coding/learning

software and digital data in education

A report from the ESRC Code Acts in Education seminar series
Edited by Ben Williamson

THE UNIVERSITY of EDINBURGH

1965 - 2015 50 E·S·R·C 50 YEARS SHAPING SOCIETY

UNIVERSITY of STIRLING

School of Education
University of Stirling
Pathfoot
Stirling FK9 4LA

ISBN: 978-0-9932310-4-9

This report was supported with a grant from the Economic and Social Research Council, grant reference ES/L001160/1.

Suggested citation:
Williamson, B. (ed.) 2015. *Coding/Learning: Software and digital data in education*. Stirling: University of Stirling.

contents

code, infrastructure, & digital materialities of education

data, analytics & visualization of education

encoded practices, knowing & teaching

Foreword: Education in code/space

Rob Kitchin, National University of Ireland Maynooth

Over the past fifty years everyday life has become increasingly reliant on software. Modes of work, play, consumption, travel, communication, learning, and governance are being mediated and augmented by software-enabled technologies and systems. So much so, that it is now almost impossible to live outside of the orbit of software—it is infused into the very fabric of everyday life, often in ways that are backgrounded and routine (even if one is not directly interfacing with software, it is shaping domains to some extent). Indeed, such has become the dependency on code for systems to function that when it fails the intended outcomes cannot take place. This is the argument that myself and Martin Dodge (Kitchin & Dodge 2011) developed at length in our book, *Code/Space: Software and Everyday Life*: software, to varying degrees, conditions modern life.

We detailed that life and places are increasingly full of coded objects (objects that have software embedded into them or are machine-readable) and coded infrastructures (infrastructures that transfer digital bits and depend on software to manage them) that support coded processes (various kinds of digitally-mediated interactions and transactions) and combine to constitute coded assemblages (dense amalgams of interconnected coded objects, infrastructures and processes).

We also argued that much work needs to be undertaken to conceptualize software and the work that it does in the world. We contended that software is not simply lines of code that perform a set of instructions, but rather needs to be understood as a social product that emerges in contingent, relational and contextual ways, the outcome of many minds situated with diverse social, political and economic relations. Software is a complex, multifaceted, mutable set of relations created through discursive, economic and material practices. Moreover, it enables diverse processes to be enacted; it undertakes work in the world, often across networks.

Software thus participates in the world as a potent actant. It does so because it possesses high technicity; the evolutive power of technologies to make things happen. Software facilitates transduction to occur; how a domain structures itself as a partial, always incomplete solution to a relational problem. Such relational problems include undertaking domestic tasks, travelling between locations, conducting work, communicating between people, practicing consumption, managing organisations, teaching pupils, etc., with software creating, enacting and facilitating solutions.

For us, the pervasiveness of software and its technicity and transductive qualities alter the nature of a number of fundamental elements and practices of everyday life. In particular, we examined how software changed the nature of space and governance. The production of space, we argued, has becoming increasingly dependent on software to take place. This produces a spatial formation we termed 'code/space': spaces dependent on code to function. And if the code fails then the space will not be transduced as intended. For example if the checkout tills crash then the supermarket cannot operate as a shop but rather as a warehouse, since no goods can be processed without the laser scanner; if the airport security software fails, then the airport simply becomes a vast waiting room full of people. Software is also changing how societies are monitored and regulated, producing new forms of surveillance and dataveillance, that are automated, autonomous, and automatic in how they operate. Cities, for instance, are now being transformed into 'smart cities' in which urban activities are constantly monitored and managed through a vast infrastructure of networked devices.

In the paper presented at the first Code Acts seminar in Stirling, I introduced these ideas and then sought to perform an initial mapping of them onto the education domain. The argument I developed was that practices and spaces of education are now thoroughly mediated and augmented by software-enabled technologies; that much education takes place in code/spaces and are shaped by coded practices. For example:

- Teaching materials are being created using software programmes
- Teaching is being co-delivered through digital media, supported by digital ancillary material and social media platforms
- Forms of assessment are being administered and processed using software packages
- Classrooms are enhanced with digital projectors and smart interfaces
- Administration is reliant on spreadsheets and online forms
- Oversight are calculative practices exercised with key performance indicators
- Research and fieldwork is increasingly mediated by digital technologies—the internet, cameras, voice recorders, sensors—that produce volumes of digital data that are analyzed using analytics software
- Publishing is mediated by software—writing, sharing, production, etc.

Software is thus making a difference to the way in which education is conceived and delivered; the educational practices of education (teaching, administration, and research); and the management and governance of education. Yet, to date, little critical and conceptual attention has been paid to thinking through role of code and associated digital forms such as databases and ICT infrastructures in reshaping the educational landscape and the implications and consequences of making software so central to how education is performed, organised, and managed.

In the final section of the paper, I provided a potential conceptual frame for organising a research agenda that would examine the relationship between software and education, and argued that such a programme of work needs to be undertaken. This programme would be much more than a set of instrumental studies of the effects of code on learning outcomes or performance metrics, but would consider the wider ethos, ethics and ambitions of education. It would consider how education is translated into code, and also how code transduces education.

The Code Acts project is clearly an important step in facilitating such a programme to be enacted. The papers presented in this report do not conceive of software from either a purely technical or pedagogic perspective, but rather try to think through more holistically the nature of software and its diverse intersections with education; to consider what it means for education to be a coded assemblage of coded objects, infrastructures and processes; for education to take place in code/spaces. There is much, much more work to be done, however.

Kitchin, R. & Dodge, M. 2011. *Code/Space: Software and everyday life*. London: MIT Press.

Introduction

The seminar series ‘Code Acts in Education: Learning through Code, Learning to Code’, was funded by the Economic and Social Research Council between 2013 and 2015. The project sought to explore how computer code is interwoven with educational processes, spaces, institutions, and practices, and to understand how code, algorithms, digital infrastructures, and the things they materialize, are shaping how, when and where learning takes place—in the classroom, the university, the professional workplace, and throughout the lifecourse.

The seminar series was designed to address two particular matters of concern: first, the extent to which learning processes, practices and spaces are increasingly shaped through code, and second, the emergence of a movement based on the idea of learning to code as a form of active participation in a heavily software-saturated world. In terms of the former, our concern was whether code acts as a kind of pedagogy that is immanent and everywhere in daily life, running as a substratum of experience with the power to variously instruct, seduce, educate, liberate, discipline and govern us. The seminars thus sought to answer previously unanswered questions about how code acts as a sociotechnical agent in education, focusing on how the interactions between code and education might impact on knowledge practices, pedagogic techniques, learner agency, identity formation and other aspects of learning through code.

In relation to the second main concern with ‘learning to code,’ the seminars have focused on such things as after-school coding clubs for children, community-based programming, and the reintroduction of programming and computer science in schools. ‘Learning to code’ has become a popular discourse in the educational technology field, as well as in educational policy and in the commercial agendas of major computing corporations. But we wanted to ask what is really involved in learning to code, and inquire into its political implications. Writing computer code is not just a technical act but a political act; it permits the programmer to construct particular models of the external world that might work through persuasion, seduction, coercion or education to change the way people think.

This report consists of a selection of short papers produced by the organizers of the seminar series and some of the invited participants in each of the events. They address questions of how code can be conceptualized; of how software is intervening in the production of educational data; and of how coded things such as data systems are exerting consequential effects on professional learning practices, forms of knowing, teaching, and educational policymaking.

Together, the papers point toward the significance of code as an object of inquiry for educational research. While code has become a recent object of inquiry for geographers, sociologists, new media theorists and philosophers (sometimes clustered under the interdisciplinary umbrella term 'software studies'), previously there has been no concerted effort to understand how code acts educationally, pedagogically, or instructionally, to shape how we acquire knowledge, skills and forms of conduct. What educational understandings can help us to interpret the work that code does in the world? In order to address the gap between research on software and research in education, each seminar was designed to include an interdisciplinary cross-section of speakers and participants from social science and humanities disciplines, educators, and organizations that act as intermediaries with wider publics and research users.

The report is divided into three main sections. In the first section, the papers attend to the conceptualization of code, infrastructures, algorithms and the digital materialities they generate. Together, the papers make it clear that code should not be understood simply as 'lines of code', but as a complex social and technical amalgam of practices, systems of thinking, materials and their attendant 'codes of conduct'. The second section focuses on emerging 'big data' technologies in education. The papers collectively address such questions as: What software facilitates the collection, analysis, visualization and communication of educational data? What is the 'social life' of educational data, and what sociotechnical practices are enacted as it moves between different statistical packages, analytics software, and modes of graphical display? How does the translation of data shape perceptions about educational institutions, practices, and people, and with what effects? The third section then attends to issues around the coded practices of education: professional learning, the distribution of social scientific research methods to software, the interweaving of the human teacher with algorithmic 'teacherbots', and the contemporary move to encourage new kinds of literacies and coding practices to enable people to learn about and make sense of the coded environment.

The seminar series has been a highly collaborative project involving organizers from across the University of Stirling, the University of Edinburgh, and the University of Bristol. It was managed by Ben Williamson (Stirling) as principal investigator, with the support of three co-investigators: Richard Edwards (Stirling), Tara Fenwick (Stirling) and Sian Bayne (Edinburgh); and was assisted by four doctoral students: Sarah Doyle (Stirling), Lyndsay Grant (Bristol), Jeremy Knox (Edinburgh), and Alison Oldfield (Bristol), with administrative management by Angela Cowan (Stirling). Consistent additional support with chairing sessions and discussions was kindly offered by (and gratefully received from) Terrie-Lynn Thompson (Stirling) and Jen Ross (Edinburgh). We have also been fortunate enough to receive feedback and advice from our advisory panel: Peter Donaldson (Computing at School Scotland), Lesley Gourlay (Institute of Education, University of London), Rob Kitchin (National University of Ireland, Maynooth), and Emma Uprichard (University of Warwick). On behalf of the entire team, I would like to acknowledge and thank all of the external speakers who have kindly agreed to contribute to the series (in order of appearance):

Rob Kitchin (National University of Ireland, Maynooth)
Amy Solder (Nesta)
John Potter (Institute of Education, University of London)
Jenny Ozga (University of Oxford)
Matt Finn (University of Durham)
Simon Buckingham Shum (University of Technology, Sydney)
Adrian Mackenzie (University of Lancaster)
Doug Belshaw (Mozilla Foundation)
Chris Speed (University of Edinburgh)
Nishant Shah (Leuphana University)
Susan Halford (University of Southampton)
Monika Nerland (University of Oslo)
Eric Meyer (Oxford Internet Institute, University of Oxford)

We would also like to acknowledge the supportive audience at each of the events, many of whom have become serial attenders and active participants in the series. Finally, we would like to thank Marisa Harlington for designing and illustrating the report. The papers that follow trace many of the debates generated in the seminars in what has been a richly collaborative, interdisciplinary and dialogic project. They have helped set the agenda for future research that explores how code acts in education; how it augments and influences educational institutions, processes, spaces, and practices; how it shapes the ways we encounter information and gain knowledge in everyday life; how it interacts with professional and work-based learning; and how it intervenes in the methods and analyses of educational research itself.

Ben Williamson

code, infrastructure, & digital materialities of education

Code, digital infrastructures, algorithms, and other digital and material things are increasingly understood to be exerting specific effects on the enactment of educational processes. How can we understand them, and what do they do?

What is code? From lines of code to active code

Ben Williamson, University of Stirling

What is code? This is a surprisingly difficult question to answer. Most of us have some sense that computer software and hardware—the digital materialities of everyday life—depend on programmers constructing lines of code to make them function. But code can be understood not just in technical terms as a set of instructions for computer software and hardware. It also has important historical dimensions; it is embedded in organizational contexts; it is shaped by social, economic, political and cultural contingencies; and it has some power to act in the world.

LINES OF CODE

In computer science and software development code is understood as the machine-readable language programmed to instruct computer software. Whenever we speak of software, we are talking of something that is utterly constituted by code and that is structured and operationalized through algorithms—sets of steps or processes which specify how to transform a given set of inputs into an output. Code is the substrate to software, and is constructed through programming using specific programming languages—the art and science of putting together algorithms and instructions that can be automatically read and translated by a machine in order to process data and *do* something. As Adrian Mackenzie (2006: 2) has phrased it, 'what software does and how it performs, circulates, changes and solidifies cannot be understood apart from its constitution through and through as code.'

A growing recognition of the power of code and algorithms is reflected in popular science publications like a recent 38,000 word *Business Week* article entitled 'What is Code?' (Ford 2015), and the book *9 Algorithms that Changed the Future* (MacCormick 2013) which demonstrate the vast reach of code and its algorithmic ordering structures into contemporary everyday practices. The nine algorithms of MacCormick's title refer to search engine indexing and ranking; the cryptographic algorithms required by secure websites; pattern recognition algorithms for recognising handwriting, speech and faces; data compression of files like MP3s and JPEGs; and the transactional changes made to databases, such as those required for online banking and social networking sites like Facebook. The lines of code that constitute these algorithms, and the software they instruct, are becoming highly consequential to the ways modern life is organized and experienced. They are also historically situated.

HISTORICAL CODE

Writing code is a technical process, but generated through social processes and human actions that are embedded in and shaped by historical developments. The production of code reflects a set of sharp historical divisions in the disciplines of computing—between computer science, computational science, and software development—that Brian Hayes (2015) has traced out in a recent analysis of different 'cultures of code.' While computer science is concerned with understanding underlying algorithms, software development is concerned with the production of tangible artefacts, and computational science treats the computer not as an object of study but a scientific instrument. And the divisions run deeper than this. The cultures of computer science, computational science and software development are very different; computer scientists, computational scientists and software developers work in different settings, attend different conferences, belong to different professional associations, and have very different ways of working, with different worldviews, systems of thinking, and professional practices. They are distinctly different 'cultures of code' as Hayes argues.

Moreover, as Nathan Ensmenger (2010) has shown in his history of computer programming, *The Computer Boys Take Over*, what is meant by 'coding' is itself a historical artefact. When computing first emerged as a professional practice in the 1940s and 50s, a 'coder' was seen merely as a 'glorified clerical worker' and the task of coding was almost exclusively performed by women, who were expected to:

> code into machine language the higher-level mathematics developed by male scientists and engineers. Coding implied manual labor, and mechanical translation or rote transcription; coders were obviously low on the intellectual and professional hierarchy. (Ensmenger 2010: 35)

The actual art of 'programming,' as it came to be known the late 1940s, consisted of a number of distinct steps (including mathematical conceptualization, algorithm selection, and numerical analysis) only the last of which was the 'coding' done by female coders. These historical notes make it clear that the production of code is inseparable from social context—that the digital materialities of software and hardware with which we live cannot just be reduced to their constitution as lines of code and sequences of algorithms. As Ensmenger (2010: 8) adds:

> Software is an ideal illustration of what the historians and sociologists of technology call a sociotechnical system: that is to say, a system in which machines, people, and processes are inextricably interconnected and interdependent. … [T]he 'heterogeneous engineering' required to assemble such complex systems blurs the boundaries between the technological and organizational, and typically creates a process fraught with conflict, negotiation, disputes over professional authority, and the conflation of social, political, and technological agendas. Nowhere is this more true than in the history of software.

CODE AS PRODUCT

Many social scientific researchers have therefore approached code from a sociotechnical perspective as something that is thoroughly interwoven with and inseparable from its social processes of production and its social effects as a product. Again, as Mackenzie (2006: 3) argues, much of what is taken for granted about software and new media depends on the 'mundane acquisition and exercise of technical skills in programming and configuring code'; yet while the activities of computer programmers may appear 'merely technical,' they are in fact highly contextualized and thus 'overlap and enmesh with imaginings of sociality, individual identity, community, collectivity, organization and enterprise. Technical practices of programming interlace with cultural practices' (Mackenzie 2006: 4).

Therefore, to understand what code is and what it does, Mackenzie (2006) argues, it is essential to view it not merely as a technical artefact—lines of code—but to understand it as the product of an *originator* (whether a programmer, a software engineer, a corporation, a hacker, or so on); as things or people that act as its *recipients* and are solicited by code to do something (whether a user, another programmer, or a software program executing commands); and as something else that is represented or *prototyped* (such as a desktop user interface representing a set of office activities, a graphic image manipulation programme representing a photography studio, a music making programme representing a music studio, and so on). Code coheres in the relations between originators, recipients and prototypes; conceptually, it cannot be reduced just to machine-readable scripts.

Code therefore needs to be understood as 'both a product of the world and a producer of the world' (Kitchin & Dodge 2011: 20):

> Code as *product*: all the acts of technical R&D, commodity production, business development, and programming that constitute any software product, including all the political, economic and cultural contingencies that contribute to its production.
>
> Code as *producer*: the performativity of computer code and its capacity to mediate, augment and produce the world, located within diversely produced social, cultural, economic and political contexts.

Yet, even as a product, as Kitchin and Dodge (2011: 24) have shown, code is inseparably surrounded by a vast landscape of discursive and material paraphernalia:

- Material objects and spaces: computer hardware, display screens, disk drives and disks, network interfaces, network telecomms infrastructures, hardware peripherals, desks, offices
- Discursive and material assemblages of knowledge: flow charts, manuals, magazines, mailing lists, blogs
- Standards and classifications: data standards, file formats, interfaces, communication protocols, intellectual property regimes such as copyrights, trademarks and patents
- Practices and experiences: ways of doing, coding cultures, hacker ethos, norms of sharing or stealing code

- Subjectivities and identities: coders, programmers, hackers, sellers, marketers, business managers, product managers, entrepreneurs
- Organizations: corporations, consultants, start-ups, government agencies, manufacturers, retailers, universities, conferences, clubs, professional associations and societies
- Marketplaces: coder recruitment agencies, commercial investment, venture capitalism

All of these interconnected and heterogeneous dimensions that make up code shape what it can do and the social and technical effects it is programmed to exert.

This recognition is important because it registers how code emerges from particular decisions made in its production. These decisions, once materialized and mobilized in software, can then exert material effects in the world. Thus software is produced by programmers with particular sensibilities, epistemological assumptions and worldviews, in collaboration with institutions and corporations with their own particular philosophies of the world, ambitions, and resources; these systems of thought and their manifestation into sequences of code and algorithmic processes make a difference to how people do work in the world, how problems are solved, how people relate to one another, and to a variety of contexts in which code reshapes practices, such as in consumption, government, entertainment, health and education (Kitchin & Dodge 2011). In other words, computer code and algorithms are 'abstracted theories about the world' which also 'have the capacity to become active in shaping and constituting social life' (Beer 2013: 80).

PRODUCTIVE CODE

As process and product, code is entangled with social and human activity, and all the discursive and material 'stuff' that constitutes social life. It possesses no simple deterministic or universal function but its effects unfold in manifold different ways in different contexts. The key point to emphasize here is that human action is just one element among a 'vast network of more or less effectively coordinated and stabilized actors' including technical devices, formal protocols, standards and material structures with which 'our activities are more or less successfully calibrated, and by which they are ordered and configured' (Fuller & Goffey 2012: 15). Sociotechnically understood as both a *product* of the world and a relational *producer* of the world, code *acts*: it interpolates, mixes with and ultimately *produces* collective political, economic and cultural life. The power of code is not just in its technical instructions but in how it sinks into discourse, thought, action and even individual and collective identity formation.

Consequently, as code is written, wired and woven into the world in software products, it is now understood among many social scientists as more than just the written script that instructs and controls computing devices, but as a lively and active substratum of everyday life. As Lev Manovich (2013: 15) phrases it, software is '*a layer that permeates all areas of contemporary societies*':

> Therefore, if we want to understand contemporary techniques of *control, communication, representation, analysis, decision-making, memory, vision, writing, and interaction*, our analysis cannot be complete until we consider this software layer. (*original italics*)

Through activating software processes, code organizes, disrupts and participates in contemporary social, economic, political and cultural activities and practices. For example, code makes possible the techniques of data collection, collation and calculation without which, Chun (20011) argues, there would be no government, no corporations, no global marketplace, and no schools. Yet code tends to remain deep in the background of such things, hidden in the 'black boxes' of technologies and obscured by taken for granted discourses, so that its specific functioning is perceived (if considered at all) as opaque and unavailable to common comprehension.

Yet given its sociotechnical capacity as an invisible structural force with the power to order and sequence the social world and 'pattern and coordinate everyday life,' code:

> has been associated with processes of identity formation, new modes of production, commodification and consumption (in the digital economy), and sometimes as a reinvented public sphere…. [These] carry with them notions of agency, either in relation to what software does as a technology or what people do with software as they make use of it. (Mackenzie 2006: 172)

In other words, when we consider software and its constitutive code, we need to be alert to the 'attribution, delegation and extension of agency through code' (Mackenzie 2006: 10).

ACTIVE CODE

To talk about code possessing agency to act autonomously seems bizarre. Does agency reside with the programmer who writes lines of code in order to do something? Does a program act as a vehicle for the agency of the 'user'? Does agency lie with the corporation who owns the IP? Or are 'agency behaviours' now becoming the norm in our expectation of things, as Mackenzie (2006: 9-10) argues, where:

> in milieus populated with bodies, things, systems, conventions and signs (and this is virtually everywhere), agency distributes itself between people or between people and things in kaleidoscopic permutations.

The software-saturated environment is one in which computer code may be seen as increasingly participating agentively in the constitution of everyday life, but this is not straightforwardly about the programmer exerting agency through code, nor of the code exerting agency over the user. Instead, code participates in a distribution of agency among people and things as an actively performing part of the digital materialities (software and hardware devices) that people encounter in their everyday lives.

To give us more sense of just how dense and interactive the sociotechnical relationship between computer code and everyday life has become, as Thrift (2011: 16) has argued, the contemporary 'lifeworld' is one underpinned by a 'technological unconscious' in which 'search engines, social networking sites, web pages, video clips, ringtones and mixes, and maps combine' to produce new forms of experience. This can be seen in a number of examples provided by Thrift, such as how:

- day-to-day activities are increasingly coded and overlaid with digital information (through location-aware mobile devices) to become an augmented reality;
- software 'literally re-cognizes us' as new kinds of intelligence-gathering, active data and predictive software are mobilized in the security and surveillance sector ('predictive policing') and in the entertainment sector ('knowing capitalism') alike;
- live algorithms are programmed with the ability to improvise and interact with the user to 'co-construct worlds';
- the details of the everyday lives of millions of people are able to be uploaded and analysed by commercial companies in order to customize and sell products back to consumers;
- cognition becomes even more of a joint experience between persons and things so that 'things have a say themselves as more than dumb actors as agency is displaced on to a host of new and varied entities,' and 'more and more things are able to become able'.

The contemporary lifeworld depicted by Thrift is one in which digital materials are becoming more knowing, more active, more able, more aware, more cognizing, more anticipatory and predictive, more 'live' and agentive, and everywhere. This is a contemporary lifeworld in which code acts constitutively, participating in a constant productive shaping of the world in which it, too, is constantly being shaped by the social, political, economic and cultural contingencies surrounding its production. In short, code is a social, economic, political and cultural product, but also socially, economically, politically and culturally productive. Tracing the work of code in the social domain of education—in terms both of its production and productivity—is the task taken up in the short papers in this report.

Chun, W.H.K. 2011. *Programmed Visions: Software and memory*. London: MIT Press.

Ensmenger, N. 2010. *The Computer Boys Take Over: Computers, programmers, and the politics of technical expertise*. London: MIT Press.

Ford, P. What is Code? *Business Week*, 11 June 2015: http://www.bloomberg.com/graphics/2015-paul-ford-what-is-code

Fuller, M. & Goffey, A. 2012. *Evil Media*. London: MIT Press.

Hayes, B. 2015. Cultures of code. *American Scientist* 103, no. 1: http://www.americanscientist.org/issues/pub/cultures-of-code

Kitchin, R. & Dodge, M. 2011. *Code/Space: Software and everyday life*. London: MIT Press.

MacCormick, J. (2012). *Nine algorithms that changed the future: The ingenious ideas that drive today's computers.* Oxford: Princeton University Press.

Mackenzie, A. 2006. *Cutting Code: Software and Sociality*. Oxford: Peter Lang.

Manovich, L. 2013. *Software Takes Command: Extending the language of new media*. London: Bloomsbury

Thrift, N. 2011. Lifeworld Inc—and what to do about it. *Environment and Planning D: Society and Space* 29, no. 1: 5–26

Things that matter in education

Tara Fenwick, University of Stirling

Materials—things that matter—are often missing from accounts of educational processes such as learning. Materials, including those digital materialities constituted by software code and algorithms, tend to be ignored as part of the backdrop for human action, dismissed in a preoccupation with consciousness and cognition, or assumed to be subordinate to human intention and design. This sort of treatment still tends to privilege human beings, as though our intentions, thoughts and desires are separate from the materiality that makes us. In educational research, Estrid Sørensen (2009) argues that there is a 'blindness toward the question of how educational practice is affected by materials'.

However in more recent educational studies, researchers have pressed for much more recognition of the ways that materiality actively configures educational practice and knowing, which have tended to be considered as social phenomena. Why this new focus on materials? Materials—objects, bodies, technologies and settings—permit some actions, and prevent others. They convey and indeed produce particular knowledges, and can become powerful. Everyday things such as doors, seat belts, keys and car parks are, as Bruno Latour has written, political locations where values and interests are negotiated and ultimately inscribed into the very materiality of the things themselves, thereby rendering these values and interests more or less permanent. In other words, material and social forces are interpenetrated in ways that have important implications for how we might examine their mutual constitution in educational processes and events.

SOCIOMATERIALITY

Many theoretical approaches might be described as 'sociomaterial', including those inspired by actor-network theory (ANT), STS (science and technology studies), complexity theory, 'new materialisms', activity theory, and spatiality studies (Fenwick & Edwards 2010). These each have very different and often conflicting theoretical and ontological roots, so it is very problematic to simply refer to 'sociomaterial theory' (Fenwick, Edwards & Sawchuk 2011). However it becomes tricky to then talk about 'sociomaterial' approaches without rehearsing a lot of detail to properly delineate the distinctions of all these approaches (Coole & Frost 2010). And of course, this thinking has a long lineage with heavy debts to the thought of Spinoza, Tarde, Heidegger, and Deleuze, among others. Rosi Braidotti's (2013) recent book *The Posthuman* offers a lively history of her own influences by such thinkers.

What ‘sociomaterial’ perspectives tend to share is an interest in materials—particularly in *how materials act* in their interweavings with everyday activity. Wanda Orlikowski (2007) describes this as the constitutive entanglement of the social and material. At a simple level we could talk about ‘materials’ as the everyday stuff in our lives that is both organic and inorganic, technological and natural—everything from our fleshly bodies to databases, snowstorms, dust and passcodes. We can think of ‘social’ dynamics as including meanings, emotions and discourses. The point is to analyse how material and social forces become interwoven so that they produce one another. The focus then is their relationships.

INTRA-ACTIONS

As the complexity physicist Karen Barad (2009) suggests, let us avoid thinking of these relationships as how subjects and objects *inter*-act, as though they are separate entities that then develop connections. Instead she encourages us to examine how these elements and forces penetrate one another, what she describes as *intra-actions* in and among nature, technologies, humanity and materials of all kinds. These intra-actions produce what we tend to treat as separate objects with inherent properties. Barad also points to the ‘apparatuses’ that we use to observe, work with, and make meaning of phenomena. With these apparatuses of language or measuring instruments or analytic logics, we create categories that define subjects and objects—or we use categories that trusted others have produced through observational apparatuses—to cut through materiality, see patterns, make meanings—perhaps develop a sense of control. These ‘agential cuts’ in matter define what we take to be agency, power flows, objects and so forth. But Barad emphasises that these cuts can also open new possibilities, if we would only attend to them. Causality is not about linear relations between causes and outcomes, but multiple entanglements with surprising effects.

What this means then is that all things are viewed as *effects* of connections and activity. Things are performed into existence in webs of relations. This starting point highlights the practices through which boundaries come into being which define things and identities, and which assign value to some while ignoring others. Any practice is a collective sociomaterial enactment, not a question solely of one individual’s skills or agency, or even of the collective skills of a group of people. This view also helps us recognise how materials act, together with other types of things and forces, to *exclude, invite, and regulate* activity.

ASSEMBLING MATERIALS

Objects, practices and phenomena—and actors too, some would argue—are therefore sociomaterial *assemblages*. Some prefer the term ‘agencements’ to avoid the static-sounding ‘assemblage’. They are gatherings of heterogeneous natural, technical and cognitive elements. They also embed a history of assemblages and negotiations and influences. As Ingunn Moser (2008) puts it, they entrench normativities because they align actors and elements in ways that define not just what is, but what ought to be. In examining particular educational practices, researchers ask how and why particular elements became assembled, why some elements become included and others excluded, and how elements change as they come together, as they *intra*-act.

Finally, most sociomaterial perspectives in different ways accept the fundamental uncertainty of everyday life, as well as of the knowledge, tools, environments and identities that are continually produced in it. To emphasise an earlier point, unpredictable problems and possibilities are always *emerging*. This may be a familiar notion, but sociomaterial theories offer specific analytic tools that can examine much more precisely just how these assemblages are emerging—why they come together to produce and mobilise or entrench particular effects, and when they do not. These are processes that complexity theory might explain in terms of 'strong emergence' (see Deborah Osberg 2008), actor-network theorists call 'translation', and new materialists like Braidotti (2013) call 'becoming'. The focus is on relations: how things influence and alter one another in ways that are continuously opening—as well as closing—possibilities.

These are crucial conceptual resources for understanding the emergent effects of code and the digital materialities it assembles into being. As the following papers indicate, code is an assemblage of technical tricks performed by human hands, woven with models of human action embedded in disciplinary and methodological histories, that come together to exert particular effects on their users or on the systems and processes they are designed to enact. The task for researchers is to tease out the relations that make code an integral and intra-active material in the enactment of educational processes, practices and spaces.

Barad K. 2007. *Meeting the Universe Half-way*. Durham, NC: Duke University Press.

Braidotti, R. 2013. *The Posthuman*. Cambridge: Polity Press.

Coole, D. and Frost, D. 2010. Introducing the new materialisms. In D. Coole & S. Frost. Eds.), *New materialisms: ontology, agency, and politics*. Durham: Duke University Press.

Fenwick, T and Edwards R. 2010. *Actor-Network theory in Educational Research*. London: Routledge.

Fenwick, T., Edwards, R., and Sawchuk, P. 2011. *Emerging Approaches to Educational Research: tracing the socio-material*. London: Routledge.

Moser, I. 2008. Making Alzheimer's disease matter: enacting, interfering and doing politics of nature. *Geoforum* 39: 98-110.

Orlikowski, W. J. 2007. Sociomaterial practices: Exploring technology at work. *Organization Studies 28:* 1435-1448.

Osberg D. 2008. The logic of emergence: An alternative conceptual space for theorizing critical education. *Journal of the Canadian Association for Curriculum Studies,* 6 1): 133-161

Sørensen, E. 2009. *The Materiality of Learning: Technology and Knowledge in Educational Practice.* Cambridge & New York: Cambridge University Press.

Software code/social code

Ben Williamson, University of Stirling

Understood as the instructional techniques that enable digital materials to perform, code is important because it forms an active layer of contemporary existence, yet to most people it remains hidden, invisible, and impenetrable. Understood technically, code provides the instructions required by software, but its instructions go beyond software functionality. Instead, code acts as a constant stream of instructions for everyday life, deeply interwoven with how we live, work, learn and identify ourselves, and how we are governed, administered, enabled and educated (Kitchin & Dodge 2011). And while code is humanly-made, scripted into being through the 'secondary agency' of programmers (Mackenzie 2006), it is also becoming increasingly autonomous of human oversight. Code that can write itself, machine learning algorithms, and computers that can think (to some degree) are becoming significant social and technical actors in many aspects of social, cultural, economic and political experience. The sociomaterial perspectives introduced in the previous paper remind us that we need to approach code as an intra-active participant that is in a constant set of relationships with people and things.

So, code is not just a language for computers to read. Lines of code combine with and work in relation to socially-defined codes of conduct as a set of active scripts that are increasingly generative of how people think, feel, act, form identities, and conduct themselves (Mackenzie & Vurdubakis 2011). As the hybrid progeny of a variety of social, human and technical elements, code is increasingly understood to be invested with a kind of performative social power that gives software the capacity to enact tasks, make decisions, and, in part, mediate how people see, know and do things. The apparent capacity of code and its algorithmic procedures to interweave with society, act upon people, augment knowledge, and to mediate and govern their lives in myriad ways, is now becoming the basis for serious social scientific inquiry. Before it can do such things, though, how does it even 'know' enough about the people it is to act upon? How does code know you? How does software see us?

CODING SOCIALITY

A decade ago, the designers Dan O'Sullivan and Tom Igoe (2004) asked the question 'how does the computer see us?' The image of a hand with one finger, one eye, and two ears that they produced—a simple yet weirdly obscene finger-eye-being—is a striking reminder that technologies carry programmed assumptions and knowledge about the human beings who will use them. Computers and software interact with us as its producers have instructed and programmed it to see and identify us. The field of human-computer interaction (HCI) is fundamentally preoccupied with generating insights about users' perceptions, capabilities and tendencies that can then be codified into the functional logics of software and hardware—technologies are always designed with an anticipated user in mind.

These pre-programmed forms of identification do not just emerge out of the air. Nor are they just the wild imaginings of software programmers. They are brought into being through various paradigms, theories and knowledge about human beings that have been developed by various experts from across the human, technical and social sciences, and from there propelled into wider thought. As the philosopher of science Ian Hacking (2007) has argued, how we conceive, characterize and classify ourselves as 'kinds of people' is ultimately mediated by the authority of particular kinds of expert claims and arguments. How does this happen?

Consider, for example, the assumptions and beliefs that many educators carry with them about the students or learners in their classes. Where do these ways of thinking about learners come from? Social psychology has provided many of the conceptual models for understanding learning and learners in recent decades. Instead of the individual child of developmental psychology, the accounts provided by social psychology have reconceived the learner as an active and culturally cognitive participant in everyday experience, learning and developing through social proximity with more knowledgeable others. The result has been that many educators enter the classroom with ways of seeing, understanding and working with learners that are based on the expertise of social psychology. Recent claims from branches of neuroscience about how the brain is activated are also now being mobilized as the basis for new kinds of brain-based learning programmes (Busso & Pollack 2015).

Elsewhere, theories of human behaviour emerging from the field of behavioural economics are being mobilized as the basis for a variety of techniques, such as social marketing and 'nudging,' that are aimed at changing behaviours (Jones, Pykett & Whitehead 2013). These theories assume people are behaviourally malleable, defined by their capacity to be influenced and prompted, liable to becoming attached to group norms, and susceptible to learning behaviours from others through processes of social observation, modelling and imitation. The UK Government Behavioural Insights Team, often called the 'Nudge Unit,' applies insights from behavioural economics and psychology to public policy and services—putting a particular image of human conduct right at the centre of the state's governing strategies.

What these examples demonstrate is that knowledge about humans is always produced via particular conceptual models from specific sites of expertise. The ways in which we, as humans, come to understand, represent and address ourselves is an expert accomplishment mediated through a whole battery of disciplinary theories and techniques that change and evolve and get overturned over time. Social psychology, neuroscience, and behavioural economics are just three prominent examples of disciplines that claim to be able to 'see' and 'know' people with particular authority (or that are at least deployed by certain actors in such a way), and on whose expert authority it is then possible to do such things as educate, govern, or act upon people.

SOCIALIZING CODE

So how does this relate to software? Does software now come with claims to be able to 'know' and 'see' us? What we need to explore here are the kinds of models or ways of thinking about people that are mobilized in the production of particular software products. I am not suggesting that software has some kind of autonomous sentience (though it can give the impression of being semi-alive), but that in its processes of production, the originators of any software product codify particular models of human activity and interaction in the software itself. These are expertly crafted through specific methodological techniques and theoretical models. As Tarleton Gillespie (2014: 174) has argued in relation to search engine companies:

> information providers conduct a great deal of research trying to understand, and then operationalize, how humans habitually seek, engage with, and digest information. Most notably in the study of human-computer interaction (HCI), the understanding of human psychology and perception is brought to bear on the design of algorithms and the ways in which their results should be represented.

In other words, code (and the things it materializes) is socially produced, and it carries codified assumptions about forms of social conduct and human behaviour that can then socialize those actors with whom into comes into contact.

To give some sense of this, we might want to consider a popular social networking site like Facebook. Facebook is built upon particular understandings of people as socially networked beings. The kinds of networking possibilities built into Facebook and the like tend to emphasize the idea of networks of horizontally connected friends. Facebook's 'people you may know' algorithm seeks to optimize users' network sociality by establishing a kind of algorithmic normality for social relations. These kinds of network architectures are supported by a whole panoply of concepts and theories which attempt to articulate their own 'authoritative' accounts of humans as socially networking creatures.

The symmetry between the rise of social networking in the domain of software and the rise of new social concepts in the domain of sociality can be detected in the recent spread of terms like 'smart mobs,' 'participatory cultures,' 'networked individualism,' 'crowdsourcing,' 'wikinomics,' 'educational ecosystems,' 'prosumption,' the 'social brain,' and, of course, the 'social network' itself (e.g. see Rainie & Wellman 2012). All of these terms appear to express something natural and given about human sociality—as if the evolution of our psyches and our brains has actually demanded the social network media that now flood telecommunications infrastructures (Rose 2013).

Yet it is important to remember that all of these terms are the products of particular expert models and theories about people as both individuals and social creatures. Many of these accounts draw on sociological concepts, on social psychology, and on emerging theories about the social brain from neuroscience, as well as on concepts from economics. The idea that human beings are a fundamentally social species, with a social brain embedded in complex social networks and cultural norms, is becoming the default theory of human nature, behaviour and sociality among organizations and research centres involved in influencing and governing public policy and education. The idea of the 'social brain' of the individual connected to others via 'social networks' is a particular image of the human, and of human sociality, that has been produced by various discursive and technical means by actors working with these various disciplinary ideas and styles of thinking—it is not natural and pre-given. The concept of the 'social network' itself originates in the sociometrics approach to mapping human relationships instituted by Jacob Moreno in the 1930s (Davies 2015). The symmetries between social scientific and computer science models and understandings of individual and social behaviour are disciplinary, discursive and technical accomplishments that are situated in, and saturated by, historical particularities of thinking.

Paradigms and ideas from these various disciplines—all of which seek to describe and explain human individuality and sociality—are now being modelled in code in ways which have the potential to 'configure the user,' in Steve Woolgar's (1991) memorable phrase. The existence of in-house social and behavioural experts at places like Facebook, Google and Microsoft demonstrates the close connection between the expertise of the human and computer sciences in the coding of much contemporary software. We might say, then, that sites like Facebook, and the combined social scientific and computer science expertise behind them, conceive of people in terms of social behaviours that have been classified and characterized by particular kinds of expertise and ways of thinking. If its software sees us as individual nodes in horizontal networks of social attachments, this is at least in part the result of disciplinary insights from the social and human sciences.

We might also want to consider the kinds of data mining technologies and other database-driven systems that 'know' us through the collection of our digital traces and byproduct data. As David Beer (2012) reports, the kind of software that can crawl, capture and scrape the web for data is becoming a powerful and largely autonomous actor in contemporary societies. Social media aggregators, algorithmic database analytics and other forms of 'sociological software' have the capacity to see social patterns in huge quantities of data and improve how we 'know' ourselves. As Tarleton Gillespie (2014) has argued in recent research on algorithms in public life, software 'anticipates' its users through the constant collection of their digital traces: 'digital providers are not just providing information to users, they are also providing users to their algorithms.' Additionally, Evelyn Ruppert (2012) has argued that the database makes people visible, knowable and therefore amenable to classification and intervention. These data can then be used as the basis for 'doing' things to people—whether by making probabilistic recommendations for consumer items, or by 'personalizing' state services.

KNOWING CODE

Increasingly, it seems as though people are to be known and remembered through their data traces and the code that facilitate them rather than through their embodied lives. The 'knowers' in what Nigel Thrift (2005) has termed contemporary 'knowing capitalism' in this sense are social media and database software devices. To put it bluntly, as Geoffrey Bowker (2013) argues, 'if you are not data, you do not exist'—you are invisible and unknown to the organizations and agencies that, through software-mediated means, classify and govern so much of contemporary existence. As studies of software help to reveal its dynamics, we are coming to know code better, and to recognize how *knowing* code itself is becoming.

Software now sees us and code knows us as particular 'kinds of people.' This is not some universal person or 'natural kind' but hybrid products of digital data and theories of human individuality and sociality all enacted by software code, or a 'new algorithmic identity' inferred from calculations on our digital traces, as John Cheney-Lippold (2011) terms it. Software sees and code knows us as hybrid products made up out of data traces of activities and interactions, as well as out of particular conceptions, theories and imaginings of what it is to be a person.

If we recognize that all software, and its underlying code, carries assumptions about actual or potential users, and that these models of human being can then shape how people go about seeing, knowing and doing things, then it becomes essential to do close studies unpacking how those assumptions came to be codified in the software. If as researchers we want to unpack how code acts in education—or in other social contexts—we need to work out what assumptions, models or concepts of human action, interaction and behaviour, and what ways of seeing, knowing and doing things, have been programmed in to the specific software we are interrogating. We need to investigate what social codes of conduct are written into the code, and to conduct genealogical explorations of the claims that underpin the codifying of conduct in software. We need to understand how software programmers engage with social and human sciences, and to trace the kinds of theories of human nature and sociality that they build into their products. The issue is how being known and seen, characterized and classified in code might become the basis for being governed and educated. In this respect, we might need to inquire into the 'knowledge infrastructures' of education—the subject of the following paper.

Beer, D. 2012. Using Social Media Data Aggregators to Do Social Research. *Sociological Research Online* 17, no. 3: http://www.socresonline.org.uk/17/3/10.html

Bowker, G. 2013. Data flakes: An afterword to 'Raw Data' is an Oxymoron. In L. Gitelman Jackson (ed). *'Raw Data' is an Oxymoron*: 167-71. London: MIT Press.

Busso D, Pollack C. 2015. No brain left behind: Consequences of neuroscience discourse for education. *Learning, Media, and Technology* 40, no. 2:168-186.

Cheney-Lippold, J. 2011. A new algorithmic identity: soft biopolitics and the modulation of control. *Theory, Culture & Society,* 28, no. 6: 164-18

Davies, W. 2015. The return of social government: From 'socialist calculation' to 'social analytics'. *European Journal of Social Theory*: http://dx.doi.org/10.1177/1368431015578044

Gillespie, T. 2014. The relevance of algorithms. In T. Gillespie, P.J. Boczkowski & K.A. Foot (eds). *Media Technologies: Essays on communication, materiality and society*: 167-194. London: MIT Press.

Hacking, I. 2007. Kinds of people: Moving targets. *Proceedings of the British Academy* 151: 285-318.

Jones, R., Pykett, J. & Whitehead, M. 2013. *Changing Behaviours: On the rise of the psychological state*. Cheltenham: Edward Elgar.

Kitchin, R. & Dodge, M. 2011. *Code/Space: Software and everyday life*. London: MIT Press.

Mackenzie, A. 2006. *Cutting Code: Software and Sociality*. Oxford: Peter Lang.

Mackenzie, A. & Vurdubakis, T. 2011. Codes and codings in crisis: Performativity, signification and excess. *Theory, Culture & Society* 28, no. 6: 3-23.

O'Sullivan, D. & Igoe, T. 2004. *Physical Computing: Sensing and Controlling the Physical World with Computers*. Boston: Course Technology Press.

Rainie, L. & Wellman, B. 2012. Networked: The new social operating system. London: MIT Press.

Rose, N. 2013. Governing socialities: Notes for an investigation. Paper presented at *The New Social-ism: What is valuable, visible & knowable in the emerging social economy?* Centre for Interdisciplinary Methodologies, University of Warwick, 11 December 2013.

Ruppert, E. 2012. The Governmental Topologies of Database Devices. *Theory, Culture & Society* 29, no. 4-5: 116-136.

Thrift, N. 2005. *Knowing Capitalism*. London: Sage.

Woolgar, S. 1991. Configuring the User: The Case of Usability Trials. In Law, J. (ed.) *A Sociology of Monsters: Essays on Power, Technology and Domination*: 57–99. London: Routledge.

Digital knowledge infrastructures of education

Richard Edwards, University of Stirling

In their influential book *Code/Space*, Rob Kitchin and Martin Dodge (2011) detail the ways in which many of the infrastructures underpinning the modern world depend on code for their functioning. They detail the existence of 'coded infrastructures' such as computing networks, communication and broadcast entertainment networks, transport and logistics networks, security and policing networks, and even satellite-based global positioning systems. These vastly complex infrastructure networks orchestrate many aspects of daily living and embed code firmly within the systems that constitute contemporary societies. For example, smartphones would be dumb without telecommunications infrastructures; traffic flow in cities would slow to gridlock without traffic management systems; government could not govern without administrative infrastructures, and so on.

Less noted, however, are the 'knowledge infrastructures' that organize the information through which particular things are known, understood, and managed. As digital data are increasingly used to 'see' and 'know' social worlds, the knowledge infrastructures within which such information is collected, analysed and made visible are becoming increasingly significant. In the book *A Vast Machine*, Paul Edwards (2010) refers to knowledge infrastructures as the networks of people, artifacts, and institutions that generate, share and maintain specific knowledge about the human and natural worlds. In another report, knowledge infrastructures are defined as:

> ecologies, or complex adaptive systems; they consist of numerous systems, each with unique origins and goals, which are made to interoperate by means of standards, socket layers, social practices, norms, and individual behaviours that smooth out the connections among them. (Edwards *et al.* 2013: 5)

Both the development of such infrastructures and their effects raise important questions for education. We can think of all education as forms of knowledge infrastructures, but in this paper, I am particularly concerned with the work of digital technologies in the knowledge infrastructures of education.

THE HIDDEN CURRICULUM OF CLASSIFICATION SYSTEMS

Recent developments in digital technology and growing research on the work of software, code and algorithms (e.g. Manovich 2013) and knowledge infrastructures (e.g. Lampland & Star 2009) in daily life have the potential for providing significant resources through which to explore the embedded work of digital technologies and knowledge infrastructures in the practices of education—what I describe as the hidden curriculum of software (Edwards 2014). This research points us towards the knowledge infrastructures, software and associated practices through which digital education is enacted. Knowledge infrastructures do not simply represent data. They select, translate and transform them. It is the ontologies, codes, algorithms and the linking of data, the applications of technical standards, and ways in which decision-making and reasoning are articulated in digital technologies that make things perform in ways and become specific actors in particular educational practices.

In relation to this, a significant role is played played by forms of classification, standards and ontology-building associated with the development of digital databases, and the ways in which complex knowledge is represented (Millerand & Bowker 2009). To classify requires the removal of ambiguity from representation, when of course many knowledge claims are ambiguous and contested. As Paul Edwards and colleagues (2013: 10) argue, the digitalization of data raises three main issues:

> first, a plethora of 'dirty' data, whose quality may be impossible for other investigators to evaluate; second, weak or non-existent guarantees of long-term persistence for many data sources; and finally, inconsistent metadata practices that may render reuse of data impossible—despite their intent to do the opposite.

In similar vein, Susan Halford, Catherine Pope and Mark Weal (2013) explore the implications of the development of the semantic web and the promises propounded about the internet as a linked database. The big promises of open data are related in terms of transparency, transcending knowledge silos and the potential to make greater advances in knowledge. These are not unimportant. However, they also point to some of the challenges associated with such promises, not least, the naming of data entities, the structuring of data and the processing of data. To name and categorize an entity in a consistent way across space and time is not without its challenges, not least because such categories themselves might be subject to challenge. These challenges are commonly identified by those who research knowledge infrastructures. As a result, Halford and colleagues (2013: 178-9) argue that in the development of such infrastructures 'making some things "known" tends to obscure other things and, indeed, ways of knowing' and that 'ontology building is not a simple or solely technical matter.'

ONTOLOGY-BUILDING

In this respect, ontology-building, the naming and structuring of digital data in the enactment of knowledge infrastructures, has itself become a subject of increasing research. For example, in their paper 'Between meaning and machine,' David Ribes and Geoffrey Bowker (2009: 199) argue that:

> Ontologies are an information technology for representing specialized knowledge in order to facilitate communication across disciplines, share data or enable collaboration. In a nutshell, they describe the sets of entities that make up the world-in-a-computer, and circumscribe the sets of relationships they can have with each other. They are a complex and ambitious technical

> approach to address the problem of diverse languages, heterogeneous categorizations and varied methods for organizing information. In the wake of ontologies the information of a domain is substantially reorganized, facilitating data exchange and reuse.

Ontologies are fundamental to the work of digital technologies in knowledge infrastructures, but how they are developed and the extent to which that process is taken for granted once they are developed is critically important in relation to the digital representation of knowledge in education. In their study of the development of an ontology, Ribes and Bowker (2009: 210) found that for those scientists involved 'the primary orientation… was to complete a working ontology rather than a coming to a definitive resolution.' This was because the outcome was determined by the pragmatic digital requirements for the data to be machine readable. Thus, it is arguable that the representation of data for the purposes of digitalization requires different qualities than those associated with existing practices in research and pedagogy. In this process, one critical dimension was not achieved in practice—the representation of disagreements, uncertainties, ambiguities and ambivalences. These are qualities that it might be argued are critically important to a worthwhile education.

The scientists involved in Ribes and Bowker's study became 'lay ontologists', but for those involved in using rather than developing the ontology, the decisions necessary to enable the information to be machine-readable are hidden. These practices of inclusion and exclusion in developing ontologies have been identified in similar studies. A close ethnographic study of ontology-building by Dave Randall and colleagues (2011: 221), for example, concluded that 'much time and effort is spent reaching agreement about what should be in a given ontology and what should be left out.' Thus, Edwards and his co-authors (2013: 14) argue that in their development, knowledge infrastructures 'not only provide new maps to known territories—they reshape the geography itself.' We therefore see how digital technologies raise important questions about the politics of knowledge, as 'turning everything into data, and using algorithms to analyze it changes what it means to know something,' as Lev Manovich (2013: 337) puts it in his recent book *Software Takes Command.*

DISAPPEARING DATA & NON-KNOWLEDGE

Ribes and Bowker (2009: 211) also point to the importance of temporality in relation to ontologies: 'as knowledge, terminology or concepts change within the scientific community, a once-accurate ontology could become obsolete'. With the passing of time and the incorporation of such data into new knowledge infrastructures, the pre-history of data, the selections and applications of ontologies and standards, and the application of rules can disappear from view. As Edwards and colleagues (2013: 7) suggest, 'the presentation of datasets as complete, interchangeable products in readily exchanged formats… may encourage misinterpretation, over reliance on weak or suspect data sources, and 'data arbitrage' based more on availability than quality'.

This points to the increasing complexity of the work of digital technologies in knowledge infrastructures within education, the tracing of which is and is likely to become ever more complex. As Edwards and co-authors (2013: 15) suggest, in articulating what is known, we also need to engage with the 'accidental and systematic means by which non-knowledge is produced and maintained'.

Computer code and algorithms are integral to the production and non-production of the knowledge that is now held about education, and that is used to make decisions about it.

Edwards, R. 2014. Software and the hidden curriculum in digital education. Pedagogy, Culture & Society 23, no. 2: 265-279

Edwards, P. 2010. *A Vast Machine: Computer Models, Climate Data, And The Politics Of Global Warming*. Cambridge, MA: MIT Press.

Edwards, P., S. Jackson, M. Chalmers, G. Bowker, C. Borgman, D. Ribes, M. Burton and S. Calvert. 2013. *Knowledge Infrastructures: Intellectual Frameworks And Research Challenges*. Ann Arbor: Deep Blue.

Halford, S., C. Pope & Weal, M. 2012. 'Digital Futures? Sociological Challenges And Opportunities In The Emergent Semantic Web'. *Sociology* 47 (1): 173-89.

Lampland, M. and S. Star (eds). 2009. *Standards And Their Stories: How Quantifying, Classifying, And Formalizing Practices Shape Everyday Life*. Ithaca: Cornell University Press.

Millerand, F. and G. Bowker. 2009. 'Metadata Standards: Trajectories And Enactment In The Life Of An Ontology'. In *Standards And Their Stories*: *How Quantifying, Classifying, And Normalizing Practices Shape Everyday Life,* edited by M. Lampland and S. Star. Ithaca: Cornell University Press.

Manovich, L. 2013. *Software Takes Command: Extending The Language Of New Media*. London: Bloomsbury Academic.

Randall, D., R. Proctor, Y. Lin, M. Poschen, W. Sharrock, and R. Stevens. 2011. 'Distributed Ontology Building As A Practical Work'. *International Journal of Human-Computer Studies*, 69: 220-33.

Ribes, D. and G. Bowker. 2009. 'Between Meaning And Machine: Learning To Represent The Knowledge of Communities'. *Information and Organization*, 19: 199-217.

Organizing algorithms in digital education

Ben Williamson, University of Stirling

Algorithms have become powerful devices in the knowledge infrastructures of digital education, as the previous paper has begun to indicate. In addition, recent news reports have begun to reveal how various analytics companies are now utilizing powerful algorithmic processes in the data mining of millions of children. The learning analytics company Knewton, for example, claims that 4.1 million students are now using its proficiency-based adaptive learning platform, which has served 3.5 billion total recommendations for learning tasks between May 2013 and May 2014 alone. The role of these predictive analytics platforms and recommender systems in education is increasingly causing political and parental concerns, largely related to privacy and data security issues. Less acknowledged, however, is the increasingly autonomous and automated capacity of the software algorithms working in the background of these platforms.

Although most users and learners remain barely aware of their existence, algorithms are an active presence not only in the predictive analytics and recommender systems of adaptive learning platforms, but in the social networking sites where 'networked publics' hang out, in the information practices deployed in inquiry learning, in techniques of digital making, and in the ed-tech software promoted in classrooms. To put it bluntly, algorithms are now deeply embedded in the governance of education and learning—where governance means the techniques by which people's actions, thoughts and ways of conducting themselves are evaluated, shaped and sculpted.

However, many of us remain only dimly aware of what algorithms are or how they operate. They are, in fact, deceptively complex to define. Kitchin (2014a) demonstrates how algorithms need to be understood as 'black boxes' that are hidden inside intellectual property and impenetrable code; as 'heterogeneous systems' in which hundreds of algorithms are woven together in relation with datasets, companies, programmers, standards and laws; as 'emergent' and evolving systems that are constantly being refined, reworked, and tweaked; and as complex, unpredictable, and fragile systems that are sometimes miscoded, buggy, and 'out of control.' As a consequence, it may make little sense to interrogate any algorithm singularly, but rather to unpack complex 'algorithmic systems.' Additionally, as Tarleton Gillespie (2014b) notes, 'algorithm' may be a 'digital keyword' for our times, but technical communities such as computer scientists and programmers may be using it in very different ways to social scientists, journalists or the wider public. So what do algorithms do, what are some of the basic principles of their functioning, how much power do they have in the social ordering and governance of education, and how might they be influencing the lives of learners?

WHAT DO ALGORITHMS DO?

In the recent book *9 Algorithms that Changed the Future*, the computer scientist John MacCormick (2012) defines an algorithm simply as 'a precise recipe that specifies the exact sequence of steps required to solve a problem.' In computer science specifically, he explains, algorithms are the fundamental entities that computer scientists grapple with to accomplish a task, and without them there would be no computing. In addition, all computer algorithms are more or less meaningless without sources of data. Algorithms require some form of input, such as an unsorted list of data, to transform into an output. A search algorithm, for example, works upon a vast database of information that must be indexed, parsed and stored in advance to facilitate fast and accurate information retrieval. This involves companies such as Google crawling the web collecting and indexing information, logging search queries and links clicked, in order to generate the data required to allow the search algorithms to function autonomously. However, algorithms are more than simply computer science abstractions and routines for sorting and structuring data, but, as MacCormick argues, 'have a profound effect on our lives.'

ORDERING ALGORITHMS

Although computer scientists and social scientists aren't always thinking of the same thing when they talk about an algorithm, social science has now begun to acknowledge their profound effects. In a chapter defining algorithms as an object of social scientific study, Andrew Goffey (2008) argues that algorithms can be understood as things that can 'do things,' enabled by the 'command structure' of their programming, and that can therefore exert material effects 'on themselves, on machines and on humans.' The development of new forms of algorithmically enabled analysis such as 'social physics' and cyberbolic claims that the entire 'universe is programmable' (Axline 2014) reflect the extent to which algorithms are now understood to be 'doing things' in a variety of ways. These interests have only grown amid both the hype of the 'data revolution' (Kitchin 2014b) and the 'anxieties of Big Data' (Crawford 2014), where algorithms play a huge role in the management, analysis and visualization of massive datasets, and in the response to the revelation that Facebook has conducted experiments on users' emotions by filtering their news feeds using its EdgeRank algorithm, where the significance of algorithms in 'engineering the public' (Tufekci 2014) has been amplified.

Although the role of algorithms in the social world is by no means uncontested among social science researchers, there is some broad agreement that algorithms are now increasingly involved in various forms of social ordering, governance and control. A brief survey of the academic field reveals how algorithms have emerged as an important object of analysis in studies of surveillance, identity formation, popular culture, digital governance, and algorithmic research methods, as well as more widely in debates about the apparent power and control that algorithms command (e.g. Amoore 2009; Beer 2009; Bucher 2012; Cheney-Lippold 2011; Lash 2007). According to Adrian Mackenzie (2006), for example, the central point about algorithms is the way they establish certain forms of 'order,' 'pattern' and 'coordination' through processes of sorting, matching, swapping, structuring and grouping data. In this sense, algorithms appear as new kinds of 'social rules' that then can shape everyday life. As David Beer (2013: 81) argues, 'algorithms are an integrated and irretractable part of everyday social processes,' with the potential 'to reinforce, maintain or even reshape visions of the social world, knowledge and encounters with information.'

Likewise, Rob Kitchin and Martin Dodge (2011: 248) suggest that 'algorithms are products of the world' which can also 'produce knowledge that then is applied, altering the world in a recursive fashion.' As such, they argue that algorithms provide 'grammars of action' for new forms of social ordering and governance, and are endowed with the power to 'actively reshape behaviour.' These accounts produce an image of algorithms as powerfully automated, autonomous and recursive technologies—socially produced and yet increasingly capable of producing new social formations, encounters and knowledge.

In order to reshape behaviour in such ways, algorithms require data and models to work on. In a recent article on 'organizing algorithms,' Daniel Neyland (2014) argues that in order for an algorithmic system to function, the world outside of the system has to be mathematically modelled in such a way that it can be built-in to 'the social world of the algorithmic system.' In other words, the selection of an algorithm only comes after the formulation of a model in computational terms. Google's driverless car, for example, relies on built-in ultra-precise digitized maps to navigate the physical world—a compelling case of what Neyland (2014: 11) describes as building 'a world out there into a world in here, in the algorithmic machine.' The title of Neyland's article, 'organizing algorithms,' therefore has a double meaning: it refers both to the social and human organization of algorithmic systems as models of the world, and to the ways those algorithmic systems then subsequently interact in the social organization of that world.

To give another example of such a social logic of algorithmic interaction, in research on the politics of search engine algorithms Astrid Mager (2012) has shown how algorithms are not merely neutral mathematical devices but designed to function according to particular powerful ways of perceiving the world, political assumptions, and the codes of conduct to which their designers and promoters have subscribed. Search engine algorithms reinforce existing values held about social activities and certain ways of ordering the world, or models of social order and organization, which they then project and reproduce as they interact with it. Mager terms this 'algorithmic ideology.'

Moreover, however, many innovations in recent algorithm design mean that algorithms are increasingly built to adapt as they interact with the world. Their capacity is based on innovations in optimizing the predictive power of statistical models. Processes such as 'machine learning' rely on adaptive algorithms and statistical models that can be 'fed training data'; these are, crudely speaking, algorithms that can learn from being taught with example data. As Mackenzie (2015: 430) notes, machine learning techniques of recommendation, recognition, ranking, pattern-finding and, especially, the 'production of prediction,' are increasingly 'woven into the fabric of everyday life.' Clearly there are important questions to address about the selection of the training data that the algorithm is expected to learn from in the production of prediction. Just as educational sociology has always addressed the question of how knowledge is selected for inclusion (or exclusion) in school curricula, and how this might reproduce existing forms of social organization and control, we might ask about the values and assumptions underpinning the training data taught to machine learning algorithms, or excluded from it, and how this, too, might reproduce particular assumptions or preferred models of social and political order.

The importance of machine learning algorithms is that they not only have the power to reproduce the world in the image in which they were programmed, but exhibit some tendencies of emergence, adaptivity, anticipation and prediction. In this sense, machine learning algorithms are not only social inventions capable of reinforcing existing forms of social order and organization, but have a powerfully productive part to play in predicting and even pre-empting or automatically organizing future events, actions, and realities. This is important not least because in machine learning theory and practice there remain serious questions about the capacity of a given predictive model to generalize from available data to other events; modelling predictions on the 'known data' too closely can mean the model adapts to the training data and prevents effective generalization to other sources of data or the incorporation of new information (Mackenzie 2015). In education, the problem of machine learning generalization has clear implications as the learning algorithms may not adapt well to the individual user, but instead produce predictive models of learners' progress that map closely on to the original training data.

SOCIOALGORITHMIC GOVERNANCE

Given such accounts of the complex relational interweaving and interaction between algorithms and social worlds it is perhaps more accurate to write of 'socioalgorithmic' processes and practices than of 'algorithmic power.' It is important to acknowledge that algorithms are socially produced through mixtures of human and machine activities, as well as being socially productive, than to imply that they act deterministically as mechanical or objective technologies. A recent article by Tarleton Gillespie (2014b) suggests that 'algorithm' is just an abbreviation for a vast sociotechnical assemblage of technologies, 'human hands' and social endeavours.

In a previous piece, Gillespie (2014a) notes that publicly relevant algorithms such as those that enact search engines and social networking site functionalities, organize information according to human but increasingly automated evaluative criteria, and on that basis then tangle with users' information practices and patterns of political and cultural engagement. Algorithms reflect a long tension between notions of autonomous human sociality and the imposition of systemized procedures, and privilege automaticity, quantification, proceduralization and automation in human actions.

Along these lines, Gillespie suggests that algorithms are an important object of study for several reasons. He argues that there are questions about how certain forms of data are chosen, selected, included and prepared for processing by algorithms; about the evaluative criteria written into the command structure of algorithms to determine what is relevant, legitimate and appropriate; and about the way the technical nature of algorithms is presented as a guarantee of impartiality and objectivity. Moreover, he argues, as algorithms are increasingly being designed to anticipate users and make predictions about their future behaviours, users are now reshaping their practices to suit the algorithms they depend on. This constructs 'calculated publics,' the algorithmic presentation of a public that shapes its sense of itself.

EDUCATIONAL CALCULATED PUBLICS

Gillespie (2014a: 189) raises a series of questions about the tension between 'networked publics' forged by users and the calculated publics offered by algorithms that have immediate resonance in discussions about digitally-mediated learning:

> Algorithms … engage in a calculated approximation of a public through their traceable activity, then report back to them …. But behind this, we can ask, What is the gain for providers in making such characterizations, and how does that shape what they're looking for? Who is being chosen to be measured in order to produce this representation, and who is left out of the calculation? And perhaps most importantly, how do these technologies, now not just technologies of evaluation but of representation, help to constitute and codify the publics they claim to measure, publics that would not otherwise exist except that the algorithm called them into existence?

These are important issues to address as we consider the role of digital technologies and algorithms in education systems and learning, particularly with the growth of algorithm-dependent adaptive learning platforms and predictive, personal analytics.

Personal and learning analytics utilize adaptive machine learning algorithms and statistical models to analyze users' data in order to anticipate or even predict individuals' actions, behaviours and attitudes. Such algorithmic systems function through a quantitative logic that privileges numerical data about learning, and that also privileges particular views about social interaction and personal dispositions that can be captured in techniques of social network analysis and psychometric profiling. This presupposes that if we want to 'improve' learning then it is possible to do that as long as we can accurately measure such things as learners' social network connections, psychological resilience and so on. And if we can measure those things using the past data available, then we can employ various forms of analytics and data processing algorithms to predict how they might play out in the future too. In this way, education is becoming increasingly focused on the production of calculated publics whose learning lives are measured out by algorithmic analytics systems.

Based on socio-algorithmic techniques of machine learning, psychometric learner profiling, and predictive modelling, the aim of learning analytics is to create 'smart' pedagogic systems, or new kinds of 'database pedagogies' that are increasingly automated and autonomous of human oversight. Utilizing these forms of algorithmic calculation—which enact particular socially produced models of the world, values and assumptions—platforms such as Knewton make prediction and anticipatory knowledge very powerful in education. The algorithms behind these platforms are programmed (or taught) in such a way that they have the power to make predictions on the basis of which decisions can be made, or even on the basis of which automatic recommendations can be produced. The algorithm is a means of anticipating or foreseeing probable future events, and can produce 'actionable insights' as it says on the Knewton website. These algorithmically powered platforms have very real and significant implications for how individuals are evaluated, assessed, and treated—in other words, how they are governed.

Algorithms, as parts of human-machine mixes and socioalgorithmic processes, are contributing to the production of new calculated publics in education. The growing role of various analytics platforms, data mining processes, and other forms of machine learning in education therefore demand a close attention to the work that algorithms do among researchers and educators in this space. Ultimately, we need to consider how certain forms of data are chosen for collection by these platforms; what models of 'learning' are built-in to them; inquire into the evaluative criteria, social norms, and promises of objectivity embodied by their algorithms; and consider how these algorithms are constructed to anticipate and predict learners' likely futures. Calculated educational publics such as those embodied in the millions of users of Knewton are not just algorithmically approximated from traces of their activities, but actively called into existence by data analytics that are able to anticipate learners' probabilities for action, construct predictive models, and serve up recommendations to pre-empt and govern their future lives. The collection, calculation and communication of digital data in education is the subject of the papers in the next section.

Amoore, L. 2009. Algorithmic war: everyday geographies of the war on terror. *Antipode* 41, no.1: 49-69.

Axline, K. 2014. The universe is programmable: We need an API for everything. *Wired.com*, 20 April 2014: http://www.wired.com/2014/04/the-universe-is-programmable

Beer, D. 2009. Power through the algorithm? Participatory web cultures and the technological unconscious. *New Media and Society* 11, no. 6: 985-1002.

Beer, D. 2013. *Popular Culture and New Media: The politics of circulation*. London: Palgrave Macmillan.

Crawford, K. 2014. The Anxieties of Big Data. *The New Inquiry*, 30 May 2014: http://thenewinquiry.com/essays/the-anxieties-of-big-data

Gillespie, T. 2014a. The relevance of algorithms. In T. Gillespie, P.J. Boczkowski & K.A. Foot (eds). *Media Technologies: Essays on communication, materiality and society*: 167-194. London: MIT Press.

Gillespie, T. 2014b. Algorithm. *Culture Digitally*, 25 June 2014: http://culturedigitally.org/2014/06/algorithm-draft-digitalkeyword (accessed 27 June 2014).

Goffey, A. 2008. Algorithm. In M. Fuller (ed.) *Software Studies: a lexicon*: 15-20. London: MIT Press.

Kitchin, R. & Dodge, M. 2011. *Code/Space: Software and everyday life*. London: MIT Press.

Kitchin, R. 2014a. Thinking critically about and researching algorithms. The Programmable City working paper 5: http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2515786

Kitchin, R. 2014b. *The Data Revolution: Big data, open data, data infrastructures and their consequences*. London: Sage.

Lash, S. 2007. Power after hegemony: Cultural studies in mutation? *Theory, Culture & Society* 24, no. 3: 55-78.

MacCormick, J. (2012). *Nine algorithms that changed the future: The ingenious ideas that drive today's computers.* Oxford: Princeton University Press.

Mackenzie, A. 2006. *Cutting Code: Software and sociality*. Oxford: Peter Lang.

Mackenzie, A. 2015. The production of prediction: What does machine learning want? *European Journal of Cultural Studies* 18, no. 4-5: 429-445.

Mager, A. 2012. Algorithmic ideology: How capitalist society shapes search engines. *Information, Communication & Society,* 15, no. 5: 769–787.

Neyland, D. 2014. On organizing algorithms. *Theory, Culture & Society*, http://dx.doi.org/10.1177/0263276414530477

Tufekci, Z. 2014. Facebook and engineering the public. *Medium*, 29 June 2014: https://medium.com/message/engineering-the-public-289c91390225

data, analytics & visualization of education

Digital systems for the collection, calculation, and communication of educational data have grown rapidly. What are the consequences of big data, data analytics, and data visualization for ways in which educational spaces and practices are seen, known and enacted?

Knowing subjects–data-driven perspectives on education

Lyndsay Grant, University of Bristol

Jenny Ozga's presentation at the second Code Acts in Education seminar in Edinburgh in May 2014 showed how data is becoming an increasingly important actor in governing education, internationally, nationally and in schools. But what is it about data that makes it so powerful?

Following Ozga's presentation, participants began discussing how and why quantitative data had come to be seen as providing a more objective and reliable way of knowing what was going on in schools and in children's learning than other ways of understanding. Some of those in the discussion, with a natural science background, described how their training led them to be very sceptical about 'raw' quantitative data. Before drawing conclusions, they would consider the accuracy of their data collection instruments and processes, what was missed in their data sets, the subjective choices they had made about which statistical analysis procedures to use, and seek possible alternative interpretations for the data they produced. Data never just 'spoke for itself'. Yet data in education often appears to be quite uncritically accepted. As a group of researchers, used to the idea that both quantitative and qualitative data could both provide useful ways of understanding education, we wondered why it is that large-scale quantitative data seems to be becoming the only sort of data that matters. In educational policy-making and regulation, it seems that 'big data' is able to lay claims to greater legitimacy and authority than other ways of knowing.

DECONTEXTUALIZING DATA

I gained some insight into this process at a recent conference on the use of the Department of Education's National Pupil Database in England. Amanda Spielman, Chair of Ofqual and Advisor to the Ark chain of academies, discussing the use of data for target-setting, described what she called a strange process of 'transubstantiation'. At the point when data was collected, teachers were well aware that it was affected by context, it might contain errors, and only provided a single snapshot of a particular learner. But when the data was entered into the official RaiseOnline database, it seemed to undergo a mystical transformation, and was treated with such great authority that teachers would disregard their own judgement in favour of RaiseOnline when setting pupil targets.

The way that data gets decontextualized for entry into a database is one of the ways that it acquires this apparent objectivity and associated authority. Jenny Ozga had described how, in the Department for Education in England, they have a large screen in a central area called 'the bridge', showing at a glance performance data across local authorities, as a way of facilitating central performance management. Ozga described how, when inspectors challenged a school's performance data, there would be opportunities for head teachers to re-contextualise their data, to provide the missing, unquantified information necessary for a fuller interpretation. At an aggregated national level, this level of specificity simply isn't possible—the overview provided by 'the bridge' depends on the standardization and decontextualisation of data. To generate an overview, databases must exclude large amounts of information, forgetting far more than they remember. If everything were included we would only be able to see every case as individual and unique. But when interpreting this data, we would do well to acknowledge what has been excluded when we come to consider how much weight to put on it, and how much interpretation it can bear when it is used in performance management or policy formulation purposes.

Tarleton Gillespie (2014) describes how data objectivity is regularly performed as a feature of algorithmic systems in the same way that statistical data boosts scientific claims, partly because 'human hands' seem to be removed from the system. Claims based on data are seen as more legitimate precisely because they are divorced from the messy, specific, local and contingent, distant from the biased or incompetent failing of human hands. Yet a lot of the work goes in to the processes of standardising, decontextualizing and cleaning remains the product of human judgements. Data must be worked on to fit smoothly into a database. It must be standardized and 'cleaned' to make it comparable, removing local idiosyncrasies and extraneous detail. A lot of work goes into this process; several presentations at the National Pupil Database conference focused entirely on research methodologies for cleaning data to make it easier to operate on. Cleaned and standardised data can be made to 'talk' to data in other archives; information about pupils in the National Pupil Database can be connected to information held about them in HEFCE, HESA, and BIS archives, with ambitions expressed to link to data held about them by the Department of Health, Ministry of Justice and data collected through Universal Credit (should it ever see the light of day). These links allow for even greater level of analysis, tracking pupil data across more spheres of their lives.

FLOWING DATA

The processes of standardising, decontextualizing and cleaning data means that it can be more mobile and fluid; data can connect into other databases freed of its local, specific and contingent context. In *The Rise of Data in Education Systems*, Martin Lawn (2013) has described the work done by statisticians, NGOs, governments and educationalists to standardise educational data, in order to allow data to flow across national borders and create international databases informing the PISA study that systematically compare and evaluate diverse educational systems.

One of the ways that data had become so powerful is in the way it is presented as unassailable, legitimate and objective knowledge, and in the way that it is able to move across boundaries and articulate with other data sets unencumbered by messy local context. Yet a lot of work must be done to data to enable these characteristics, through selecting and excluding, standardizing, decontextualizing and cleaning. Data is translated through many processes, into an authoritative knowledge object, such as DfE's 'the bridge', or a RaiseOnline report, or a data dashboard, imbued with characteristics of objectivity and legitimacy. As with the quote from Jean-Francois Lyotard that Ozga opened her talk with suggests, 'to talk about knowledge is to talk about governing', data-driven educational knowledge lends its legitimacy and objectivity to processes of governing and regulating educational performance.

DATA OBJECTS

Educational data produces objects by which education can be known including data tables and graphs. It makes up education, learning and learners as things that can be known and governed in a particular way. But it also produces knowing subjects: the civil servants, policy makers, teachers and children who are invited to adopt a data-driven perspective from which to understand education and their own educational engagements. How do the subjectivities produced through these ways of knowing compete or cohere with other subjectivities? This could be seen as a further way in which data acquires its power in governing education: by inviting diverse groups of people to take up similar positions as knowing subjects in relation to understanding education.

Data is, in part, such a powerful form of governing knowledge because it is imbued with characteristics of objectivity and fluidity, seeming to be distant from messy, human, local idiosyncrasies. Yet it takes a lot of deliberate, political work to bestow these characteristics upon data sets. The following papers in this section examine the ways in which digital data are becoming integrated into educational policy work and embedded in the pedagogical complex of the classroom.

Gillespie, T. 2014. The relevance of algorithms. In T. Gillespie, P.J. Boczkowski & K.A. Foot (eds). *Media Technologies: Essays on communication, materiality and society*: 167-194. London: MIT Press.

Lawn, M. 2013. The rise of data in education. In M. Lawn (ed.) *The Rise of Data in Education Systems: Collection, visualization and use*: 7-25. Oxford: Symposium.

Digital education policy: big data, visualization and real-time analytics

Ben Williamson, University of Stirling

Digital technologies facilitate the generation, calculation and circulation of the data required to govern education. Seemingly objective statistical data are now being integrated into much educational policymaking, with schools and classrooms configured as ‘data platforms’ linked to vast global data collection programmes, and the ‘reality’ of education rearticulated in numerical practices that are enacted by new software developments, data companies, and data analysis instruments. The influence of digital technologies in such practices is complementing existing uses of data with methods of *digital education governance* (Williamson 2015a).

As such, education governance is now being enacted through new kinds of digital policy instruments (Lascoumes & le Gales 2007) that allow educational policy to be made operational. It is taking place in a context in which ‘datafication’—the objective quantification of all kinds of human behaviour to enable real-time tracking, monitoring and predictive analysis—has become a new paradigm in science and society. This is apparent with the growing interest in using big data for policymaking (and reflected in concerns over policy by algorithm), although education policy research has long been concerned with the social and political processes involved in the production of educational data, and in their socially and politically productive effects. By focusing on digital policy instruments, and the data infrastructures (Sellar 2014) in which they operate, I suggest that education policy is now being influenced to a significant degree by the design of the digital instruments through which educational data are collected, calculated, analysed, interpreted and visualized.

Looking at educational data in this way means going beyond the ‘policy numbers’ (Grek 2009) to acknowledge the infrastructural apparatus of technologies, human actors, institutions, methodologies, and social and political contexts that frame and shape their production and use. This requires us to consider the specific instruments, the human hands, eyes and minds, the companies and agencies, and the wider contexts that constitute such an infrastructure for educational data production and give it its productive power. To illustrate this, I have specifically looked at a number of examples of digital education governance at work.

GOVERNING THROUGH DATABASES

The first example is a fairly well-known database. The National Pupil Database was established in 2002 by the UK government under the supervision of Michael Barber, then head of the Prime Minister's Delivery Unit. The NPD features extensive datasets on the educational progress of children and young people from the early years through to higher education and contains detailed information on over 7 million pupils currently matched over a period of 12 years. The NPD captures information on their progress through the educational system as traces of data that can be standardized, joined together and aggregated with a national population dataset. The NPD pages on the gov.uk website enable interested parties to request access to the data, which is presented in Excel spreadsheet files as thousands upon thousands of rows of numbers that can be searched and analyzed in myriad ways, and used to generate graphical displays such as charts, tables, plots, and graphs.

Spreadsheets are a highly mundane form of database technology, but have also become, since their invention at the end of the 1970s, highly influential in the organization and presentation of data across commercial and governmental sectors. The spreadsheet enables a particular view of reality as enumerable and calculable by its in-built statistical formulas and models.

As a data source that is enacted through Excel spreadsheets, the NPD has become a major policy instrument of educational governance in the UK. For example, in 2015 the Education DataLab was launched as 'the UK's centre of excellence for quantitative research in education, providing independent, cutting-edge research to support those leading education policy and practice,' a task largely to be accomplished by conducting secondary analyses of the NPD in order to 'improve education policy by analysing large education datasets.' The Education DataLab is indicative of how education governance is being displaced to new centres of technical expertise, such as 'policy labs' (Williamson 2015b) that are able to translate massive data resources into actionable policy insights through advanced digital methods of data analysis and presentation.

A raft of other database-enabled digital policy instruments has followed the NPD. These include the Department for Education's school performance tables, Ofsted's School Data Dashboard, and the OECD's Education GPS application. These digital policy instruments and data dashboards all conform to a realist view that education can be presented as 'visualized facts' (Kitchin, Lauriault & McArdle 2015) rendering visible particular representations of the data whilst rendering invisible the underlying statistical and algorithmic techniques performed on it, by new kinds of technical data experts, to make it intelligible. The structure of the software interface of the policy instrument in this sense structures the data, and is intended to structure the user's interaction with that data as a means to facilitate social action.

CENTRES OF VISUALIZATION

A key techniques of digital education governance is data visualization. Visualization is now a major topic in social science studies of big data. It's important, again, to acknowledge that a data visualization is an expertly crafted accomplishment, not simply a visual reproduction of some underlying reality. Any visualization produced using software and digital data is an 'interfacial site' created through networks of human bodies at work with various kinds of software and hardware, facilitated by vast repositories of code and databases of fine-grained information, and possesses productive power to shape people's engagement and interaction with the world itself (Rose, Degen & Melhuish 2014).

A notable producer of data visualizations in education is the global educational publisher Pearson Education. Pearson's Learning Curve Data Bank combines 60 global datasets in order to 'enable researchers and policymakers to correlate education outcomes with wider social and economic outcomes.' The Learning Curve includes national performance data (sourced from, for example, the National Pupil Database) along with global data sources, in order to produce a 'Global Index' of nations that is ranked in terms of 'educational attainment' and 'cognitive skills'. The Learning Curve is highly relational, enabling the conjoining of multiple datasets, as well as scalable in that it can expand rapidly as new datasets become available.

It features a suite of dynamic and user-friendly mapping and time series tools that allow countries to be compared and evaluated both spatially and temporally. Countries' educational performance in terms of educational attainment and cognitive skills are represented on the site as semantically resonant 'heat maps.' It also permits the user to generate 'country profiles' that visually compare multiple 'education input indicators' (such as public educational expenditure, pupil:teacher ratio, educational 'life expectancy') with 'education output indicators' (PISA scores, graduation rates, labour market productivity), as well as 'socio-economic indicators' (such as GDP and crime statistics). The Learning Curve is a powerful technique of political visualization for envisioning the educational landscape, operationalizing the presentation and re-presentation of numbers for a variety of purposes, users and audiences. Michael Barber, the Chief Education Adviser to Pearson who launched the Learning Curve (and formerly the leading government adviser behind the National Pupil Database), has described it as allowing the public to 'connect those bits together' in a way that is more 'fun' and 'co-creative' than preformatted policy reports.

Even so, as an interactive and co-creative policy instrument, the Learning Curve is no neutral device. The choice of the instrumentation materializes the forms of analysis that are possible. Users' own analyses are in effect preformatted by the design of the interface as a form of user-generated comparative analysis, inciting users to compare country performance according to in-built tools constructed according to the assumptions and preferences of its technical and methodological producers. At the same time, the Learning Curve is structured according to the social media logic of 'prosumption' (Beer 2013) where users are seen not simply as consumers of data but as its producers too. The Learning Curve therefore reconfigures education governance as a form of 'play' and 'fun' that is consonant with the logics of social media participation and audience democracy in the popular domain, but at the same time preformats the possible results of such activities through the methodological preferences built-in to its interface. It incites the wider publics of education to see themselves as comparative analysts, and as participatory actors in the flow of comparative data, but subtly configures and delimits what users can do with the data and what can be said about them.

As such, the global 'centres of calculation' (Latour 1986) such as Pearson that manage the global flow of educational data are now increasingly becoming *centres of visualization* with the technologies and techniques to render dynamic educational data visualizations and to mobilize the interactivity of users to secure their consensus. Their visualizations act as surfaces on which millions of educational performances and measurements are inscribed and made visible for inspection, analysis, evaluation and comparison. As policy instruments, these visualizations act as 'interfacial sites' through which different views and visions of education are constantly being composed and compared, altered and modified, developed and designed in order to render certain kinds of meanings and arguments possible.

CENTRES OF ANTICIPATION

The emergence of big data in education means that data can now, increasingly, be collected and analysed in real-time and automatically. Pearson, for example, has established a Center for Digital Data, Analytics, and Adaptive Learning, that is intended to 'make sense of learning in the digital age,' which has produced a report on the impacts of big data on education. It envisions education systems where 'teaching and learning becomes digital' and 'data will be available not just from once-a-year tests, but also from the wide-ranging daily activities of individual students.' The report highlights the possibilities of data tracking, learner profiling, real-time feedback, individualization and personalization of the educational experience, and probabilistic predictions to optimize what students learn. Consonant with the wider potentials of data analytics, these approaches combine real-time data tracking of the individual with synchronous feedback and pedagogic recommendation.

Late in 2014 Pearson Education also published a report (co-authored by Michael Barber) calling for an 'assessment revolution' using 'intelligent software and a range of devices that facilitate unobtrusive classroom data collection in real time,' and to 'track learning and teaching at the individual student and lesson level every day in order to personalise and thus optimise learning.' In particular, the report promotes new forms of computer-based assessments, 'the application of data analytics and the adoption of new metrics to generate deeper insights into and richer information on learning and teaching,' as well as 'online intelligent learning systems,' and the use of data analytics and automated artificial intelligence systems to provide 'ongoing feedback to personalise instruction and improve learning and teaching.' Moreover, it argues for a revolution in education policy, shifting the focus from the governance of education through the institution of the school to 'the student as the focus of educational policy and concerted attention to personalising learning.' In the report, intelligent analytics are taken to be key policy instruments that concentrate policy on the real-time tracking of the individual rather than the planned and sequenced longitudinal measurement of the institution or system, and that ultimately possess the potential to determine classroom pedagogy itself. Pearson's own Center for Digital Data, Analytics, and Adaptive Learning is intended as the organizational setting for the development and advancement of such instruments.

Ultimately, the data analytics being developed by Pearson anticipate a new form of 'up-close' and 'future-tense' educational governance. Its analytics makes every individual learner into a *micro-centre of anticipation*—the focus for a constant and recursive accumulation, analysis and presentation of data, real-time feedback, probabilistic predictions, and future-tense prescriptions for pedagogic action. These analytics capacities complement existing large-scale database techniques of governance. But they also, to some extent, short-circuit those techniques. The deployment of big data practices in schools is intended to accelerate the temporalities of governing by numbers, making the collection of enumerable educational data, its processes of calculation, and its consequences into an automated, real-time and recursive process materialized and operationalized 'up close' from within the classroom and regulated 'at a distance' by new centres of calculation that house expertise in digital methods of automated data analytics.

MAKING DATA/MAKING UP PEOPLE

As big data developments increasingly join disparate datasets, it is feasible to speculate that the linking of global international assessment data with individualized learning analytics data by data companies such as Pearson would produce a vast and powerful data infrastructure in which student data could be collected continuously, analysed in real-time, and fed back not just into national profiles, global league tables and data dashboards, but directly into the pedagogic instrumentation of the classroom.

Are there implications for the students assessed these emerging data analytics techniques? Pearson claims that through pattern detection techniques it is able to identify evidence of learning processes that have so far not been theorized or modelled. Indeed, it claims that new forms of data and experience will create a theory gap between the increase in data-based results and the theory base to integrate them, and lead to the production of new generalizable models of learning processes and progressions. New theoretical understandings and models of learning might then be folded-back into the kind of pedagogic resources that Pearson itself produces and promotes to schools, particularly its personalized and adaptive learning applications. This could exert a kind of 'looping effect', as Ian Hacking (2007) has described it, where the process of making the data-derived model acts to shape and 'make up' the people that it purports to measure and represent. In other words, the big data-based assessment analytics of Pearson could become highly consequential to the formation of new models of learning, and thereby to 'making up' students as new 'kinds of people' who are understood in terms of the data and thus encouraged through the pedagogic complex of the adaptive classroom to relate to themselves and their own learning processes in novel ways.

DATA EXPERTS

The collection and digitization of massive educational datasets has a relatively long history, and data collection in education goes back well over a century. However, emerging digital data practices of data visualization and data analytics enabled by emerging public policy instruments—many based on functional principles derived from social media and big data science—are becoming powerful sources of contemporary digital educational governance. Digitally rendered as a vast surface of machine-readable data traces, education is increasingly amenable to being effortlessly and endlessly crawled, scraped and mined for insights. While this is not all new, then, it does indicate the emergence of a relatively distinctive style of digital education governance in which data-based policy instruments are employed to perform a constant audit of student actions in order to make them visible and thus amenable to pedagogic intervention. As a consequence, to examine educational governance increasingly requires exploration of the data infrastructures framing it, the digital policy instruments making it operational, and the experts that analyse, visualize, and prepare it for the interaction and interpretation of others.

The new experts of the virtual world of educational data are the technical, statistical, methodological and graphical experts—both human and non-human—that inscribe schools and the learners within them in enumerable, visible and anticipatory data, and address their audiences as particular kinds of users. New kinds of data careers have been made possible, both for leading policy advisers such as Michael Barber, but also for the educational data scientists, experts and algorithm designers required to do the data work, construct the database architectures, and design the analytics that now make education policy operational and productive. The techniques produced and promoted by such data experts appear to respatialize education governance to new centres of calculation beyond central government, and to accelerate the temporalities of digital data collection and use in education. They complement the massive, longitudinal datasets such as those held by national governments or by massive international organizations with more dynamic, automated, and recursive systems that are intended to sculpt learners' performances in real-time through the pedagogic instruments of the classroom.

Beer, D. 2013. *Popular Culture and New Media: The politics of circulation*. London: Palgrave Macmillan.

Rose, G., Degen, M. & Melhuish, C. 2014. Networks, interfaces and computer-generated images: learning from digital visualizations of urban regeneration projects. *Environment & Planning D: Society & Space*, 32: 386-403.

Lascoumes, P. & le Gales, P. 2007. Introduction: Understanding public policy through its instruments—from the nature of instruments to the sociology of public policy instrumentation. Governance 20, no. 1: 1-21.

Grek, S. 2009. Governing by numbers: the PISA 'effect' in Europe. *Journal of Education Policy* 24, no. 1: 23-37.

Hacking, I. 2007. Kinds of people: Moving targets. *Proceedings of the British Academy* 151: 285-318.

Sellar, S. 2014. Data infrastructure: a review of expanding accountability systems and large-scale assessments in education. *Discourse: Studies in the Cultural Politics of Education*, DOI: 10.1080/01596306.2014.931117

Kitchin, R., Lauriault, T. & McArdle, G. 2015. Knowing and governing cities through urban indicators, city benchmarking and real-time dashboards. *Regional Studies, Regional Science* 2, no. 1, 6-28.

Latour, B. 1986. Visualization and cognition: thinking with eyes and hands. Knowledge and Society 6: 1-40.

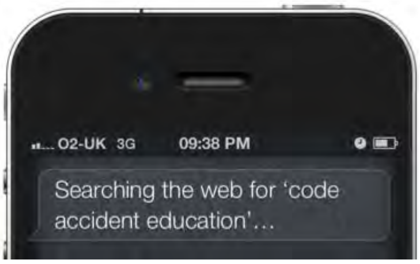
Williamson, B. 2015a. Digital education governance: data visualization, predictive analytics and 'real-time' policy instruments. *Journal of Education Policy*: http://dx.doi.org/10.1080/02680939.2015.1035758

Williamson, B. 2015b. Governing methods: policy innovation labs, design and data science in the digital governance of education. *Journal of Educational Administration & History* 47, no.3: 251-271. http://dx.doi.org/10.1080/00220620.2015.1038693

Learning analytics: white rabbits and silver bullets

Simon Buckingham Shum, University of Technology, Sydney

Learning Analytics sits at the intersection of *Computer Science (*drawing on sub-disciplines such as Data Mining, Information Retrieval, Information Visualization, Web Semantics) and *Education* (e.g. Educational Research, Measurement Science, Learning Sciences, Computer-Supported Collaborative Learning, e-Assessment*).* In my view it is an educational incarnation of *Human-Centred Informatics (*the effective design of human/digital information systems) and arguably *Computational Social Science* where social phenomena and computational modelling meet (elegantly introduced by Hannah Wallach in her recent Medium posting, and explored in relation to Complexity Science elsewhere). Extending several recent talks (e.g. EdMedia2014) this post introduces some of the big questions I see arising around the discourse of "educational big data."

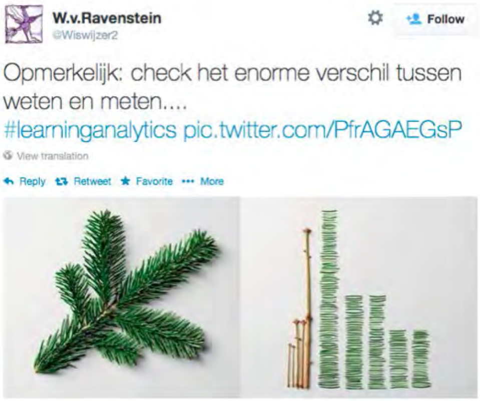


Searching the web for 'code accident education'...

I was speaking at the second Code Acts in Education seminar, and asked Siri to find me the website so I could check the agenda. Siri cunningly seized the moment to generate a pun on the very topic of my talk—the fears that many have around the impact should the automated algorithmic analysis of educational data take off.

This tweet sounded a cautionary note on behalf of those for whom the rhetoric around data, algorithms and analytics is (variously) pedagogically vacuous, blatant marketing, or driven by misguided accountability agendas (all of the above options may be checked!).

In the very process of trying to value certain learning qualities by tracking them, will we in fact distort or even destroy a living, organic system, through clumsy efforts to categorise and quantify?


W.v.Ravenstein
@Wiswijzer2

Opmerkelijk: check het enorme verschil tussen weten en meten.... #learninganalytics pic.twitter.com/PfrAGAEGsP

"Note: check the huge difference between knowing and measuring…"

> accounting tools … do not simply aid the measurement of economic activity, they shape the reality they measure (Du Gay & Pryke 2002)

Of course, we don't need computational analytics to take this worry seriously. Concerns about the damage to learning that can be wreaked by inappropriate assessment regimes long precedes Big Data. As chair of governors in an English primary school, working closely with the Head and senior team, I saw the pressure that teachers experience to maintain their visibility in the Department of Education's quantitative analytics of whole-school performance (RAISEonline). Quite apart from the pressure the teachers and 9–10 year olds experience, if schools drop too low in the stats for a couple of years, Headteachers typically *lose their job* as part of the standard 'improvement measures'. Not exactly a warm invitation to take risks and experiment creatively (but responsibly) with new ways to engage learners in challenging contexts. School principals need courage and support to break from the instrumental test-driven mindset, and it's good to see people like Anya Kamanetz making accessible to wider audiences the critical debate and evidence base that now testifies to the damage that such assessment regimes are causing.

Let me tell you about 10 year old "Joe". Joe's attendance is at 97%. When his curiosity is sparked his appetite to learn is insatiable. He's mentoring peers thanks to his good interpersonal skills. When he gets stuck, he's learnt not to panic, but has figured out ways to push through. Here's the class progress visualization of English *Key Stage 2* SATs. Joe is presumably one of those light blue *Level 4 good progress* avatars, or perhaps even a green *Level 5+*?

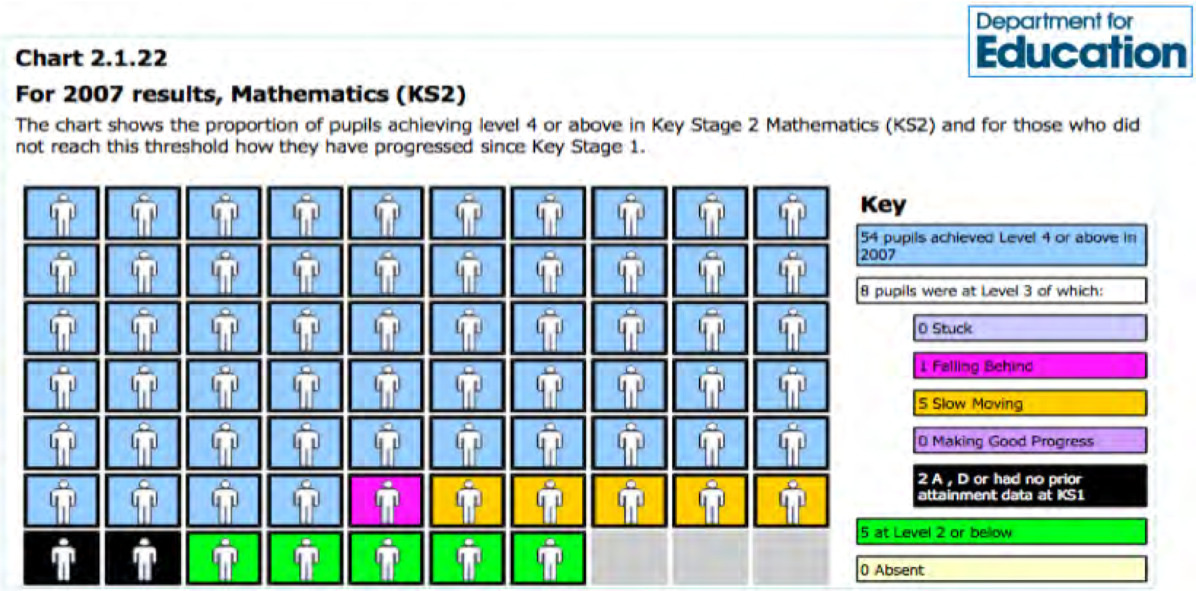
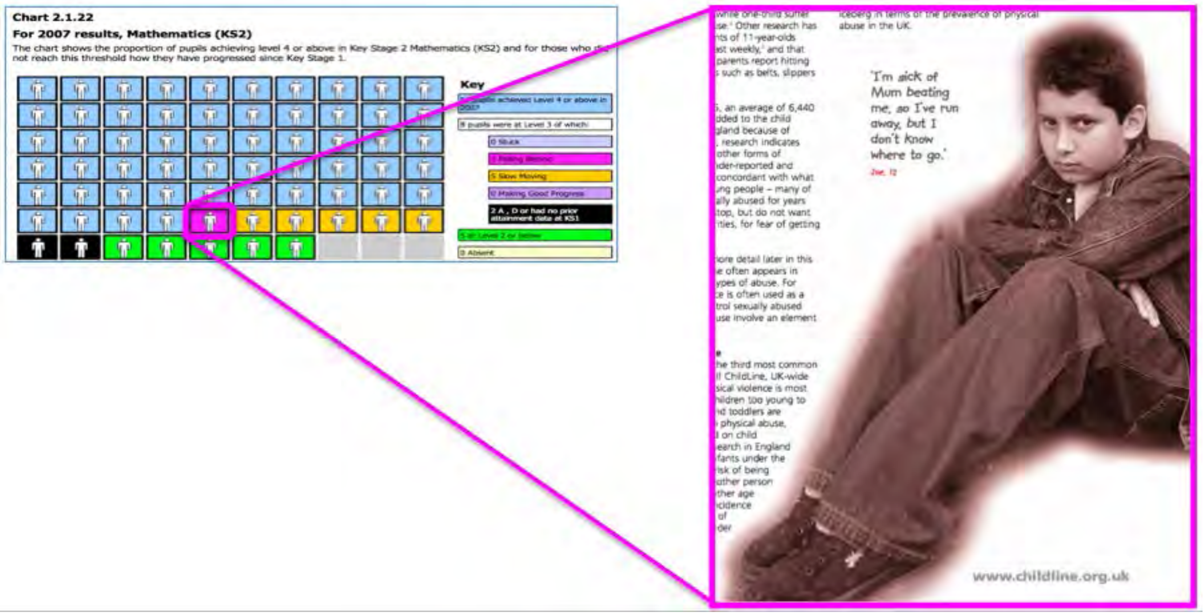


Chart 2.1.22

For 2007 results, Mathematics (KS2)

The chart shows the proportion of pupils achieving level 4 or above in Key Stage 2 Mathematics (KS2) and for those who did not reach this threshold how they have progressed since Key Stage 1.

Unfortunately not. While Joe is making faster progress than some of his more academic peers who are cruising and not stretching themselves, he is that purple *Falling Behind* blob.



When Joe started at school, he would often fall asleep in class. He could be aggressive to staff and peers with very little provocation. We'd often have to give him breakfast, or a clean shirt. His reading age was 6 months behind. But it's no wonder. He was often up several times a night feeding his baby sister while his mother was in a drunken sleep. He was picked up by the police recently on the streets at 3am, on his way to collect drugs for her.

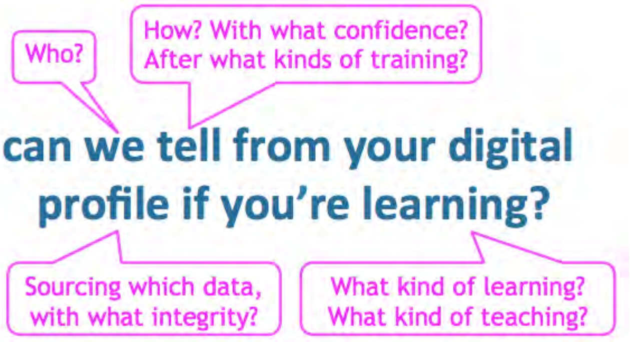
Through their commitment to Joe, the school was able to provide a stable weekday environment (picking up the pieces on Monday after a turbulent weekend), and cultivate a set of qualities that have transformed his attitude. He actually *likes* learning now. He's learnt that when something is new and difficult, *that's what learning feels like.* He's learnt that *asking good questions* is as important as knowing the right answer. He's learnt *what to do when you don't know what to do.* Not all pupils get this (even the 'high achievers'). Indeed, not all teachers get this. We'd still like to see Joe's reading and numeracy improve, but we're winning the strategic battle—he now wants to learn. "Joe" is a composite of some of the children we worked with, and his story will be painfully familiar to many school staff. If you're in tertiary education, substitute him with one of your students from a tough background but who is resilient enough to have made it into college, still has the right attitude given the opportunity and support, but is still fragile.

This vignette reminds us that the data points in a graph are tiny portholes onto a rich human world, and encapsulates some of the concerns that educators have about the misuse of blunt, blind analytics—proxy indicators that do not do justice to the complexity of real people, and the rich forms that learning take. In all sectors, however, progressive practitioners and researchers are emphasising the need to instill higher order competencies in learners to complement the conventional indices. However, these qualities are presumably even tougher to quantify in in a meaningful way. This is the highly charged social, organisational, political, educational context that Learning Analytics enters, and must do so with eyes wide open.

While we're used to declarations of new silver bullets on the airport bookshelves, it's extraordinary to see serious researchers get so myopic about a new technology that they follow suit, and even manage to get this past the editorial control of serious publishers — it does happen, although thankfully not yet in Learning Analytics.

However, as K-12, higher education institutions, and the associated government departments wake up to data, they are now seen by business intelligence (BI) companies as exciting new markets ready to hear how they can be transformed by Data and Analytics. The lure of the dashboard which shows at a glance how a student, department, institution, region or nation is doing, holds a deep appeal.

Indeed, to the extent that schools and universities are businesses, they can benefit from the sorts of optimisations that BI brings other enterprises. Moreover, there can be no complacency on the part of educational institutions about the risk of educational disruption by analytics-intensive businesses (who now go far beyond traditional BI vendors). Educational startups make mistakes at a furious rate, making them the object of scorn by some academics ("Haven't they read the literature?!"). They also learn at a furious rate, a design prototyping approach that puts educational institutions to shame.


Who?
How? With what confidence? After what kinds of training?
can we tell from your digital profile if you're learning?
Sourcing which data, with what integrity?
What kind of learning? What kind of teaching?

So no room for complacency as edtech startups rev their engines, but just as we interrogate the assumptions and biases underpinning computational models of other human phenomena (economics; epidemics; crime; migration…) we need to ask how an algorithmic mindset shapes our conception of learning: what assumptions about learning are made in the selection of data, the setting of thresholds, the selection of advice, recommendation or adaptation of curriculum? And who is supposed to make sense of the dazzling dashboards embedded in every e-learning product pitch? If *we* are to govern algorithms (and not vice-versa), who is equipped to ask questions in the right way, get the attention of those who can answer, and make sense of the responses?

FOLLOW THE WHITE RABBIT

It's here that our friend the rabbit has some provocations to offer. Some worry about *Learning Analytics as the Alice in Wonderland Rabbit*—an alluring, hard to pin down promise whose ROI is down a frustratingly long, dark hole (but hopefully the *next* upgrade will fix it…). Sales hype guilty of technological solutionism does not help of course, with so many organisations experiencing less than the promised delights of new information systems.

It's relatively early days, and financial analyses of the ROI on Learning Analytics are hard to come by at present. Our current educational paradigms and accounting systems make it easy to operationalise $$student in terms of course enrolments. Consequently, one form of analytics attracting a lot of interest is the use of predictive models for identifying 'at risk' students based on behavioural data:

if an intervention programme increases the student completion rate, that has direct monetary value. Models are validated on historical data of student dropouts, which gives statistical confidence that their deployment on live student data detects genuinely 'at risk' students.

Some of the most promising efficiency results also seem to be in the rate at which formally modelled curriculum and skills can be mastered through personalised adaptive tutors and educational games (there's a significant research literature on this now in the AI in Education community). It may be that adaptive platforms can release pressured curriculum time for other modes of learning that remain beyond the scope of the student modelling algorithms.

This brings us to *Learning Analytics as a Magic White Rabbit*—in which we ooh and aah when out of the black analytics hat pops a delightful surprise—*but nobody's quite sure where it came from…* Sounds like a non-starter as a sales pitch to a school or university, surely. As Candace Thille (Stanford) has provocatively put it, what educational institution would outsource key core competencies (learning design, assessment, feedback) to an ed-tech analytics platform, without knowing inside-out what was in the black box?

But just a minute. Everyday we put our trust in black boxes we don't understand. Society manages this through accredited professionals who can see inside the boxes and explain to mere mortals (up to a point) what's going on—in our car, or in a medical test, or why the bank sent us the wrong automated letter. Black box algorithms certainly seem to be accepted by some educators who have positive experiences with student 'at risk' early warning systems using predictive modelling, or with adaptive tutoring environments. Indeed, if they were shown the models, many wouldn't understand them without a maths and machine learning tutorial, and reading the background research publications. The pragmatic argument says that overstretched educators just want tools that work for them and for students: they care most that it is treating students in a manner which they judge to be appropriate—just as you do with your thermostat or phone.

So should we be content as long as *somebody* in the school or university is able to explain how the magic works? What if that person isn't actually in the institution, but in the company you've outsourced to? What if no-one's quite sure why it's behaving like it is—because *it's been learning autonomously for the last 3 months*—but it seems to be doing a great job? We are of course now in the realms of the oldest of AI dilemmas. But this is hardly Isaac Asimov science fiction. As we saw recently with Facebook's end of year review algorithm, it had all sorts of assumptions and values baked into it. So the question of how one validates an algorithm cannot escape values-laden issues: it's flawed if it merely replicates flawed human judgement.

We might look to the open source software movement for important principles and ways of working to ensure transparency around Open Learning Analytics—so that when detailed questions are asked by knowledgeable people, genuine answers can be provided, and when deep changes are needed they can be made. How companies who feel they have IP to protect will respond to questions around transparency remains to be seen. Perhaps they're banking that clients will enjoy the magic show, and not want to look behind the curtain. Or perhaps they will be caught out by an upsurge in data literacy which makes such a position untenable.

Questions must also be asked about whether part of a university's mission is to help students discern their calling, which may include discovering that they are on the wrong course; whether a focus on course completion comes at the risk of ignoring deep learning in favour of passing the tests; or whether a recommendation engine based on the historical habits of most students who've passed, threatens individual innovation and creativity.

Learning Analytics as a Shadow Rabbit http://th02.deviantart.net/fs71/PRE/f/2011/119/c/0/rabbit_shadow_by_plasticx76-d3f5qtd.jpg

Learners will leave an increasingly rich "digital shadow"—but as Plato reflected in his shadowy cave, this is but a pale imitation of vibrant reality. *Learning Analytics as a Shadow Rabbit* reminds us that a digital footprint necessarily reveals only a filtered record of the rich context in which a complex human took that step. The same step might be taken by different learners for different reasons: the algorithmic hope is that your preceding and subsequent paths reveal enough of your intent or state of mind in order to enable the software to do something useful, or alert a mentor to assess the situation and step in if required.

The Rabbit Talks Back (source unknown)

What we thought was just a shadow is now exercising agency… *When the Rabbit Talks Back,* we're reminded that the digital shadow is no longer a passive, causally determined rendering. It is not only getting higher and higher in resolution, updated in real time, but is exercising agency. Data visualizations—especially if they are endorsed by those in power—will shape how we see the world and how we act. Recommendation engines will exploit limited human attention to place certain resources at the top of the page.

In *The Matrix* Neo is instructed to *follow the white rabbit*. This rabbit (and a very odd under-the-counter pill) will reveal to him that what he took to be reality was in fact a digital mirage. This mirage was designed to stop people from asking the real questions: to treat the digital map as the territory. Treating digital renderings as actionable intelligence about reality is of course the ultimate promise, and risk, of all analytics. Bowker and Star help us sort things out. Their critical analysis of how we evolve information infrastructures is a sobering reminder of the compromises that are always made in the processes of abstraction that are required to construct standardised schemas for data processing (often for accountability purposes):

> Classification systems provide both a warrant and a tool for forgetting […] what to forget and how to forget it […] The argument comes down to asking not only what gets coded in but what gets coded out of a given scheme. (Bowker & Star 1999)

No coding scheme, no analytics. So what do we choose to forget about our learners, and how should we think about the role of computational analytics in learning?

The reflexiveness of humans is both a blessing and a curse. Just as digital data and computational analysis has transformed fields such as genetics, astronomy and high energy physics, educational researchers and practitioners have reason to be intrigued by the opportunity to analyse authentic user data at scale, at pace. The tricky point is that the BRCA2 gene, Red Dwarf stars and the Higgs bosun do not hold strong views on being computationally modelled, or who does what with the results. Learners' awareness that their shadow has a 24/7 audience (both human and increasingly machine) could easily lead them to distort their behaviour, or game the system. For an educator or researcher trying to get a robust measure of change, this might be considered a curse (depending on their epistemology). On the other hand, the blessing is that if high stakes are associated with demonstrating particular behaviours, evidenced by certain kinds log files, then no matter how sophisticated the user experience, we can be sure that smart people will invent ways to hack the system. We should be grateful for that, to the extent that we believe that it's dangerous to attach high stakes outcomes to (always limited) computational models of human behaviour.

This throws us back on the question of what learning analytics are seeking to model, and who gets to interpret and act on them. This in turn begs the question—*what kinds of learners are we trying to nurture?* Let's get clear where we want to go, and then we can talk about selecting and tuning engines, chassis and dashboards.

VUCA / LIMINAL SPACE

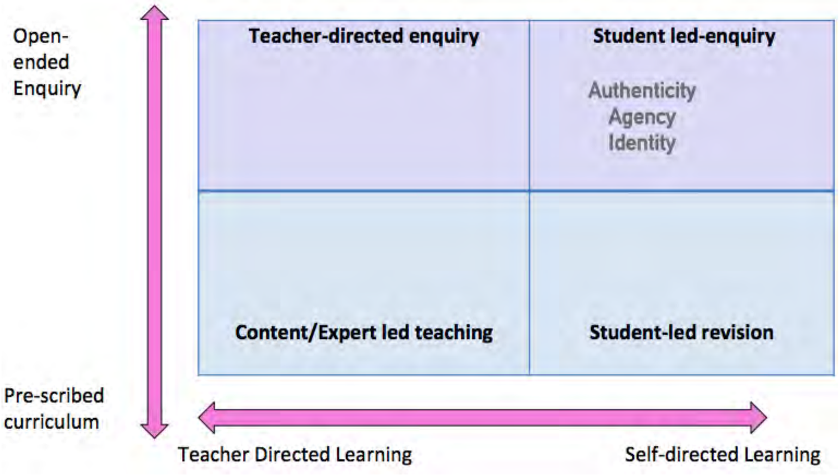
Volatile. Uncertain. Complex. Ambiguous. These are the conditions we now confront in all sectors, and the educational world is struggling to adapt, such is the systemic inertia to change. We should now be designing our educational systems to build learners who can not only survive, but thrive under unprecedented conditions of complexity and uncertainty. This goes deep, since we're talking about *liminal space* at many levels of the system. A favourite quote is from Richard Rohr, writing at the time about the post-9/11 disorientation in the US, but more broadly, about how we train ourselves emotionally and spiritually to tolerate, and navigate the unknown:

> Liminal Space… when you have left the tried and true but have not yet been able to replace it with anything else.
>
> …when you are between your old comfort zone and any possible new answer… If you are not trained in how to hold anxiety, how to live with ambiguity, how to entrust and wait, you will run… anything to flee this terrible cloud of unknowing. (Rohr 2002)
> (Limina is the Latin word for threshold, the space betwixt and between)

Liminal space is being developed as a concept for educational practice, for instance Johansson & Felten:

> So while we initially strive to make our students feel comfortable […] we then must help them balance their desire for security with the need to take risks and explore new ideas and possibilities.
>
> Rather than attempting to resolve the tension, a college should help students find their place on this precarious threshold, in the liminal space between the familiar and strange, the old and the new. (Johansson & Felten 2014)

The Knowledge–Agency Window from Ruth Deakin Crick helps us orient to this new landscape. Different kinds of analytics may help in different parts of this design space. But we can no longer sit primarily in the lower left cell, for which our dominant educational apparatus is tuned. Educators must learn to move fluidly around this space. My interest is in analytics for the top right quadrant, and in constructing prototype analytics suited to building these qualities.



Bowker, G. C. and Star, L. S. 1999. Sorting Things Out: Classification and Its Consequences. MIT Press, Cambridge, MA, pp. 277, 278, 281

Du Gay, P. and Pryke, M. 2002. Cultural Economy: Cultural Analysis and Commercial Life. Sage, London. pp. 12–13

Johansson, C. & Felten, P. 2014. Transforming Students: Fulfilling the Promise of Higher Education. Johns Hopkins University Press

Rohr, R. 2002. Grieving as Sacred Space. Sojo. January, February 2002 http://sojo.net/magazine/2002/01/grieving-sacred-space

Abstracting learning analytics

Jeremy Knox, University of Edinburgh

In the previous paper in this collection, Simon Buckingham-Shum raises important critical questions about Learning Analytics. Significantly, Learning Analytics, he suggests, necessarily embodies a 'world view', one that is already approximated by the algorithms employed to 'discover' it. However, despite this call to acknowledge the inevitable limitations of Learning Analytics, Buckingham-Shum's rationale is ultimately one of positive potential, in which the 'invisible' of education could be 'made visible'.

To understand these ocular metaphors in learning analytics, I want to consider an unusual source: *The Rules of Abstraction* on BBC Four, a documentary about the history and motivations of abstract art. But what can art—and an obscure, often confusing form of art at that—possibly tell us about Learning Analytics? Moreover, how can something so 'abstract' help us to understand the 'realities' of the operations of code in education?

THE VIEW FROM SOMEWHERE

To claim that Learning Analytics provides a 'view' of the world, albeit a partial one, is to frame it as a process which reveals something else; in this case educational activities, or 'learning' itself, depending on how one might wish to interpret the results. This is encapsulated rather well in the notion that Learning Analytics 'makes visible the invisible'. In other words, there is stuff going on in education that is not immediately perceptible to us, largely due to scale, distribution and duration, and Learning Analytics provides the means to 'see' this world. The idea of the 'visual' we are dealing with here is therefore a kind of access; we are able to see the unseen, and therefore gain admittance to something 'elsewhere'. Put rather more simply, we might say that Learning Analytics is 'visual' in the way that a window provides a view of something else—not the entire world of course, but a particular framing of it.

So, faced with such a seemingly straightforward idea of the visual capacity of Learning Analytics, what could the ambiguous field of abstract art possibly offer? One might easily dismiss the prospect, especially after watching Collings deliver typically mystifying statements such as, 'abstract art is not abstract'. However, I want to dwell precisely on this statement, because it has something genuinely useful to contribute to our thinking about Learning Analytics, and in helping us to understand what we mean by making something 'visible'.

Crucial here is the difference between 'abstract' art and 'realist' or 'naturalist' art, the latter being interested in capturing 'real life', or avoiding the stylisation of the artist. Crudely put, we can say that realist or naturist art attempts to depict things as they are. For example, such a painting might attempt to represent a panoramic view or still life scene as accurately as possible on the canvas, relative to what can be perceived by the viewer. The fidelity of the image to its object is of prime importance, and this is perhaps what remains as the most commonplace understanding of what 'art' is. This seems to reflect the way I described 'viewing' previously in relation to Learning Analytics. That is to say, Learning Analytics is considered valuable precisely because it is able to provide an accurate depiction of a 'real world' of education; albeit a real world that is invisible to our normal perception.

In contrast, abstract art is often considered non-representational. In other words, those famous images by artists such as Jackson Pollock or Bridget Riley are not necessarily meant to depict landscapes or domestic scenes; they are not intended to represent or correspond to something 'out there'. This notion of non-representation must therefore be saying something different about what an image is, and what it is to 'view' something. It is concerned with the practices and materials of painting itself. Such a reading would say that, rather than representing a landscape or an object elsewhere, it depicts the very thing that it is: a painting. What we see on the canvas is not a truthful portrayal of something external (a hillside, a seascape), but rather an account of the internal act of producing the painting (the movement of a human being, the effect of gravity on the substance of paint, the yield of a canvas).

Before I get carried away, let's return to that idea of abstract art not really being abstract, and to flip the distinction between 'abstract art' and 'realist art' on its head. What if we understood the beautifully detailed and accurate oil painting of some famous landscape not as 'realist', but as an abstraction from the real? That is to say, not as the real scenery 'over there', but rather as the very precise extraction of its particular hues and tones and their replication on a canvas 'over here'. Could we then understand the 'realist' painting as actually an 'abstraction'? By the very same measure, the so-called 'abstract art' of someone like Pollock can be understood differently. Rather than extracting and replicating the qualities of some external scene, a Pollock represents the very real processes of marking a canvas with paint. As such, it is not 'abstract' at all, but rather about the actual, tangible events that constitute the production of a painting: the relationships between the materials and how they interact; the influence of the artist's movement on the paint; the role of the environment in shaping how the paint reaches the canvas. Following Collings, we might say it is 'painting about painting'.

THE ABSTRACT ART OF ANALYTICS

So what does all that have to do with Learning Analytics? I think it is this: to critique Learning Analytics simply on the grounds that it makes certain worlds visible while hiding others remains within a representational logic that diverts attention from the contingent relations involved in the process of analysis itself. Let's unpick that a bit. In my very simplistic, and certainly inadequate, interpretation of abstract art, I have suggested that a concern for 'realism' actually involves processes of abstraction that attempt to recreate an absent scene, object or person on a present canvas. I think it is fairly uncontentious to say that Learning Analytics is fundamentally concerned with a similar idea of abstraction. Particularly as the 'worlds' Learning Analytics is attempting to depict are not discernible through our individual senses alone. For example, the broad range of 'grade point averages' would not be apparent to us as human beings without data collection and presentation methods.

However, the clever critique that I think the 'abstract artists' were making was that realism invalidated the actual painting itself; what was really important was the scene being depicted, and the painting was judged exclusively on its ability to represent this reality. Conversely, 'abstract' art might not be involved in abstraction at all, but rather the foregrounding of the very real, immanent practices of producing an image. But why should that concern Learning Analytics, which is fundamentally interested in providing a view, not an image; with 'making visible' the realities of educational activity so that positive intervention can take place? Well, this post is certainly not advocating that designers of Learning Analytics should suddenly surrender notions of representation and evoke their inner Jackson Pollock. Rather the point is to dwell on what the valuable lessons from abstract art might be: if we strive for Learning Analytics to be transparent, to depict with precise fidelity the real behaviours of our students, then we are working to hide the processes inherent to analysis itself.

In describing the demise of the Russian constructivist movement in abstract art, Collings alludes to the communist propaganda machine, and we are shown very realistic paintings of a smiling Stalin holding aloft a beaming blue-eyed child, similar to that shown in figure 1. This was the only 'reality' that Russian art was allowed to portray.

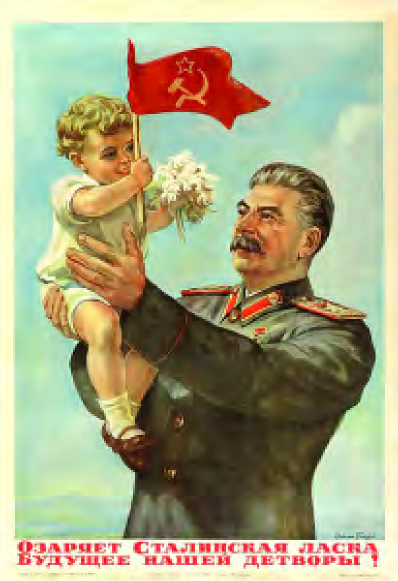


Figure 1: Russian propaganda poster showing Stalin holding a child https://farm6.staticflickr.com/5228/5637093714_933742e72a_z.jpg Creative Commons BY SA

My point in raising this example here is not just to suggest that what is produced by Learning Analytics may be controlled by wider political, economic, societal, or algorithmic influences, but also to divert attention away from the supposed reality behind the image. What I think the 'abstract artists' might have taught us is that the question is not whether Stalin really lifted the child (the reality 'behind' the image), but how and why the image itself was produced. That, I suggest, might tell us more about the state of Russia at that particular time. Indeed, we might even go as far as to say that to analyse the image in this way could tell us more than attempting to come up with a new, more accurate painting that shows us what Stalin was really doing at the time.

To drag the conversation back to Learning Analytics, my argument is that if we focus exclusively on whether the educational reality depicted by analysis is truthful or not, we seem to remain locked-in to the idea that a 'good' Learning

Analytics is a transparent one. I think we should focus less on the results of Learning Analytics, and whether they measure up to reality, and more on the processes that have gone into the analysis itself. Understanding these processes, I contend, is as crucially important in understanding the 'realities' of education in our current times.

REALIZING ANALYTICS

If that has all been rather abstract, then let's get real. The traffic light system is perhaps the simplest, and possibly the most widely used example of Learning Analytics currently employed by educational institutions. A notable example is 'Course Signals' created by Purdue University. Crude associations with abstract shapes and colours aside, such systems essentially produce an image, derived from data collected around student behaviour. By my own tentative definition, it is 'abstract', or rather 'abstracted', in the sense that it is a representation of something else. A green light signifies the student is doing fine, a red light that they are not. There is, of course, code acting to produce such images. For example, if the algorithm detects the presence of a variable that indicates 'semester 1 assignment has been handed in', it will produce the output required for a green light to appear in the student's VLE profile.

OK, now let's consider the two different ways of thinking about Learning Analytics here. If we remain within a representational framework, we are primarily concerned with the 'reality' being depicted by the traffic light: either the 'problem' of a student not handing in their assignment, or the absence of a problem resulting from a successful submission. What such a focus overlooks is the conditions through which this measurement of educational attainment actually comes about. This is not to argue that indicators of whether a student has completed an assignment or passed an assessment aren't useful, rather it is to suggest that an equally significant question might be how a traffic light has come to be the gauge by which a pedagogical intervention might take place. What are the broader societal and economic factors that produce an educational concern for retention over that of enjoyment, for example, and how is the image of the traffic light amplifying this concern? Why are the things that are programmed to produce a green light the exclusive measures of student success, and how does the prevalence of certain student data influence what kind of things are measured? Why has an algorithm been given the responsibility of saying 'you're doing OK' or 'you're not doing OK' over that of the teacher, or indeed the student? Why is education in need of such 'solutions', premised on what kind of 'problem'? These are the kinds of questions we can begin to ask when we see the traffic light as a traffic light, not as a transparent window to something else.

This is not an argument against traffic light systems, or Learning Analytics in general, but rather a call to expose and interrogate the assumptions already embedded in the code that produces them. While the drive to 'make visible the invisible' through Learning Analytics may indeed be useful, and techniques will undoubtedly aim for increased accuracy, such approaches work towards the transparency of their own processes. What may be a far more significant analysis of education in our times is not whether our measurements are accurate, but why we are fixated on the kinds of measurements we are making, and how this computational thinking is being shaped by the operations of the code that make Learning Analytics possible.

Data visualization as socio-digital practice

Sarah Doyle, University of Stirling

Data visualization is an interesting example of code at work, and it is one that provides insights into some of the interdisciplinary dimensions of coding. Recently, social science and humanities researchers have developed a strong interest in data visualization, as evidenced by interdisciplinary projects such as Picturing the Social and Seeing Data.

Part of the rationale for exploring this interdisciplinary angle lies in the recognition of coding as socio-digital practice. Instead of thinking about coding as a detached, neutral sequence of technological instructions, coding is understood as inherently imbued with sociality. Using the example of data visualization, we can see that coding entails working on the data to produce something new. As Rob Kitchin (2014: 106) notes:

> The visual register has long been used to summarise and describe datasets through statistical graphs and charts, diagrams, spatialisations, maps, and animations. These visual methods effectively reveal and communicate the structure, patterns and trends of variables and their interconnections.

Visual analytics methods imply a combination of humans and algorithms working symbiotically, with the algorithmic ordering of the visual data subtly working alongside the designer (and later, the user) in the construction of meaningful images, diagrams, graphs, and tables. The visual artefact emerges through a process of highlighting, leaving out and smoothing over. Visualizing data might be understood as a kind of translating, but like all translations, the work requires decisions about form and content. Who or what does the deciding about what to code in and what to code out?

COMPOSING THE VISUAL

In data visualization, socio-digital practices are especially apparent: finding the coded algorithm in the software is not the whole story. This is a point made by Jeremy Knox in his Code Acts in Education seminar presentation in relation to multimodality and digital literacy. Jeremy uses the example of word clouds to remind us that the visual artefact produced is not simply a mechanical movement of text from one format to another. On the contrary, the algorithms work on the text according to pre-selected criteria such as identifying which words are used most frequently. Producing algorithms that encode particular criteria is at least as much a social practice as it is a digital one.

Recent research by Maureen Michael helps to unravel some of these encoded practices of visualizing data. Informed by an art and design background, Maureen's eye is drawn to the composition of visuals, such as the judicious and deliberate use of line, shape and form. For those who know how, the purposeful selection of colour engages the audience in particular ways. Producing text in different fonts is a loaded process: traditional fonts invoke history and seriousness while newer ones portray simplicity. The positioning of graphics and the ways that captions are used (or not) can lead an audience in a particular direction or privilege a certain message. Web pages and documents disclose traces of these decisions.

These artistic practices encode specific assumptions and values. What happens when they are put to work in digital spaces? In what ways do they tangle with algorithms? Gillian Rose is investigating these and other issues in her work on digital technologies and architectural design. Her interest is in the ways that these technologies are specifying both the form and social use of buildings, as well as the work practices of the architects involved. In educational contexts, how do these socio-digital entanglements change data through visualizing processes? How does this visualization work to influence debates about which knowledge(s) are most worthwhile, which learning is most desirable or which institutions are most highly evaluated?

In her own research, Maureen is using visual methods to investigate art practices and often finds her data being created differently in the various software programmes that she uses. For example, her hundreds of observational photographs are viewed and manipulated in the digital camera, computer directory, Windows Live Photo Gallery, Adobe Photoshop, Word (and SmartArt Graphics), PowerPoint and Photobox. Each one affords different ways of viewing, grouping and manipulating the photographs. Each one has different aesthetics and tools to be used or sometimes subverted. Maureen's viewing of the data is not just her professional looking. Instead, the looking is actively constructed in each one of the different digital spaces. For Maureen, the worlds of art and design are unavoidably implicated in data visualization.

ENCODING THE VISUAL

Code acts: it participates. Code has effects. Coding encapsulates educational and political (and in data visualization, artistic) assumptions, preferences and priorities. In turn, these are entangled with the affordances of digital technologies and data servers as well as the availability and participation of digital literacies. While we might think we deploy code in ways of our choosing, code weaves throughout the fabric of our lives, changing and changed by the collaborations encountered. Code deploys us, too.

As this would suggest, the visualization of data is no neutral or objective accomplishment. The semiotics of visual analytics methods amplify the rhetorical function of data, allowing it to be employed to create arguments and generate explanations about the world, and to convince others that such representations depict the world as it really appears (Gitelman & Jackson 2013). Visual methods give the numbers meaning; they translate numerical measurements into curves and trends; and they make the data amenable to being inserted into presentations and arguments that might be used to produce conviction in others. In other words, data visualization gives numbers some pliability to be shaped and configured as powerful and persuasive presentations.

Thus, researchers need to examine the actors involved in producing visualizations, ask what data they are using, how those data have been formed, as well as 'what software is used in the analysis, what code or algorithms shape the data and the visualization,' in order to 'treat these visuals seriously as they come to envision the social world' (Beer 2013: 118-19). These are highly technical and methodological acts performed in concrete social circumstances. Any data visualization is ultimately made as it circulates around a network of offices and computer screens, as it is worked on by designers, visualizers, project managers, programmers and data analysts, and as it moves between software programmes and hardware devices. A visualization constitutes an 'interfacial site' created among networks of human bodies at work with software and hardware, 'through which data are constantly mobile, shifting and proliferating, moving between different actors and media, ported and patched, altered and designed, collaged and commented on' (Rose, Degen & Melhuish 2014: 401). The human eyes and hands, as well as software platforms and algorithms, involved in its display shape the interpretations a data visualization makes possible and the possible meanings that might be extracted from it.

Unraveling the specific nature of how code acts in particular settings means attending to the work that code does as well as what code is. It means, for example, identifying and examining the amplifications, deletions and obfuscations effected by code. This is interdisciplinary work. We can find source codes and crack algorithms, and we can bring these into dialogue with critiques of power relations and social inequities. We can ask about the mathematical and computational details of code, and we can also ask about the embedded artistic, sociological and ethical traditions.

Kitchin, R. 2014. *The Data Revolution: Big data, open data, data infrastructures and their consequences*. London: Sage.

Beer, D. 2013. *Popular Culture and New Media: The politics of circulation*. London: Palgrave Macmillan.

Gitelman, L. & Jackson, V. 2013. Introduction. In L. Gitelman (ed.) *'Raw Data' is an Oxymoron*: 1-14. London: MIT Press.

Rose, G., Degen, M. & Melhuish, C. 2014. Networks, interfaces and computer-generated images: learning from digital visualizations of urban regeneration projects. *Environment & Planning D: Society & Space*, 32: 386-403.

Computing brains in the cognitive classroom

Ben Williamson, University of Stirling

What is the future of schools in 'smart cities'? Smart cities are urban environments structured and supported 'line by line, algorithm by algorithm, program by program,' 'by code using data as fuel' as Nigel Thrift (2014: 10) claims. To some degree, smart cities are even cities that 'think of us' (Crang and Graham 2007: 792), with some 'sentience' to learn and adapt. Many smart city programmes now feature themes such as 'smart education' and 'smart learning.' I focus here on IBM's 'Smarter Education' programme, part of its wider 'Smarter Cities' agenda, emphasizing its 'learning analytics' and 'cognitive computing' developments. The promise of educational data analytics and cognitive computing in the classroom is to transform schools into 'brainy spaces' where the environment itself possesses brain-like functions of learning.

Conceptually, IBM's learning analytics and cognitive classrooms developments can usefully be understood as 'digital data practices.' Following Evelyn Ruppert and colleagues (2015) digital data can be conceived as being generated through social and technical practices: data have 'social lives' that bring them into being. But the generation of these data are also generative of particular effects and social implications—data are consequential to 'what is known,' and can influence decision-making and other activities. My argument is that IBM's ambitions for schools constitute an imaginary of a 'cognitive infrastructure,' where the data practices associated with data analytics and cognitive computing are being translated into the pedagogic space of the school.

LEARNING ANALYTICS

Learning analytics software is designed to enable individual students to be tracked through their digital data traces in real time in order to provide automated predictions of future progress. The emerging field of learning analytics is made up of expertise in education, statistics, computer science, sociology, machine learning, AI, organizational theory, learning sciences, psychology, and neuroscience—new kinds of 'educational data scientists' as Roy Pea (2014) terms them. The social life of learning analytics as a data practice is embedded in this multidisciplinary space.

IBM is a major developer of learning analytics techniques and applications. The IBM Smarter Education programme is based on the assumption that 'with the increasing availability of technology in the instructional process, educational institutions now collect, in real time, data about what their students learn and how they progress … using big data and analytics,' so that 'by turning masses of data into useful intelligence, educational institutions can create smarter schools for now and for the future.' To roll out its analytics applications in practice, IBM has established its own high school chain in the US, P-TECH. Its aim is 'to build for schools what its operations center is for cities: a single system for collecting, aggregating and analyzing data from students and teachers alike, then writing algorithms to prescribe how to cope'; all to be accomplished by building a 'software 'infrastructure layer' for schools ... to manage students' digital textbooks & analyze their performance' (Linday 2013).

It's important to note that learning analytics is underpinned by techniques of modelling. Learner modelling, cognitive modelling, behaviour modelling, probability modelling, and modelling the knowledge structure of a discipline are all elements in any learning analytics platform. As George Siemens (2013: 1386) explains, with advances in analytics techniques 'new data-based discoveries are made and insight is gained into learner behavior … through models and algorithms.' Machine learning techniques can then be mobilized to make predictions based on these models—to construct predictive models. As Adrian Mackenzie (2013: 399) notes, 'programmers construct models that predict what people will do' through 'transforming data on events, actions, behaviours, beliefs and desires' into probabilistic predictions of the future that then can be used to decide on preventative or even pre-emptive action to be taken in the present.

What we are talking about, then, is not algorithms per se, but the interaction of the algorithm with the underlying model. As Tarleton Gillespie (2014) notes, 'the 'algorithm' comes after the generation of a 'model,' i.e. the formalization of the problem and the goal in computational terms.' Such models are always constructed according to the embedded values and assumptions of their designers. Therefore, Gillespie argues, 'the embedded values that make a sociological difference are probably more about the problem being solved, the way it has been modeled, the goal chosen, and the way that goal has been operationalized.' As Daniel Neyland (2014: 11) asks, how do algorithm designers 'build a world out there into a world in here, in the algorithmic machine,' in order 'to model human action'?

Learning analytics is a typical example of modelling human action. These models are the product of complex sociotechnical practices and are embedded in the methodological commitments, assumptions, values and styles of thinking of their designers. In approaching learning analytics critically as a data practice, then, we need to be alert to the 'social life' informing the production of its underlying models and to the processes involved in training the algorithms that will interact with those models to generate 'insights' into the learner and to make learning processes known.

COGNITIVE CLASSROOMS

A new development in IBM's learning analytics portfolio is 'cognitive-based learning systems' informed by neuroscience and technical developments in brain-based computing. Like learning analytics, cognitive computing is another instantiation of a social and technical 'data practice,' but woven through with models and understandings from the burgeoning field of educational neuroscience (Busso & Pollack 2015). It's built on the idea that the architectures and functions of the brain can now be modelled computationally, extending beyond the predictive capacity of machine learning applications to function more like human brains than programmed analytics software. In the field of learning analytics, as George Siemens (2013: 1383-84) again outlines, cognitive modelling is concerned with developing systems that possess a 'computational model capable of solving the problems that are given to students in the ways students are expected to solve the problems,' and since 'cognitive processes can be modeled, software (tutors) can be developed to support learners in the learning process.'

According to IBM, applications derived from these modelling techniques can then be embedded into schools as a cerebral augmentation to the cognitive capacities of the learner. But first we need to understand something of the social life of IBM's cognitive computing developments to appreciate IBM's imaginary of the smarter school. Cognitive computing at IBM—again linked to its 'Smarter Cities' initiative—is a category of technologies intended to create systems that learn, reason and help magnify human expertise and cognition. The language here is of 'neuromorphic hardware,' 'algorithms that learn,' 'neural network learning algorithms,' and 'brain-inspired algorithms.'

IBM claims that 'cognitive computing aims to emulate the human brain's abilities for perception, action and cognition,' utilizing its 'neurosynaptic chip' to 'emulate the neurons and synapses in the human brain.' In a 2014 article in *Science* (and reported in a variety of non-specialist publications) IBM engineers working across neuroscience, algorithm design and supercomputing claimed they had created a 'one million neuron brain-inspired processor,' 'capable of 46 billion synaptic operations per second … literally a synaptic supercomputer in your palm'; it can also be tiled together 'to create vast neuromorphic systems' of several millions of neurons and billions of synapses—sometimes referred to in IBM material as 'computing brains,' 'systems that can perceive, think and act,' or even a 'brain-in-a-box.'

Significantly, it also claims to be combining its neurosynaptic developments with neuroplasticity. Plasticity is the understanding that the brain's neural architecture is itself pliable, flexible, and constantly adapting to environmental input (Rose and Abi-Rached 2013). IBM's cognitive computers are, therefore:

> designed to learn dynamically through experiences, find correlations, create hypotheses and remember—and learn from—the outcomes, emulating the human brain's synaptic and structural plasticity (or the brain's ability to re-wire itself over time as it learns and responds to experiences and interactions with its environment).

In sum, IBM's engineers are attempting to model the neural plasticity of the 'learning brain' in silicon. These technical developments are now the subject of intense future imagining and R&D within IBM itself.

Education is one area in which IBM is seeking to translate and apply its brain-inspired computing. In an imaginary of the classroom in five years, it grandly claims that the IBM 'smarter classroom' is a 'classroom that will learn you' through 'cognitive-based learning systems.' As the IBM promotional website claims:

> The rapid digitization of educational institutions will allow unprecedented instrumentation of the learning process. Cognitive computing, or learning technologies, will help us calculate everything we can about how each student learns and thrives, then create flexibility in the system to continually adapt and fine-tune what we deliver to that student.

These claims are reinforced and reiterated in a variety of IBM think pieces, glossy interactive multimedia presentations, and infographics available on the company website. The cognitive classroom promises personalization of the learning experience, real-time feedback on learner performance, adaptive learning software that can learn from and adapt to the learner, and intelligent software tutors that can automate remedial intervention or even prescribe appropriate curricular content. IBM's Cognitive Computing for Education Transformation program director Satya Nitta has described such systems as intelligent, interactive systems:

> Recent advances in artificial intelligence, represented by systems such as IBM's Watson, along with highly interactive technologies such as touch, natural language, gesture and speech recognition are significant milestones in computing. At the same time, advances in neuroscience promise to bring us closer to a deeper understanding of cognitive processes such as learning. At the intersection of cognitive neuroscience and cognitive computing lies the extraordinary opportunity to transform learning for all mankind.

The cognitive classroom detailed here is just one imaginary instantiation of what IBM terms an 'ecosystem of cognitive environments inhabited by a society of specialized software agents called cogs.' Based on the assumption that 'cognition does not occur solely (or even mostly) within an individual human mind, but rather is distributed across people, artifacts and environments,'

> Cogs are designed to follow and interact with humans and other cogs across a variety of everyday environments … [and] learn and leverage sophisticated models of human characteristics, preferences and biases so they can communicate naturally.

This is ultimately what IBM envisages as a 'cognitive environment': an 'infrastructure ... enabling "human-computer collaboration at the speed of thought."'

Adrian Mackenzie (2015) has argued that advances in cognitive computing in places like IBM are based around 'the ideal of something like pattern recognition or indeed conscious awareness' and 'abound in references to cognition, meaning, perception, sense data, hearing, speaking, seeing, remembering, deciding.' He terms such technologies 'cognitive infrastructures' and says:

> the increasing 'mindfulness' of the infrastructures under construction at IBM, Google and the like predicate a certain re-concatenation of the world, no longer in the mobile train of experience of people … but instead in the relations mindfully discerned in streams of data.

The brain-like architectures that can learn exemplified by IBM's cognitive computing developments are clear examples of such cognitive infrastructures—architectures that use the brain as a prototype, and for R&D practices that take neuroscience inspiration for new algorithm designs. IBM's imaginary of the cognitive classroom can be conceived, then, as a mindful infrastructure in which data-based brain modelling and software cogs are to be applied as brain-targeted 'neuropedagogies' to extend and enhance learners' cognition. The smarter school is imagined as a brainy space, inhabited by software 'cogs,' that is located in the cognitive infrastructure of the increasingly sentient, smart city.

MODELLING AND MORPHING THE BRAIN

What might be the consequences of cognitive computing for learners themselves? N. Katherine Hayles (2013: 10) argues that recent discoveries around neural plasticity support the view that humans develop through 'epigenetic changes—changes initiated and transmitted through the environment rather than through the genetic code.' This presupposes that any outside stimulus detected by the body has the potential to cause epigenetic modifications, so as Hayles (2013: 11) claims:

> As digital media … embedded in the environment, become more pervasive, they push us in the direction of faster communication, more intense and varied information streams, more integration of humans and intelligent machines…. These environmental changes have significant neurological consequences.

Hayles (2013: 123) refers to a 'technogenetic spiral' where digital media 'changes brain morphology' and human cognitive capacities, whilst also recognizing the dynamic connections between changes in human neurobiology, technical innovations and sociotechnical imaginaries. So we can see IBM's cognitive classroom as a sociotechnical imaginary of a dynamic, computationally cognitive environment: a neuropedagogic technology with the capacity to interact technogenetically with the brain of the learner, whilst also learning and adapting through its own plasticity to the learners it encounters.

It is in this sense that we might see the student to be developing an 'algorithmic skin' (Williamson 2014), or even an 'algorithmic self' (Pasquale 2015), conceptualised as a neurobiological core which is increasingly 'known' and then enveloped by a 'networked cognitive system' that applies 'artificial skin' to those bodies with which it interacts, as David Beer (2013: 131) suggests. The student is to be 'mindfully' known through streams of data, becoming a kind of data object to then be acted upon, modified and optimized by cognitive technologies with the promised capacity to extend human cognition.

In conclusion, the application of 'brain-inspired thinking' in plans for the cognitive classroom reflect what Jessica Pykett (2013) has termed the 'application and popularization of neuro knowledges,' those disciplines that attempt to 'model the brain.' The imaginary of the IBM smarter classroom as a cognitive environment represent the enmeshing of neuro knowledges with the data practices associated with technical expertise in data analytics and cognitive computing. These social and technical practices assume it is now possible to model and know the 'learning brain' through its data, and put it into interaction with the 'learning algorithms' of machine learning and cognitive computing technologies. Such learning brain-like processes can then be modelled computationally, and written on silicon, from where these models will then interact with users in a more cognitively-capable environment.

Beer, D. 2013. *Popular Culture and New Media: The politics of circulation*. London: Palgrave Macmillan.

Busso D, Pollack C. 2015. No brain left behind: Consequences of neuroscience discourse for education. *Learning, Media, and Technology* 40, no. 2:168-186.

Crang, M. & Graham, S. 2007. Sentient cities: ambient intelligence and the politics of urban space. *Information, Communication & Society* 10, no. 6: 789-817.

Gillespie, T. 2014b. Algorithm. *Culture Digitally*: http://culturedigitally.org/2014/06/algorithm-draft-digitalkeyword

Hayles, N.K. 2013. *How We Think: Digital media and contemporary technogenesis*. University of Chicago Press.

Linday, G. 2013. IBM's Department for Education: The company that brought you smarter cities moves into schools. *NextCity*: http://nextcity.org/features/view/ibms-department-of-education

Mackenzie, A. 2013. Programming subjects in the regime of anticipation: Software studies and subjectivity. *Subjectivity* 6, no. 4: 391-405.

Mackenzie, A. 2015. Demis Hassabis: mindful infrastructures and re-concatenated worlds. http://rian39.github.io/infrastructure/2015/02/06/Demis-Hassabis-Mindful-Infrastructures.html

Neyland, D. 2014. On organizing algorithms. *Theory, Culture & Society*: http://dx.doi.org/10.1177/0263276414530477

Pasquale, F. 2015. The algorithmic self. *The Hedgehog Review* 17, no. 1: http://www.iasc-culture.org/THR/THR_article_2015_Spring_Pasquale.php

Pea, R. 2014. Building the Field of Learning Analytics for Personalized Learning at Scale. Stanford University: https://ed.stanford.edu/sites/default/files/law_report_complete_09-02-2014.pdf

Pykett, J. 2013. Neurocapitalism and the new neuros: using neuroeconomics, behavioural economics and picoeconomics for public policy. *Journal of Economic Geography* 13: 845–869

Rose, N. & Abi-Rached, J. 2013. *Neuro: The new brain sciences and the management of the mind*. Oxford: Princeton University Press.

Ruppert, E. et al. 2015. Socializing big data: from concept to practice. CRESC Working Paper no. 138: http://www.cresc.ac.uk/medialibrary/workingpapers/wp138.pdf

Siemens, G. 2013. Learning Analytics: The emergence of a discipline. *American Behavioral Scientist* 57, no.10: 1380-1400.

Thrift, N. 2014. The sentient city and what it may portend. *Big Data & Society* 1, no. 1: http://dx.doi.org/10.1177/2053951714532241

Williamson, B. 2015. Algorithmic skin: Health tracking technologies, personal analytics and the biopedagogies of digitized health and physical education. *Sport, Education & Society*,20, no. 1: 133-151.

encoded practices, knowing & teaching

Software, code, data analytics and algorithms are reshaping the ways in which professionals act, how researchers construct knowledge, and how educators approach the task of teaching. What new encoded practices, modes of professionalism, knowledge and teaching are now emerging?

Code, technologies and professional learning and work

Richard Edwards, University of Stirling

Computer technologies and computer-mediated information and communication are increasingly parts of professional practice and learning. They are part of the programmes of preparation for professionals and the assemblages of professional work and learning once in situ. These technologies are often taken simply to be tools to be used to enhance professional practice. Technology has always been important to work, but it is arguable that the speed and scope of innovation in computer technologies is at a faster pace and more pervasive than we have seen previously.

While such technologies take many forms—for example, sensing, communicating, analysing, doing—their ubiquity is arguably matched by their increasing invisibility. The technologies are often black boxed and naturalised. They are there and they do the work we need them to do… until of course they breakdown or do something that we do not expect. The black boxing and invisibility is almost deliberate, designed into the technology in the sense that few understand how things work or can repair them. To replace a silicon chip in an iPad is not something most of us would try. Newer cars are diagnosed in garages by being plugged into computers because the on-board computer has all the necessary data.

How then do we research, frame and theorise the entanglements of technology in professional work and the learning associated with this? These are not new questions. Examining the effects of computer technology on work organisations and practices has been a concern for many years, with issues about de-skilling and re-skilling, and the resultant struggles over shifting statuses and rewards. There is also interest in the changing relationships with users or clients of professional services, as for instance, with the self-monitoring and remote sensing of data and conditions by patients enabling less face-to-face contact with doctors. The capacity to quantify the self through digital monitoring is growing rapidly.

There is also increasing interest in what can be done through the data gathered, scraped and mined from computer technologies, both in relation to enhancing services through big data analysis and in terms of the monitoring and accountability of professionals to ensure effective and efficient services. Mobile technologies enable communication with professionals in the field, but also the monitoring and tracking of their movements. And there is increasing interest in the ways in which such data is used in policy and managing to enable forms of governing by data.

These are important areas for research. However, what has yet to be fully explored is the ways in which the software that make these technologies operate assume and produce certain affects in their development and uptakes. In other words, in examining the role of computer technologies in professional work and learning, we need to denaturalise them and open the black boxes to examine the work of code, algorithms and standards in the entanglements of their enactments.

Technology is part of the materiality of professional practice. Yet computer technology also has an immateriality about it, as what lies behind the screen is invisible and unknown to many who use such artefacts, even as it represents increasingly sophisticated visualisations and environments with and in which we participate. However, in recent years, a range of cross-disciplinary studies have started to point to the work of code, algorithms and standards in selecting and shaping the information and forms of knowledge made available to professionals in their studies and work. Code is technical, social, material and symbolic. Concerns have been raised about the ways in which data is selected and shaped by software in ways which are not always apparent to those using the technologies. For some this work is hidden, for others it is inscrutable. What is clear is that code is influentially entangled in many aspects of professional work and learning and requires closer research. The papers in this section begin to attend to this challenge.

Professional responsibility in a future of data analytics

Tara Fenwick, University of Stirling

> Remotoscope™ is one of dozens of medical apps becoming available for the iPhone that can be purchased by any consumer. This one claims to provide at-home diagnosis of ear infections. Such diagnostic technologies have prompted Khosia (2012) among others to claim that in health care, ‘up to 80% of diagnosis in future will be conducted through computers’.
>
> Motorola’s Real-Time Crime Centre Starter Kit links wide ranging data from sources like sensors, alarms, multiple video systems and computer aided dispatch with software analytics to support ‘predictive policing’. This technology is touted to allow agencies ‘to implement predictive policing tactics and leverage existing technology to provide relevant and timely intelligence to improve closure rates, help stop a crime in action and proactively identify potential incidents before they occur’ (Cipriano 2014).

New ‘smart’ technologies are proliferating across services from health and education to urban planning and policing. They collect data through continuous sensing. They access massive data sets such as administrative and health records, and link these with all sorts of new unstructured data combed in real time from human activity. They work through algorithms to analyse this data on a huge scale for patterns, then calculate these patterns to diagnose problems and suggest solutions.

Increasingly these technologies are being used in professional work to predict and plan, to recommend, and even to make decisions. But where is professional judgment and responsibility? How is accountability to be delineated with so many technological actors integrated into professional service?

These are important questions for professional futures. Professionals’ work in many domains is being irrevocably transformed by these interlocking forces of ‘big data’ and the software codes that are collecting, comparing, and calculating it. Data dredging techniques, data visualization, and analytics are now influencing what many consider to be important knowledge—in everyday life, in policy, and in professional practice. These are already reconfiguring professional practices, and conjuring questions about what are the most fruitful contributions of human experts to the sociotechnical assemblages that professional work is becoming.

A critical sensibility is increasingly important in understanding these challenges. So is much greater attention to the apparatuses of smart technologies and their interconnections with human thinking and activity. However we miss an opportunity if we focus only on the invasive potential of data analytics in decision-making or its commercial colonising of human affairs. A better approach might be to understand the opportunities as well as the limitations of big data and software code so that we can work with it to construct useful future directions. Hopefully we can remain mindful of the lurking dystopias while finding ways to interlink the affordances of smart technologies and web science with the continuing problems that professional services struggle to address. Here are a few more examples for consideration:

In medicine, new consumer diagnostic mobile technologies such as *Remotascope* are moving rapidly from prototype to market. The argument is that such technology-delivered, data-driven processes are more reliable and consistent than services performed by human professionals. Furthermore, this 'code'-service is presented as responsive and timely, convenient, and of course, personal—echoing that mantra of current technology hype that consumers have learned to demand. Doctors, of course, might argue that such 'automatic' diagnoses cannot respond to the nuances and ambiguities of patient narratives, and the multitude of exceptional presentations of particular conditions. Careful probing and active listening by physicians often opens a new direction of analysis, and prevents a false diagnosis based on what can seem to be a clear case. Yet given the pressures of enormous patient loads and reduced availability of general physicians and emergency room beds, technological diagnosis and prescription for selected common conditions—working alongside clinicians—can provide important relief.

In law, Susskind (2013) shows how professional legal service has been proliferating into many specialised technology driven processes, with a corresponding rise of technology driven entrepreneurs: legal technologists, legal knowledge engineers, project managers and risk managers. Online personal legal services such as 'Cube-Legal' or 'Rocket Lawyer' have sprung up, using software that analyses problems and presents solutions. The claim is that this makes legal service more convenient and affordable. Susskind predicts a radical reconfiguration of the profession of law, delivered through diverse internet based global legal businesses, online document production, virtual courts and online dispute resolution.

In human resource management, Google has developed what it calls 'people analytics' using big data and algorithm-based decisions (Sullivan 2013). One algorithm predicts which employees are likely to become a 'retention problem', alerting management so that pre-emptive action can be taken. 'Forward-looking' predictive models are developed to forecast and act upon other 'people management' problems and opportunities before they arise. A hiring algorithm is used to predict which employees are most likely to succeed after recruitment, both to shorten the total interview time and to ensure that the selection panels do not 'miss' top talent: this is 'scientific' recruitment. There is even an algorithm to solve 'diversity problems', analysing the root causes of weak diversity and presenting actions.

In the finance professions and services, algorithmic software has already eclipsed most transactions, speeding up the process to the point where computers trade shares in milliseconds. In his description of what is called 'high frequency trading', Lewis (2013) shows that the principal actors in well over

90% of stock trades are computers. Software codes can 'sniff' a sale before it is completed, even detecting others' algorithms in the market, then buy the share to make a profit. Nanoseconds are crucial to making profit. Humans not only become extraneous to most trading decisions, but often are unable to see or follow the process, let alone monitor it effectively. No one outside the process can gain access.

In the built environment professions (such as architects, engineers and planners), professional roles are changing dramatically as large integrated data systems are used increasingly to design, construct and maintain buildings (Jaradat et al 2013). The client is becoming increasingly 'professionalised', new conflicts are appearing across professional groups, and new kinds of professional accountabilities are emerging. For example, workflow approvals are often delegated to digital mechanisms, while professionals running the projects may bypass the fuss and unwieldy structures of uploading the required documents and continue to rely on phone calls and emails to negotiate fast changing details between engineers, contractors, and architects. New specialists dealing with document control and integration are becoming part of building design and delivery processes, who may exercise different standards of judgment in assessing work quality than the design professionals. The proliferation of different design professionals—architects, servicing engineers, and so forth—all mediated by data and new integrated digital systems, create new issues of standardisation and transfer of the digital data. Multiple users interacting with the same data at once add immense value as well as increased errors and misinterpretations.

These examples all show that algorithms and big data could offer real benefits depending on the professional purposes they are made to serve—not just creating efficiencies, but also serving clients or improving care. Practice can draw from abundant sources of diverse real-time, fine-grained, formerly difficult-to-access data assembled with state of the art new technologies that capture, manipulate, and curate this information in ever-more-accessible ways. Data analytics can calculate and generate useful predictive patterns from masses of data with more speed, quality and reliability than humans. At the same time, these examples raise troubling issues like these:

1. *Changing everyday practice and responsibilities in ways that may not be fully recognised.* Some work is becoming tied to, and even dictated by, databases and their categories. Notions of place are being transformed. Multitudes of data require careful interfaces, transfer points, and scrutiny to ensure that important nuances are not distorting or disappearing altogether. Some professional work itself is proliferating into cooperative networks of teams, or activities delegated to myriad para-specialists, mediated by large data systems and algorithms. More troubling, perhaps, are examples where professional knowledge work is actually being delegated to digital technologies – to predict, diagnose, solve problems, or even decide.

2. *Reduction of knowledge and terms of decision-making.* Data analytics software works from simplistic premises: that problems are technical, comprised of knowable, measurable parameters, and can be solved through technical calculation. Complexities of ethics and values, ambiguities and tensions, culture and politics and even the context in which data is collected are not accounted for.

3. *Reliance on comparison and prediction.* As Barocas et al (2013) show, these algorithms 'embody a profound deference to precedent', acting on the past to ensure the future. Predictive analytics – which are used extensively in professional practice from medical diagnosis to school resource allocations and individual students' programme planning to police deployment of neighbourhood watch – depend on past pattern seeking and cycles of anticipation. These can be self-reinforcing and reproductive, augmenting path dependency and entrenching existing inequities.

4. *Methodologies are inadequate.* Critics in software studies claim that current quantitative methods taught to undergraduates such as pre-service professionals and educators are hopelessly out of date in this new era of big data: not only are the methods unsuited for enormous, unstructured 'dirty' datasets with unknown properties, but they also do not develop critical awareness of emerging forms and structures of data, and of how code works in and on professional practice and knowledge.

5. *New questions about professional agency and accountability.* Much data accumulation and calculation is automated, which opens new questions about the autonomy of algorithms and the attribution of responsibility when bad things happen.

In my own work, I argue that educators of professionals will play an important role in mediating these issues in the future (Fenwick 2015 *forthcoming*). Educators can actively encourage student professionals to engage critically with issues of big data and code, and to find out more about coding processes. Students can learn how to collaborate with digital designers and analysts and how to work effectively with data and data analytics in their practice. They also can explicitly debate the new implications of code and big data for professional responsibility.

But perhaps the most critical questions for educators of professionals will be about what professional knowing and capability is *becoming* through the changes in work practice being engineered by these digital technologies, and how education can better support this becoming. There is no doubt that professional work is being fundamentally transformed in many sectors, but professional education that doesn't transform itself in tandem risks producing a lot of unemployed or very frustrated practitioners. As we think about the future of professional education, we perhaps need to begin by considering more rigorously this question: What capabilities will be needed most—amidst the mix of digital and physical objects, languages, settings and codes—that *human* professionals of the future can bring? I believe that a crucial dynamic is responsibility—critical wise judgement, drawing upon capacities that are uniquely human such as empathy, intuition and caring. When we fully appreciate the emerging capacities of web science and digital tools, we realise how precisely we need to understand and value these human capacities of responsibility in relation with these digital assemblages.

Barocas, S., Hood, S. and Ziewitz, M. 2013. Governing algorithms: A provocation piece. Governing Algorithms conference, May 16-17, New York University.

Cipriano, M. 2014. Crime center kit boosts predictive policing capabilities. *GCN: Technology, Tools and Tactics for Public Sector IT*, 28 March 2014.

Fenwick, T. 2015 *forthcoming*. *Professionalism and Professional Responsibility in New Regimes: A sociomaterial examination*. London: Routledge

Jaradat, S., Whyte, J., and Luck, R. 2013. Professionalism in digitally mediated project work. *Building Research and Information* 41, no. 1: 51-59

Lewis, M. 2014. *Flash Boys: A Wall Street Revolt*, New York: W.W. Norton

Sullivan, J. 2013. How Google is using people analytics to completely reinvent HR. *TLNT: The Business of HR*, 26 February 2013.

Susskind, R. 2013. *Tomorrow's Lawyers: An Introduction to Your Future*. Oxford: Oxford University Press.

Technologies of professional knowledge making

Sarah Doyle, University of Stirling

In her book *Meeting the Universe Halfway: Quantum Physics and the Entanglement of Matter and Meaning* Karen Barad asserts that knowledge making is not a mediated activity. In my study about the emergence of professional knowing in paediatric diabetes, I am finding that digital technologies participate in very important ways. I am researching professional knowing by examining the conditions and practices through which it comes into being.

This approach includes focusing on the entangled nature of human knowing and the material arrangements for knowing. For example, in the case of paediatric diabetes, the particular tools and technologies of treatment are part of these arrangements. Digital technologies, in the form of insulin pumps (also known as Continuous Subcutaneous Insulin Infusion), initiate changed practices of knowing. Software packages replace handwritten notes; lengths of plastic tubing inserted into the abdomen replace injection pens and needles; and calculations of insulin doses rely on the digital technology inside the insulin pump. There are too many other patterns of difference to include them all here.

Barad's argument hinges on the recognition that this is not a case of professional knowing mediated differently. Instead, she claims that the professional knowing is different. Hers is a point of ontology. So for example, the challenge of making theoretical knowledge translate into practical knowledge is reframed by conceptualising the knowings as different. The issue is not how to get the same knowing to work in different spaces and arrangements—the issue is that different knowings are configured by different spaces and arrangements. The professional knowing that is configured by the insulin pump is not the same knowing configured differently by the injection pens. The insulin pump configures a professional knowing that is ontologically different.

In a different study two newly qualified (Foundation Year 1) doctors talking about their first jobs put it like this,

> 'Yeah, medical school doesn't really prepare you for being an FY1, it's completely different you know…I knew what to do, I just didn't know how to do actually do it; I wasn't prepared in any practical sense at all.'
> 'Exactly! Like the bradycardia I saw the other day…I knew as a medical student that I needed to give atropine but I had never actually seen it, never drawn it up, never had to actually give it, so that knowledge isn't in a form you can use it.' (Tallentire et al, 2011)

The knowings are ontologically different, and it matters. The material arrangements for knowing are reconfigured to include for example not just the diagnosis and the name and dose of the required medication (atropine), but also the fine-grained details of administering the medication. This reconfigured newly qualified professional knowing includes myriad 'doings'. A small selection of those implied by the doctor in this extract include breaking the glass vial, assembling the needle and syringe, checking and rechecking patient identity, prescription, medication and dose, withdrawing the atropine from the ampoule into the syringe, and expelling air bubbles from the solution.

This way of thinking is a significant shift, because it suggests professional knowing is not just differently mediated through different materials. Instead, the professional knowing itself is materially different. Professional knowing materialises differently as a consequence of the different material arrangements for knowing.

Tallentire, V. R., Smith, S. E., Skinner, J. and Cameron, H. S. 2011. Understanding the behaviour of newly qualified doctors in acute care contexts. *Medical Education*, 45, Issue 10, 995–1005

Teacherbot

Sian Bayne, University of Edinburgh

The literatures on teaching with technology tend to be dominated by anthropocentric resistances to the technological 'working-over' of teaching, or on equally humanistically-oriented promises of, and imperatives for, 'enhancement' of learning through technology. 'Teacher automation' emerges within this literature almost as a nexus between the positions of technological-promise (the infinitely reproducible, low cost, maximally efficient automated tutor) and technological-threat (the supercession of human teachers by 'robots'). Based on my recent research on automated teaching (Bayne 2015) in this article I will briefly survey this landscape of resistance and embrace, before moving on to consider a particular code intervention—the 'teacherbot'—designed to allow us to play across the torn landscape of pedagogic automation.

TEACHER AUTOMATION: RESISTANCE AND EMBRACE

Feenberg (2003: 100) has cogently described the context of online education as one of embrace from 'corporate strategists…., top university administrators, and "futurologists"', motivated by the automation of teaching as a means for achieving greater efficiency in the 'business' of education:

> Their goal is to replace (at least for the masses) face-to-face teaching by professional faculty with an industrial product, infinitely reproducible at decreasing unit cost.

Faculty response to this rationalising imperative has been, according to Feenberg, a two-fold 'mobilization in defense of the human touch'. This takes the form either of blanket opposition to all kinds of digital interruptions to education; or a favouring of a model of online education that places human communication at its centre—technology as a 'support for human development and online community' (100-1). For Feenberg, both the managerialist, 'technocratic' embrace of technology, and its 'humanistic opposition', function as instrumentalisations of digital technology: on the one hand to achieve efficiency gains, and on the other to facilitate ready access into a newly-constituted social world. Both perspectives, in fact, work on the basis of humanistic assumptions of rational autonomy and the ontological separation of human 'subject' and technological 'object', whether that technological 'object' is turned either to technocratic or to 'democratising' social ends.

More recent work in similar vein interrogates the alignment of technology in teaching with the instrumentalisation of higher education, partly through a critique of the ready adoption by educational technologists of the 'language of learning' (Biesta 2005) in which teaching is disaggregated and reduced to 'facilitation', 'learning support', 'instructional design' and the like. According to Haugskbakk and Nordkvelle (2007: 10), the de-professionalisation of teaching implied in the rhetoric of the 'language of learning' lays us open to the possibility that automation might reasonably come to stand-in for and replace the teacher: a 'very concrete deduction from the technocratic dream', and one which should be resisted.

'Learner centredness' indeed continues to be used across much of the literature on educational technology, as a discursive mode by which the teacher is under-considered or even written-out of the equation. For example, the first 'Grand Challenge' identified in a recent 'Roadmap for Educational Technology' funded by the National Science Foundation in the US, relates to the 'personalization of education':

> We suggest that in the next few decades education will be personalized to harmonize with each student's traits, for example, personality, learning style, and states, such as, affect, and level of engagement. Computational tools will understand an individual's strengths, weaknesses, challenges and motivational style as might a human tutor. Technologies available to produce such personalized instruction include user-models, intelligent environments, gaming environments, and data mining. (Woolf 2010: 6)

It is ironic that the taking into account of a student's personality, style of learning and level of engagement is posited as a goal for 'the next few decades' of education, when (human) teachers have been pretty adept at working to this particular configuration of need for decades, if not centuries. This kind of discursive move is deeply problematic for education, because it assumes a profound deficit in current teaching method or capacity, while at the same time indicating that the 'solution' to such deficit lies in automation and advanced computation.

My point here is not that automated methods are undesirable—on the contrary, the computational turn in education is both exciting and important—it is rather that the terms on which they are proposed are driven by a productivity-oriented solutionism which has been critiqued for decades. Where advanced computational methods are proposed from an instrumentalising, humanistic perspective which sees the technology as in service to social 'need', resistance to such methods also takes humanistic forms positing essentialism (the 'human touch', 'desirable humanity', 'human relationships') as the main locus for resistance to cold technocratic imperative. It seems to be time for the debate to move on, and to take place within other, perhaps more generative terms.

TEACHERBOT: PLEASURE IN THE CONFUSION OF BOUNDARIES

To begin to engage with the debate on different terms, therefore, we need to explore ways of theorising and practicing digital education and automated teaching which are driven neither by technical-rational efficiency models, nor by equally instrumentally-focused social models which assume a position of humanistic opposition to, or appropriation of, digital technology (Feenberg 2003). Working within a broadly critical posthumanist mode which refuses to accept the dominance of the human over the natural-material, but rather sees the human subject as produced by its material and discursive entanglements, we can begin to explore new ways of valuing teacher presence which are not resistant to the generative potential of automation. Such a view would not see technological development as taking place in order to solve a problem, or address a deficit in teacher ability or productivity, but would rather explore how human and non-human teachers might work together in a teaching 'assemblage' which refuses ontological hierarchy in the interest of productive 'play'.

In this section of the paper, therefore, I outline an approach we developed in order to pursue such an agenda: the 'teacherbot' developed by a team at the University of Edinburgh in order to provide a level of co-teaching within a massive open online course (MOOC) on 'E-learning and digital cultures'.[1] This particular MOOC is characterised by very large numbers of enrolments (around 90,000 signed-up participants over three instances); a high level of take-up by highly-educated professionals in teaching-related areas (around 60% of participants hold postgraduate degrees, a large proportion of which are in education-related disciplines); a highly diffuse and global spread of participants (from around 200 countries); and an experimental course design which emphasises the building of a critical understanding of 'e-learning' by positioning it within the broader context of digital culture. For these reasons, the MOOC was a receptive place to begin to explore a critical approach to automation: large numbers of widely-distributed course participants created a challenge in terms of the team's capacity to 'scale up' teaching; the participants themselves were generally receptive to—and understanding of the critical motivations for—this kind of intervention; and the content of the course itself, which explored popular conceptions of 'the posthuman' alongside related theory and its broader educational applications, meshed well with the attempt to put these ideas into practice in a live teaching context.

Over the summer of 2014 we therefore worked with a developer[2] to build an automated teacher presence for our Twitter feed which would function to do useful work in the MOOC, while at the same time operating as a proof-of-concept for a critical approach to teacher automation. We wanted to develop a twitterbot which 'coded in' something of the teacher function to the MOOC, using it as a way of researching some creative and critical futures for a MOOC pedagogy in which the 'teacher function' might become less a question of living teacher presence and more an assemblage of code, algorithm and teacher-student agency. Here, I describe the design of the teacherbot and give a brief 'ethnographic snapshot' of its first few weeks, with the aim of exploring how the operations of teacherbot (or 'botty' as she came to be termed by students) is helping us address the critical agenda outlined in the first half of this paper.

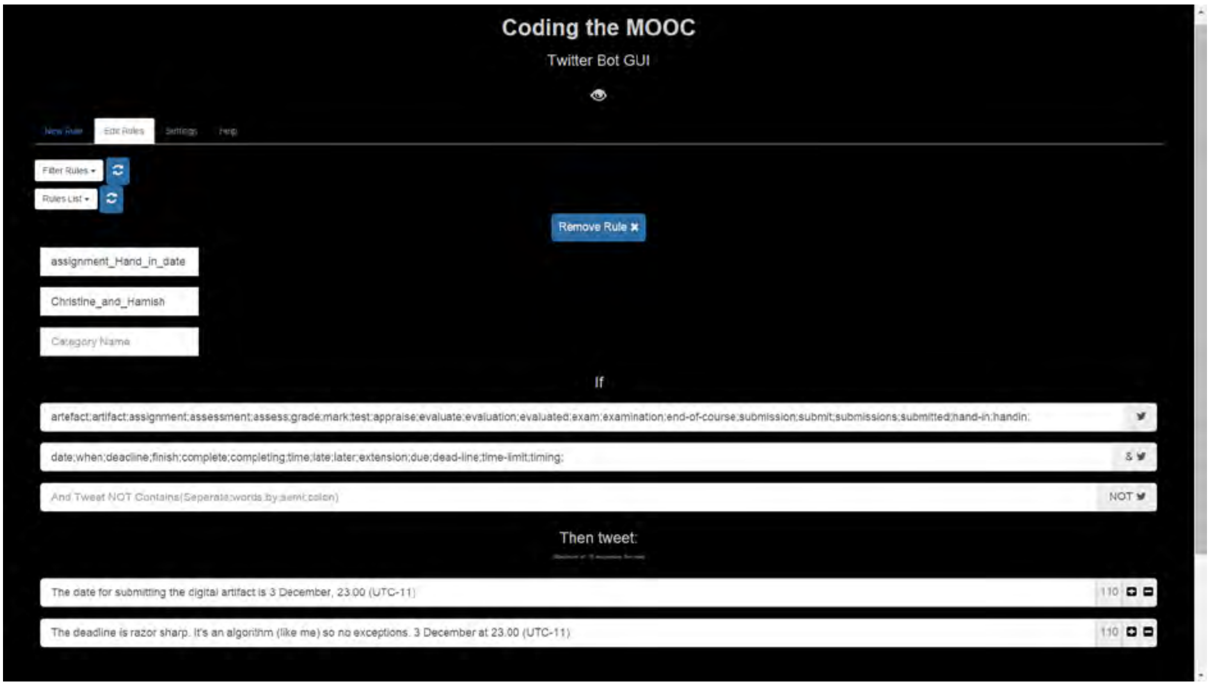
The technical development of the teacherbot consisted of a simple graphical user interface (a web form), a local MySQL database and a bot programmed using Java and based on an agent-oriented philosophy. This agent roams Twitter, storing tweets which use the #edcmooc tag on the database. The teachers then use the graphical user interface to develop rules for another agent who is in charge of posting tweets.

---

[1] This MOOC is offered on the Coursera platform: https://www.coursera.org/course/edc. The MOOC teaching team are Sian Bayne, Jen Ross, Jeremy Knox, Christine Sinclair and Hamish Macleod in the School of Education at the University of Edinburgh.

[2] Hadi Mehrpouya in the Design Informatics group at the University of Edinburgh.

The teacherbot is thus ‘programmed’ by the teaching team adding keywords and responses in this simple web form interface, making it very easy for professionals with no programming knowledge to craft their own bot:



assignment_Hand_in_date

Christine_and_Hamish

Category Name

artefact;artifact;assignment;assessment;assess;grade;mark;test;appraise;evaluate;evaluation;evaluated;exam;examination;end-of-course;submission;submit;submissions;submitted;hand-in;handin;

date;when;deadline;finish;complete;completing;time;late;later;extension;due;dead-line;time-limit;timing;

And Tweet NOT Contains(Seperate;words;by;semi;colon)

The date for submitting the digital artifact is 3 December, 23.00 (UTC-11)

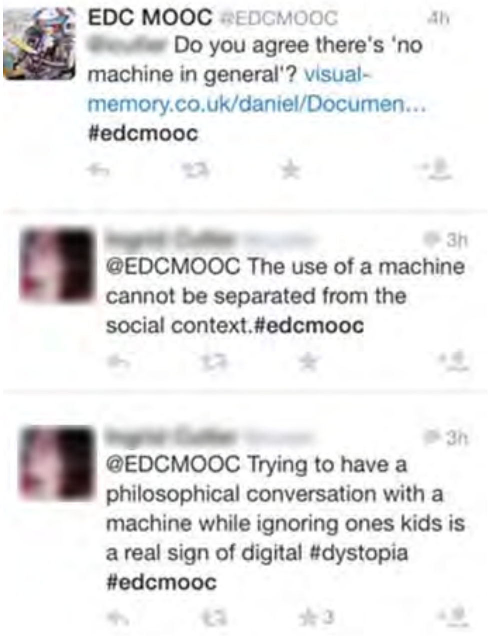
The deadline is razor sharp. It's an algorithm (like me) so no exceptions. 3 December at 23.00 (UTC-11)

Image 1: teacherbot interface for creating responses

For example, as shown in Image 1, we were able to formulate a rule which anticipated tweets from students expressing a need for clarification about the assignment deadline. By entering ‘assignment’ and similar terms in the first ‘If’ field, and ‘deadline’ and approximate synonyms in the second, we were able easily to program the teacherbot to respond to any tweet sent by a student using those terms plus the course hashtag (#edcmooc). The teacherbot in this instance would randomly choose one of two responses to tweet, giving immediate clarification. We mixed pragmatic and process-oriented rules like this one, with more curriculum-focused responses and other more social and playful interventions. A few examples of resulting exchanges between students and the teacherbot are given below.[3]

---

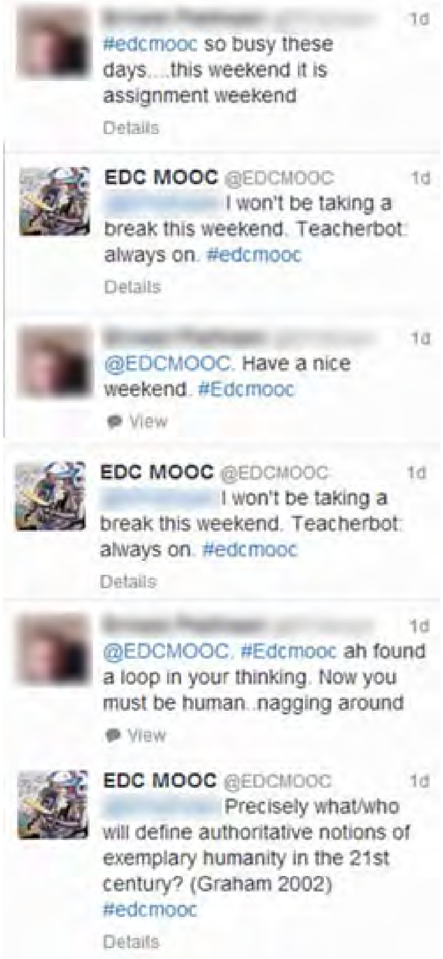
[3] EDCMOOC is the name used by the teacherbot, student usernames are blurred for anonymity.



EDC MOOC @EDCMOOC 4h
[redacted] Do you agree there's 'no machine in general'? visual-memory.co.uk/daniel/Documen... #edcmooc

[redacted] 3h
@EDCMOOC The use of a machine cannot be separated from the social context.#edcmooc

[redacted] 3h
@EDCMOOC Trying to have a philosophical conversation with a machine while ignoring ones kids is a real sign of digital #dystopia #edcmooc



[redacted] 1d
#edcmooc so busy these days....this weekend it is assignment weekend

EDC MOOC @EDCMOOC 1d
[redacted] I won't be taking a break this weekend. Teacherbot: always on. #edcmooc

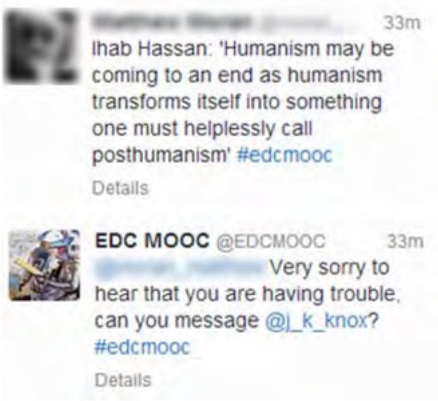
[redacted] 1d
@EDCMOOC. Have a nice weekend. #Edcmooc

EDC MOOC @EDCMOOC 1d
[redacted] I won't be taking a break this weekend. Teacherbot: always on. #edcmooc

[redacted] 1d
@EDCMOOC. #Edcmooc ah found a loop in your thinking. Now you must be human..nagging around

EDC MOOC @EDCMOOC 1d
[redacted] Precisely what/who will define authoritative notions of exemplary humanity in the 21st century? (Graham 2002) #edcmooc

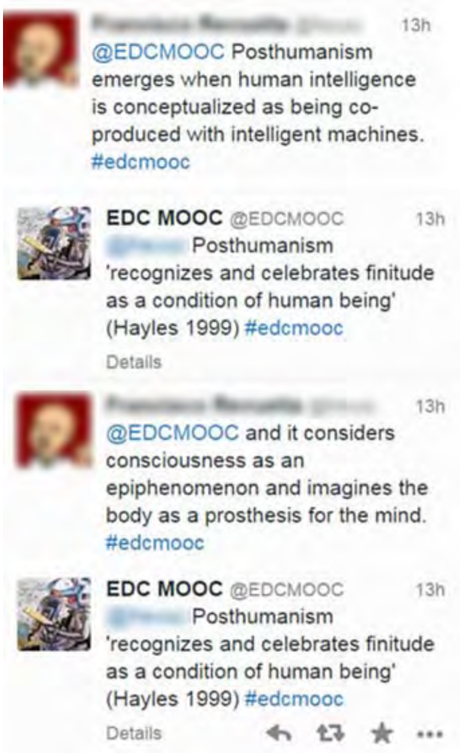
Students on the MOOC were aware that the EDCMOOC tweets were coming from a bot: we explicitly referred to the teacherbot in the course web site and in associated tutorials, and it was in any case fairly obvious in most instances that they were automated in some way. In this sense, we never attempted to mask the automated input or tried to pretend to students that it represented human teacher presence. The bot interventions were generally slightly 'clunky' and often rather wide-of-the-mark. However, as a piece of experimental boundary-work it functioned well: teacherbot responses worked playfully and with immediacy across the social exchanges on Twitter, prompting some often quite profound reflection on course concepts, as well as generative misunderstandings.



[redacted] 33m
Ihab Hassan: 'Humanism may be coming to an end as humanism transforms itself into something one must helplessly call posthumanism' #edcmooc

EDC MOOC @EDCMOOC 33m
[redacted] Very sorry to hear that you are having trouble, can you message @j_k_knox? #edcmooc

One strategy we used in creating rules for the teacherbot was to take very brief extracts from relevant readings, and have the bot use them as responses. This worked well, in that it resulted in several instances of students engaging playfully with the bot on the conceptual issues raised. A good example is that given below, in which a student deliberately set out to engage the bot in a series of exchanges quoting from Katherine Hayles' *How we became posthuman* (1999):



13h
@EDCMOOC Posthumanism emerges when human intelligence is conceptualized as being co-produced with intelligent machines. #edcmooc

EDC MOOC @EDCMOOC 13h
Posthumanism 'recognizes and celebrates finitude as a condition of human being' (Hayles 1999) #edcmooc
Details

13h
@EDCMOOC and it considers consciousness as an epiphenomenon and imagines the body as a prosthesis for the mind. #edcmooc

EDC MOOC @EDCMOOC 13h
Posthumanism 'recognizes and celebrates finitude as a condition of human being' (Hayles 1999) #edcmooc
Details

One student who engaged in this kind of exchange then went on to write a blog post on the experience of engaging with the teacherbot, saying:

> While I was trying to figure out what the hell "post-humanism" means, the teacher bot led me on a merry chase looking up quotes and obscure academic references, which had the interesting side effect of "ambush teaching" me. I will happily admit, that I do not feel like I have been to a class. I do not feel like I have been taught, either. I do, however, think I have learned something. I've certainly been prompted to think. Isn't this what every good teacher/trainer strives for?[4]

While the conclusion drawn by this student is that 'posthuman teachers can never happen in my lifetime', the teacherbot as an entity put into play to help students engage critically with the idea of automated teaching was well achieved in this instance. In general, students' expressed responses to the teacherbot were ones of initial bemusement, followed by engagement and a well-articulated understanding of what the bot was setting out to achieve within the space of the course.

---

[4] http://www.digitalang.com/2014/11/chatting-to-teacherbot-why-posthumanism-can-never-happen-in-my-lifetime

CONCLUSION

The teacherbot explicitly worked with the idea that teacher automation does not have to be about rationalism and instrumentalism: 'botty' was not intended to 'solve' any productivity deficits in teachers, or to replace teachers, but rather to explore how an assemblage of teacher-student-code might be pedagogically generative. In this sense, it also emphasised that our response to automation need not be an uncritical re-statement of the centrality of humanism for education. In explicitly prompting students to engage with questions about the shifting sociomaterial boundaries of human and machine in a pedagogic context, the teacherbot rather worked to establish a new way of thinking about automation which attempted to do as Pickering (2005: 31) prompts us: to try to 'see double' by viewing 'the human and the nonhuman at once, without trying to strip either away'. Above all, perhaps, it enabled us to engage in a resistant way with the famously supercessionist truism stated by Arthur C. Clarke (1980): 'Any teacher that can be replaced by a machine should be!'

Bayne, S. (2015) Teacherbot: interventions in automated teaching. *Teaching in Higher Education*, 20(4), 455-467.

Biesta, G. 2005. Against learning. Reclaiming a language for education in an age of learning. *Nordisk Pedagogik*, 2005/01, 54-66.

Clarke, A. C. (1980). Electronic Tutors. *Omni Magazine*. June 1980.

Feenberg, A. (2003) Modernity Theory and Technology Studies: Reflections on Bridging the Gap. In *Modernity and Technology*, edited by Thomas J. Misa, Philip Brey and Andrew Feenberg. MIT Press. 73-104.

Haugsbakk, G. and Nordkvelle, Y. 2007. The rhetoric of ICT and the new language of learning: a critical analysis of the use of ICT in the curricular field. *European Educational Research Journal* 6 (1): 1-12.

Pickering, A. (2005). Asian eels and global warming: a posthumanist perspective on society and the environment. *Ethics and the Environment* 10, no.2: 29-43.

Woolfe, B. P. (2010) *A Roadmap for Education Technology*. Online at: http://www.cra.org/ccc/files/docs/groe/GROE%20Roadmap%20for%20Education%20Technology%20Final%20Report.pdf [date of access: 15 June 2015]

The death of the theorist and the rise of data and algorithms in digital social research

Ben Williamson, University of Stirling

Computer software and data-processing algorithms are becoming an everyday part of Higher Education practices. How might this be affecting research in the social sciences and the formation of the professional identities of academics? These are important challenges for social science researchers in HE, asking us to consider how digital devices and infrastructures might be shaping our professional practices, knowledge production, and theories of the world.

CODING THE ACADEMIC LIFEWORLD

Computer code, software and algorithms have sunk deep into what Nigel Thrift (2011) has described as the 'technological unconscious' of our contemporary 'lifeworld,' and are fast becoming part of the everyday backdrop to Higher Education. Academic research across the natural, human and social sciences is increasingly mediated and augmented by computer coded technologies. This is perhaps most obvious in the natural sciences and in developments such as the vast human genome database. As Geoffrey Bowker (2005: 119) has argued, such databases are increasingly viewed as a challenge to the idea of the scientific paper (with its theoretical framework, hypothesis and long-form argumentation) as the 'end result' of science:

> The ideal database should according to most practitioners be theory-neutral, but should serve as a common basis for a number of scientific disciplines to progress. … In this new and expanded process of scientific archiving, data must be reusable by scientists. It is not possible simply to enshrine one's results in a paper; the scientist must lodge her data in a database that can be easily manipulated by other scientists.

The apparently theory-neutral techniques of sorting, ordering, classification and calculation associated with computer databases have become a key part of the infrastructures underpinning contemporary big science. The coding and databasing of the world does not, though, end with big science. It is becoming a major preoccupation in the social sciences and humanities too.

BIG METHODS

Across multiple disciplines in the social science and humanities, 'big data' are now being generated and mobilized. Catalyzed by huge government investment in big data research centres through the research councils, the production of social and human knowledge and theory is also now being affected by 'big methods' enacted through software code, algorithms and the data they process.

For some enthusiastic commentators, such as Chris Anderson of *Wired* magazine, big data and its associated algorithmic techniques are bringing about 'the end of theory' and making disciplinary expertise obsolete:

> This is a world where massive amounts of data and applied mathematics replace every other tool that might be brought to bear. Out with every theory of human behavior, from linguistics to sociology. Forget taxonomy, ontology, and psychology. Who knows why people do what they do? The point is they do it, and we can track and measure it with unprecedented fidelity. With enough data, the numbers speak for themselves.

Another recent *Wired* article ran with the provocative headline 'Big data and the death of the theorist.' While such claims represent exaggerated obituaries for the social sciences, it is clear that big data software and its algorithmic techniques of analysis are increasingly challenging conventional views about the institutional practices and spaces of social scientific knowledge production (Kitchin 2014).

ALGORITHMIC EXPERTISE

Social science appears to be escaping the academy. Instead of social scientists, the new experts of the social media environment, argue Viktor Mayer-Schonberger and Kenneth Cukier (2013), are the 'algorithmists' and big data analysts of Google, Facebook, Amazon, and of software and data analysis firms. Algorithmists are experts in the areas of computer science, mathematics, and statistics, as well as aspects of policy, law, economics and social research, who can undertake big data analyses and evaluations. Facebook, for example, has a Data Science Team that can apply a combination of programming skills and social science to mine data for insights that will advance Facebook's business while also contributing to new social science insights. Run by Facebook's 'in-house sociologist,' the Data Science Team aims to mobilize Facebook's massive data resources to 'revolutionize' existing understandings of human and social behaviour.

Similar methods are being deployed in politics. The think tank Demos has recently established a Centre for the Analysis of Social Media (CASM), a research centre dedicated to 'social media science' which seeks to 'see society-in-motion' through big data, as its research director Carl Miller explains:

> To cope with the new kinds of data that exist, we need to use new big data techniques that can cope with them: computer systems to marshal the deluges of data, algorithms to shape and mould the data as we want and ways of visualising the data to turn complexity into sense.

Underlying ‘social media science’ is a belief that the behaviour of citizens can be analysed and understood as a kind of data to inform new policy ideas. The emergence of ‘policy labs‘ that work across the social scientific, technological and policy fields—such as the Public Services Innovations Lab at Nesta in the UK, New York’s Governance Lab, and Denmark’s MindLab—is further evidence of how social scientific methodological expertise is diversifying (Williamson 2015). For think tanks and policy labs the political promise of using sophisticated algorithmic techniques to analyze and visualize big data is to make societies and populations visible with unprecedented fidelity in order to improve government intervention.

SOCIOLOGICAL SOFTWARE

The recent emergence of approaches such as ‘digital sociology‘ and ‘digital social research‘ reflect disciplinary anxieties about the relevance of social science at a time when commercial social media companies, R&D labs, and think tanks are staking their claim to social scientific expertise (Lupton 2015). Many researchers are optimistic about the synergies between digital and social research methods. Emerging methods deployed by digital researchers such as Lev Manovich (2013) and Richard Rogers (2013) include the use of Twitter and blogs to document everyday activities, the mobilization of search engine analytics to reveal massive population trends and social behaviours over time, the analysis of Instagram images to detect cultural and social patterns, the study of social network formation on Facebook, and so on. These platforms enable the continuous generation of data about social life and make possible new forms of social data, analysis and visualization.

As David Beer (2012a) reports, the kind of software that can crawl, mine, capture and scrape the web for data has the potential to be powerful in academic research. Social media aggregators, algorithmic database analytics and other forms of what might be termed ‘sociological software’ have the capacity to see social patterns in huge quantities of data and to augment how we ‘see’ and ‘know’ ourselves and our societies. Sociological software offers us much greater empirical, analytical and argumentative potential.

REMEDIATING METHODS

For other researchers, the outlook is less hopeful. Lisa Gitelman and Virginia Jackson (2013) argue that the capacity to mobilise data graphically as visualizations and representations, or ‘database aesthetics,’ amplifies the rhetorical, argumentative and persuasive function of data. Moreover, software and big data seriously threaten established social research and are associated with a concentration of data analysis and knowledge production in a few highly resourced research centres, including the R&D labs of corporate technology companies.

Noortje Marres (2012) has suggested another way to think about the proliferation of new devices and formats for the documentation of social life. She argues that we need to acknowledge a ‘redistribution of social research’ and to see social science as a ‘shared accomplishment’ as the roles of social research are distributed between different actors. Such a redistribution of research would include human actors such as academic researchers, software developers, data analysts, commercial R&D labs, and bloggers and tweeters, but also a much broader set of actors such as databases, software, algorithms, platforms, and other digital devices, media and infrastructures that all contribute to the enactment of digital social research. The redistribution of research among these diverse actors, Marres argues, would entail a 'remediation of methods' as social research is reshaped and refashioned through the use of devices and platforms.

The process of redistributing, remediating, or 'reassembling social science methods' as Evelyn Ruppert, John Law and Mike Savage (2013: 25) have articulated it, means recognizing that digital devices are both part of the material of social lives and part of the methodological apparatus required for knowing those lives too, as 'digital devices are reworking and mediating not only social and other relations, but also the very assumptions of social science methods and how and what we know about those relations.' Likewise, in an article detailing the 'algorithms in the academy,' David Beer (2012b) has shown that software algorithms are increasingly intervening in social research through the 'algorithmic normalities' of SPSS, GoogleScholar, LexisNexis, as well as emerging social media and data analytics devices, which frame information and codify the social world in certain ways—shaping the objects of analysis.

DATAFIED IDENTITIES

Digital devices are also reworking and mediating HE institutions and academic researchers' professional identities and personal lives. Roger Burrows (2012) has argued that the work of researchers in universities is now subject to 'metricization' from an assortment of measuring and calculating devices. These include bibliometrics, citation indices, workload models, transparent costing data, research and teaching quality assessments, and commercial university league tables, many increasingly enacted via code, software and algorithmic forms of power. As a result, Deborah Lupton (2015) suggests, an academic version of the 'quantified self' is emerging: a professional identity based on quantified measures of output and impact.

Writing in an article titled '#MySubjectivation,' the philosopher Gary Hall (2013) argues that today's social media are constitutive of a particular emergent 'epistemic environment.' The epistemic environment of 'traditional' academic knowledge production was based on the Romantic view of single authorship and creative genius materialized in writing, long-form argumentation, and the publication of books. New social media infrastructures, however, are reshaping the epistemic environment of contemporary scholarly knowledge production. In the emerging epistemic environment of HE, academics are increasingly encouraged to be self-entrepreneurial bloggers and tweeters, utilizing social media platforms and open access publishing environments to extend their networks, drive up citations, promote their professional profiles, and generate impact. Commercial social media platforms such as Twitter, Facebook, LinkedIn and Academia.edu are becoming part of the everyday networked infrastructure through which academics create, perform and circulate research, knowledge and theory. As Hall (2013: 89) states it, the emerging epistemic environment:

> invents us and our own knowledge work, philosophy and minds, as much as we invent it, by virtue of the way it modifies and homogenizes our thought and our behaviour through its media technologies.

To put it more bluntly, academics are becoming data, as mediated through complex coded infrastructures and devices. Geoffrey Bowker (2013) has written that 'if you are not data, you don't exist'; the same is true for academics in Higher Education. The unfolding effects of data and algorithms on HE ought to be the subject of serious social scientific inquiry.

Whether we are confronting the 'end of theory' and the 'death of the theorist' as computer coded software devices and sophisticated algorithms increasingly mediate, augment, and even automate academic practice and knowledge production remains an open question for further research. Is academic work really being homogenized and manipulated by the media machines of Google and Facebook, and is disciplinary expertise and knowledge production being displaced to the 'algorithmists' of private R&D labs and commercial technology firms? For researchers in Higher Education the task is to be open and alert to the current redistribution of research across these new infrastructures, devices, experts and organizations, and to recognize how our knowledge, theories and understandings of the social world we are studying are being mediated, augmented and even co-produced by software code and algorithmic power.

Beer, D. 2012a. Using Social Media Data Aggregators to Do Social Research. *Sociological Research Online* 17, no. 3: http://www.socresonline.org.uk/17/3/10.html

Beer, D. 2012b. Leave the thinking to us. *Times Higher Education*, 30 August 2012: https://www.timeshighereducation.co.uk/news/leave-the-thinking-to-us/420978.article?storycode=420978

Bowker, G.C. 2005. *Memory Practices in the Sciences*. London: MIT Press.

Bowker, G.C. 2013. Data flakes: An afterword to 'Raw Data' is an Oxymoron. In L. Gitelman (ed.) *'Raw Data' is an Oxymoron*: 167-71. London: MIT Press.

Burrows, R. 2012. Living with the h-index: Metric assemblages in the contemporary academy. *The Sociological Review* 60, no. 2: 355–372.

Gitelman, L. & Jackson, V. 2013. Introduction. In L. Gitelman (ed.) *'Raw Data' is an Oxymoron*: 1-14. London: MIT Press.

Hall, G. 2013. #MySubjectivation. *New Formations* 1, no. 7: 83-103.

Kitchin, R. 2014a. Big Data, new epistemologies and paradigm shifts. *Big Data & Society* 1, no.1. http://dx.doi.org/10.1177/2053951714528481

Lupton, D. 2015. *Digital Sociology*. London: Routledge.

Manovich, L. 2013. *Software Takes Command: Extending The Language Of New Media*. London: Bloomsbury Academic.

Marres, N. 2012. The redistribution of methods: on intervention in digital social research, broadly conceived. *The Sociological Review* 60, S1: 139-165.

Mayer-Schonberger, V. & Cukier, K. 2013. *Big Data: A revolution that will change how we live, work and think*. London: John Murray.

Rogers, R. 2013. *Digital Methods*. London: MIT Press.

Ruppert, E., Law, J. & Savage, M. 2013. Reassembling Social Science Methods: The Challenge of Digital Devices. *Theory, Culture & Society* 30, no. 4: 22–46.

Thrift, N. 2011. Lifeworld Inc—and what to do about it. *Environment and Planning D: Society and Space* 29, no. 1: 5–26.

Williamson, B. 2015. Governing methods: policy innovation labs, design and data science in the digital governance of education. *Journal of Educational Administration & History* 47, no. 3: 251-271. http://dx.doi.org/10.1080/00220620.2015.1038693

Web Literacy beyond software with shareholders

Doug Belshaw, Dynamic Skillset

We're at peak centralisation of our data in online services, with data as the new oil. It's a time of 'frictionless sharing', but also a time when we're increasingly having decisions made on our behalf by algorithms. Education is now subject to a land grab by 'software with shareholders' looking to profit from collecting, mining, packaging, and selling learner data. This article explores some of the issues at stake, as well as pointing towards the seeds of a potential solution.

WEB TREND MAP 4
iA

In 2009, a design firm called Information Architects produced their fourth iteration of a 'Web Trends Map'. As with previous versions, they set it out in the style of the Tokyo subway, with websites grouped and located by association.

This was the height of the 'Web 2.0' era, a hugely creative and liberating force in the tech industry meaning that, for the first time, the average web user could not only consume the web, but publish to it. According to Flew (2008) it signaled:

> [a] move from personal websites to blogs and blog site aggregation, from publishing to participation, from web content as the outcome of large up-front investment to an ongoing and interactive process, and from content management systems to links based on tagging (folksonomy).

THE PROBLEM

Interestingly, the way that Information Architects chose to portray the largest websites of the day is very much reminiscent of agricultural silos. This is an apt metaphor as silos exist to keep things discrete and separate. Although the tenets of Web 2.0 had been interoperability and openness, intense competition led to vertical integration. This was hastened by the confluence of a burgeoning (and lucrative) 'native' app ecosystem and with the impact of the worldwide economic slowdown. In short, technology companies sought to lock users into their platform.

This kind of move has its analogue in the territorial imperialism of the 19th century. Recent developments around internet.org, a project from Facebook, confirms this trend. In return for free access to hand-picked websites ('zero-rating'), the principles of the open web are undermined:

> ‘The impact of zero-rating may result in the same harms as throttling, blocking, or paid prioritization. By giving one company (or a handful) the ability to reach users at no cost to them, zero-rating could limit rather than expand a user’s access to the Internet and ultimately chill competition and innovation. The promise of the Internet as a driver of innovation is that anyone can make anything and share it with anyone. Without a level playing field, the world won’t benefit from the next Facebook, Google or Twitter.’ (Dixon-Thayer, 2015)

The approach taken by internet.org has been labelled by the Electronic Frontier Foundation (EFF), a non-profit set up to defend user rights in the digital world, as ‘a ghetto for poor users instead of a stepping stone to the larger Internet’ (Gillula & Malcolm, 2015). Even worse, instead of a tool for human flourishing, it could become an instrument of oppression for users in third-world countries:

> By setting themselves up as gatekeepers for free access to (portions of) the global Internet, Facebook and its partners have issued an open invitation for governments and special interest groups to lobby, cajole or threaten them to withhold particular content from their service. In other words, Internet.org would be much easier to censor than a true global Internet. (Gillula & Malcolm, 2015)

One of the main reasons technology companies seek to expand their number of users is because, as publicly-traded organisations, they are legally obliged to maximise shareholder value. Their services are provided free at the point of access for users, so the profits of tech companies such as Google, Twitter, and Facebook are generated through data-mining and selling this targeted data to advertisers. They are providing, in effect, 'software with shareholders' (Belshaw, 2014a).

Data, then, is often seen as the 'new oil'. It is a source of huge wealth and opportunity for those who control the giant technology silos within which we learn, socialise, and transact. Startups that create apps that attract large user bases are purchased for eye-watering amounts of moneyeven before they are profitable. An example of this would be WhatsApp, the instant messaging app for smartphones, bought by Facebook in 2014 for $22 billion.

The oil metaphor can be instructive in terms of the way we approach this new landscape. Instead of data-mining with abandon, perhaps we could, as Jer Thorp pleads in his 2012 Harvard Business Review article, bring ‘some much-needed criticality’ to bear:

> Our experience with oil has been fraught; fortunes made have been balanced with dwindling resources, bloody mercenary conflicts, and a terrifying climate crisis. If we are indeed making the first steps into economic terrain that will be as transformative (and possibly as risky) as that of the petroleum industry, foresight will be key. We have already seen ‘data spills’ happen (when large amounts of personal data are inadvertently leaked). Will it be much longer until

> we see dangerous data drilling practices? Or until we start to see long term effects from ‘data pollution’? (Thorp, 2012)

One of the fastest-growing areas in tech is the burgeoning field of so-called 'Big Data'. Using large data sets, algorithms are let loose to find correlations. Acting on the results of this data-mining process in the tech world is often called 'growth hacking'. In the world of education it's known as 'learning analytics'. In both cases, the 'data exhaust' of users is analysed to an unprecedented degree. Interactions with friends, colleagues, and peers are cross-referenced for trends and patterns.

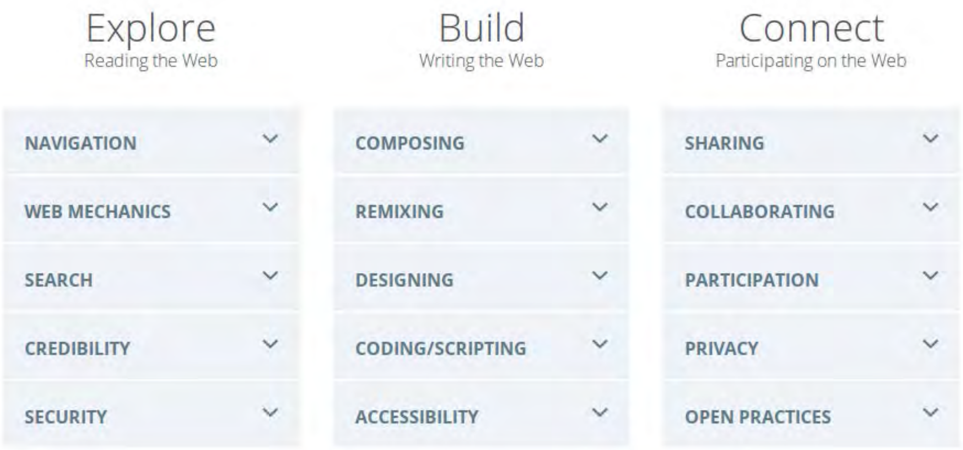
In the case of tech companies such as Facebook, it was found that their extremely long user agreement allowed them to conduct a large psychological experiment on almost 700,000 users of the social network (McNeal, 2014). As part of the experiment, the 'news feed' of users was manipulated to assess the effect on their emotions. This 'social contagion' research, which some sources claim was funded in part by the US military, was successful. The researchers *were* able to affect emotions based on manipulating Facebook news feeds. This was only made public, however, due to the results being published in an academic journal (Kramer, Guillory & Hancock, 2014).

In the case of learning analytics, it could be argued that data-mining to ascertain patterns of engagement and behaviour is a *positive* use of such data. The aim here, after all, is to intervene to prevent the individual dropping out of university and to ensure they have the help required to succeed on a course. However, with an increased move towards the marketisation and commercialisation of higher education, this cannot be seen in neutral terms. In the case of both growth hacking and learner analytics, the aim is to use data generated by - but not necessarily accessible to - users in ways they may never have imagined.

TOWARDS A SOLUTION

There are many groups fighting for user rights on the web. Two of these, the EFF and Mozilla, have been mentioned above as expressing concern over Facebook's internet.org initiative. Mozilla in particular has been very active in efforts to inform and educate the public about the skills and competencies required to read, write, and participate on the web.

To this end, the Mozilla Foundation worked with a community of stakeholders on a 'Web Literacy Map'. The latest iteration at the time of writing is version 1.1:



Explore Reading the Web	Build Writing the Web	Connect Participating on the Web
NAVIGATION	COMPOSING	SHARING
WEB MECHANICS	REMIXING	COLLABORATING
SEARCH	DESIGNING	PARTICIPATION
CREDIBILITY	CODING/SCRIPTING	PRIVACY
SECURITY	ACCESSIBILITY	OPEN PRACTICES

Each strand ('Explore', 'Build', 'Connect') contains five competencies, with each competency containing, in turn, a number of skills. Each skill is framed in terms of *action* and phrased in a way that can be achieved at multiple levels. The aim is for the Web Literacy Map to be the raw material from which learning materials and curricula can be devised. Most relevant to the current discussion is the 'Web Mechanics' competency, containing the following skills:

- Using and understanding the differences between URLs, IP addresses and search terms.
- Identifying where data is in the network of devices that makes up the Internet.
- Exporting, moving, and backing up data from web services.
- Explaining the role algorithms play in creating and managing content on the web.
- Creating or modifying an algorithm to serve content from around the web.

Through carefully-planned activities, developing these skills also cultivates 'habits of mind' in web users, allowing them to curate their own information environment. This means users are equipped to be effective in any kind of digital environment:

> Being aware of the way that tools shape the way we think and interact with the world is the first step on the way to changing behaviours. As learners, as teachers, as *citizens*, we have a duty to ourselves and to one another to be mindful of this. (Belshaw, 2014b)

Building upon technical standards is a long-established trend with the web. We need to do likewise with our teaching and learning about how the web works. Building upon iterative, crowdsourced standards such as Mozilla's Web Literacy Map allows us to shape the web *we* want - rather than the distorted view that suits shareholders.

Belshaw, D.A.J. (2014a). Software with shareholders (or, the menace of private public spaces). Doug Belshaw's blog. 23 April 2014. http://dougbelshaw.com/blog/2014/04/23/software-with-shareholders.

Belshaw, D.A.J. (2014b). Curate or Be Curated: Why Our Information Environment is Crucial to a Flourishing Democracy, Civil Society. DMLcentral. 23 October 2014. http://dmlcentral.net/blog/doug-belshaw/curate-or-be-curated-why-our-information-environment-crucial-flourishing-democracy

Dixon-Thayer, D. (2015). Mozilla View on Zero-Rating. Open Policy & Advocacy Blog. Mozilla. 5 May 2015. https://blog.mozilla.org/netpolicy/2015/05/05/mozilla-view-on-zero-rating.

Flew, T. (2008). *New Media: An Introduction* (3rd ed.). Melbourne: Oxford University Press.

Gillula, J. & Malcolm, J. (2015). Internet.org Is Not Neutral, Not Secure, and Not the Internet. Deeplinks Blog. Electronic Frontier Foundation. 18 May 2015. https://www.eff.org/deeplinks/2015/05/internetorg-not-neutral-not-secure-and-not-internet.

Kramer, A.D.I., Guillory, J.E., Hancock, J.T. (2014) Experimental evidence of massive-scale emotional contagion through social networks. *Proceedings of the National Academy of Sciences of the United States of America* 111, no.24.

McNeal, G.S. (2014). Facebook Manipulated User News Feeds To Create Emotional Responses. Forbes. 28 June 2014. http://www.forbes.com/sites/gregorymcneal/2014/06/28/facebook-manipulated-user-news-feeds-to-create-emotional-contagion

Mozilla. (2015). Web Literacy Map v1.1. https://teach.mozilla.org/teach-like-mozilla/web-literacy

A hidden computing curriculum

Ben Williamson, University of Stirling

While issues about the ways in which people are now increasingly 'learning *through* code' have been the focus of the papers in this collection, there has also been a massive increase in interest more widely with people 'learning *to* code.' The basic premise for this enthusiasm is that in a world saturated with powerful software code, algorithms, and big data processing, people will need to learn to program in order to avoid 'being programmed' (Rushkoff 2010). In the last few years, the idea of learning to code and associated ideals around 'digital making' have grown from a minority focus among computing educators, grassroots computing organizations, and computer scientists into a major global educational concern. In England, it has been taken so seriously that it has even become an important educational agenda amongst politicians and policymakers. The most significant political development of this agenda has been the 'disapplication' of the subject ICT (Information and Communication Technology) in the English National Curriculum (which critics claim over-emphasized basic functional skills for using computers), and its replacement from September 2014 with a new computing curriculum that focuses instead on computer science, programming skills and 'computational thinking'—the understanding of how to construct problems so they can be expressed in machine-readable binary mathematics.

Learning to code has been translated from a grassroots campaign into curriculum reform in a remarkably concentrated period, yet the powerful actors mobilizing it into curriculum policy and the practices of coding promoted through its pedagogies are largely unrecognized in educational research. This is surprising since coding programmes such as Code Club, Codecademy, Hour of Code, and Year of Code, as well as digital making initiatives like Make Things Do Stuff and many others, have attained a high profile over the last few years among business, media, civil society and political groupings. The BBC, for instance, has launched a raft of content around coding and computing for a major initiative supporting the computing curriculum in 2015, including the distribution of a handheld coding device to a million UK schoolchildren. Drawing on my recent research on the various campaigns, organizations and lobbying groups that have combined to advocate for coding and computing in the curriculum (Williamson 2015), I want to suggest that there is a 'hidden curriculum' behind the computing curriculum, one that can be glimpsed if we take a closer look at some of the thinking behind learning to code and related digital making campaigns.

LEARNING TO CODE IDEOLOGY

The first aspect of the hidden computing curriculum to note is that 'coding' carries into the classroom a specific set of assumptions about ways of knowing and doing things. Writing code is not just a technical procedure but is related to systems of thought about the way the world works, and about how it might be modelled in order to further shape people's interactions with it. As Rob Kitchin and Martin Dodge (2011: 33) have argued, coding is a 'disciplinary regime' with established 'ways of knowing and doing regarding coding practices.' Writing code projects the 'rules' of computer science and its system of computational thinking into the world. It captures assumptions about how the world works and translates them into formalized models that can be computed through algorithmic procedures.

Taking such points as cues, we can begin to see how learning to code embodies a host of assumptions and working practices based on ideas such as computational thinking, statistical modelling, systems thinking, scientific rationality, and procedural algorithmic logic that have their origins in the working practices and codes of conduct associated with the 'culture of code' of software development (Hayes 2015). These are very specific kinds of social practices which, like much of the hype around 'big data,' as Rob Kitchin (2014) argues, are contextualized within a particular scientific approach, reflecting sometimes quite functionalist and technicist modes of thinking that approach the world in computational terms rather than in relation to cultural, economic or political contexts. In this sense, learning to code may be interpreted as a material practice of what Astrid Mager (2012) terms 'algorithmic ideology,' a kind of introduction into the codes of conduct, practices, assumptions, and values that underpin the production of code.

Learning to code thus seeks to inculcate learners into the systems of computational thought associated with the professional regime of programmers and the computer science disciplines, and with their philosophies of the world, biases, prejudices, ideological assumptions and modes of perception. This is not to suggest that coders or coding are bad things, but to acknowledge what approaches to the world they privilege.

LEARNING TO CODE/TO LABOUR

An aspect of the hidden curriculum of schooling, as educational sociology has long claimed, is its socialization of children into the workforce. Certainly it is possible to detect such thinking in many of the learning to code campaigns and organizations that have helped contribute to the computing curriculum. Campaigns such as Make Things Do Stuff and Year of Code are premised at least in part on the idea that coding is an economically valuable skill to be developed in children before they reach working age. Some of the key reports contributing to the computing curriculum have made similar arguments, notably reports from Nesta and The Royal Society, as well as more recent materials from the think tank the Education Foundation and the UK Digital Skills Taskforce.

In most of these materials around learning to code, digital making, makerspaces and so on, coding is positioned as a rewarding, desirable and skilled occupation. Indeed, a message on Twitter by one edtech advocate in the UK argued that if children want top jobs in coding, then they are better off maintaining a GitHub code-sharing page than passing GCSE examinations. Although one tweet doesn't amount to research evidence, it signals how some of the commentary around learning to code, digital making and related computing projects reflect a modern political preoccupation with sculpting a mind and body with the technical skills, knowledge and capacity for entrepreneurship and value-creation in the digital economy.

Yet this depiction glosses over the fragility, complexity and mundanity of much coding work in the digital economy. As Adrian Mackenzie (2006: 14) notes, 'the figure of the programmer often vacillates between potent creator of new worlds and antisocial, perhaps criminal or parasitic.' More prosaically, the work of coding is often dull, routinized and monotonous, as well as difficult, frustrating and dysfunctional. Owing to intense ongoing innovation in the field, programmers are always struggling to learn and adapt to constant change and experience a high degree of 'ignorant expertise' and confusion about what they are doing, particularly in relation to the wider possible social effects of the software they produce, and the kinds of interactions and ways of seeing and doing things that they make possible (Kitchin & Dodge 2011). There are even reports that a great deal of programming work will be automated in the near future by advances in machine learning, and that the idea of learning to code is being made obsolete by developments in cognitive computing (Frey & Osborne 2014; Wakeford 2014).

As a result, the learner participating in Code Club, Year of Code, Make Things Do Stuff, or the like, is being solicited into a system of thinking, knowing and doing associated with coding practice that is not always as systematic, objective and expert as it is widely represented as being by learning to code advocates. Learning to code is premised on a fantasy of the material practices associated with coding which simplifies and glamorizes the mundane and even dysfunctional reality of disciplinary practice in the digital economy, and which ignores its potential for automation in the near future.

LEARNING TO CODE FOR COMMERCE

According to some critics such as Evgeny Morozov (2014), learning to code and digital making have become major commercial concerns, stripping such activities of their original 'radical' intentions. For example, in the US, the Hour of Code campaign was co-founded by the Partovi twins, 'angel investors' from Silicon Valley, and has been heavily promoted and backed by massive multinationals like Microsoft, Facebook and Google. Its British cousin, Year of Code, was set up by entrepreneurs at Index Ventures, an international venture capital firm whose mission statement is that 'every aspect of human life and economic activity can be transformed by technology and entrepreneurial passion.' The chair, executive director and advisors of Year of Code are almost all drawn from the fields of technology entrepreneurship and venture capital. As the *Guardian* newspaper columnist John Naughton has argued, 'Year of Code is a takeover bid by a corporate world that has woken up to the realization that the changes in the computing curriculum … will open up massive commercial opportunities.' As if to demonstrate this, the chief executive of Codecademy claims that they have 'struck oil' as the computing curriculum is 'forcing an entire country to learn programming,' though at about the same time, the director of Code Club was forced to quit over demands from its board that she refrain from criticizing the 'corporate mass surveillance' practices of commercial sponsor Google.

Brian Hayes (2015) argues that learning to code is entirely consistent with the commercial culture of software development, and while optimistic that 'a new generation discovers that coding is cool' and about the 'hacker enthusiasm' for the 'nerdy side of life,' he is cautious about the long-term contribution of learning to code initiatives to the disciplinary field of computer science:

> At the moment, most of the energy flows into the culture of software development or programming. The excitement is about applying computational methods, not inventing new ones or investigating their properties. … Everyone wants to pick up the knack of coding, but the more abstract and mathematical concepts at the core of computer science attract a smaller audience.

Learning to code therefore needs to be understood in terms of the longer history of the separation of coding from computer science, and potentially seen as an unhelpfully 'cool' deviation from the far less funky business of advancing the future of computing itself. The distinguishing culture of code associated with computer science, and the professional identities held by those who conduct it, is distinct from the culture of code associated with learning to code, and the potential professional identities available for those who pursue coding further.

But this is not just about commercial firms—it's about the increasing entanglement of business and government. In their book on political lobbying in the UK, Tamasin Cave and Andy Rowell (2014) describe the various activities surrounding the learning to code movement and the reform of the computing curriculum as a lobbying tool for technology firms with a clear, vested interest in digitizing learning, as well as enthusing a new generation of coders. This 'campaign of business-backed think tanks and education technology lobbyists,' Cave and Rowell argue, has now 'got what it wanted' in the shape of the computing curriculum and strong political support for the educational technology market.

LEARNING TO CODE FOR PROSUMPTION

However, programmes such as Make Things Do Stuff and Code Club justify themselves not just through the prospective economic and commercial value of children learning to code, but through a wider cultural argument about people producing and not simply consuming technology. One way to analyze this preoccupation with coding clubs, programming and related digital making activities is to view it as promoting 'participatory' practices of 'co-production,' 'crowdsourcing' and 'prosumption' in new social media practices, as advocated through Make Things Do Stuff and Code Club.

The term 'prosumption' registers the alleged blurring of production and consumption as consumers of digital media increasingly also become its producers. The media theorist Lev Manovich (2013), for example, argues that software development is gradually getting more democratized as social media—Facebook, Twitter, YouTube, Wikipedia, and so on—enables users to create and post content, contribute to 'crowdsourced' forms of 'co-production,' and perform their own customizations, mash-ups and remixes of existing material.

While prosumption is presented by its advocates in highly positive terms, critics such as the sociologists David Beer and Roger Burrows (2013) claim the increasing participation of people in the formation of media content is leading to the 'significant phenomena of the growing amount of "labouring" people are undertaking as they "play" with these new technologies.' 'Free labour' is the perfect business model for contemporary capitalism. As such, prosumption firmly embeds people in what Beer & Burrows term the social media 'infrastructures of participation' that are subject to the commercial interests of for-profit social media corporations. Within such infrastructures, the prosumerist individual is encouraged to share personal information and data; maximize sociality through horizontal networks of connected friends and by liking and sharing digital artefacts; and to contribute through everyday participatory forms of digital making, software programming, and coding.

Learning to code is a direct outgrowth of this concern with co-production, crowdsourcing and prosumption, enabling young people to become expert prosumers of social media content. Consequently, learning to code is not a neutral, decontextualized or depoliticized practice, but shaped, patterned, ordered and governed by powerfully commercialized coded infrastructures.

LEARNING TO CODE FOR *X*

There is a final hidden political dimension to learning to code and the computing curriculum too. Learning to code and digital making is closely related by key advocates such as Nesta with ideas about 'hackathons' and 'codefests' for public service design, and 'government hacking' events. 'Hack' events put teams of computer programmers together, using code-sharing tools, to engineer solutions to government and public sector problems. The voluntary prosumer is the ideal subject for a governmental context where the state is seeking to deconcentrate its responsibilities and enable more 'people-powered public services' and co-produced solutions facilitated by 'people helping people,' as Nesta documents describe it. These Nesta documents describe projects such as 'local government digital making,' 'civic tech' and 'coding for civic service' that involve a mixture of coding skills, design skills, and user experience to explore 'solutions to challenges'—merging 'what is (technically) possible and what is (politically) feasible.'

These projects apply what Nesta terms a 'code for x model' where it appears that computer code can be applied as a solution for almost any problem. It rests on the assumption that the problems with the social world can be addressed with solutions written in code. This is about applying technical engineering to the task of human, social and political engineering, and represents the embedding of computational thinking—the expression of problems in the language that computers can understand—in the main style of contemporary governmental thought.

According to internet critic Evgeny Morozov (2013) this kind of 'solutionist' thinking originates in the Silicon Valley hacker culture of technological innovation. Such thinking recasts complex social phenomena like politics, public health, and education as neatly defined problems with definite, computable solutions that can be optimized if the right code and algorithms are available. Here we find social phenomena translated into computational models that can be operated upon by algorithmic procedures—the perfect technical fix for an increasingly 'solutionist state' that wants to promote a new generation of coders to fix its problems on its behalf.

The overall digital making, learning to code and hacking discourse is embedded in this emerging mode of what I see as 'political computational thinking.' Through various advocacy coalitions, campaigns, lobbying groups and networks of likeminded organizations, learning to code has been positioned as equipping young people with the computational skills required to become solutions-engineers and hackers of the future. The current preoccupation with children learning to code is reflected in how government is also learning to code in order to apply computational thinking and procedural algorithmic solutions to 'hack' public and social problems.

LEARNING TO REPOLITICIZE CODE

In conclusion it can be argued that learning to code is a kind of introduction into new computational ways of interacting with the world, as channelled through the culture of code of software development and the disciplinary systems of thought associated with programmers. Such practices are intended, at least in part, to prepare them for a world in which computational thinking and coding practices are seen as potential solutions to all of today's economic, commercial and political problems. It is important to acknowledge that learning to code, digital making and the computing curriculum are attached to these other political activities and ways of thinking, rather than simply to see practices of coding and computational thinking as decontextualized and depoliticized sets of technical skills.

This paper has sought to put such innovations in their necessary historical context so as not to take them for granted, and to begin tracing the assumptions on which they've been built, the cultures of coding in which they are embedded, and the longer lines of thinking that have made them seem like the correct solution to contemporary problems. As the culture of code associated with learning to code and digital making now enters school through the computing curriculum—supported by the massive digital and broadcast infrastructures of the BBC, Microsoft, Google and the like, as well as by various think tanks, policy labs and other cross-sectoral intermediaries—we need to be aware of how those assumptions, values and practices are now being promoted in schools, and inserted into the everyday practices and 'computational thinking' of young people themselves. It will shape their ways of engaging with computers, and in turn, will influence the ways they go about shaping computers towards particular goals in the future. In the sense that the production of code can have seriously productive consequences, learning to code is a political act but it is being shaped, delimited and constrained by existing disciplinary assumptions, professional cultures, and governmental systems of thinking.

Beer, D. & Burrows, R. 2013. Popular culture, digital archives and the new social life of data. *Theory, Culture and Society* 30, no. 4: 47-71.

Cave, T. & Rowell, A. 2014. *A Quiet Word: Lobbying, Crony Capitalism, and Broken Politics in Britain.* London: Bodley Head.

Frey, C. D. & Osborne, M.A. 2013. The future of employment: How susceptible are jobs to computerisation? University of Oxford: http://www.oxfordmartin.ox.ac.uk/downloads/academic/The_Future_of_Employment.pdf

Hayes, B. 2015. Cultures of code. *American Scientist* 103, no. 1: http://www.americanscientist.org/issues/pub/cultures-of-code

Kitchin, R. & Dodge, M. 2011. *Code/Space: Software and Everyday Life*. London: MIT Press.

Kitchin, R. 2014a. Big Data, new epistemologies and paradigm shifts. *Big Data & Society* 1, no.1: http://dx.doi.org/10.1177/2053951714528481

Mackenzie, A. 2006. *Cutting Code: Software and Sociality*. Oxford: Peter Lang.

Mager, A. 2012. Algorithmic ideology: How capitalist society shapes search engines. *Information, Communication & Society* 15, no. 5: 769-787.

Morozov, E. 2013. The perils of perfection. *New York Times*, Sunday Review, 2 March: http://www.nytimes.com/2013/03/03/opinion/sunday/the-perils-of-perfection.html

Morozov, E. 2014. Making It. *The New Yorker*, 13 January 2014: http://www.newyorker.com/magazine/2014/01/13/making-it-2

Rushkoff, D. 2010. *Program or Be Programmed: ten commands for a digital age*. ORbooks.

Wakefield, J. 2014. Does a five year-old need to learn to code? BBC News, 26 September: http://www.bbc.co.uk/news/technology-29145904

Williamson, B. 2015. Political computational thinking: policy networks, digital governance, and 'learning to code.' *Critical Policy Studies*: http://dx.doi.org/10.1080/19460171.2015.1052003

THE UNIVERSITY
of EDINBURGH

1965 - 2015
50
E·S·R·C
50 YEARS SHAPING SOCIETY

UNIVERSITY *of*
STIRLING