Machine Learning, Knowledge Risk, and Principal-agent Problems in Automated Trading

Borch, Christian

Document Version
Final published version

Published in:
Technology in Society

DOI:
10.1016/j.techsoc.2021.101852

Publication date:
2022

Technology in Society 68 (2022) 101852

Contents lists available at ScienceDirect

Technology in Society

journal homepage: www.elsevier.com/locate/techsoc

Machine learning, knowledge risk, and principal-agent problems in automated trading

Christian Borch

Department of Management, Politics and Philosophy, Copenhagen Business School, Porcelaenshaven 18A, DK-2000, Frederiksberg, Denmark

ARTICLE INFO

Keywords:
Automated trading
Financial markets
Knowledge risk
Machine learning
Principal-agent problems

ABSTRACT

Present-day securities trading is dominated by fully automated algorithms. These algorithmic systems are characterized by particular forms of knowledge risk (adverse effects relating to the use or absence of certain forms of knowledge) and principal-agent problems (goal conflicts and information asymmetries arising from the delegation of decision-making authority). Where automated trading systems used to be based on human-defined rules, increasingly, machine-learning (ML) techniques are being adopted to produce machine-generated strategies. Drawing on 213 interviews with market participants involved in automated trading, this study compares the forms of knowledge risk and principal-agent relations characterizing both human-defined and ML-based automated trading systems. It demonstrates that certain forms of ML-based automated trading lead to a change in knowledge risks, particularly concerning dramatically changing market settings, and that they are characterized by a lack of insight into how and why trading rules are being produced by the ML systems. This not only intensifies but also reconfigures principal-agent problems in financial markets.

Introduction

On May 6th, 2010, the US financial markets experienced what has become known as a major "Flash Crash": within a few minutes the interaction of fully automated trading algorithms produced massive losses and left some market participants with the sense of the markets disappearing [1,59]. It has been argued that the Flash Crash was the "first generalized crisis" of automated trading [2], and it certainly triggered a broader discussion of the risks presented by new algorithmic technologies and how to possibly address them through new types of market regulation [3]. To address this new reality where, in many markets, most of the volume is traded by fully automated algorithms [4], the EU introduced a new legislative framework, the "Markets in Financial Instruments Directive (recast)" (MiFID II), which strengthened the regulation of automated trading, including forcing upon trading firms stronger internal testing procedures. Importantly, however, when MiFID II took effect in 2018, the algorithmic systems of some trading firms had already superseded the conception of algorithmic trading underpinning the regulatory framework: Where MiFID II's underlying frame of reference is human-defined automated systems of the kind involved in the Flash Crash—that is, systems defined end-to-end by humans and whose trading rules are an implementation of human-conceived ideas—-an increasing number of firms are beginning to deploy machinelearning (ML) techniques (ML is a branch of artificial intelligence (AI)), where the idea is that the automated ML trading system develops its trading rules independently [5–7].

This study argues that the shift from human-conceived to ML-based trading is in need of scrutiny, as it transforms automated trading in significant ways. Specifically, it changes the knowledge risks and principal-agent problems automated trading firms are facing. Where knowledge risks refer to any adverse effects relating to the use or absence of particular forms of knowledge, principal-agent problems concern the goal conflicts and costs that may arise once decision-making authority is delegated. Principal-agent problems might be seen as a subset of knowledge risk, concerning the specific problem that the delegation of decision-making authority can generate information asymmetries and moral hazard, leading the agent to take too much risk or engage in malfeasant action because it does not bear the costs of any failures.

Knowledge risks and principal-agent problems concerning ML systems obviously have salience beyond the field of automated trading, including management [8], manufacturing [9], and medicine [10], and this paper's analyses may therefore help to pinpoint knowledge risks and principal-agent problems in other ML application domains. That said, knowledge risks and principal-agent problems are particularly relevant to discuss in the context of automated trading. Not only are automated

E-mail address: cbo.mpp@cbs.dk.

https://doi.org/10.1016/j.techsoc.2021.101852
Received 5 October 2021; Received in revised form 9 December 2021; Accepted 30 December 2021
Available online 4 January 2022

trading firms confronted by both—and addressing them insufficiently can lead to overly perilous behaviors on both firm and market levels—but the rise of ML also entails a transformation of them and, therefore, the measures needed to deal with them.

To understand the specific knowledge risks and principal-agent problems that pertain to ML-based automated trading, it is helpful to compare them with those associated with non-ML, human-defined automated trading. To this end, this study draws upon fieldwork in the automated trading industry, including a comprehensive pool of interviews with market participants who design and deploy such trading systems (covering both human-defined and ML-based systems). Mobilizing these interview data, this study teases out how ML-based trading differs from human-defined automated trading. Although both types of algorithmic systems are rules-based, they are so in different ways and with different associated knowledge risks. In human-defined trading systems, the rules informing and constituting the trading strategies are conceived by human traders and then translated into code by software developers but without the former necessarily knowing precisely whether the translation is correctly conducted. Furthermore, many trading firms deliberately make traders compete with one another and limit their access to the firm's collated strategies, meaning that individual teams of traders might unknowingly develop algorithmic strategies that interfere with others', potentially putting the firm at risk. In much ML-based trading, the rules constituting the trading strategies are internally generated by the ML system itself. However, in sophisticated ML architectures such as deep neural networks, humans struggle to understand how the system extracts patterns from data to come up with its predictions. In other words, despite ongoing work seeking to enhance explainability, for these kinds of opaque ML systems, humans have dramatically limited knowledge of how and why their systems arrive at their specific decisions—with potentially high risk.

As for principal-agent problems, this study argues that human-defined automated trading systems share with pre-automated securities trading those types of principal-agent problems that revolve around brokers' and fund managers' (agents) fiduciary duties toward their clients (principals), or those concerning managers (principals) and staff (agents) within organizations. However, automated trading also prompts the question of whether conventional principal-agent relationships apply when the emphasis is on the relationship between humans and algorithms. Can the humans who conceive and design the algorithms be seen as principals and the automated trading systems as agents? This study discusses this question in relation to both types of automated systems, arguing that, for human-defined automated trading systems, in theory, algorithms can be seen as merely extending human principal-agent problems. In this sense, they can be analyzed within a standard human-oriented principal-agent framework. However, in practice, trading firms are often organized in ways that generate information asymmetries to the disadvantage of the humans who defined the algorithmic trading rules. In ML-based trading, the situation is fundamentally different. Here, the ML systems are more like proper agents but in ways that conventional solutions to principal-agent problems cannot easily address. For example, given the opacity of these systems, better designed contracts between the principal and agent or more deterministically construed relationships between them offer no viable solutions.

Automated trading has attracted particular attention within financial economics (e.g. [1]), but its workings are also being surveyed more broadly [11]. This study contributes to two types of literature, in particular. One is sociological discussions about automated trading. Previous sociological work has focused primary attention on human-defined automated trading systems in the form of so-called high-frequency trading (HFT), that is, automated trading systems operating at extreme speed. Notably, MacKenzie has detailed how HFT relies on complex material infrastructures such as high-speed fiber-optic and microwave data transmission systems [4,12]. Other scholars have studied regulatory aspects of automated trading [3]; how securities exchanges adapt to automation [13]; the ways in which the organizational design of HFT firms both deliberately and unintentionally fosters knowledge conflicts and knowledge limitations with associated risks [14–18]; as well as how flash crashes might be immanent to automated trading [59–61]. Common to this sociological literature is that it analyzes automated trading empirically by focusing on the practical considerations and concerns of market professionals when they design, develop, and deploy automated trading systems. This study continues the empirical interest in the practical aspects of automated trading. However, most of the existing sociological research revolves around human-defined trading algorithms and only a few studies have examined the uptake of ML-based trading systems. The main exception is Hansen [19,20] who demonstrates that many traders are reluctant to deploy sophisticated ML systems characterized by opacity and that these traders often prefer simpler models (see also [59]). Opaque ML models are nonetheless gaining traction in automated trading [5] which raises hitherto unexplored questions concerning their knowledge risks and principal-agent implications. By addressing the latter, this study also extends recent attempts to re-think principal-agent theory considering AI and ML [21]. In doing so, the paper's analysis combines empirical sociological examinations of automated trading with insights from organization and management studies.

In addition, this study contributes to sociological and anthropological work that attends to the practical contexts in which ML systems are being deployed [22,23]. The proprietary nature of many ML systems makes access to carry out ethnographic fieldwork on them difficult—an obstacle that is especially weighty in financial markets since many market professionals are reluctant to being interviewed and, even more so, to being studied in observational work. This suggests that social science examinations of ML and AI systems may need to assume a pragmatic approach and suffice with fragmented observations compiled in different fields. To put it with Hannerz [24], "ethnography is an art of the possible, and it may be better to have some of it than none at all" (p. 213). Given this, the aim of this study is not to provide a novel discussion of how to approach ML systems ethnographically, as conventional methods continue to have something to offer. However, the study does provide new insights into how market participants conceive of their automated trading systems—human-defined and ML-based alike—in the hope that this will add productively to the pool of data jointly collated by social science researchers in their quest to understand better contemporary ML and AI systems, their deployment, workings, contexts, and implications.

The paper is structured as follows. First, it offer a conceptual discussion of knowledge risk and principal-agent problems. Next, it presents the methods and data. Then, it analyzes human-defined and ML-based trading in turn, examining each considering their knowledge risks and principal-agent problems. The conclusion summarizes the argument and findings.

2. Knowledge risk and principal-agent problems

Instead of seeing knowledge as something inherently positive, the notion of knowledge risk serves to draw attention to the potentially negative aspects of knowledge or insufficient knowledge management in organizations. Knowledge might leak; it might be misleading, overly limited, or based on poor foundations; it might not be properly shared; it might rely on external sources over which the organization har little control; or it might be absent. Highlighting such issues and synthesizing a larger body of literature on knowledge risk management, Durst and Zieba [25] define knowledge risk as "a measure of the probability and severity of adverse effects of any activities engaging or related somehow to knowledge that can affect the functioning of an organisation on any level" (p. 2; see also [26]). There are many aspects of securities trading that can be analyzed in the light of knowledge risk. For example, given that the future is unknown, any market prediction might be wrong, potentially generating substantial losses—despite the amount and quality of knowledge on which it is based.

That said, and of particular importance to this paper, the precise manifestations of such types of knowledge risk will likely depend on the technologies deployed in different market settings. So, although classical inter-human trading (where human traders interacted directly with each other) was supported by certain technologies—for example, the stock ticker and the telegraph played important roles in the late nineteenth century [27,28]—more heavily technologically inculcated forms of securities trading involve new knowledge risk stemming from how technology makes certain types of knowledge available and disregards others. In addition, there is a more inter-organizational—even systemic—side to increasing technologization, as the electronification of markets has pulled a larger number of market participants closer to one another, meaning that market fluctuations are likely to be exacerbated faster and more intensely [29,30,59,61]. This risk may be further intensified if rival firms are deploying financial models based on the same or similar forms of knowledge: if trading decisions are based on overall similar assumptions or expectations, their "resonance" may reinforce sudden ruptures [31–34].

As mentioned earlier, principal-agent problems might be seen as a particular instantiation of knowledge risk, as they concern how the delegation of decision-making authority might create problematic information asymmetries. The central tenets of principal-agent theory are analyzed in Jensen and Meckling's [35] classical discussion of the ownership structure of firms. They "define an agency relationship as a contract under which one or more persons (the principal(s)) engage another person (the agent) to perform some service on their behalf which involves delegating some decision making authority to the agent" (p. 308). Although Jensen and Meckling's analysis pays particular attention to the relationship between the owners (shareholders as principals) and top management (agent) of a corporation, it is applicable to a wider range of situations involving cooperation and delegation. In the context of securities trading, for example, it captures the relationship between a broker (agent) who is acting on behalf of some investor (principal). Where a broker is obliged to execute the principal's orders in the best possible way, rather than exploit information asymmetries to their own advantage, history is replete with concerns about whether these fiduciary duties have, in fact, been met [36]. Similarly, within trading firms, principal-agent problems can be identified when traders misuse their positions to make overly risky, unauthorized, or fraudulent trades—whether this be in the form of fund managers acting in the disinterest of external investor principals [37] or rogue traders who act far beyond the risk limits set by their management principals.

Principal-agent theory rests on the assumption that both principal and agent engage in "maximizing behavior," for example, by maximizing profit ([35], p. 307). This, combined with delegation, is precisely what generates the goal conflicts and the potential losses incurred by the principal: as the agent is focused on maximizing their own behavior, they might not be acting in ways that benefit the principal. In recognition of this, the principal might take steps to monitor the agent's action or incentivize agent behaviors more aligned with the principal's interest—thereby also addressing problems arising from information asymmetries between the parties. In the words of Jensen and Meckling:

> The *principal* can limit divergences from his [sic] interest by establishing appropriate incentives for the agent and by incurring monitoring costs designed to limit the aberrant activities of the agent. In addition in some situations it will pay the *agent* to expend resources (bonding costs) to guarantee that he [sic] will not take certain actions which would harm the principal or to ensure that the principal will be compensated if he does take such actions. However, it is generally impossible for the principal or the agent at zero cost to ensure that the agent will make optimal decisions from the principal's viewpoint. ([35], p. 308, original emphasis).

It has been argued that the goal conflict between principal and agent might be (partly) resolved if their contractual relationship is defined such that their interests are better aligned [38]. Similarly, it has been proposed that principal-agent problems are particularly prominent under certain conditions and that they might be alleviated through a pre-specification of agent tasks. For example, Eisenhardt ([38], p. 62) suggests that a high degree of task programmability—defined as "the degree to which appropriate behavior by the agent can be specified in advance"—makes the "agent's behavior […] more readily determined" and therefore also "easier to observe and evaluate."

Underlying these kinds of discussions is another assumption of conventional principal-agent theory, namely that principal-agent relationships involve "two or more people" ([35], p. 309). In other words, classical principal-agent theory concerns *inter-human* relationships. But what about situations as in automated trading where trading decisions are not delegated to humans but are made instead by machines? Can human-machine relationships be seen as principal-agent relationships?

In one of the few existing attempts to answer these questions, Bostrom ([56], pp. 155–157) differentiates between what he calls the "first principal-agent problem" and the "second principal-agent problem." The former refers to situations where a human appoints another human to act on their behalf and thus encompasses classical principal-agent relationships as described above. The second principal-agent problem is one where the agent is an AI/ML system (what Bostrom terms "superintelligent" systems), and where the agent's behavior and decision-making logic are not a result of human instructions. In this case, the agent is not a mere human proxy. Bostrom argues that addressing the second principal-agent problem requires new techniques that go considerably beyond those suggested for first principal-agent problems. Specifically, he proposes to wrap AI/ML technologies in various control systems.

Kim [21] has taken Bostrom's discussion several steps further by aligning it with insights from the field of new materialism which grants algorithms agency instead of seeing them as, for example, representations of political interests. Similar to Bostrom, Kim argues that the use of algorithmic systems, including deep-learning-based ones, can be studied from two principal-agent perspectives. One concerns the relationship between the data scientist(s) developing and deploying the algorithmic systems and their manager(s), where the former can be seen as agent(s) and the latter as their principal(s). Where this corresponds to Bostrom's first principal-agent relationship, the theoretically more challenging situation is where algorithmic systems assume independent agency and are thus, in effect, constituted as agents vis-à-vis the human principals who designed their algorithmic architecture, curated the data, and in other ways enabled their agency. Kim's central achievement is to provide theoretical backing for identifying classical principal-agent issues such as information asymmetry and malfeasance in the context of deep-learning systems as well as to discuss these in the light of algorithmic governance. Specifically, he suggests that the control wrappings proposed by Bostrom need to be supplemented with an emphasis on only using ML systems in an incremental and precautionary manner and comparing the results of various ML systems. Where Kim's discussion is mainly theoretical, this paper offers an empirical analysis of knowledge risk and principal-agent relationships as they pertain to different kinds of automated trading systems.

3. Methods and data

The paper's analysis of human-defined and ML-based automated trading is based on interviews and ethnographic observations. From 2014 to 2020, colleagues (Kristian Bondo Hansen, Pankaj Kumar, Ann-Christina Lange, Bo Hee Min, Nicholas Skar-Gislinge, and Daniel Souleles) and the author conducted 213 interviews with market participants involved in automated trading. Out of these, the author conducted 70 interviews individually or jointly. Informants were mainly working in Chicago, New York, London, or Amsterdam and represented a wide range of a total of 146 institutions, including proprietary trading firms, banks, hedge funds, institutional investors, brokers, exchanges,

regulators, and data and technology providers. Interviews typically lasted approximately 1 h, with some being significantly longer. Most of the interviews were with single informants, but, occasionally, two or more informants were interviewed at once. Interviews typically focused on informants' backgrounds and daily work, including the kinds of knowledge and expertise they employ; how automated trading systems are designed; what concerns informants have regarding automated trading and its risks; how trading firms are internally organized; and so on. All participants gave their informed consent.

Most of the informants directly involved in automated trading (as, for example, traders, developers, or executives in trading firms) focused on human-defined automated trading systems. Of the 213 interviews, 94 were with informants working with ML-based trading systems. These include informants from firms that use ML in addition to human-defined trading systems as well as from firms that exclusively trade using ML systems. This study focuses particularly on the latter group. Specifically, interviews with three firms are drawn upon whose entire trading operations revolve around sophisticated ML systems. These firms are Clark Investment (pseudonym), Launch Capital Markets (pseudonym), and Tyler Capital Limited. Clark Investment (approximately 100 staff) and Launch Capital Markets (14 staff) are US-based hedge funds whose ML architectures are built upon genetic programming (to be explained below). Tyler Capital (approximately 50 staff) is a proprietary trading firm based in London and its trading activities are centered around a deep neutral network-based trading system that is active on several markets around the globe and is progressively being deployed to trade on new markets.

The author personally interviewed informants from all three firms and the discussion in this paper is based on repeat interviews with people from each firm. These three firms were selected because their core management teams have long-term hands-on experience with ML systems in finance and beyond and, therefore, have relevant expertise when it comes to assessing the key facets of ML. To further understand the role and consequences of ML-based trading, this study also draws upon ethnographic observations at Tyler Capital conducted jointly with Bo Hee Min. From 2017 to 2019, we paid the firm three ethnographic visits, each lasting a couple of days, in which we were allowed to follow all parts of the organization. A planned visit in spring 2020 was converted to online interviews due to COVID-19 lockdown measures. During our visits, we spent time with traders and ML/data engineers but also had several meetings with management including the Chief Technology Officer (CTO) who is the mastermind behind the firm's ML system. In addition to the ethnographic observations, we interviewed 18 firm employees and were granted access to more than 300 pages of internal documents which describe various parts of the ML system, procedures for dealing with risk, financial performance, and much more. These sources—internally from Tyler Capital and across the larger pool of interviews—were triangulated to identify aspects of ML-based trading that are common across firms. That said, there is a variety of ML architectures available (including some not addressed here), and the study does not claim to be generalizable beyond the kinds of systems analyzed in this paper. Note also that the applications of particular ML techniques in finance might differ from the use of those same techniques in other domains.

4. Human-defined automated trading

The quest for market automation has a long history, dating back to the 1970s [13]. However, it was only in the early 2000s that fully automated trading really took off [4]. Its first *human-defined* phase is characterized by automated trading systems that enact strategies which are defined end-to-end by humans. Specifically, these algorithmic trading systems implement human-defined *rules*. These rules can take the form of highly elaborate instructions which might follow a crude scheme such as: "if C conditions are in place, place O_1 orders to buy/sell N_1 number of S_1 securities at P_1 price at T_1 trading venue, while simultaneously hedging that position by placing O_2 orders to sell/buy N_2 number of S_2 securities at P_2 price at T_2 trading venue." Whether the conditions to send certain orders are in place is something the algorithmic system determines on the basis of a flow of continuously incoming *data*, typically in the form of the orders stacked in exchanges' "electronic order books," which list any pending orders to buy or sell a security at any given time, thereby providing an immediate sense of where the market is [12,13].

Rules and data generate particular knowledge risks for human-defined automated trading firms. Two knowledge risks are especially important. One is rules-related and concerns the epistemic obstacles and incongruences among human staff when it comes to implementing rules as well as staff's—sometimes deliberately organized—inability to comprehend how rules might co-exist in the algorithmic "black box" which collates all the individual strategies of an individual firm. Where the rules-related knowledge risks arise from intra-organizational structures, "data-related" knowledge risk concerns the external data on which trading firms rely.

4.1. Rules-related knowledge risk

As Lange [15] observes on the basis of her ethnographic fieldwork in the HFT industry, the development of human-defined automated trading systems usually takes place as a team effort. This observation is corroborated by the data informing the present study. A trader with quantitative proficiency would come up with a suggestion for a trading strategy which he—or, much more rarely, she (this is a heavily male-dominated industry)—would express mathematically. A developer with programming expertise would then translate the strategy into code. Before being made live, the algorithmic strategy would be tested against historical data and its performance subsequently further tested in a simulation environment containing current market data.

There are variations to this basic team structure, and much depends on firm size and capacity for implementing a division of labor. For example, a smaller automated trading firm visited by the author had inserted a layer between traders and developers to ease the translation of human-defined rules into code. However, it is generally the case in small firms that the trader and developer functions are less divided and more likely to be fulfilled by the same person. In larger firms, teams may consist of more traders and developers just like additional functions may be included. For example, Seyfert ([18], p. 263) notes that sometimes traders' strategies may be conceived in collaboration with "quantitative researchers, usually mathematicians who create models to predict movements in the financial market." Regardless of their specific mode of organizing, potential knowledge risks in the form of misunderstandings and epistemic misalignments are inherent in this type of team collaboration. Seyfert captures this concisely as a matter of epistemic conflicts between dissimilar "codes" or "numerical languages":

> Conversions and translations of numerical languages introduce mistranslations and misunderstandings precisely because these actors do not perfectly understand one another. As an example, a trader who invents a quantitative trading strategy might not be fully qualified to check its translation into a mathematical code by a financial engineer, who in turn is not fully able to guarantee that the translation into a trading algorithm by a developer has been performed flawlessly. Thus, bugs are not simple technical mistakes but might arise from "conflicting codes." ([18], p. 263).

Collaboration across epistemic regimes is obviously not unique to human-defined automated trading [39]. However, given that the human traders or quants are not trading directly in markets but indirectly through their algorithms, these epistemic incongruences can be consequential. Thus, the translation issue noted by Seyfert means that, at a fundamental level, the design of human-defined automated trading systems may not turn out as intended. Lacking a rigorous control

environment which can identify and rectify any minor changes that may occur along the chain of translation, the trading system may therefore behave differently than expected.

This knowledge risk is exacerbated by two intra-organizational features often found in automated trading firms. First, in many firms, traders are competing internally against each other without sharing their strategies. In the words of the CEO of Tyler Capital, "from an organizational design perspective, [many firms] are Balkanized portfolios of individual self-interest." Second, and partly related, firms protect their trading strategies vigorously and not even their own traders and developers may be granted access to the entire source code of the black box [15]. To illustrate, one of my informants, a senior quant analyst and developer who had been part of setting up a new HFT hedge fund, described how its staff would work compartmentally, each developing their separate elements of the black box system and each working on encrypted machines. The main purpose for this and for only giving traders and developers fragmented views of the black box is to prevent people from taking the firm's strategies to competitors: "You have to think about the safety, people stealing things. So how do you split the work? What access do you give to some people? Who's allowed on the production machine?" (senior quant). The downside of this compartmentalized approach is that rules-related knowledge risk is greatly enhanced since new rules might work against existing ones and a global overview of the complex system resides with just a few people who may not even engage with the system on a daily hands-on basis.

4.2. Data-related knowledge risk

In addition to the rules conceived and defined by human traders, automated trading systems are critically dependent on market data, which are used for four main purposes. First, they comprise the universe human quantitative traders and analysts parse when searching for new trading opportunities. Second, they are used when testing and simulating new strategies and their profitability and viability. Third, once automated strategies are running, market data inform the algorithms about the current market developments that these algorithms are designed to exploit. Such data form the essential input to the "if" part of algorithmic strategies. Fourth, in the US, quotes and trades data for any equities listed on national exchanges are aggregated in official "consolidated" feeds. Such consolidated feeds are much slower than the direct exchange data feeds and therefore are unusable for trading itself. However, HFT firms use them "as a data-integrity check on the faster direct feeds that they purchase from exchanges" ([4], p. 229). If the direct exchange data feeds and the slower consolidated feeds are inconsistent, this may be a concern for trading firms since it suggests that their trading systems may be receiving incorrect data from one or more sources and this may seriously jeopardize their strategies [62].

Since data are critical on several levels, firms specializing in human-defined automated trading are exposed to multiple types of data-related knowledge risks, that is, operative risks caused by any issues with the data on which the trading systems rely. These risks vary with the type of data dependence mentioned above and may be broadly differentiated into two forms concerning data quality and connectivity. Given the first two data dependencies listed earlier, poor data quality can negatively affect the strategies traders may come up with and the testing of them. In other words, the knowledge human quantitative traders and analysts seek to obtain from market data will suffer from any data imperfections and the same applies to the data-based testing of strategies. Seyfert [18] correctly observes that good data hygiene is therefore crucial. Trading firms spend substantial resources obtaining market data from data vendors, but they sometimes struggle with the quality of the data they receive and put considerable efforts into cleaning data—which, itself, generates the risk of inadvertently removing important data points ([18], pp. 266–7).

Risk exposure is even greater for the third and fourth data dependencies, since these concern the real-time data to which the automated trading systems respond. Given that these data are obtained through external sources and relayed to trading firms via technological infrastructures such as fiber-optic and micro-wave data transmissions [12], any data incidents and outages in these infrastructures can critically affect human-defined automated trading systems [62]. In other words, firms are exposed to a form of data risk that revolves around connectivity.

What is common to all four data dependencies is that poor or absent data negatively affects the market knowledge informing the trading rules and concerning the market settings to which the algorithms respond. Both these knowledge issues generate risks relating to implementing unfavorable strategies and being unable to respond to potential market havoc quickly.

4.3. Principal-agent problems

The rise of human-defined automated trading both replicates principal-agent problems known from earlier forms of (inter-human) trading and introduces new ones. For example, when a broker (agent) is executing an order on behalf of a client (principal) today, they would do so through tailored execution algorithms which, given the fiduciary responsibility brokers have, must be designed to give the client the best possible result. However, many contemporary brokers also act as dealers and operate so-called "dark pools," which are algorithmic trading platforms where, contrary to "lit" exchanges, market participants cannot see each other's orders [40]. This multiplication of roles creates imminent principal-agent problems. As Mattli summarizes,

> broker-dealers may be tempted to send all client orders first to their own dark pools [earning fees for this] even though these orders may receive better execution if routed to other trading venues. Or a dark pool operator could use confidential client trading information to trade ahead of clients. The operator could even sell confidential trading information to predatory high-speed traders lurking in the dark pool, just waiting for a propitious moment to step ahead of institutional order flow. ([41], p. 127)

These are not mere hypothetical risks. Mattli [41] lists several cases where major banks and broker-dealers have taken undue advantage of incoming order flow to the detriment of their clients.

In addition to such technologically transformed variants of previous principal-agent problems, human-defined automated trading also raises principal-agent discussions of a rather different kind. Particularly, this form of trading invites reflections on the relationship between the human(s) who define the algorithmic rules and the automated trading system that implements them. On one level, it may be argued that this relationship is not, in fact, a principal-agent one. Where the human quants may be agents in relation to their management principals, the algorithm itself does not pursue an independent goal, nor does it make sense to talk about a contractual relationship between human traders and their algorithms. The algorithm is a mere human proxy that enacts human-defined instructions. Furthermore, even if the trading algorithm were seen as an agent, the principal-agent relationship between trader (principal) and algorithm (agent) is arguably characterized by a high degree of programmability which, following Eisenhardt [38] diminishes, or even eliminates, the principal-agent problem. In other words, to the extent that the algorithm in this form of trading is an agent, its pure enactment of human-defined rules would entail complete programmability and thus make the principal-agent problem vanish.

In theory, therefore, the rise of automated trading does not create a further multiplication of technological principal-agent problems. Adding algorithms to the equation merely amounts to *extending human* principal-agent problems. In practice, however, things are more complicated. As the discussion of rules-related knowledge risk makes plain, there are several organizational features of trading firms that cause information asymmetries to proliferate. Epistemic incongruences,

a competitive culture, and compartmentalized access to the black box system all demonstrate that human-defined automated trading may be characterized by a low degree of programmability.

Importantly, moreover, the degree of programmability does not affect liability. When a human trader (principal) develops a trading strategy which is then implemented in a hand-coded algorithm (agent), the human principal remains responsible for its action. Dignum puts it in the following way:

> When a person delegates some task to an agent, be it artificial or human, the result of that task is still the responsibility of the delegating person (principal), who is the one who will be liable if things do not go as expected. The agent, however, must be able to give a report on how the task was executed and to explain eventual problems with this execution. ([58], p. 219)

Since a human-defined algorithm cannot give a report on its action, the human principal carries that responsibility as well, that is, the human must be able to present the underlying source code and its rationale. For a larger human-defined black box system, management would constitute the principal responsible for explaining how the various algorithmic strategies work together. Since this might not be easy for complex black box systems, requests have been put forth that financial regulators should be granted access to the source codes of trading algorithms [3,42]. This idea follows more general calls for greater algorithmic transparency, also beyond securities trading. For example, Citron and Pasquale [43] argue that since the decisions of credit-scoring algorithms have significant implications for individuals' lives, and since these decisions may be unfavorably biased against certain groups of people, it is important for the public to know the source codes of these algorithms. At the very least Pasquale ([44], p. 141) suggests, the source code should be made available to "some trusted auditor." However, Seyfert argues that such calls are, at best, only partially helpful when it comes to human-defined automated trading systems:

> A demand to reveal the inner workings of algorithmic codes is formulated from a very particular perspective that assumes that operations are in fact operating from inside the [black] box. For trading firms, algorithms are relational entities and can only be [understood] in context and in relation to other algorithms. […] Thus, attempts to understand the complexity of HFT cannot consist in revealing as much code as possible. Rather, it must consist of looking at the way code operates and interacts in "real" life. ([18], p. 273)

It is true that human-defined trading algorithms are typically designed to take other algorithms into account, and that their interaction patterns are complex and not always foreseeable [45,59–61]. Although it might be argued that, without a proper understanding of individual automated trading systems, their interaction patterns will be even less intelligible, the central implication of Seyfert's observation for the present discussion is another one: it might be extremely difficult to establish the extent to which a trading algorithm engages in behaviors that are consistent with the rules defined for it—and whether a principal-agent problem, therefore, can be said to exist. It is conceivable, for example, that a trading algorithm might be making profits for the firm deploying it without this being a direct consequence of its human-defined rules. In other words, due to epistemic incongruences, intricate black box systems, and the complex interaction patterns in markets—of which human traders have only limited purview—an algorithm might fortuitously be making money for the "wrong" reasons, without this implying self-interested agent behavior that runs counter to the principal's instructions.

5. ML-based automated trading

ML-based securities trading has gained increasing momentum since the mid-2010s, reflecting the proliferation of ML technologies in society more broadly (e.g., [46,47]). ML techniques come in many shapes and forms and comprise vastly different degrees of complexity, stretching from easily interpretable decision-tree models to difficult-to-explain deep-learning techniques. Common to the ML trading systems analyzed in this study is that like their human-defined predecessors, they are rules-based and data-driven—but in ways that differ markedly from their forerunners. The central dissimilarity between the two forms of automated trading is that, in the ML-based one, trading rules are developed internally by the ML systems themselves, rather than by humans, based on the data they are fed. This is the case regardless of the type of ML architecture deployed. For example, in the fieldwork informing this study, interviews were conducted with firms that specialize in genetic programming. This is an evolution-inspired technique where a population of optimal trading rules is cultivated through numerous iterations of reproduction, crossover, and mutation [48]. Other firms would specialize in various deep neural network architectures. Deep neural nets consist of several layers of information-processing neurons connected through synapses: an input layer, several intermediate "hidden" layers, and an output layer. The central idea behind this architecture is to train the system such that succeeding layers extract increasingly abstract features from data and eventually come up with predictions on that basis. In other words, once given input data, the algorithm should learn to make an appropriate output prediction of some form.

Regardless of their specific architecture, the idea behind these ML-based trading systems is that humans would design the system's overall algorithmic architecture, select the data on which to train it, as well as define an objective function for it (which would be some variant of "maximize profits under given risk parameters"), but, then, it would be the task of the ML system to produce a set of rules independently—the trading strategy—that would fulfil the objective function. Illustratively, when asked whether it would be the ML system itself (rather than humans) that develops the trading rules, the CEO of Clark Investment responded:

> Yeah, yeah. Absolutely. The system itself would come up with the rules, the rules were evolved from scratch. So, when the system started, it would randomly put together some of these indicators and actions. And the ones that did a little bit better were parents of new generations. And they were all tried out on unseen data over many, many generations until we actually got these elaborate, complex rules.

Although in the firms analyzed in this study, humans would not define the rules, they might seek to influence the ML systems in various ways. For example, at Tyler Capital, staff would seek to influence its deep learning trading system by adding particular penalties and rewards, thereby hoping to coax it in certain desired directions. That said, the firm's CTO stressed that, in the end, it would be up to the ML system "entirely to come up with the trading policy. We can't do it." So, while other trading firms might have a greater level of human influence—including, in the case of genetic programming, using human knowledge to initialize the population to some known solution and then evolve the trading rules to improve them—the firms analyzed in this study take a different approach where human knowledge plays a much less significant role. Given this way their ML trading systems are designed to operate, it is possible to single out three central knowledge risk corollaries of ML-based trading.

5.1. Re-constituting relevant knowledge

First, *questions about what constitutes relevant market knowledge—the*

knowledge utilized for trading purposes—and who possesses it are reshuffled. Previously, including in human-defined automated trading, knowledge resided in human traders who would conceive a strategy and implement it. This did not preclude that individual traders might have questioned their own knowledge and tested it against that of peers [31,39]. In ML-based trading, by contrast, knowledge is a function of market data. Indeed, in the trading-related adoption of ML as in other fields, it is a truism that ML systems perform no better than the data they are fed. The basic raison d'être of ML systems is that, if fed sufficient data, they are superior to humans when it comes to identifying profitable opportunities (or "signals") in markets. In Hansen's ([19], p. 4) phrasing, these models "are technical aids firms use as a way of compensating for the limited information processing, calculative, and information storage (memory) capacity of humans."

The exceptional data-processing capacity of ML systems holds new potential that some firms are trying to seize. It was mentioned earlier that for most firms, order-book data constitute their key data pool. The granular, sub second supply and demand changes registered in the order book's listing of pending orders to sell and buy securities is the primary information used by many firms when making their own investment decisions. However, with the rise of big data and ML, new possibilities have arisen which make so-called "alternative data" potentially valuable for trading purposes [49,63]. Alternative data are non-order-book data and may include satellite imagery of warehouse parking lots or social media data which firms then deploy to predict the price of securities. For example, satellite imagery detailing an increase in unused warehouse parking lots might suggest that the warehouse's next quarterly earnings will take a dip, and particular trades may then be formed around that prediction. Similarly, sentiment analysis of tweets about a company may be used as a prediction of its shares' subsequent performance.

Regardless of whether an ML trading strategy is based on order-book or alternative data, it is thoroughly data dependent. Indeed, ML-based automated trading is exposed to types of data-related knowledge risk that are not faced to the same degree by human-defined trading algorithms: while both types of automated trading require an inflow of continuous market data feeds (information about the market conditions to which the algorithms respond), the underlying trading policies differ in their data dependence. Although poor data quality is a major problem for human-defined systems, it might be partly compensated for by human knowledge, including human traders' knowledge about prior or currently running strategies. If an automated ML-based trading system is trained on poor or corrupt data, this may manifest in the predictions and strategies it produces, with less ability for humans to capture this early on, and with potentially devastating consequences for the firm deploying such algorithms and possibly also for the market more broadly. Realizing this, firms specializing in ML-based trading are exceptionally concerned with data hygiene. In the words of the CEO of Clark Investment:

> The biggest headaches that we have had have to do with data; how well curated the data is and all that sort of stuff. Lots and lots of headaches. […] In actual fact, a good 65–70% of our time goes to the mundane.

The CTO of Launch Capital Markets echoed this, illustrating the problem the following way:

> In the finance and investment world, there's plenty of data. It can be expensive to get, but there's quite a bit of it. The real challenge is that it's very, very easy to accidentally time travel one way or another. You need to have an infrastructure that completely prevents that. And it can be very subtle. For example, many [data] vendors when they're reporting on dividends, they will—if a company has never issued a dividend—put "NaN" and not a number for the value of that dividend. As soon as that company offers a dividend in the future […], they will backfill all those "NaNs" with zeroes. Your clever little machine learning algorithm could pick out if a dividend is zero that that company's going to do well because at some point in the future they're going to offer a dividend.

To address such data issues organizationally—including the risk that the data-derived knowledge the ML system generates is fundamentally flawed or based on faulty inferences—firms such as Tyler Capital have several data engineers employed whose main role is to detect and correct, through triangulation, any omissions within and inconsistences across the data they receive from different data vendors. This demonstrates that human knowledge and input are far from absent in ML-based trading. However, rather than constituting central elements in the direct crafting of trading rules as was the case in previous forms of trading, human knowledge is now utilized for creating proper conditions—be they in terms of the overall ML architecture or granular data cleaning—on which the ML systems are then expected to excel.

5.2. Non-adaptable knowledge

Second, and another variant of the notion that ML systems perform no better than the data they are fed, *an ML-based system's knowledge—what it has learned from data—is constrained by the type and content of data on which it was trained, and it cannot suddenly jump beyond these bounds*. In the context of securities trading, this can be highly consequential. Although the entire purpose of ML is to make predictions on previously unseen data, trading algorithms based on ML architecture face limitations when confronted with market events that differ markedly from those captured by the data on which they were trained. Their trading strategies may, therefore, be highly risky if pursued in the context of such radically new market situations. The CEO of Clark Investment illustrates:

> [if] there is fundamental change [in the market], and if your system isn't able to keep up with that, then it goes stale very quickly. Your processes and your system have to be able to keep up with that. If you look at a timeframe 2003 or 2004 to 2006 or early 2007, the market is just doing the same thing. There is no volatility. If I were training something on that period of time, again deceptively I would think, "Oh, OK, it's pretty simple, it's pretty tame," but then suddenly 2007 and 2008 hits and things change drastically.

In other words, ML-based trading systems' knowledge is constrained by the specific market settings reflected in the training data. This makes them at once good and poor at adapting to changes. In human-defined trading systems, any adaptations to changing market conditions must to be hand-coded, which is both labor intensive and potentially risky given that even small perturbations may have larger effects on the overall black box system. In contrast, since ML-based trading systems learn from their actions in markets on a continuous basis—several of the firms interviewed for this study had designed their systems to update their key parameters daily and automatically—this renders them adaptable to smaller, gradual changes in markets. As long as market changes are incremental, ML systems are geared to pick up on these and quickly adapt their actions accordingly. However, if market changes are dramatic, ML systems are at risk since they may enter territory distinctly different from what is reflected in their training data without being able to build new and useful knowledge from what they have learned from their previous actions in markets. The CTO of Tyler Capital stressed this problem and the temporal adaptability disadvantage faced by otherwise high-speed ML-based systems when compared to human traders:

> Humans can adapt extremely quickly and you could literally diverge from your past within an instant. Try to make a decision, "I'm going to change from that to that." Machine learning, the way that it relies on consistency through data over long periods of time, it's very difficult for it to do that just on the flip of a coin.

There might be different ways of addressing this risk. At Tyler Capital, they introduced an organizational design where human traders help

monitor market developments and intervene if markets suddenly change (or are believed to be at the risk of changing) in ways their ML system is unable to adequately cope with. While recognizing that humans might also have a hard time navigating under radically changing conditions, the point of this organizational design was to mobilize human experience in situations where the adaptability of the ML system is considered insufficient to cope with a new reality (such as Brexit).

5.3. Not knowing what actually guides trading decisions

The third knowledge risk corollary of ML-based trading, and in many ways the most complex one, is that since an ML system is designed to be a self-learning entity, questions arise about how and what it actually learns. Translated into the field of automated trading, it is a matter of *how an ML system develops knowledge to make tradeable market predictions*. Whereas in some ML techniques, such as decision trees, it is easy to see why the system arrived at a particular prediction, in others answering the question about how ML systems learn and acquire knowledge is anything but trivial. Deep neural network architectures, in particular, are associated with opacity. Given that they often juggle millions of non-linearly linked variables, it is exceptionally difficult to understand how and why they arrive at their predictions [50].

The opacity of deep neural networks was widely recognized as a key knowledge risk by those of my informants who specialize in ML-based trading. The CEO of Clark Investment addressed the problem through a distinction between white and black box systems:

> There are two primary tracks around explainability in AI. One is if your system is inherently explainable already. We call these white box systems. Here, the substrate that is used in order to map context to actions, or map data to predictions, is itself explainable. You can look at, for example, decision trees as a very, very simple version of that. When you look at a decision tree, you can map that to an "if" statement, and it's relatively explainable. Unfortunately, when things get a little more complicated and you move on to, for example, random forests, or you move on to neural networks and deep learning-based systems, they're inherently unexplainable. They're black box, very, very difficult to explain. In those cases, the substrate itself is a black box, and so what [people] do [when seeking explainability] is try to make the system explainable after the fact. So, it's more of an interrogation and investigation, and some guesswork as to what is it that triggers the black box substrate to actually trigger. [… However, this] is not giving you a generalized explanation of its behavior. So, while there's a lot of work being done on that sort of explainability, it's difficult and not completely satisfying, and sometimes doesn't really pass muster with, for example, regulators that might be asking for explainability in their systems.

The opacity of deep neural networks introduces a level of rules-related knowledge risk that is unparalleled in the earlier human-defined automated trading systems. To be sure, the decision-making logic of human-defined systems may also be difficult to understand in practice, especially in firms with an intra-competitive culture where staff have a compartmentalized access to the overall black box system and therefore do not necessarily know how their individual strategies interact and affect its overall dynamics. Still, given that these systems consist of human-defined rules, it is, in principle, possible to disentangle their decision-making structure. Since this is vastly more challenging for deep neural network architectures, it is also more difficult to put in place adequate safety mechanisms which can prevent these algorithms from either triggering or exacerbating market crashes (a concern expressed in various ways in, e.g., [51,52]). Indeed, the lack of explainability of advanced ML systems is a widespread cause of concern among market participants, and as Hansen [19,20] notes, many hesitate to deploy ML-based trading systems precisely because their decision-making logics are not intuitively (see similarly [59]).

5.4. Principal-agent problems

This study has argued that the central difference between human-defined and ML-based trading systems is that, in the latter, rules are developed internally by the ML system rather than by humans. This difference largely maps onto the difference Bostrom identifies between first and second principal-agent problems. Although Bostrom takes for granted that AI systems introduce particular types of principal-agent problems, it is worth considering the extent to which these are genuine principal-agent problems. Doing so is particularly warranted considering the unclarity demonstrated in this respect when it comes to human-defined automated trading systems. What subverts the notion of second principal-agent problems is that AI/ML systems cannot be said to engage meaningfully in a contractual relationship with their human principals. That said, there are other dimensions pointing towards the advantage of conceiving of ML-based trading systems as part of principal-agent relationships. Most importantly, they can be seen as pursuing some degree of rational self-interest. According to Husted [53], "[r]ationality in principal-agent models consists of the assumption that agents and principals maximize their interests as defined by some objective function" (p. 180). In ML-based trading, human principals can be said to maximize an objective function (generating profit), and the same applies to the agent ML system (as mentioned earlier, this would typically be some variant of "maximize profits under given risk constraints"). Although the ML system's objective function is defined by human principals, the ways in which the objective function is fulfilled are not. This is precisely where ML systems are given latitude: they are tasked with finding the best possible way to fulfil the objective function, but how they do this, is not defined by humans. The principal-agent problem arising here is that, for deep learning systems in particular, the human principals may not know whether, despite particular risk parameters, the objective function is fulfilled in an overly risky or even illegal way. Indeed, in principle, the ML system could learn to maximize profits by trading illegally, say, by placing manipulative orders, against the human principals' intentions and knowledge.

Kim [21] argues that theories of principal-agent relationships applied to debates about algorithmic governance usually rest on "the assumption that experts know the precise working principle of these algorithms and can fully control and dictate them." (p. 6). As argued earlier, this is a problematic assumption even for human-defined trading systems. However, as Kim [21] rightly notes, "[t]hat assumption now becomes increasingly precarious, due to the opacity of deep learning algorithms." (p. 6). Not even the computer science experts who develop deep neural network-based systems may know precisely how these arrive at their predictions, which is jarring not just in a trading context but also in other fields, such as medicine, where ML systems might inform decision-making (including diagnosis and treatment). Although an entire computer science subfield known as "explainable AI" has emerged (e.g., [54,55]), which seeks to devise tools and methods with which to arrive at some degree of explainability and interpretability of especially deep neural networks, and although some important progress has been made in this field, no uniformly accepted approaches to explainability and interpretability yet exist.

One consequence of this is that the fundamental solution to the risks raised by ML-based systems—be they applied in trading, automated warfare, or elsewhere—cannot be adequately addressed through conventional principal-agent principles concerning the contractual relation between the parties [38]. Nor is programmability an option when it comes to addressing second principal-agent problems. Similarly, while wrapping ML systems in various control systems [56] or using them only in an incremental and precautionary manner [21] are undoubtedly important, these suggestions do not fully address the fundamental issue of ML systems' decision-making opacity. Equally unhelpful is the idea of giving regulators or trusted auditors access to the source code of ML systems in the hope that they would be able to detect the decision-making logics of these. As Kim [21] notes, "The disclosure of

codes alone does not automatically increase their explainability" (p. 6). Instead, what is needed is that real progress is made in "explainable AI" such that, eventually, humans will be able to understand how and why deep neural networks extract patterns in data and make predictions on that basis.

6. Conclusion

By comparing two overall forms of automated trading, this study identified knowledge risks and principal-agent relationships that look similar, yet are different, across the two. In human-defined automated trading, rules-related knowledge risk and principal-agent problems are a function of: (1) the epistemic obstacles and incongruences among differently trained human staff which render the translation and subsequent implementation of human-defined trading rules highly complex; (2) the data-dependence of the trading systems which create data quality and connectivity risks; and (3) the intra-competitive culture of many trading firms and their deliberate curbing of traders' and developers' access to the overall black box system which make human-defined automated trading replete with information asymmetries.

ML-based automated trading systems are data-dependent, too, but in more consequential ways. The knowledge they generate is derived entirely from the data they are fed, meaning not only that poor data quality and flawed inferences may have a direct negative impact on their trading policies, but also that ML-based trading systems may face severe risks when confronted with rapidly changing market settings that differ from those reflected in the training data. Most importantly, however, complex ML-based automated trading systems built on deep neural network architectures, are characterized by opacity: it is, as of yet, exceedingly difficult to understand how they arrive at their predictions and trading policies. This creates a novel type of principal-agent problem where the agent (ML system) might not act as expected by the principal (the human ML architects), and where traditional means of overcoming principal-agent problems face limitations. To solve key knowledge risks and principal-agent problems of ML-based securities trading, significant additional advances in explainable AI are therefore needed. Only when humans start to understand fully how neural nets arrive at their predictions can central challenges of this form of ML-based automated trading be addressed.

Funding

This study was supported by the funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No 725706).

Declaration of competing interest

There are no conflicts of interest to declare.

Acknowledgements

I am grateful to Bo Hee Min, Kristian Bondo Hansen, and the anonymous reviewers for their helpful comments. I also want to thank my informants for their time and help.

References

[1] Andrei Kirilenko, Albert S. Kyle, Mehrdad Samadi, Tugkan Tuzun, The flash crash: high-frequency trading in an electronic market, J. Finance 72 (3) (2017) 967–998.
[2] Donald MacKenzie, Mechanizing the Merc: the Chicago Mercantile Exchange and the rise of high-frequency trading, Technol. Cult. 56 (3) (2015) 646–675.
[3] Nathan Coombs, What is an algorithm? Financial regulation in the era of high-frequency trading, Econ. Soc. 45 (2) (2016) 278–302.
[4] Donald MacKenzie, Trading at the Speed of Light: How Ultrafast Algorithms Are Transforming Financial Markets, Princeton University Press, Princeton and Oxford, 2021.
[5] Bank of England, & The Financial Conduct Authority, Machine Learning in UK Financial Services, Bank of England and the Financial Conduct Authority, London, 2019.
[6] Tony Guida (Ed.), Big Data and Machine Learning in Quantitative Investment, Wiley, Chichester, 2019.
[7] Marcos López de Prado, Advances in Financial Machine Learning, Wiley, Hoboken, NJ, 2018.
[8] Xinlei Wang, Jianing Zhi, A machine learning-based analytical framework for employee turnover prediction, Journal of Management Analytics 8 (3) (2021) 351–370.
[9] Donghong Ding, Fengyang He, Lei Yuan, Zengxi Pan, Lei Wang, Montserrat Ros, The first step towards intelligent wire arc additive manufacturing: an automatic bead modelling system using machine learning through industrial information integration, J. Indus. Inform. Integrat. 23 (2021) 100218, https://doi.org/10.1016/j.jii.2021.100218.
[10] Kanchan Pradhan, Priyanka Chawla, Medical internet of things using machine learning algorithms for lung cancer detection, Journal of Management Analytics 7 (4) (2020) 591–623.
[11] Boming Huang, Yuxiang Huang, Li Da Xu, Lirong Zheng, Zhuo Zou, Automated trading systems statistical and machine learning methods and hardware implementation: a survey, Enterprise Inf. Syst. 13 (1) (2019) 132–144.
[12] Donald MacKenzie, Material signals: a historical sociology of high-frequency trading, Am. J. Sociol. 123 (6) (2018) 1635–1683.
[13] Juan Pablo Pardo-Guerra, Automating Finance: Infrastructures, Engineers, and the Making of Electronic Markets, Cambridge University Press, Cambridge, 2019.
[14] Armin Beverungen, Ann-Christina Lange, Cognition in high-frequency trading: the costs of consciousness and the limits of automation, Theor. Cult. Soc. 35 (6) (2018) 75–95.
[15] Ann-Christina Lange, Organizational ignorance: an ethnographic study of high-frequency trading, Econ. Soc. 45 (2) (2016) 230–250.
[16] Ann-Christina Lange, Marc Lenglet, Robert Seyfert, On studying algorithms ethnographically: making sense of objects of ignorance, Organization 26 (4) (2019) 598–617.
[17] Marc Lenglet, Conflicting codes and codings: how algorithmic trading is reshaping financial regulation, Theor. Cult. Soc. 28 (6) (2011) 44–66.
[18] Robert Seyfert, Bugs, predations or manipulations? Incompatible epistemic regimes of high-frequency trading, Econ. Soc. 45 (2) (2016) 251–277.
[19] Kristian Bondo Hansen, The virtue of simplicity: on machine learning models in algorithmic trading, Big Data Soc. 7 (1) (2020), https://doi.org/10.1177/2053951720926558.
[20] Kristian Bondo Hansen, Model talk: calculative cultures in quantitative finance, Sci. Technol. Hum. Val. 46 (3) (2021) 600–627.
[21] Eun-Sung Kim, Deep learning and principal-agent problems of algorithmic governance: the new materialism perspective, Technol. Soc. 63 (2020) 101378, https://doi.org/10.1016/j.techsoc.2020.101378.
[22] Angèle Christin, The ethnographer and the algorithm: beyond the black box, Theor. Soc. 49 (2020) 897–918.
[23] Nick Seaver, Algorithms as culture: some tactics for the ethnography of algorithmic systems, Big Data Soc. 4 (2) (2017), https://doi.org/10.1177/2053951717738104.
[24] Ulf Hannerz, Being there… And there. And there!: reflections on multi-site ethnography, Ethnography 4 (2) (2003) 201–216.
[25] Susanne Durst, Malgorzata Zieba, Mapping knowledge risks: towards a better understanding of knowledge management, Knowl. Manag. Res. Pract. 17 (1) (2019) 1–13.
[26] Peter Massingham, Knowledge risk management: a framework, J. Knowl. Manag. 14 (3) (2010) 464–485.
[27] John Handel, The Material Politics of Finance: the Ticker Tape and the London Stock Exchange, 1860s–1890s, Enterprise & Society, 2021, pp. 1–31.
[28] Alex Preda, Socio-technical agency in financial markets: the case of the stock ticker, Soc. Stud. Sci. 36 (5) (2006) 753–782.
[29] Karin Knorr Cetina, From pipes to scopes: the flow architecture of financial markets, Distinktion Scand. J. Soc. Theory (4) (2003) 7–23.
[30] Karin Knorr Cetina, Urs Bruegger, Global microstructures: the virtual societies of financial markets, Am. J. Sociol. 107 (4) (2002) 905–950.
[31] Daniel Beunza, David Stark, From dissonance to resonance: cognitive interdependence in quantitative finance, Econ. Soc. 41 (3) (2012) 383–417.
[32] Donald MacKenzie, Taylor Spears, 'A device for being able to book p&l': the organizational embedding of the Gaussian copula, Soc. Stud. Sci. 44 (3) (2014) 418–440.
[33] Donald MacKenzie, Taylor Spears, 'The formula that killed Wall Street': the Gaussian copula and modelling practices in investment banking, Soc. Stud. Sci. 44 (3) (2014) 393–417.
[34] Ekaterina Svetlova, *Financial Models and Society: Villains Or Scapegoats?* Northampton, Edward Elgar, MA, 2018.
[35] Michael C. Jensen, William H. Meckling, Theory of the firm: managerial behavior, agency costs and ownership structure, J. Financ. Econ. 3 (4) (1976) 305–360.
[36] Donald MacKenzie, Dark markets, Lond. Rev. Books 37 (11) (2015) 29–32.
[37] Diane-Laure Arjaliès, Philip Grant, Iain Hardie, Donald MacKenzie, Ekaterina Svetlova, Chains of Finance: How Investment Management Is Shaped, Oxford University Press, Oxford, 2017.
[38] Kathleen M. Eisenhardt, Agency theory: an assessment and review, Acad. Manag. Rev. 14 (1) (1989) 57–74.
[39] Leon Wansleben, Cultures of Expertise in Global Currency Markets, Routledge, London and New York, 2015.
[40] Donald MacKenzie, Market devices and structural dependency: the origins and development of 'dark pools, Finan. Soc. 5 (1) (2019) 1–19.

[41] Walter Mattli, Darkness by Design: the Hidden Power in Global Capital Markets, Princeton University Press, Princeton and Oxford, 2019.
[42] Gregory Meyer, US Regulators Propose Powers to Scrutinise Algo Traders' Source Code, The Financial Times, 1 December, 2015.
[43] Danielle K. Citron, Frank Pasquale, The scored society: due process for automated predictions, Wash. Law Rev. 89 (1) (2014). https://digitalcommons.law.uw.edu/wlr/vol89/iss81/82.
[44] Frank Pasquale, The Black Box Society: the Secret Algorithms that Control Money and Information, Harvard University Press, Cambridge, MA, 2015.
[45] Donald MacKenzie, How algorithms interact: Goffman's 'interaction order' in automated trading, Theor. Cult. Soc. 36 (2) (2019) 39–59.
[46] Kushwaha, Bahl Shashi, Shashi, Ashok Kumar Bagha, Kulwinder Singh Parmar, Mohd Javaid, Abid Haleem, Ravi Pratap Singh, Significant applications of machine learning for covid-19 pandemic, J. Indus. Integrat. Manag. 5 (4) (2020) 453–479.
[47] Wanigasekara, Chathura, Ebrahim Oromiehie, Akshya Swain, B. Gangadhara Prusty, Sing Kiong Nguang, Machine learning-based inverse predictive model for afp based thermoplastic composites, J. Indus. Inform. Integrat. 22 (2021) 100197, https://doi.org/10.1016/j.jii.2020.100197.
[48] John R. Koza, Riccardo Poli, Genetic programming, in: Edmund K. Burke, Graham Kendall (Eds.), Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques, Springer, Cham, 2014, pp. 143–185.
[49] Alexander Denev, Saeed Amen, The Book of Alternative Data: A Guide for Investors, Traders, and Risk Managers, Wiley, Hoboken, NJ, 2020.
[50] Jenna Burrell, How the machine 'thinks': understanding opacity in machine learning algorithms, Big Data Soc. 3 (1) (2016), https://doi.org/10.1177/2053951715622512.
[51] Cambridge Centre for Alternative Finance, & World Economic Forum, Transforming Paradigms: A Global AI in Financial Services Survey, Cambridge Centre for Alternative Finance and World Economic Forum, Cambridge and Geneva, 2020.
[52] World Economic Forum, The New Physics of Financial Services: Understanding How Artificial Intelligence Is Transforming the Financial Ecosystem, World Economic Forum, Geneva, 2018.
[53] Bryan W. Husted, Agency, information, and the structure of moral problems in business, Organ. Stud. 28 (2) (2006) 177–195.
[54] Fang Chen, Jianlong Zhou (Eds.), Human and Machine Learning: Visible, Explainable, Trustworthy and Transparent, Springer, Cham, 2018.
[55] Samek, Wojciech, Grégoire Montavon, Vedaldi, Hansen Andrea, Lars Kai, Klaus-Robert Müller (Eds.), Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, Springer, Cham, 2019.
[56] Nick Bostrom, Superintelligence: Paths, Dangers, Strategies, Oxford University Press, Oxford, 2014.
[58] Virginia Dignum, Responsibility and artificial intelligence, in: Markus D. Dubber, Frank Pasquale, Sunit Das (Eds.), The Oxford Handbook of Ethics of AI, Oxford University Press, Oxford, 2020, pp. 215–231.
[59] Christian Borch, High-frequency trading, algorithmic finance, and the Flash Crash: reflections on eventalization, Econ. Soc. 45 (3–4) (2016) 350–378, https://doi.org/10.1080/03085147.2016.1263034.
[60] Christian Borch, Machine learning and social theory: Collective machine behaviour in algorithmic trading, Eur. J. Soc. Theor. (2021), https://doi.org/10.1177/13684310211056010.
[61] Christian Borch. Social Avalanche: Crowds, Cities and Financial Markets, Cambridge University Press, Cambridge, 2020.
[62] Bo Hee Min, Christian Borch, Systemic failures and organizational risk management in algorithmic trading: Normal accidents and high reliability in financial markets, Soc. Stud. Sci. (2021), https://doi.org/10.1177/03063127211048515.
[63] Kristian Bondo Hansen, Christian Borch, Alternative data and sentiment analysis: Prospecting non-standard data in machine learning-driven finance, Big Data Soc. (2022), https://doi.org/10.1177/20539517211070701.

Christian Borch is Professor of Economic Sociology and Social Theory at the Copenhagen Business School, Denmark. His work focuses on automated trading, machine learning, and social theory. He is the Principal Investigator of the European Research Council-funded research project "Algorithmic Finance." His latest book is *Social Avalanche: Crowds, Cities and Financial Markets* (Cambridge University Press, 2020).