (2018b), *The Shared-Use City: Managing the Curb*, Corporate Partnership Board Report, International Transport Forum, Paris, https://www.itf-oecd.org/shared-use-city-managing-curb-0.

Janssen, M., Y. Charalabidis and A. Zuiderwijk (2012), "Benefits, adoption barriers and myths of open data and open government", *Information Systems Management*, Vol. 29/4, pp. 258–268, https://doi.org/10.1080/10580530.2012.716740.

Janssen, M. and G. Kuk (2016), "The challenges and limits of big data algorithms in technocratic governance", *Government Information Quarterly*, Vol. 33/3, pp. 371–377, https://doi.org/10.1016/j.giq.2016.08.011.

Kemper, J. and D. Kolkman (2018), "Transparent to whom? No algorithmic accountability without a critical audience", *Information, Communication and Society*, pp. 1–16, https://doi.org/10.1080/1369118X.2018.1477967.

Kira, A. (2019), "Managing Uber's data workflows at scale", Uber Engineering, https://eng.uber.com/managing-data-workflows-at-scale/ (accessed on 29 April 2019).

Kitchin, R. (2014), *The Data Revolution: Big Data, Open Data, Data Infrastructures and Their Consequences*, SAGE Publications, Los Angeles, California.

Kitchin, R. (2017), "Thinking critically about and researching algorithms", *Information, Communication and Society*, Vol. 20/1, pp. 14–29, https://doi.org/10.1080/1369118X.2016.1154087.

Klinedinst, D. and C. King (2016), *On Board Diagnostics: Risks and Vulnerabilities of the Connected Vehicle,* No. DM-0003466, Software Engineering Institute, Carnegie Mellon University, https://resources.sei.cmu.edu/asset_files/WhitePaper/2016_019_001_453877.pdf (accessed on 29 April 2019).

Konečný, J. et al. (2016), "Federated learning: Strategies for improving communication efficiency", Cornell University, http://arxiv.org/abs/1610.05492 (accessed on 29 April 2019).

Kowalski, R. (1979), "Algorithm = logic + control", *Communications of the ACM*, Vol. 22/7, pp. 424–436, https://doi.org/10.1145/359131.359136.

Kroll, J. (2016), "Accountable Algorithms (A Provocation)", Media Policy Project Blog, London School of Economics and Political Science, https://blogs.lse.ac.uk/mediapolicyproject/2016/02/10/accountable-algorithms-a-provocation/ (accessed on 29 April 2019).

Krzyk, K. (2018), "Coding deep learning for beginners: Types of machine learning", Towards Data Science, https://towardsdatascience.com/coding-deep-learning-for-beginners-types-of-machine-learning-b9e651e1ed9d (accessed on 29 April 2019).

Kuhn, M. and K. Johnson (2013), *Applied Predictive Modeling*, Springer, https://doi.org/10.1007/978-1-4614-6849-3.

LADOT (2016), "Urban mobility in a digital age: A transportation technology strategy for LADOT", Los Angeles Department of Transport, http://www.urbanmobilityla.com/strategy (accessed on 14 May 2019).

LADOT (2019a), "LADOT dockless mobility program", City of Los Angeles Department of Transportation, https://ladot.io/programs/dockless/ (accessed on 29 April 2019).

LADOT (2019b), "LADOT transportation technology strategy", City of Los Angeles Department of Transportation, https://ladot.io/ (accessed on 29 April 2019).

LADOT (2019c), "Mobility data specification GitHub", City of Los Angeles Department of Transportation, https://github.com/CityOfLosAngeles/mobility-data-specification (accessed on 29 April 2019).

LADOT (2019d), "LADOT data protection principles", City of Los Angeles Department of Transportation*,* https://ladot.io/wp-content/uploads/2019/03/2019-04-12_Data-Protection-Principles.pdf.pdf (accessed on 29 April 2019).

LADOT (2019e), "LADOT data protections principles public comment", City of Los Angeles Department of Transport, https://ladot.io/wp-content/uploads/2019/04/2019-04-12_Data-Protections-Principles-Public-Comments.pdf (accessed on 29 April 2019).

LADOT (2019f), "LADOT Master Data License and Protection Agreement", City of Los Angeles Department of Transport, https://ladot.io/wp-content/uploads/2019/04/City-of-Los-Angeles-Master-Data-License-Protection-Template_15-Apr-2019.pdf (accessed on 29 April 2019).

LabPlus (2018), "Better Rules for Government Discovery Report", Service Innovation Lab, Government of New Zealand, https://www.digital.govt.nz/dmsdocument/95-better-rules-for-government-discovery-report/html (accessed on 29 April 2019).

Landecker, W. et al. (2013), "Interpreting individual classifications of hierarchical networks", *2013 IEEE Symposium on Computational Intelligence and Data Mining*, Institute of Electrical and Electronics Engineers, https://doi.org/10.1109/cidm.2013.6597214.

Li, Y. (2017), "Deep reinforcement learning: An overview", Cornell University, http://arxiv.org/abs/1701.07274 (accessed on 29 April 29 2019).

Litescu, S. et al. (2015), "Information impact on transportation systems", *Journal of Computational Science*, Vol. 9, pp. 88–93, https://doi.org/10.1016/j.jocs.2015.04.019.

Lou, Y. et al. (2013), "Accurate intelligible models with pairwise interactions", proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '13 (p. 623), ACM Press, https://doi.org/10.1145/2487575.2487579.

Lou, Y., R. Caruana and J. Gehrke (2012), "Intelligible models for classification and regression", proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '12 (p. 150), ACM Press, https://doi.org/10.1145/2339530.2339556.

Lu, J. et al. (2017a), "No need to worry about adversarial examples in object detection in autonomous vehicles", Cornell University, http://arxiv.org/abs/1707.03501 (accessed on 29 April 2019).

Lu, J. et al. (2017b), "Standard detectors aren't (currently) fooled by physical adversarial stop signs", Cornell University, http://arxiv.org/abs/1710.03337 (accessed on 29 April 2019).

Makridakis, S. (2017), "The forthcoming Artificial Intelligence (AI) revolution: Its impact on society and firms", *Futures*, Vol. 90, pp. 46–60, https://doi.org/10.1016/j.futures.2017.03.006.

Maxmen, A. (2018), "Self-driving car dilemmas reveal that moral choices are not universal", *Nature*, https://www.nature.com/articles/d41586-018-07135-0 (accessed on 29 April 2019).

McCarthy, J. et al. (2006), "A proposal for the Dartmouth Summer Research Project on Artificial Intelligence - August 31, 1955", *AI Magazine*, Vol. 27/4, http://jmc.stanford.edu/articles/dartmouth/dartmouth.pdf (accessed on 29 April 2019).

McCullom, R. (2017), "Facial recognition technology is both biased and understudied", Undark.org, https://undark.org/article/facial-recognition-technology-biased-understudied/ (accessed on 29 April 2019).

McMahon, H. et al. (2017), "Communication-efficient learning of deep networks from decentralized data", Cornell University, https://arxiv.org/abs/1602.05629 (accessed on 29 April 2019).

Montavon, G., W. Samek and K. Müller (2018), "Methods for interpreting and understanding deep neural networks", *Digital Signal Processing*, Vol. 73, pp. 1–15, https://doi.org/10.1016/j.dsp.2017.10.011.

Morra, J. (2018), "NXP Semiconductors takes machine learning to the edge," Electronic Design, https://www.electronicdesign.com/embedded-revolution/nxp-semiconductors-takes-machine-learning-edge (accessed on 29 April 2019).

Moses, L. (2013), "How to think about law, regulation and technology: Problems with 'technology' as a regulatory target", *Law, Innovation and Technology*, Vol. 5/1, pp. 1–20, https://doi.org/10.5235/17579961.5.1.1.

Mozur, P. (2018), "Inside China's dystopian dreams: A.I., shame and lots of cameras", *The New York Times*, https://www.nytimes.com/2018/07/08/business/china-surveillance-technology.html (accessed on 29 April 2019).

Muncrief, R., J. German and J. Schultz (2016), "Defeat devices under the U.S. and EU passenger vehicle emissions testing regulations", International Council on Clean Transportation, https://www.theicct.org/publications/defeat-devices-under-us-and-eu-passenger-vehicle-emissions-testing-regulations (accessed on 29 April 2019).

Munn, L. (2018), "Rendered inoperable: Uber and the collapse of algorithmic power," *A Peer-Reviewed Journal About - Research Values*, Vol. 7/1, https://www.aprja.net/rendered-inoperable-uber-and-the-collapse-of-algorithmic-power/ (accessed on 29 April 2019).

New, J. and D. Castro (2018), "How policymakers can foster algorithmic accountability," Information Technology and Innovation Foundation, https://itif.org/publications/2018/05/21/how-policymakers-can-foster-algorithmic-accountability (accessed on 29 April 2019).

New Tech Dojo (2018), "List of machine learning algorithms," https://www.newtechdojo.com/list-machine-learning-algorithms/#Unsupervised%20Learning (accessed on 29 April 2019).

Neyland, D. and N. Möllers (2017), "Algorithmic IF … THEN rules and the conditions and consequences of power," *Information, Communication and Society*, Vol. 20/1, pp. 45–62, https://doi.org/10.1080/1369118X.2016.1156141.

NHTSA (2017), "ODI resume: Automatic vehicle control systems", National Highway Traffic Safey Administration, United States Department of Transportation, https://static.nhtsa.gov/odi/inv/2016/INCLA-PE16007-7876.PDF (accessed on 29 April 29 2019).

Niler, E. (2019), "Can AI be a fair judge in court? Estonia thinks so," *Wired*, https://www.wired.com/story/can-ai-be-fair-judge-court-estonia-thinks-so/ (accessed on 29 April 2019).

NTSB (2018), "Preliminary report highway: HWY18MH010", United States National Transportation Safety Board, https://www.ntsb.gov/investigations/accidentreports/pages/hwy18mh010-prelim.aspx (accessed on 29 April 2019).

OECD (2017), *OECD Digital Economy Outlook 2017*, OECD Publishing, Paris, https://doi.org/10.1787/9789264276284-en.

OECD (2019), *Artificial Intelligence in Society*, OECD Publishing, Paris, https://doi.org/10.1787/eedfee77-en.

O'Neil, C. (2016), *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy* (First edition.), Crown, New York.

Goldstein L. and L. Dyson (eds.) (2013), *Beyond Transparency: Open Data and the Future of Civic Innovation*, Code for America Press, San Francisco, CA, http://beyondtransparency.org/chapters/part-5/open-data-and-algorithmic-regulation/ (accessed on 29 April 2019).

Papernot, N. et al. (2016), "Towards the science of security and privacy in machine learning", Cornell University, http://arxiv.org/abs/1611.03814 (accessed on 29 April 2019).

Papernot, N. (2018), "A marauder's map of security and privacy in machine learning: An overview of current and future research directions for making machine learning secure and private", proceedings of the 11th ACM Workshop on Artificial Intelligence and Security - AISec '18, pp. 1, ACM Press, https://doi.org/10.1145/3270101.3270102.

Pasquale, F. (2015), *The black box society: The secret algorithms that control money and information*, Harvard University Press, Cambridge.

Perez, C. (2017), "Why we should be deeply suspicious of BackPropagation", medium.com, https://medium.com/intuitionmachine/the-deeply-suspicious-nature-of-backpropagation-9bed5e2b085e (accessed on 29 April 2019).

Raine, L. and J. Anderson (2017), "Code-dependent: Pros and cons of the algorithm age", Pew Research Center, https://www.pewinternet.org/2017/02/08/code-dependent-pros-and-cons-of-the-algorithm-age/ (accessed on 29 April 2019).

Rajaniemi, H. (2018), "Unchained: A story of love, loss, and blockchain", *MIT Technology Review*, https://www.technologyreview.com/s/610831/unchained-a-story-of-love-loss-and-blockchain/?source=download-metered-content (accessed on 29 April 2019).

Rankin, K. (2017), "The dark secret at the heart of AI", *MIT Technology Review*, Intelligent Machines, https://www.technologyreview.com/s/604087/the-dark-secret-at-the-heart-of-ai/ (accessed on 29 April 2019).

Reddy, E., B. Cakici and A. Ballestero (2019), "Beyond mystery: Putting algorithmic accountability in context", *Big Data and Society*, Vol. 6/1, https://doi.org/10.1177/2053951719826856.

Reinhold, E. (2016), "Rewriting Uber engineering: The opportunities microservices provide", Uber Engineering, https://eng.uber.com/building-tincup/ (accessed on 13 May 2019).

Reisman, D. et al. (2018), *Algorithmic Impact Assessments: A Practical Framework for Public Agency Accountability*, AI Now Institute, New York University, https://ainowinstitute.org/aiareport2018.pdf (accessed on 29 April 2019).

Ribeiro, M., S. Singh and C. Guestrin (2016), "'Why should I trust you?': Explaining the predictions of any classifier", proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16, pp. 1135–1144, ACM Press, https://doi.org/10.1145/2939672.2939778.

Romei, A. and S. Ruggieri (2014), “A multidisciplinary survey on discrimination analysis”, *The Knowledge Engineering Review*, Vol. 29/05, pp. 582–638, https://doi.org/10.1017/S0269888913000039.

Salian, I. (2018), “SuperVize Me: What’s the difference between supervised, unsupervised, semi-supervised and reinforcement learning?”, NVIDIA, https://blogs.nvidia.com/blog/2018/08/02/supervised-unsupervised-learning/ (accessed on 29 April 2019).

Seaver, N. (2013), “Knowing Algorithms”, Media in Transition 8, MIT, Cambridge, MA, http://nickseaver.net/papers/seaverMiT8.pdf (accessed on 29 April 2019).

SharedStreets (2018a), “A Shared Language for the World’s Streets”, *SharedStreets*, http://sharedstreets.io/ (accessed on 29 April 2019).

SharedStreets (2018b), “SharedStreets GitHub”, https://github.com/sharedstreets (accessed on 29 April 2019).

Sharif, M. et al. (2016), “Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition”, proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 1528–1540, ACM Press, https://doi.org/10.1145/2976749.2978392.

Sheehan, B. et al. (2018), “Connected and autonomous vehicles: A cyber-risk classification framework”, *Transportation Research Part A: Policy and Practice*, https://doi.org/10.1016/j.tra.2018.06.033.

Shen, X. (2019), “Facial recognition camera catches top businesswoman ‘jaywalking’ because her face was on a bus”, Abacus, https://www.abacusnews.com/digital-life/facial-recognition-camera-catches-top-businesswoman-jaywalking-because-her-face-was-bus/article/2174508 (accessed on 29 April 2019).

Shneiderman, B. (2016), “Opinion: The dangers of faulty, biased, or malicious algorithms requires independent oversight”, *Proceedings of the National Academy of Sciences*, Vol. 113/48, pp. 13538-13540, https://doi.org/10.1073/pnas.1618211113.

Shokri, R. and V. Shmatikov (2015), “Privacy-Preserving Deep Learning”, proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, pp. 1310–1321, ACM Press https://doi.org/10.1145/2810103.2813687.

Silver, J. (2013), “Is your turn-by-turn navigation application racist”, Free Future – American Civil Liberties Union blog, ACLU, https://www.aclu.org/blog/national-security/your-turn-turn-navigation-application-racist (accessed on 29 April 2019).

Simonite, T. (2017), “Even artificial neural networks can have exploitable ‘backdoors’”, Wired.com, https://www.wired.com/story/machine-learning-backdoors/ (accessed on 29 April 2019).

Singh Gill, N. (2019), “Artificial neural networks and neural networks applications”, Xenonstack: Data Science, https://www.xenonstack.com/blog/artificial-neural-network-applications/ (accessed on 29 April 2019).

Skymind (2019), “Symbolic reasoning (symbolic AI) and machine learning”, A.I. Wiki, https://skymind.ai/wiki/symbolic-reasoning (accessed on 29 April 2019).

Smith, A. (2018), “Franken-algorithms: The deadly consequences of unpredictable code”, *The Guardian*, https://www.theguardian.com/technology/2018/aug/29/coding-algorithms-frankenalgos-program-danger (accessed on 29 April 2019).

Smith, D. (2000), “Changing Situations and Changing People”, in Von Hirsch, A. et al. (eds.), *Studies in Penal Theory and Penal Ethics: Ethical and Social Perspectives on Situational Crime Prevention*, Hart, Oxford.

Somers, J. (2017), "Is AI riding a one-trick poney?," *MIT Technology Review*, https://www.technologyreview.com/s/608911/is-ai-riding-a-one-trick-pony/ (accessed on 29 April 2019).

Song, D. (2017), "AI and security: Lessons, challenges and future directions", presentation from the ARO Workshop on Adversarial Machines Learning held at Stanford University on 14 September 2017, https://seclab.stanford.edu/AdvML2017/slides/dawn-stanford-ai-security-workshop-short-sep-2017.pdf (accessed on 29 April 2019).

SRS Inc. (2013), *Toyota Unintended Acceleration and the Big Bowl of "Spaghetti" Code*, Safety Research and Strategies, Inc., http://www.safetyresearch.net/blog/articles/toyota-unintended-acceleration-and-big-bowl-%E2%80%9Cspaghetti%E2%80%9D-code (accessed on 29 April 2019).

Su, Y., M. Lyu and I. King (2018), "Communication-efficient distributed deep metric learning with hybrid synchronization", proceedings of the 27th ACM International Conference on Information and Knowledge Management, pp. 1463–1472, ACM Press, https://doi.org/10.1145/3269206.3271807.

Supergovernance (2018), "A Canadian algorithmic impact assessment", medium.com, https://medium.com/@supergovernance/a-canadian-algorithmic-impact-assessment-128a2b2e7f85 (accessed on 29 April 2019).

Szegedy, C. et al. (2013), "Intriguing properties of neural networks", Cornell University, http://arxiv.org/abs/1312.6199 (accessed on 29 April 2019).

Thierer, A. (2016), "Innovation arbitrage, technological civil disobedience & spontaneous deregulation", The Technology Liberation Front, https://techliberation.com/2016/12/05/innovation-arbitrage-technological-civil-disobedience-spontaneous-deregulation/ (accessed on 29 April 2019).

Thing, V. and J. Wu (2016), "Autonomous vehicle security: A taxonomy of attacks and defences", presented at the 2016 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), pp. 164–170, Institute of Electric and Electronic Engineers, https://doi.org/10.1109/iThings-GreenCom-CPSCom-SmartData.2016.52.

Thornton, P. and Danaher, J. (2018), "On the wisdom of algorithmic markets: Governance by algorithmic price", *SSRN Electronic Journal*, https://doi.org/10.2139/ssrn.3314078.

Tutt, A. (2016), "An FDA for algorithms", *SSRN Electronic Journal*, https://doi.org/10.2139/ssrn.2747994.

Ullman, E. (1997), *Close to the Machine: Technophilia and Its Discontents*, City Light Books.

United States White House Office (2016a), *Big Data: A Report on Algorithmic Systems, Opportunity, and Civil Rights*, Executive Office of the President of the United States, Washington, DC, https://obamawhitehouse.archives.gov/sites/default/files/microsites/ostp/2016_0504_data_discrimination.pdf (accessed on 29 April 2019).

United States White House Office (2016b), *Preparing for the Future of Artificial Intelligence*, National Science and Technology Council Committee on Technology, Executive Office of the President of the United States, Washington, DC, https://obamawhitehouse.archives.gov/sites/default/files/whitehouse_files/microsites/ostp/NSTC/preparing_for_the_future_of_ai.pdf (accessed on 29 April 2019).

US EPA (2016), "Learn about Volkswagen violations", United States Environmental Protection Agency, https://www.epa.gov/vw/learn-about-volkswagen-violations (accessed on 29 April 2019).

Wachter, S., B. Mittelstadt and C. Russell (2017), "Counterfactual explanations without opening the black box: Automated decisions and the GDPR", *SSRN Electronic Journal*, https://doi.org/10.2139/ssrn.3063289.

Webb, K. (2018), "Getting started with SharedStreets," https://observablehq.com/@kpwebb/sharedstreets-api (accessed on 29 April 2019).

Webster, N. (2018), "LabPlus: Better rules for government discovery report", https://www.digital.govt.nz/blog/labplus-better-rules-for-government-discovery-report/ (accessed on 29 April 2019).

Williams, B. and M. Ingham (2002), "Model-Based Programming: Controlling Embedded Systems by Reasoning About Hidden State", in Goos, G.et al. (eds.), *Principles and Practice of Constraint Programming - CP 2002,* Vol. 2470, pp. 508–524, Springer, https://doi.org/10.1007/3-540-46135-3_34.

Willson, M. (2017), "Algorithms (and the) everyday", *Information, Communication and Society*, Vol. 20/1, pp. 137-150, https://doi.org/10.1080/1369118X.2016.1200645.

Wilson, B., J. Hoffman and J. Morgenstern (2019), *Predictive Inequity in Object Detection*, Cornell University, https://arxiv.org/abs/1902.11097?utm_campaign=the_algorithm.unpaid.engagement&utm_source=hs_email&utm_medium=email&utm_content=70390107&_hsenc=p2ANqtz–QeYz8_DR85xu4f90lXEXuL8URA7ivc6R3ryJ0CzTHB7tfzJ1qg2nBexlKLU7x3dlySKIYArF-nStNCeFhf_sQkWTjIQ&_hsmi=70390107 (accessed on 29 April 2019).

World Wide Web Foundation (2017), "Algorithmic accountability: Applying the concept to different country concepts", https://webfoundation.org/docs/2017/07/WF_Algorithms.pdf (accessed on 29 April 2019).

Wright, R. (2018), "Interpreting black-box machine learning models using partial dependence and individual conditional expectation plots", Paper no. SAS1950-2018, SAS Institute Inc., https://www.sas.com/content/dam/SAS/support/en/sas-global-forum-proceedings/2018/1950-2018.pdf (accessed on 15 May 2019).

Yeung, K. (2011), "Can we employ design-based regulation while avoiding *Brave New World*?", *Law, Innovation and Technology*, Vol. 3/1, pp. 1–29, https://doi.org/10.5235/175799611796399812.

Yeung, K. (2017), "'Hypernudge': Big Data as a mode of regulation by design," *Information, Communication and Society*, Vol. 20/1, pp. 118–136, https://doi.org/10.1080/1369118X.2016.1186713.

Yeung, K. (2018), "Algorithmic regulation: A critical interrogation", *Regulation and Governance*, Vol. 12/4, pp. 505–523, https://doi.org/10.1111/rego.12158.

Yu, H. and D. Robinson (2012), "The new ambiguity of 'open government'", *SSRN Electronic Journal*, https://doi.org/10.2139/ssrn.2012489.

Yuan, X. et al. (2019), "Adversarial examples: Attacks and defenses for deep learning", *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–20, https://doi.org/10.1109/TNNLS.2018.2886017.

Zambonelli, F. et al. (2018), "Algorithmic governance in smart cities: The conundrum and the potential of pervasive computing solutions", *IEEE Technology and Society Magazine*, Vol. 37/2, pp. 80–87, Institute of Electric and Electronic Engineers, https://doi.org/10.1109/MTS.2018.2826080.

Zanzotto, F. (2019), "Human-in-the-loop Artificial Intelligence", *Journal of Artificial Intelligence Research*, Vol. 64, pp. 243–252, https://doi.org/10.1613/jair.1.11345.

Zeiler, M. and R. Fergus (2013), "Visualizing and Understanding Convolutional Networks", Cornell University, http://arxiv.org/abs/1311.2901 (accessed on 29 April 2019).

Zipper, D. (2019), "Cities can see where you're taking that scooter", Slate, https://slate.com/business/2019/04/scooter-data-cities-mds-uber-lyft-los-angeles.html (accessed on 29 April 2019).

International Transport Forum

Governing Transport in the Algorithmic Age

This study explores where automated decision-making systems impact transport activity, and how. More and more transport activity is influenced by algorithms. Automated decision-making is taking a hold in areas from health care and housing to media and mobility. In transport, algorithms are a core feature for services from public transport scheduling to routing apps, bicycle sharing to self-driving technology, parcel delivery to the dispatching of ride services. How can policy makers ensure mobility driven by algorithmic code supports societal objectives?

International Transport Forum
2 rue André Pascal
F-75775 Paris Cedex 16
+33 (0)1 73 31 25 00
contact@itf-oecd.org
www.itf-oecd.org

OECD

Published: 05/2019 | Photo credit: Philippe Crist