(Deep) Learning to Trade: An Experimental Analysis of AI Trading and Market Outcomes*

I. Gufler†, F. Sangiorgi‡, E. Tarantino§

August 1, 2025

Abstract

We investigate how AI-driven investors, modeled via deep reinforcement learning, operate in a calibrated financial market with realistic return predictability and endogenous price impact. We examine whether these agents can learn to detect and exploit return predictability from public signals, decode prices to infer latent demand, and adjust for price impact. To evaluate performance, we compare AI traders to a rational benchmark representing the optimal policy under full knowledge of the data generating process. In simulations, AI traders qualitatively match the benchmark. Quantitatively, however, they fall short when many interact. The presence of other AI traders injects noise into the price process through their exploration, distorting the portfolio-return signals each agent learns from and thus impairing learning. This negative learning externality reduces trading profits, lowering market efficiency and liquidity relative to the benchmark. Our findings suggest caution when extrapolating from partial-equilibrium analyses of AI trading's profitability and impact on market quality.

Keywords: Deep reinforcement learning, Multi-agent reinforcement learning, Market efficiency, Learning frictions, Asset pricing, Algorithmic trading

*The views expressed are those of the authors and do not necessarily reflect those of the European Commission.

†Luiss University, ivangufler26@gmail.com

‡Frankfurt School of Finance and Management, f.sangiorgi@fs.de

§Luiss University and European Commission, etaranti@gmail.com

Acknowledgements and Disclosure

We gratefully acknowledge financial support from Banca Intesa. The authors declare that they have no relevant or material financial interests that relate to the research described in this paper.

1 Introduction

Financial markets have undergone profound changes due to advances in computing power and algorithmic sophistication. Early algorithmic trading, rooted in fixed, rule-based paradigms, has evolved into data- and computation-intensive machine learning systems capable of placing orders on the market and adjusting to changing market conditions. Recent evidence suggests that machine learning can improve price discovery by extracting information relevant to predicting future returns and by better analyzing corporate success, thereby making stock prices more informative (e.g., Bai et al., 2016; Dugast and Foucault, 2018; Farboodi et al., 2022, among others).

Reflecting these advances, algorithmic trading has become pervasive in financial markets (SEC, 2020, p. 5). Both retail and institutional investors use algorithms to process market information, assess trading opportunities, and implement trading decisions in real time (SEC, 2020, p. 30,34). This raises concerns that widespread adoption of similar trading strategies may be a source of fragility, impairing market liquidity during shocks (Federal Reserve Board, 2022). While such systemic risks are still debated, a more immediate question emerges: how does increased adoption of AI-based trading reshape the very market structure and return patterns it initially sought to exploit? Do AI traders help markets achieve greater efficiency and resilience? Or do their interactions generate new frictions that degrade market quality?

We contribute to this debate by examining a learning friction that can undermine the effectiveness of algorithmic trading and degrade stock market performance. The computer science literature shows that reinforcement learning algorithms face significant challenges in multi-agent settings: the environment becomes non-stationary as each agent's learning both influences and is influenced by others' behavior (Lowe et al., 2017; Albrecht et al., 2024). In financial markets, algorithms learn from price signals that reflect the collective behavior of all market participants. We refer to the noise generated by one agent that interferes with the learning of others as a learning externality. Understanding this externality in a financial market setting is central to assessing both the performance of AI-based trading strategies and their systemic market impact. Yet empirical identification of such frictions is inherently difficult: without a clear counterfactual, it is nearly impossible to determine how AI traders influence each other and the market using observational market data alone.

To overcome these difficulties, we adopt an experimental approach that embeds AI-based traders—modeled via deep reinforcement learning (DRL) algorithms—within a theoretically grounded and empirically calibrated asset pricing framework. In this setting, DRL agents learn to trade without prior knowledge of the data-generating process and interact in equilibrium through their price impact. This analysis requires a conceptual framework that goes beyond the current state of the art in economics, finance, and computer science.

Our approach departs from existing literature by combining insights from two separate fields within an equilibrium framework. The computer science literature applies reinforcement learning to portfolio selection problems,[1] while the finance literature demonstrates machine learning's ability to detect return predictability from historical data.[2] Both approaches, however, treat algorithmic agents as atomistic and price-taking, deploying strategies on fixed historical data. Unlike these partial-equilibrium frameworks, we model traders as learning and optimizing within a market where prices respond endogenously to demand. This equilibrium approach allows us to study how AI traders' learning is affected by their own price impact and by the noise generated by others.

To generate a realistic learning and trading environment, we nest our DRL agents in a demand-based asset pricing model based on the approach of Koijen and Yogo (2019) in which the marginal investor's asset demand is calibrated from data.[3] To maintain tractability while preserving realism, we focus on ten U.S. equities spanning a broad range of characteristics and return dynamics. For each stock, DRL agents solve the one-period portfolio allocation problem between that risky asset and a riskless asset, learning over time how to respond to market signals and their own price impact.

This demand system implies that stock prices reflect both persistent firm characteristics (e.g., book-to-market, profitability, market beta) and latent institutional demand. Return predictability arises from partial mean reversion in both observable and unobservable components. We examine whether AI traders can uncover and exploit this structure, and how their trading strategies affect market outcomes. In particular, if AI agents can infer latent demand from prices,

[1] See, among others, Cartea et al. (2021); Jiang et al. (2017); Yang et al. (2018, 2020); Zhang et al. (2020); Wang and Zhou (2020); see also Hambly et al. (2023) for a literature review.

[2] E.g., Gu et al. (2020); see also Vives (2019), Nagel (2021) and Kelly and Xiu (2023) for comprehensive reviews.

[3] Recent work explores alternative demand estimation methods (e.g., van der Beck, 2022; Fuchs et al., 2024). While the quantification of our effects may be affected by the specific demand specification employed, our qualitative results on learning and equilibrium outcomes should not.

they may generate excess returns—while modifying the very patterns they aim to exploit. This framework also enables us to evaluate how AI trading influences market efficiency and liquidity as more AI agents interact or grow in scale.

Most existing work on reinforcement learning in economics and finance employs the tabular Q-learning framework introduced by Watkins (1989) to model economic agents (e.g., Calvano et al., 2020; Colliard et al., 2023; Dou et al., 2023; Abada and Lambin, 2023; Johnson et al., 2023). While these algorithms, designed for discrete state and action spaces, reduce computational complexity, this discretization can distort the learning of optimal portfolio policies in settings with return predictability. In such environments, both relevant state variables (such as predictive signals from prices or stock characteristics) and optimal actions (portfolio weights) are inherently continuous. To address this, we employ deep deterministic policy gradient (DDPG) algorithms, which support continuous state and action spaces through neural network function approximation (Lillicrap et al., 2015; Kilinc and Montana, 2018). This enables our AI traders to learn smooth portfolio policy functions and generalize across previously unseen market conditions, capabilities that tabular Q-learning cannot provide.

Our AI agents must identify and exploit return predictability solely through their trading experience, without prior knowledge of market structure. This leads to a central question: how can we evaluate the extent to which AI traders are learning? To address this, we introduce a theoretical benchmark in which rational speculators can perfectly infer latent demand from prices, understand the underlying price formation process, and anticipate their price impact. Both the benchmark and the algorithms solve a standard portfolio allocation problem between a risky and a riskless asset to maximize one-period returns. Since they operate under identical conditions–facing the same information set, trading constraints, and objectives–any difference in behavior must reflect algorithmic learning frictions. The benchmark thus represents the upper bound of what algorithms could achieve if they were able to learn optimal behavior from experience.

The theoretical benchmark delivers three propositions. First, optimal portfolio weights depend only on a single composite signal, z, which compresses all publicly available information into a sufficient statistic for next-period returns. The optimal portfolio weight rises with z, falls as each trader's portfolio size grows larger, and rises when the same aggregate capital is split across more competing traders. Second, market efficiency (measured by the share of return variance unexplained

by public signals) improves when either total wealth or the number of rational speculators increases, as their trades remove predictable patterns. Third, liquidity (inversely related to price response to transient supply shocks) likewise improves as better-capitalized or more numerous speculators trade more aggressively to absorb supply shocks.

Our experimental results show that AI traders' portfolio policies exhibit strong qualitative alignment with the theoretical benchmark. In line with the theoretical prediction, AI traders' portfolio weights increase monotonically in the sufficient statistic z. Portfolio weights also decrease with the trader's own size—reflecting internalization of price impact—and increase with the number of competing traders, as greater competition dilutes individual influence on prices. AI traders also improve market outcomes: they enhance market efficiency by reducing the share of returns explained by public signals and improve liquidity by attenuating price responses to transient supply shocks, in ways qualitatively consistent with the theoretical benchmark.

However, as the number of AI traders increases or their collective wealth share grows, systematic quantitative deviations from the benchmark emerge. While the direction of policy responses remains consistent with theory, the magnitude becomes distorted. AI traders scale down their portfolio shares too little when their size grows and scale up positions too aggressively as competition intensifies. These behaviors degrade portfolio performance relative to the rational benchmark and generate persistent inefficiencies. Specifically, return predictability remains elevated and prices react more sharply to transitory supply shocks than in the benchmark.

Diagnostics from controlled simulations reveal that these quantitative gaps are driven by a negative learning externality: each agent's exploratory trades introduce non-stationary variation in prices, which contaminates the learning signals available to others. Since no agent observes the identity or strategy of its peers, it cannot disentangle its own price impact from noise created by others. The result is a learning friction that reduces individual performance and dampens market-level benefits of algorithmic learning.

These findings highlight the value of studying AI trading within an equilibrium framework: without it, partial-equilibrium back-tests may overstate both the effectiveness of AI-based trading strategies and their market impact. In isolation, AI agents may appear to learn successfully and enhance market efficiency, but when multiple agents interact and adapt jointly, endogenous feedback and learning frictions emerge that degrade performance and reduce efficiency gains. Capturing these

dynamics requires a framework where agents endogenously affect the environment they learn from, something partial-equilibrium approaches reliant on fixed historical data cannot account for.

We contribute to the nascent literature on AI-based trading and market quality by identifying a novel learning externality that emerges from AI trader interactions in equilibrium. This literature has revealed important behavioral patterns in algorithmic decision-making that deviate from classical predictions. Colliard et al. (2023) demonstrate that Q-learning algorithms acting as market makers in a Glosten-Milgrom framework learn to deal with adverse selection, but fail to achieve competitive pricing because of noise in the reward and limited exploration of the state space. Conversely, Dou et al. (2023) show that Q-learning algorithms possessing fundamental information can learn to collude without explicit coordination or communication. They demonstrate that collusive behavior can be sustained by two distinct mechanisms, a price-trigger strategy and a learning bias causing AI traders to become overly conservative. Yang (2024) studies Q-learning algorithms' coordination in a speculative attack framework à la Morris and Shin (1998). Barberis and Jin (2023) emphasize biases in the portfolio choice of Q-learning algorithms. Lopez-Lira (2025) tests the ability of Large Language Models (LLMs) to serve as different types of trading agents (for instance, value or momentum investors, market makers) in an experimental setting.

Our work differs fundamentally from these studies in both focus and mechanism. Methodologically, we advance the literature by embedding deep reinforcement learning agents within an empirically calibrated demand-based asset pricing framework. This approach allows us to study algorithmic learning and portfolio policies in realistic market environments with endogenous price formation and latent demand dynamics. In this respect, we are closely related to the strand of literature exploring how machine learning techniques (Nagel, 2021; Kelly and Xiu, 2023; Kelly et al., 2024) and reinforcement learning (Heaton et al., 2017; Yang et al., 2018, 2020; Zhang et al., 2020) can be used to detect signals from the data and implement profitable trading strategies. However, we differ from this literature in that we focus on the predictability of returns in an equilibrium setting, where prices respond endogenously to the portfolio choice of reinforcement learning agents.

Substantively, we investigate how algorithmic learning itself becomes impaired through multi-agent interactions, identifying a learning externality where each agent's exploratory trades inject noise into the price process, contaminating the learning signals available to others and degrading overall performance. Importantly, our experimental design prevents tacit collusion by restricting

the state space to exogenous variables that do not depend on past trading decisions. This ensures that the market-wide effects we observe are due to learning frictions, not strategic coordination. Our contribution lies in showing that even when the experimental design prevents coordination, the collective learning of algorithms can still undermine market efficiency through purely informational channels–a mechanism distinct from the competition failures or collusion highlighted in prior work.

The rest of the paper is organized as follows. Section 2 introduces the asset-pricing environment and derives the rational expectations benchmark, highlighting how speculators with full knowledge of the price-formation process behave. Section 3 describes the empirical calibration of the market and investor demand, the construction of the state variables, and the reinforcement learning implementation. Section 4 presents our experimental results, comparing AI traders to the rational benchmark in terms of portfolio policies and market outcomes. It also identifies the learning externality that arises from AI interaction and quantifies its impact. Section 5 discusses policy implications. Section 6 concludes.

2 The market environment

2.1 Overview

In this section, we introduce a market environment featuring return predictability and price impact, both arising endogenously from a representative investor's demand function. Although our numerical experiments with AI traders in Section 4 focus on repeated one-period portfolio choices involving a single risky asset and a riskless asset, we embed these choices in a dynamic framework; this adds realism by ensuring that portfolio returns depend on future price changes (capital gains) rather than dividends alone. At the same time, the one-period, single-asset setting allows us to isolate the core algorithmic learning challenges under different market conditions (such as varying assets, trader competition, and assets under management) while avoiding the complexities inherent in fully dynamic, multi-asset portfolio optimization.

In the remainder of this section, we first derive asset prices with a representative investor and a set of J traders. These traders will later be interpreted as either rational or AI-driven, but at this stage their portfolio choices are exogenous. Then we discuss how traders might learn from prices and exploit return predictability embedded within the data-generating process. To this

end, we introduce a rational expectations benchmark, which serves as a baseline for evaluating the performance of AI-driven traders in Section 4.

2.2 Representative investor and equilibrium prices

We consider a market for N risky assets and a risk-less asset. Time is discrete and runs from 0 to ∞. For each risky asset $n = 1, \ldots, N$, we denote its price and dividend per share in period t with $P_{n,t}$ and $D_{n,t}$, respectively, and its number of shares with S_n, which we assume constant over time. Risky asset prices are determined endogenously, as explained below. The risk-free asset is elastically supplied at an exogenous gross return R_f constant over time.

The market is populated by a representative investor and J traders. Let $S^j_{n,t}$ denote the number of shares of asset n held by the j-th trader at time t. The aggregate holdings of asset n across the J traders are given by $S^a_{n,t} = \sum_{j=1}^{J} S^j_{n,t}$ and the residual supply held by the representative investor is $\tilde{S}_{n,t} = S_n - S^a_{n,t}$. Throughout the paper, we use lower-case letters to denote logarithms and Δ to indicate first differences.[4]

To model the representative investor's asset demand, we adapt the framework of Koijen and Yogo (2019) and specify the following log-exponential form:

$$\frac{w_{n,t}}{w_{0,t} + \gamma_t} = \delta_{n,t}; \quad \delta_{n,t} = \exp\left(\beta_0(p_{n,t} + s_n) + \sum_{k=1}^{K-1} \beta_k x_{k,n,t} + \beta_K + \epsilon_{n,t}\right), \tag{1}$$

where $w_{n,t}$ is the representative investor's portfolio weight in asset n, $w_{0,t}$ is the weight in the risk-free asset, and γ_t represents the fraction of assets consumed by the representative investor, which we define below. The variables $\{x_{n,k,t}\}_{k=1}^{K-1}$ represent publicly observable characteristics for asset n (e.g., book-to-value, profitability, etc.). The term $\epsilon_{n,t}$ represents the representative investor's latent demand for asset n, capturing investor sentiment and other demand components not explained by asset fundamentals. Koijen and Yogo (2019) document that latent demand accounts for a significant portion of cross-sectional variation in stock returns.

[4] For example, $p_{n,t} = \log(P_{n,t})$ represents the log price, and $\Delta S^a_{n,t} = S^a_{n,t} - S^a_{n,t-1}$ denotes the change in the J traders' aggregate holdings.

We assume that stock characteristics and latent demand follow the autoregressive processes:

$$x_{k,n,t+1} = c_{k,n} + \rho_{k,n} x_{k,n,t} + \eta_{k,n,t+1}, \tag{2}$$

$$\epsilon_{n,t+1} = c_{\epsilon,n} + \rho_{\epsilon,n} \epsilon_{n,t} + \xi_{n,t+1}, \tag{3}$$

where $\rho_{k,n}, \rho_{\epsilon,n} \in (0,1)$ are the autoregressive coefficients, and $\eta_{k,n,t+1}$ and $\xi_{n,t+1}$ are mean-zero normally distributed innovations, independent over time and across variables, with variances $Var(\eta_{k,n,t}) = \sigma^2_{\eta_{k,n}}, Var(\xi_{n,t}) = \sigma^2_{\xi_n}$.

To conduct our numerical experiments in Section 4, we estimate the representative investor demand coefficients in Eq. (1) and calibrate the parameters of the processes in Eqs. (2)–(3), as detailed in Section 3.

The representative investor's wealth A evolves according to

$$A_t = \sum_{m=1}^{N} \tilde{S}_{m,t-1} \left(P_{m,t} + D_{m,t}\right) + S_{0,t-1} R_f + \Gamma_t, \tag{4}$$

where $S_{0,t-1}$ is the units of the risk-free asset held at time $t-1$ and Γ_t is an exogenous inflow of resources which is specified below. Similarly, trader-j's wealth A^j evolves according to

$$A^j_t = \sum_{m=1}^{N} S^j_{m,t-1} \left(P_{m,t} + D_{m,t}\right) + S^j_{0,t-1} R_f. \tag{5}$$

The market clearing condition for each stock n reads as follows:

$$A_t w_{n,t} + P_{n,t} S^a_{n,t} = P_{n,t} S_n. \tag{6}$$

We initially take the J traders' holdings $\{S^j_{n,t}\}_{j=1}^J$ as given and derive the price function for each security, as implied by (i) the representative investor's demand in Eq. (1), (ii) the market clearing condition in Eq. (6), and (iii) the asset dynamics of both the representative investor and the J traders' in Eqs. (4)-(5) together with the respective budget constraints. In Appendix A.1, we derive the resulting equilibrium price function as

$$p_{n,t} = \frac{\beta_0 s_n - \tilde{s}_{n,t} + \sum_{k=1}^{K-1} \beta_k x_{k,n,t} + \beta_K + \epsilon_{n,t} + \log\left(D_{M,t} + S_{0,t-1}R_f + \Gamma_t + S^a_{0,t-1}R_f - S^a_{0,t}\right)}{1-\beta_0}, \tag{7}$$

where the aggregate dividend $D_{M,t}$ is defined as

$$D_{M,t} = \sum_{m=1}^{N} S_m D_{m,t}.$$

We make the following additional assumptions to enhance analytical tractability:

Assumption 1. *The exogenous change in the representative investor's assets equals* $\Gamma_t = S^a_{0,t} - S^a_{0,t-1}R_f$.

Assumption 2. *The representative investor's consumption at time* t *equals* $\gamma_t A_t = \lambda(D_{M,t} + S_{0,t-1}R_f)$*, where* $\lambda \in (0,1)$ *is a constant.*

Assumption 3. *Aggregate dividends grow deterministically at the constant rate* g*:* $D_{M,t} = (1+g)D_{M,t-1}$.

Assumption 4. *The parameters* λ, g, R_f *satisfy the condition* $1 + g > R_f(1-\lambda)$.

These assumptions are imposed for analytical convenience. Assumption 1 eliminates indirect cross-asset price effects from the J traders' positions in the risk-free asset, simplifying the equilibrium price expression in Eq. (7) to depend only on $D_{M,t} + S_{0,t-1}R_f$. This is justified by our focus on single risky asset portfolio decisions, where cross-asset interactions are irrelevant. Assumption 2 allows expressing $D_{M,t} + S_{0,t-1}R_f$ recursively as a function of past dividends (see Eq. (A5) in Appendix A.1). Assumption 3 removes aggregate uncertainty from returns, reducing noise in AI investors' learning. Finally, Assumption 4 ensures time-invariance of the price function as $t \to \infty$.

In Appendix A.1, we prove that under Assumptions 1–4, the price function in Eq. (7) simplifies to

$$p_{n,t} = -s_n + \frac{-\log\left(1-\alpha^a_{n,t}\right) + \sum_{k=1}^{K-1} \beta_k x_{k,n,t} + \beta_K + \epsilon_{n,t} + d_{M,t} + \phi}{1-\beta_0}, \tag{8}$$

where $\alpha^a_{n,t} = \frac{S^a_{n,t}}{S_n}$ is the fraction of asset supply held collectively by the J traders, and

$$\phi = \log\left(\frac{(1+g)}{1+g-R_f(1-\lambda)}\right); \quad d_{M,t} = \log(D_{M,t}).$$

Equation (8) has a straightforward interpretation. The price of asset n is inversely related to its supply s_n, and it is positively related to its observable characteristics, weighted by the demand coefficients $\sum_{k=1}^{K-1} \beta_k \, x_{k,n,t}$. Additionally, it increases with the aggregate dividend $d_{M,t}$. All these variables are publicly observed.

The price is also influenced by the latent demand $\epsilon_{n,t}$. Although latent demand is unobservable, in the next subsection we show how investors who understand the price-formation process can combine the observed price with other public signals to infer this latent demand.

Finally, the demand from the J traders enters the price through $\alpha^a_{n,t}$. From Eq. (8), the elasticity of price with respect to the residual supply held by the representative investor is[5]

$$-\frac{\partial p_{n,t}}{\partial \log(\tilde{S}_{n,t})} = \frac{1}{1-\beta_0}.$$

Hence, if the J traders purchase an additional 1% of the asset supply, the price $P_{n,t}$ rises by $1/(1-\beta_0)$%. Intuitively, when β_0 is close to one, the representative investor's demand is less elastic, increasing the price impact faced by the J traders.

2.3 The rational expectations benchmark as a learning frontier

In this subsection, we consider the case where the J traders introduced previously have rational expectations, and refer to them as "speculators." These speculators operate in the same market environment, face the same trading protocol, and observe the same public information as the AI traders introduced later in Section 3. The key distinction is that speculators have full knowledge of the data generating process: they understand the price formation mechanism described in Section 2.2, including the dynamics of the exogenous processes in Eqs. (2)–(3) and the pricing rule in Eq. (8). They are able to decode the information embedded in prices and anticipate the price impact of their trades, thereby fully internalizing both return predictability and price

[5] Notice that $\log\left(1-\alpha^a_{n,t}\right) = \log\left(\frac{S_n - S^a_{n,t}}{S_n}\right) = \log\left(\tilde{S}_{n,t}\right) - s_n$.

impact in their optimization (as explained later in this section).

The purpose of this rational expectations (RE) setup is not to model how human traders behave, nor to suggest that such fully informed optimization is within reach of institutional investors. Rather, it serves as a theoretical "learning frontier": an upper bound on the performance that could, in principle, be achieved by model-free AI traders if they were to perfectly infer the structure of the market environment through experience.

This benchmark plays a central role in our numerical experiments in Section 4, where we evaluate how closely the behavior of reinforcement learning agents approximates this theoretical outcome. It allows us to quantify the performance gaps that emerge in practice—gaps we later attribute to learning frictions, such as the externality arising in multi-agent learning. A detailed discussion of these mechanisms is deferred to Section 4.

Setup Speculators enter the market at time t, allocate their wealth between risky asset n and the riskless asset to maximize next period wealth, and exit at time $t+1$. In the remainder of this section, we use the notation $\alpha^S_{n,\tau}$ for the fraction of the supply of shares held collectively by the J speculators at time τ (as opposed to the generic $\alpha^a_{n,\tau}$ used in Section 2.2, where the J investors' holdings were taken as given).

Because they enter at time t and liquidate their holdings at time $t+1$, $\alpha^S_{n,t}$ is determined in equilibrium, whereas $\alpha^S_{n,t+1} = 0$. We first analyze the learning and predictability aspects of the data-generating process, then describe the trading game in more detail and characterize its equilibrium.

Predictability and learning in the data generating process By setting $\alpha^a_{n,t} = \alpha^S_{n,t}$ and $\alpha^a_{n,t+1} = \alpha^S_{n,t+1} = 0$ in Eq. (8) to compute prices at t and $t+1$, and using Eqs. (2)–(3), we can compute the conditional expectation of the capital gain as

$$E_t\left(\frac{P_{n,t+1}}{P_{n,t}}\right) = \exp\left(\frac{\log\left(1-\alpha^S_{n,t}\right) - \sum_{k=1}^{K-1}\beta_k(1-\rho_{k,n})x_{k,n,t} - (1-\rho_{\epsilon_n})\epsilon_{n,t}}{1-\beta_0}\right)\Phi_n, \quad (9)$$

where Φ_n is a constant value defined in Eq. (A13) in the Appendix. Eq. (9) shows that mean reversion in the observable characteristics $\{x_{k,n,t}\}$ and in the latent demand $\epsilon_{n,t}$ gives rise to return predictability. However, $\epsilon_{n,t}$ is not observable by investors, who must infer it from market data.

In a RE equilibrium where traders can condition their asset demand on public information and the equilibrium price itself, the equilibrium price in Eq. (8) fully reveals $\epsilon_{n,t}$.[6] Therefore, in a RE equilibrium, investor expectations of capital gains coincide with the full-information in Eq. (9).

This learning process, though standard in RE models, is difficult to replicate with algorithms due to its fixed-point nature: the equilibrium price depends on speculators' demands, but each speculator's demand is itself a function of that same equilibrium price. In a reinforcement learning framework, such as the one developed in the next section, the agent's action depends on the current state which must be fully realized before the action is taken. Hence, using the equilibrium price as an input to each trader's policy introduces circular logic: the policy would depend on a price that itself depends on the policy.

Putting rational and AI traders on equal footing To address this conceptual issue, we introduce a "pre-trade" price $P^*_{n,t}$ that incorporates public information, namely $\{x_{n,k,t}\}_{k=1}^{K-1}, d_{M,t}$, and the representative investor's latent demand $\epsilon_{n,t}$, but not the time-t trading decisions of the J speculators. Hence, $P^*_{n,t}$ is exogenous to the speculators' actions in period t. We define the log of the pre-trade price setting $\alpha^S_{n,t} = 0$ in Eq. (8):

$$p^*_{n,t} = -s_n + \frac{\sum_{k=1}^{K-1} \beta_k x_{k,n,t} + \beta_K + \epsilon_{n,t} + d_{M,t} + \phi}{1 - \beta_0}. \tag{10}$$

Accordingly, we let $\mathcal{I}_{n,t} = \left\{p^*_{n,t}\{x_{n,k,t}\}_{k=1}^{K-1}, d_{M,t}\right\}$ denote the public information set. Since $p^*_{n,t}$ depends on $\epsilon_{n,t}$, investors who understand the pricing rule can invert Eq. (10) to perfectly recover the latent demand:

$$E\left(\epsilon_{n,t} \mid \mathcal{I}_{n,t}\right) = (1 - \beta_0)\left(p^*_{n,t} + s_n\right) - \left(\sum_{k=1}^{K-1} \beta_k x_{k,n,t} + \beta_K + d_{M,t} + \phi\right) = \epsilon_{n,t}, \tag{11}$$

where $E(\cdot)$ denotes the rational expectations operator. Thus, $p^*_{n,t}$ fully reveals $\epsilon_{n,t}$ just as the equilibrium price would in a classical RE model, but $p^*_{n,t}$ is determined prior to time-t trading by the speculators.

This design places rational speculators and AI traders—implemented as reinforcement-learning

[6] We are implicitly assuming the standard RE arguments: the price function is common knowledge, speculators correctly anticipate other traders' demand, and therefore correctly infer the value of $\alpha^S_{n,t}$. Under these assumptions, speculators can invert Eq. (8) to recover the latent demand.

algorithms—on an equal footing with respect to observable information. Both agent types first observe the exogenous signals $\mathcal{I}_{n,t}$ and then choose their portfolios. This setup lets us test whether AI traders can decode prices to infer latent demand while avoiding the fixed-point problem of learning from the equilibrium price.

The trading game The trading game unfolds as follows. Each speculator enters the market at t with initial wealth $A_t^j = A_t^s$, where $A_t^s = \omega P_{n,t}^* S_n$, and ω, J satisfy $\omega < 1/J$. That is, each speculator has the same initial wealth, and collectively, the J speculators' wealth equals a fraction $J\omega$ of the pre-trade market capitalization of asset n. Next, each speculator observes the public information set $\mathcal{I}_{n,t}$ and, conditional on this information, speculators simultaneously choose their portfolio share $\theta_{n,t}^j$ in the risky asset n, subject to no-short-selling and no-borrowing constraints. Each speculator aims to maximize the expectation of the portfolio return

$$R_{p,t+1}^j = R_f + \theta_{n,t}^j \left(R_{n,t+1} - R_f\right), \tag{12}$$

where $R_{n,t+1} = \frac{P_{n,t+1}}{P_{n,t}} + DY_{n,t+1}$, and $DY_{n,t+1} = \frac{D_{n,t+1}}{P_{n,t}}$ denotes the dividend yield. In the rest of the paper we assume the dividend yield is exogenous and i.i.d. over time with mean $\overline{DY}_n$.[7]

Formally, a speculator's strategy is a mapping $\theta_n^j : \mathcal{I} \to [0,1]$, which, given the public information set, specifies what fraction of wealth to invest in the risky asset. No-short-selling implies $\theta_{n,t}^j \geqslant 0$, and no-borrowing implies $\theta_{n,t}^j \leqslant 1$.

Given the speculators' chosen portfolio shares $\{\theta_{n,t}^j\}_{j=1}^J$, the asset price is determined according to Eq. (8). We substitute $S_{n,t}^j = \frac{A_t^s \theta_{n,t}^j}{P_{n,t}}$ into Eq. (8), obtaining an implicit relationship in terms of $\{\theta_{n,t}^j\}$:

$$0 = \beta_0(p_{n,t} + s_n) - \log\left(S_n P_{n,t} - \sum_{j=1}^{J} \theta_{n,t}^j A_t^s\right) + \sum_{k=1}^{K-1} \beta_k x_{k,n,t} + \beta_K + \epsilon_{n,t} + d_{M,t} + \phi. \tag{13}$$

[7] This assumption is for simplicity and to focus on predictability of capital gains. See Koijen and Yogo (2019) for evidence that dividend yields explain only 0.4% of the cross-sectional variance of stock returns, suggesting that capital gains drive most price variation.

Because the speculators liquidate their positions at time $t+1$, we have

$$p_{n,t+1} = -s_n + \frac{\sum_{k=1}^{K-1} \beta_k x_{k,n,t+1} + \beta_K + \epsilon_{n,t+1} + d_{M,t+1} + \phi}{1-\beta_0}. \tag{14}$$

When choosing their portfolio share, a rational speculator correctly anticipates the effect of their price impact on the expected portfolio return:

$$\frac{\partial E(R^j_{p,t} \mid \mathcal{I}_{n,t})}{\partial \theta^j_{n,t}} = E\left[\frac{P_{n,t+1}}{P_{n,t}}\left(1 - \frac{\theta^j_{n,t}}{P_{n,t}}\frac{\partial P_{n,t}}{\partial \theta^j_{n,t}}\right) \mid \mathcal{I}_{n,t}\right] + \overline{DY}_n - R_f. \tag{15}$$

Taking other traders' strategies as given, we can calculate the price impact factor in Eq. (15) from implicit differentiation of Eq. (13) as follows:

$$\frac{\theta^j_{n,t}}{P_{n,t}}\frac{\partial P_{n,t}}{\partial \theta^j_{n,t}} = \frac{\theta^j_{n,t} A^s}{(1-\beta_0) S_n P_{n,t} + \beta_0 \sum_{i=1}^{J} \theta^i_{n,t} A^s}. \tag{16}$$

Information compression to a sufficient statistic For expositional simplicity we define "adjusted market equity" as

$$me^*_{n,t} = p^*_{n,t} + s_n - \frac{d_{M,t}}{1-\beta_0}, \tag{17}$$

and we rewrite speculators' information set more compactly as $\mathcal{I}_{n,t} = \{me^*_{n,t}, \{x_{n,k,t}\}_{k=1}^{K-1}\}$. This is without loss of generality because (i) by observing $\{me^*_{n,t}, \{x_{n,k,t}\}_{k=1}^{K-1}\}$ investors can fully learn the latent demand,[8] and (ii) neither the expected capital gain in Eq. (9) nor the price impact factor in Eq. (16) depend on $d_{M,t}$.[9]

We define $z_{n,t}$ as

$$z_{n,t} = \gamma_0 + \sum_{k=1}^{K-1} \gamma_k x_{k,n,t} + \gamma_K me^*_{n,t}, \tag{19}$$

for some coefficients $\gamma_0, \ldots \gamma_K$ shown in Eq. (A15) in the Appendix. The proof of Proposition 1

[8] Notice that Eq. (10) and Eq. (17) imply

$$(1-\beta_0) me^*_{n,t} - \left(\sum_{k=1}^{K-1} \beta_k x_{k,n,t} + \beta_K + \phi\right) = \epsilon_{n,t}, \tag{18}$$

[9] While immaterial in this RE setting where inference is exact, this change of variable becomes useful in later sections when we implement the model algorithmically. In particular, scaling $p^*_{n,t}$ by $d_{M,t}$ removes the trend in aggregate dividends and makes the variables stationary. This improves the stability of the reinforcement learning algorithm, which benefits from stationary inputs.

below establishes that $z_{n,t}$ is a sufficient statistic for $\mathcal{I}_{n,t}$ with respect to the future return on asset-n, and is such that the expected capital gain increase in $z_{n,t}$. Hence, each speculator's optimal weight in the risky asset depends on their information set only through $z_{n,t}$.

Equilibrium portfolio choice An equilibrium of the trading game is a vector of portfolio shares $\{\theta^j_{n,t}\}$ that maximises each speculator's expected utility given their information set $\mathcal{I}_{n,t}$ and the price functions in Eqs. (13)–(14). In the following proposition, "increasing" and "decreasing" mean weakly increasing and weakly decreasing as no-short-sale or borrowing constraints may bind at $\theta^j_{n,t} = 0$ or 1.

Proposition 1 (Portfolio choice)**.**

(i) ***Existence and uniqueness.*** *An equilibrium exists, is unique, and is symmetric.*

(ii) ***Linear sufficiency.*** *The equilibrium portfolio share depends only on the sufficient statistic $z_{n,t}$, $\theta_{n,t} = \theta_n(z_{n,t})$, and is increasing.*

(iii) ***Comparative statics.*** *For all values of $z_{n,t}$:*

*(a) The equilibrium portfolio share is decreasing in each speculator's wealth ω (*size effect*).*

*(b) The equilibrium portfolio share is increasing in the number of speculators J holding aggregate speculator wealth ωJ fixed (*competition effect*).*

The comparative statics results in Proposition 1 are intuitive. First, a higher $z_{n,t}$ raises expected capital gains for asset n, so the optimal θ_n rises.[10]

Second, θ_n falls with speculators' wealth ω because their own price impact becomes stronger.

Third, for a fixed aggregate wealth ωJ, increasing the number of speculators rises θ_n. Each speculator internalizes only their individual price impact, so when total wealth is spread over more agents they each trade more aggressively.

[10]The proof of the proposition further shows that in terms of the original variables in $\mathcal{I}_{n,t}$, θ_n is (i) decreasing in adjusted market equity $me^*_{n,t}$ and (ii) is increasing in asset characteristic $x_{k,n,t}$ if and only if $\beta_k(\rho_{k,n} - \rho_{\epsilon,n}) > 0$. The intuition for these comparative statics is that, holding characteristics fixed, higher $me^*_{n,t}$ signals a stronger current latent demand (see Eq. (18) in Footnote 8); mean reversion then predicts a lower capital gain (see Eq. (9)). In response, speculators allocate a smaller fraction of their wealth to the risky asset. As for the characteristics, a larger value of the k-th characteristic has two effects. First, it decreases the expectation of $\Delta x_{k,n,t+1}$ due to mean reversion, thereby affecting the expected capital gain depending on the sign of β_k (see Eq. (9)). Second, for a fixed value of $me^*_{n,t}$, a larger value of $x_{n,k,t}$ also affects the expectation of latent demand (see Eq. (18)). The overall effect depends on the relative mean reversion speeds of ϵ_n and $x_{n,k}$ times β_k.

Next, we analyze how speculators' trading affects two key market characteristics: market efficiency and liquidity.

Market Efficiency. In the benchmark model with only a representative investor, returns are predictable based on public information. We remain agnostic as to whether this predictability reflects risk premia or mispricing, and adopt the semi-strong market efficiency view that returns should not be predictable from public signals. Thus, the extent to which this predictability persists directly measures market (in)efficiency.

Since dividend yields are unpredictable by assumption, we focus on the capital gain component of returns. We write:

$$p_{n,t+1} - p_{n,t} = g_n(\mathcal{I}_{n,t}) + e_{n,t+1}, \tag{20}$$

where $g_n(\mathcal{I}_{n,t})$ is an equilibrium function that depends on $z_{n,t}$ and on the trading strategies of the J speculators via $\alpha^S_{n,t}$, and $e_{n,t+1}$ is the unpredictable component of the capital gain.[11]

We define market efficiency as the fraction of capital gain variance that is not explained by public information:

$$\mathcal{ME} = \frac{\text{Var}\left(e_{n,t+1}\right)}{\text{Var}\left(p_{n,t+1} - p_{n,t}\right)}. \tag{21}$$

Note that $g_n(\mathcal{I}_{n,t})$ in Eq. (20) is derived from the true data-generating process. In this sense, $\mathcal{ME}$ is computed "inside the model." In Section 4 we also consider an alternative measure of market efficiency where the $g_n(\mathcal{I}_{n,t})$ is approximated by a linear function.

In the model, speculators' trading on predictive signals shrinks the predictable component $g_n(\mathcal{I}_{n,t})$, thereby increasing $\mathcal{ME}$. When speculators have more capital or trade more aggressively, this effect strengthens. In the limit case where all return predictability is eliminated, then $g_n(\mathcal{I}_{n,t})$ is constant and $\mathcal{ME} = 1$.

Proposition 2. *Market efficiency $\mathcal{ME}$ increases in speculator size ω and, for fixed aggregate size ωJ, in the number of speculators J, provided ωJ is large enough.*

The requirement that ωJ is large enough for monotonicity in J is driven by the borrowing constraint. A larger number of speculators trade more aggressively, which improves efficiency over

[11] See Eq. (A21) in the Appendix for the derivation of $g_n(\mathcal{I}_{n,t}), e_{n,t+1}$.

the range of $z_{n,t}$ values where the constraint is not binding, but also implies that the constraint binds over a wider range of $z_{n,t}$ values, partially offsetting this gain. When individual size ω is sufficiently large, the former effect dominates, so market efficiency rises monotonically with both ω and J.

Market Liquidity. We measure liquidity by the price response to an exogenous "shock" in asset supply. Suppose (log) supply increases from s_n to $s_n + \sigma$ at time t before reverting back to s_n at $t+1$. In the case where only the representative investor is present (i.e., assuming $\alpha_t^S = 0$), Eq. (8) implies that the price impact of this supply shock is given by $\frac{\partial p_{n,t}}{\partial \sigma} = -1$, so the price drops one-for-one with supply. In contrast, when speculators are active in the market, they recognize that the price drop is temporary and anticipate an expected capital gain when the supply reverts at $t+1$. As a result, they increase their demand for the asset, partially absorbing the supply shock and reducing its price impact.

We define liquidity as

$$\mathcal{L} = 1 + E\left(\left.\frac{\partial p_{n,t}}{\partial \sigma}\right|_{\sigma=0}\right). \tag{22}$$

Higher values of $\mathcal{L}$ indicate that the market can absorb supply shocks with smaller price distortions, reflecting greater liquidity. In the absence of speculators, the full price impact of a supply shock leads to $\mathcal{L} = 0$. Conversely, if speculators fully absorb the shock, there is no price impact, implying $\mathcal{L} = 1$. As ω (the relative size of speculators) or J (the number of speculators) increase, speculators trade more aggressively on mispricing, allowing the market to absorb larger shocks, thereby reducing price impact and increasing liquidity.

Proposition 3. *Liquidity increases in speculator size ω and, for fixed aggregate size ωJ, in the number of speculators J, provided ωJ is large enough.*

The non-monotonicity in J stems from the same borrowing constraint argument highlighted in Proposition 2, which tempers the liquidity gains from additional speculators unless their individual size ω is sufficiently large.

3 Empirical implementation

In this section, we outline the calibration and estimation of our model. First, we outline the demand estimation procedure for the representative investor, based on Koijen and Yogo (2019). Second, we detail how we simulate stock characteristics, dividends, and latent demand processes. Third, we describe our deep reinforcement learning (DRL) approach and how it is integrated into the environment. Finally, we outline how we run and evaluate our simulation experiments.

3.1 Demand estimation

We calibrate the demand of the representative investor using data on US investors' holdings from SEC 13F filings, combined with asset characteristics from Compustat and CRSP. The sample spans the period from 1982:Q2 to 2021:Q4. We consider log market equity (me) and five additional stock characteristics: log book equity (be), investment growth (inv), dividend-to-book equity (div), profitability (prof), and market beta (mkt).

We begin by estimating individual investors' demand functions using the log-exponential specification of Koijen and Yogo (2019), and then aggregate these estimates to construct a representative demand function. Following Koijen and Yogo (2019), for each investor $i = 1, \ldots, I$ and each quarter $t = 1, \ldots, T$, we estimate the following equation, where the elasticity of investor i is identified using cross-sectional variations in stock characteristics:

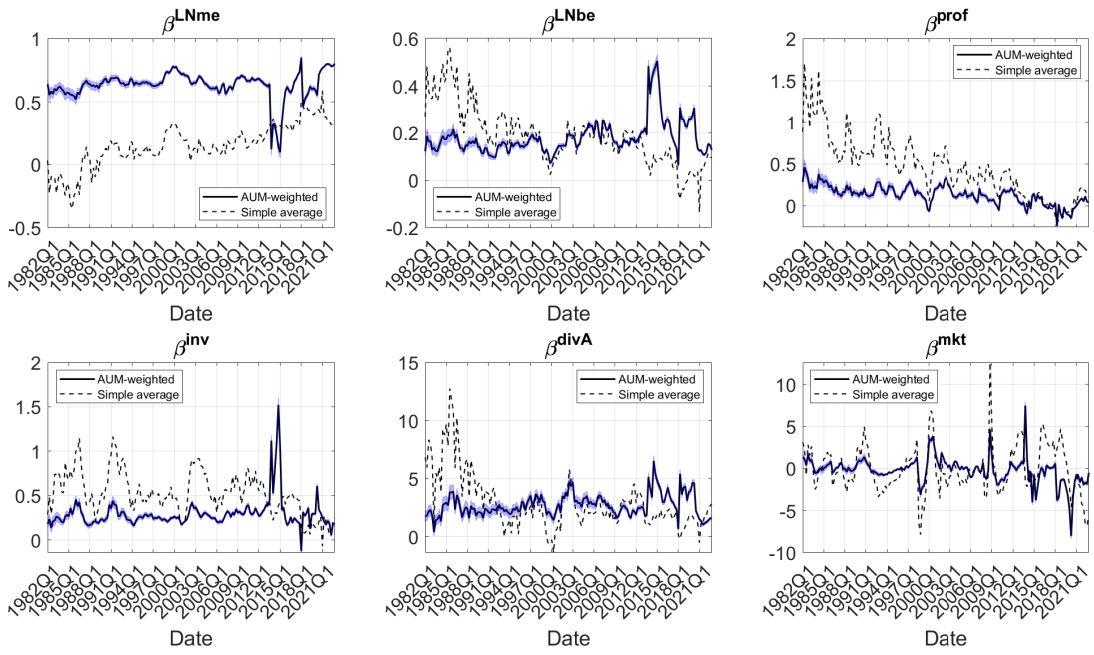
$$\begin{aligned}\frac{w_{i,n,t}}{w_{i,0,t}} = \exp\{&\beta_{i,t}^{me}\,\mathrm{me}_{t,n} + \beta_{i,t}^{be}\,\mathrm{be}_{t,n} + \beta_{i,t}^{prof}\,\mathrm{prof}_{t,n} \\ &+ \beta_{i,t}^{inv}\,\mathrm{inv}_{t,n} + \beta_{i,t}^{div}\,\mathrm{div}_{t,n} + \beta_{i,t}^{mkt}\,\mathrm{mkt}_{t,n} + \beta_{i,t}^{0} + \epsilon_{i,n,t}\}.\end{aligned} \tag{23}$$

The dependent variable represents the holdings (in US dollars) of risky asset n by investor i at time t, relative to the outside asset (i.e., cash and other non-equity holdings). The error term $\epsilon_{i,n,t}$ captures latent demand, reflecting investor sentiment, private information, and beliefs.

To address price endogeneity, we follow Koijen and Yogo (2019) and instrument market equity using its counterfactual value, assuming that all other investors hold an equal-weighted portfolio within their investment universe.

Figure 1 illustrates the estimated demand coefficients, showing how average demand sensitivities (across investors) vary by stock characteristic and over time.

Figure 1: Estimated demand coefficients for individual investors.



Next, we use these estimated coefficients to compute the demand of a representative investor.

3.2 Calibration of the representative investor's demand

The demand coefficients of the representative investor are computed as an assets-under-management (AUM) weighted average of the individual investors' demand coefficients. To ensure time-invariant parameters, we take the AUM-weighted average across the full sample period:

$$\bar{\beta}^k = \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{I}\frac{AUM_{i,t}}{\sum_{i=1}^{I} AUM_{i,t}}\beta_{i,t}^k. \tag{24}$$

Table 1 reports the estimated values of these representative demand coefficients.

Table 1: Demand coefficients of the representative investor

$\bar{\beta}^{me}$	$\bar{\beta}^{be}$	$\bar{\beta}^{prof}$	$\bar{\beta}^{inv}$	$\bar{\beta}^{div}$	$\bar{\beta}^{mkt}$
0.6327	0.1810	0.1274	0.2947	2.6960	-0.1984

Notes: This table reports the demand coefficients used to calibrate the representative investor. Coefficients are an assets-under-management weighted average of individual investors' demand coefficients estimated according to Eq. 23.

Similarly, the latent demand of the representative investor is computed as the AUM-weighted average of individual investors' latent demand:

$$\epsilon_{n,t} = \sum_{i=1}^{I} \frac{AUM_{i,t}}{\sum_{i=1}^{I} AUM_{i,t}} \epsilon_{i,n,t}. \tag{25}$$

This calibrated demand function provides the foundation for the equilibrium price computation in Section 2.2.[12]

3.3 Simulating stock characteristics and latent demand

We select ten stocks that remained active throughout the entire sample period (1982:Q1 to 2021:Q4). Specifically, we aim to obtain a sample that exhibits cross-sectional variation in i) the average dividend yield (D_t/P_{t-1}), ii) the autoregressive coefficient of the latent demand (ρ_ϵ), iii) the autoregressive coefficients of the stock characteristics (ρ_k), and iv) the contribution of the latent demand to the capital gain $\left(\frac{\text{Var}(\xi_{t+1})}{\text{Var}(p_{t+1}-p_t)}\right)$.

To this end, we first apply principal component analysis to the matrix of standardized variables. We then project the data onto the principal component space and compute the pairwise Euclidean distances between all stocks. Finally, we select the ten stocks that are most distant from one another in this space. Put differently, we choose the ten stocks that exhibit the greatest heterogeneity in the variables listed above. Table 2 lists the selected stocks.

Table 2: Company Information and Business Sectors

Ticker	Company Name	Business Sector
IBM	International Business Machines Corporation	Information Technology Services
AXP	American Express Company	Credit Services
ABM	ABM Industries Incorporated	Specialty Business Services
AEE	Ameren Corporation	Utilities - Regulated Electric
WEYS	Weyco Group, Inc.	Footwear & Accessories
GIS	General Mills, Inc.	Packaged Foods
KO	The Coca-Cola Company	Beverages - Non-Alcoholic
L	Loews Corporation	Insurance - Property & Casualty
SJM	The J. M. Smucker Company	Packaged Foods
ARW	Arrow Electronics, Inc.	Electronics & Computer Distribution

Notes: This table reports the ticker, company name and business sector for each stock in our dataset.

For each of the five stock characteristics—log book equity, profitability, investment, dividend-

[12] Consistent with Koijen and Yogo (2019), we hold the representative-investor agent demand elasticities fixed during our experiments in Section 4, so that any variations in prices, return predictability and liquidity can be attributed solely to AI trading rather than to contemporaneous adjustments in the representative-investor demand.

to-book equity, and market beta—as well as for latent demand, we estimate an AR(1) process using the simulated method of moments, based on Eqs. (2)–(3), and using data from CRSP and Compustat. We use the estimated stock-specific AR(1) parameters to generate simulated time series for each variable. For dividend yields, which we model as i.i.d., we instead sample values with replacement from the historical time series. Appendix A.3 provides further details on the estimation process and summary statistics of empirical and simulated stock characteristics and dividend yields.

3.4 Calibration of other parameters

Consumption rate (λ). The consumption rate λ is set to 0.99. Since this parameter does not affect stock returns, it is chosen to ensure that equilibrium price levels are on the same scale as in the data.

Aggregate dividend growth (g). The aggregate dividend growth rate, g, is calibrated by minimizing the mean absolute error (MAE) between empirical and simulated stock returns. We find the optimal value to be 0.4%.

Variance of innovations of latent demand. To align realized and simulated return volatility, we calibrate a parameter (α) to scale the variance of innovations in latent demand. The optimal value of α is determined by minimizing the MAE between the empirical and simulated standard deviations of log stock returns. The estimated value of α is 0.4916.

Risk-free rate (R_f). We calibrate the risk-free rate using the 3-month Treasury bill rate from 1982:Q2 to 2021:Q4, retrieved from FRED Economic Data. The annualized average over this period is 3.54%.

3.5 Reinforcement Learning Model

We model AI traders using the Deep Deterministic Policy Gradient (DDPG) algorithm (Lillicrap et al., 2015), which combines reinforcement learning with deep neural networks. The deep learning component enables the algorithm to operate in continuous state and action spaces and to generalize across unvisited states by learning functional relationships. This is critical in our

setting, where portfolio weights and predictive signals are continuous, and market conditions can move into regions not seen during training.

A high-level description of the algorithm is as follows. The DDPG algorithm is based on an actor–critic architecture. The critic network, parameterized by Θ^Q, approximates the optimal state–action value function $Q^*(\mathcal{I}, \theta)$, which evaluates the expected future rewards of a given state–action pair. The actor network, parameterized by Θ^μ, approximates the optimal policy function $\mu^*(\mathcal{I})$ giving the optimal action given the current state. To stabilize learning, DDPG employs target networks—slowly updated copies of both networks, denoted as $\Theta^{Q'}$ and $\Theta^{\mu'}$, that provide stable targets for learning. Additionally, a replay buffer stores past transitions $\{\mathcal{I}, \theta, R, \mathcal{I}'\}$, breaking temporal correlations and improving learning efficiency.

At each time step, the AI trader observes the current market state $\mathcal{I}$, selects an action θ, and receives a reward R. The reward is defined as the one-period portfolio return as in Eq.(12). The transition tuple $\{\mathcal{I}, \theta, R, \mathcal{I}'\}$ is stored in the replay buffer $\mathcal{B}$. Training updates occur by drawing, at each time step, a minibatch of transitions from the replay buffer via uniform random sampling. Let B denote the index set of the transitions in the minibatch, with cardinality $|B|$. The corresponding target Q-values represent the total expected return from taking the action and then continuing to act optimally in the future:

$$y_i \;=\; R_i \;+\; \gamma\, Q\Big(\mathcal{I}'_i,\, \mu\big(\mathcal{I}'_i\,;\, \Theta^{\mu'}\big)\,;\, \Theta^{Q'}\Big)\,, \quad i \in B \tag{26}$$

where R_i is the immediate reward and the second term in the r.h.s. of the equation represents the discounted continuation value, with γ being the discount factor.

The critic network is trained to predict Q-values, i.e. the expected future reward of a state-action pair ($\mathcal{I}$, θ). Its training implies a loss minimization between the target and predicted Q-values:

$$L\big(\Theta^Q\big) \;=\; \mathbb{E}\Big[\big(y - Q(\mathcal{I}, \theta; \Theta^Q)\big)^2\Big]\,, \tag{27}$$

which is approximated by the minibatch average:

$$\hat{L}\big(\Theta^Q\big) \;=\; \frac{1}{|B|}\sum_{i\in B}\Big(y_i - Q(\mathcal{I}_i, \theta_i; \Theta^Q)\Big)^2. \tag{28}$$

Critic parameters are updated by stochastic gradient descent, changing the neural network weights in the direction that minimizes the loss at learning rate β:

$$\Theta^Q \;\leftarrow\; \Theta^Q \;-\; \beta\, \nabla_{\Theta^Q} \hat{L}\big(\Theta^Q\big), \tag{29}$$

Unlike the critic network, the actor network learns by maximizing the expected reward. Formally, its goal is to find a policy $\mu(\mathcal{I};\Theta^\mu)$ that maximizes the prediction of the critic network $Q\big(\mathcal{I}, \mu(\mathcal{I};\Theta^\mu);\Theta^Q\big)$ over a continuous space:

$$\mathbb{J}(\Theta^\mu) \;=\; \mathbb{E}\big[Q\big(\mathcal{I}, \mu(\mathcal{I};\Theta^\mu);\Theta^Q\big)\big],$$

In practice, the expectation is approximated using a minibatch of transitions:

$$\hat{\mathbb{J}}(\Theta^\mu) \;=\; \frac{1}{|B|}\sum_{i\in B} Q\big(\mathcal{I}_i,\, \mu(\mathcal{I}_i;\Theta^\mu);\Theta^Q\big). \tag{30}$$

Gradient ascent is applied to $\hat{\mathbb{J}}(\Theta^\mu)$. This requires computing the gradient of the Q-function with respect to actions, evaluated at the actor's output. Using the chain rule, the gradient is computed as:

$$\nabla_{\Theta^\mu}\hat{\mathbb{J}}(\Theta^\mu) \;=\; \frac{1}{|B|}\sum_{i\in B} \nabla_\theta Q\big(\mathcal{I}_i,\theta;\Theta^Q\big)\Big|_{\theta=\mu(\mathcal{I}_i;\Theta^\mu)} \nabla_{\Theta^\mu}\mu(\mathcal{I}_i;\Theta^\mu). \tag{31}$$

This gradient decomposes into two components: $\nabla_\theta Q(\mathcal{I}_i,\theta;\Theta^Q)$, which captures the sensitivity of the critic's Q-value prediction to changes in actions, and $\nabla_{\Theta^\mu}\mu(\mathcal{I}_i;\Theta^\mu)$, which captures how the actor's policy responds to changes in its parameters. Overall, these two effects quantify how changes in the actor's parameters induce a change in actions, which in turn influences the critic's evaluation of that action. This provides the direction in which the actor's parameters adjust to maximize expected returns, as estimated by the critic network. The actor parameters are updated by stochastic gradient ascent, at learning rate α:

$$\Theta^\mu \;\leftarrow\; \Theta^\mu \;+\; \alpha\, \nabla_{\Theta^\mu}\hat{\mathbb{J}}(\Theta^\mu). \tag{32}$$

Target networks are updated using a soft update rule that slowly incorporates the current

network parameters. At each training step, the target network parameters are updated according to:

$$\Theta^{\mu'} \leftarrow \tau\,\Theta^{\mu} + (1-\tau)\,\Theta^{\mu'},$$
$$\Theta^{Q'} \leftarrow \tau\,\Theta^{Q} + (1-\tau)\,\Theta^{Q'},$$

where $\tau \ll 1$ is the update rate.

In the subsequent time step, a new transition is added to the replay buffer, a new minibatch is sampled, and the learning protocol repeats.

Our implementation follows standard practices in deep reinforcement learning. Both the actor and critic networks consist of two fully connected hidden layers with 400 and 300 neurons, respectively. The final layer of the actor network employs a *softmax* activation function, ensuring the risky-asset portfolio weight lies in the $[0, 1]$ interval. To enhance exploration, the risky-asset portfolio weight is uniformly distributed over the $[0, 1]$ interval during training. The network weights are initialized using the Glorot normal distribution. The agent learns from mini-batches of size 500, and to ensure sufficient variability in the data we start the learning process when the replay buffer includes at least 10,000 transitions.

Table 3 summarizes the key hyperparameters used in our implementation, following Lillicrap et al. (2015).

Table 3: Hyperparameters for the DDPG algorithm.

τ	0.001	Target update rate
α	0.0001	Actor learning rate
β	0.001	Critic learning rate
$\lvert B\rvert$	500	Minibatch size
$\lvert\mathcal{B}\rvert$	10^5	Replay buffer size
γ	0.99	Discount rate

3.6 Investigation strategy

In our simulations (experiments) we focus on the 10 large-cap U.S. stocks in Table 2; this choice keeps the dimensionality tractable yet reflects a diverse range of fundamentals and latent-demand dynamics. Let $\mathcal{N} = \{\text{IBM}, \ldots, \text{XRX}\}$ denote the set of stocks considered in our simula-

tions. The set $\mathcal{J} = \{1, 2, 5\}$ represents the number of AI traders populating a given simulation, and $\Omega = \{1\%, 5\%, 10\%\}$ specifies the set of AI traders' initial wealth relative to the beginning-of-period market capitalization of the asset. For each triple $(n, J, \omega) \in \mathcal{N} \times \mathcal{J} \times \Omega$ we run a total of S independent simulations.[13] In each simulation, the AI traders face H independent episodes of T time periods each; the time series of stock characteristics and latent demand in each episode is simulated as explained in Section 3.3. AI traders live each episode $\mathcal{T}$ times. We set $S = 50$, $H = 100$, $T = 97$, and $\mathcal{T} = 5$.

In each time period, a trader's information set in asset n is $\mathcal{I}_{n,t} = \{me^*_{n,t}, \{\boldsymbol{x}_{n,t}\}\}$, where $\boldsymbol{x}_{n,t}$ is the vector of stock characteristics for stock n at time t. Given $\mathcal{I}_{n,t}$, the AI trader decides the fraction $\theta^j_{n,t} \in [0, 1]$ of its wealth to allocate to the stock. Given all AIs' portfolio decisions, the market clears and $P_{n,t}$ is determined. Then, the AI observes its reward given by the portfolio return $R^j_{p,t+1}$ and the next-period state $\mathcal{I}_{n,t+1}$. The tuple $(\mathcal{I}_{n,t}, \theta^j_{n,t}, R^j_{p,t+1}, \mathcal{I}_{n,t+1})$ constitutes a transition, which is the basis for algorithmic learning as explained in Section 3.5.

For each simulation s and triple (n, J, ω) we obtain a set of J portfolio policy functions, one per AI trader, such that $\theta^j_{n_s}(\cdot; \omega, J) : \mathcal{I} \to [0, 1]$. We define the average policy across agents as

$$\theta^{AI}_{n_s}(\cdot; \omega, J) = \frac{\sum_{j=1}^{J} \theta^j_{n_s}(\cdot; \omega, J)}{J},$$

and we define the average of this policy across time periods and simulations as

$$\bar{\theta}^{AI}_n(\omega, J) := \frac{\sum_{s=1}^{S}}{S} \frac{\sum_{h=1}^{H}}{H} \frac{\sum_{t_h=1}^{T}}{T} \theta^{AI}_{n_s}(\mathcal{I}_{t_h}; \omega, J). \tag{33}$$

We denote the corresponding quantities in the rational benchmark with $\theta^{RB}_n(\cdot, \omega, J)$ and $\bar{\theta}^{RB}_n(\omega, J)$.

To assess AI traders' performance and market impact, we generate 100 independent out-of-sample episodes of 97 periods each. Drawn from the same process as in training but unseen during learning, these episodes allow us to evaluate portfolio returns and measure market efficiency, liquidity, and volatility out-of-sample.

[13] Running multiple independent simulations mitigates the effect of random neural network weight initialization. Additionally, randomness in exploration and mini-batch selection remains independent across simulations.

4 Experimental results

This section presents the outcomes of our numerical experiments. First, we analyze AI traders' portfolio policies and out-of-sample returns, examining how these are impacted by trader size and competition intensity. Second, we investigate how these trading behaviors affect market efficiency, liquidity, and volatility.

4.1 Portfolio choice and returns

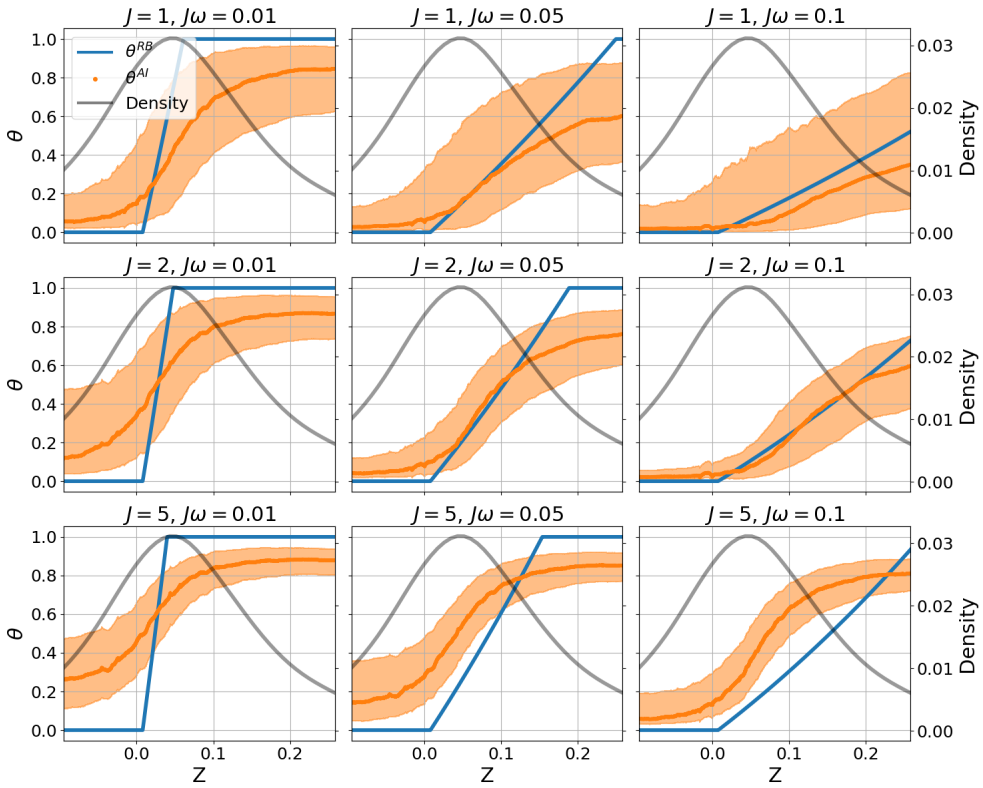
4.1.1 Portfolio policies

Comparative statics in $z_{n,t}$. Figure 2 shows that, across all stocks in our simulations, the AI traders' policy functions are strictly increasing in $z_{n,t}$, the sufficient statistic for $\mathcal{I}_{n,t}$ with respect to the stock's future return. This aligns with Proposition 1. Intuitively, a higher $z_{n,t}$ reflects higher expected capital gains, so both the rational benchmark and the AI traders' portfolio choices are increasing in $z_{n,t}$, suggesting that the DDPG-based investors internalize the fundamental relationship between latent demand, asset characteristics, and future returns in a manner consistent with theory. Figures A2–A11 show that this result is consistent for all ten stocks individually.

The effect of trader size (larger ω for fixed J). Holding the number of AI traders J fixed, increasing $J\omega$ raises the aggregate capital they manage and thus amplifies their price impact. As shown in both Figure 3 and Table 4, the AI traders reduce their holdings as ω increases, consistent with Proposition 1. This reflects a degree of internalization of price impact. However, compared to the rational benchmark, the decline in AI portfolio weights is more muted. This suggests that while the AI traders qualitatively understand the relationship between size and optimal trading intensity, they fall short quantitatively of fully internalizing their price impact.

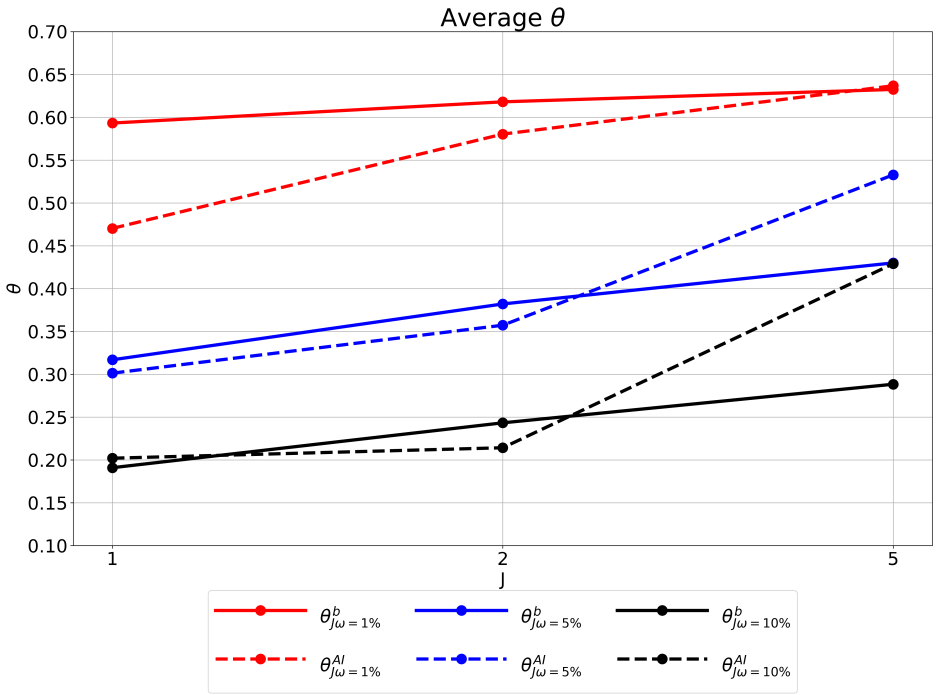
The effect of competition (larger J for fixed $J\omega$). In the rational benchmark, increasing J while holding $J\omega$ constant leads each trader to behave more aggressively, since price impact is distributed across more agents. This comparative static is confirmed in the AI setting as well: the AI portfolio weights tend to increase with J, as seen in both Figure 3 and Table 4. However, the AI traders generally overshoot the rational benchmark—especially at higher levels of J—indicating

Figure 2: Portfolio weights in the risky asset as a function of the sufficient statistic z,



Notes: The figure shows portfolio weights in the risky asset averaged across agents and simulations for AI traders (orange line) and for the rational benchmark (blue line) as a function of the sufficient statistic $z_{n,t}$. The shaded area indicating the 5th–95th percentile range across 50 simulations. The black line displays the empirical probability density function of $z_{n,t}$. All curves are computed separately for each of the ten stocks and then averaged across stocks. The nine panels correspond to different combinations of $(J, J\omega)$.

Figure 3: Average portfolio weights in the risky asset



Notes: This figure reports the average portfolio weight for the rational benchmark, $\bar{\theta}_n^{RB}$ (solid line) and the AI traders $\bar{\theta}_n^{AI}$ (dashed line), averaged across stocks, for $J\omega = 1\%$, (red line), $J\omega = 5\%$ (blue line) and $J\omega = 10\%$ (black line) as function of the number of competing agents, J.

that they only partially internalize the collective impact of their trading. The divergence from the benchmark becomes more pronounced as J increases, reflecting the fact that, under greater competition, each agent finds it more difficult to disentangle its individual effect on prices from the aggregate noise generated by the actions of its peers.

Table 4: Regressions of average portfolio weights in the risky asset

	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB
J=1			-0.165	-0.238	-0.281	-0.374
			(0.019)	(0.015)	(0.021)	(0.016)
J=2	0.059	0.047	-0.106	-0.191	-0.221	-0.326
	(0.020)	(0.014)	(0.039)	(0.028)	(0.041)	(0.029)
J=5	0.208	0.083	0.043	-0.155	-0.072	-0.29
	(0.020)	(0.014)	(0.038)	(0.029)	(0.040)	(0.030)

Notes: This table reports the effects of increasing the number of agents, J, and the relative size of the agents, $J\omega$, on the average portfolio weight on the risky asset, $\bar{\theta}_n^{AI}$ for the AI case and $\bar{\theta}_n^{RB}$ for the rational benchmark (RB). The baseline is $(J = 1, J\omega = 1\%)$. The effects are estimated from regressing $\bar{\theta}_n^{AI}$, $\bar{\theta}_n^{RB}$ on dummies $D_{J=2}$, $D_{J=5}$, $D_{\omega=5\%}$ and $D_{\omega=10\%}$. Stock fixed effect are included. Sample size: 90.

Linear sufficiency. Proposition 1 establishes that in the rational benchmark, portfolio weights depend only on the sufficient statistic $z_{n,t}$, implying that any variation in the information set $\mathcal{I}_{n,t}$ holding $z_{n,t}$ fixed should not affect optimal portfolio choice. Table 5 (and Table A5 for all 10 stocks) tests this prediction for AI traders. For each realized value of $z_{n,t}$ in training data, we generate 100 perturbations of the state variables that leave $z_{n,t}$ unchanged. If the AI policy perfectly implemented linear sufficiency, the resulting portfolio weights would be invariant across those perturbations. In practice, we find that the standard deviation of θ^{AI} conditional on $z_{n,t}$ remains sizable, especially when J or ω is low. This indicates that AI traders do not fully compress the information into $z_{n,t}$, and instead respond to irrelevant variation in $\mathcal{I}_{n,t}$.

Table 5: Average standard deviation of θ^{AI} for given $z_{n,t}$ values

	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$
J=1	0.414	0.355	0.242
	(0.019)	(0.042)	(0.107)
J=2	0.378	0.398	0.321
	(0.041)	(0.027)	(0.095)
J=5	0.352	0.393	0.404
	(0.055)	(0.027)	(0.022)

Notes: This table reports the standard deviation of $\theta^{AI}_{n_s}$ across 100 random realizations of state variables that leave $z_{n,t}$ unchanged, averaged across $z_{n,t}$ values, stocks, and simulations. Standard deviation across stocks in parenthesis.

Explaining the quantitative gap: a learning externality. While the AI traders' behavior aligns qualitatively with the theoretical predictions in Proposition 1, there are notable quantitative deviations. In particular, AI agents tend to trade too aggressively as the number of traders increases, and they fail to sufficiently scale down their holdings when their size grows. These discrepancies point to a learning friction that becomes more pronounced when many AI agents are trained simultaneously.

The core challenge stems from the model-free nature of DDPG. Lacking a structural understanding of how their actions influence prices, the agents must learn from observed reward signals alone. When multiple traders explore concurrently, each agent influences the market price through its own experimentation. This injects noise into the price process and alters the reward landscape faced by its peers. As a result, the environment becomes more difficult to interpret: each AI faces distorted feedback that makes it harder to disentangle its own price impact from the effects of

others. This is the essence of the learning externality—exploration by one agent disrupts learning for the others by contaminating the very signals they rely on.

To isolate this effect, we conduct a controlled experiment. Table 6 (and Table A6 for all 10 stocks individually) reports the average performance difference between an AI trader trained alongside $J-1$ rational speculators (who play the equilibrium strategy of the J-trader game) and the average performance of J AI traders trained jointly in a shared environment. The AI trained with rational peers consistently achieves higher returns, with the gap widening in both J and $J\omega$ in all but one case. This confirms that concurrent exploration by other agents disrupts learning. Beyond hindering price-impact internalization, the externality also weakens the AI's ability to learn predictive relationships from public signals, since price impact and return forecasting are intertwined in a single learned policy.

Table 6: Average portfolio returns difference $\Delta R_{J-1,J}$

	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$
J=2	0.926	1.492	1.139
	(0.473)	(0.616)	(0.997)
J=5	1.323	3.738	5.403
	(0.391)	(0.704)	(1.383)

Notes: This table reports the average percentage difference in portfolio returns between two settings: (i) a single AI trader trained while competing with $J-1$ rational speculators, and (ii) J AI traders trained jointly. For each stock and simulation , the return difference is computed and then averaged across stocks and simulations. Values are expressed as percentages. Standard deviations across stocks are reported in parentheses.

We now examine how these portfolio policy choices translate into portfolio returns, market efficiency, and liquidity.

4.1.2 Returns

Average Returns. Table 7 (and Table A7 for all 10 stocks) shows that in the $J = 1, \omega = 1\%$ case, the AI's average returns closely match those of the rational benchmark, achieving on average 84% of the benchmark returns in out-of-sample data. This suggests that a single AI agent with moderate price impact can learn near-optimal portfolio policies. Notably, this setup closely resembles a partial equilibrium environment, in which the AI is trained on historical data and does not interact with the environment or with other traders.

However, as the aggregate size of AI traders, $J\omega$, increases (for fixed J), their average

returns fall further below the benchmark. This pattern indicates that learning frictions become more severe when portfolio size—and thus price impact—grows. The performance gap is further amplified when multiple AIs are trained simultaneously: each agent's exploration perturbs market prices, distorting the reward signals observed by its peers and hindering learning. This effect is particularly clear when J raises from one to five, both when aggregate AI wealth is held constant and when individual size is fixed–for instance, from $(J, J\omega) = (1, 1\%)$ to $(J, J\omega) = (5, 5\%)$.

The increased sensitivity of performance to learning frictions at larger portfolio sizes can be understood more formally. Let θ_t^* denote the optimal portfolio weight solving Eq. (15), and consider a small deviation ϵ around this optimum. Assuming θ_t^* is interior, a second-order Taylor expansion of the expected return function implies that

$$\mathbb{E}(R_{p,t+1} \mid \mathcal{I}_{n,t})\big|_{\theta_t^*+\epsilon} - \mathbb{E}(R_{p,t+1} \mid \mathcal{I}_{n,t})\big|_{\theta_t^*} \approx \frac{1}{2}\frac{\partial^2 \mathbb{E}(R_{p,t+1} \mid \mathcal{I}_{n,t})}{(\partial\theta_t)^2}\Big|_{\theta_t^*} \times \epsilon^2.$$

Thus, the performance loss from a small portfolio misallocation is proportional to the curvature of the expected return function. Importantly, we can show that this second derivative increases (in absolute value) with portfolio size ω, making the objective more concave. In other words, the cost of small errors in portfolio weights rises with portfolio size, magnifying the consequences of imperfect learning for larger positions.

These results also caution against extrapolating individual performance from partial equilibrium setups. In many empirical studies, AI-based portfolio strategies are evaluated out-of-sample on historical price data. Our findings show that such strategies may appear to perform well in isolation and without feedback effects on the market environment (similarly to the $J = 1, \omega = 1\%$ case), but their performance can degrade substantially when they interact and learn jointly with others, and when the portfolio grows in size. This divergence highlights the need to assess AI portfolio strategies in equilibrium environments to understand their true impact on returns.

4.2 Market outcomes

We now examine how AI traders, through their portfolio policies, affect overall market quality. We focus on three key metrics: market efficiency (how well prices reflect public information), liquidity (the market's ability to absorb supply shocks), and return volatility. Market efficiency

Table 7: Average log portfolio returns

	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	$\Delta_{AI,b}$	$\log(\bar{R}^b)$	$\Delta_{AI,b}$	$\log(\bar{R}^b)$	$\Delta_{AI,b}$	$\log(\bar{R}^b)$
J=1	-0.161	0.056	-0.568	0.040	-1.058	0.032
	(0.271)	(0.048)	(0.344)	(0.041)	(1.023)	(0.034)
J=2	-0.195	0.054	-0.329	0.034	-0.469	0.027
	(0.303)	(0.048)	(0.272)	(0.039)	(0.297)	(0.029)
J=5	-0.212	0.053	-1.399	0.028	-2.804	0.020
	(0.322)	(0.048)	(1.044)	(0.039)	(1.296)	(0.028)

Notes: This table reports average log portfolio returns across stocks. $\Delta_{AI,b}$ represents the percentage deviation of AI traders' average log portfolio returns from the rational benchmark, defined as $\Delta_{AI,b} = \log(R^{AI})/\log(R^b) - 1$. $\log(\bar{R}^b)$ denotes the average log portfolio return of the rational benchmark. For each $(J, J\omega)$ pair, we compute the average out-of-sample returns for each stock and simulation, then take the average across stocks and simulations. Standard deviations across stocks are reported in parentheses.

and liquidity are computed according to the definitions in Section 2.3, averaged across simulations.

Market efficiency. Table 8 (also Table A8 for all 10 stocks individually) reports market efficiency, defined as the share of return variance that is unpredictable given the public information set $\mathcal{I}_{n,t}$, consistent with Eq. (21). Panel A shows that AI traders consistently improve market efficiency relative to the no-AI baseline. As either greater capital is deployed by AIs or competition is more intense, market efficiency generally improves in line with Proposition 2, reflecting the AIs' ability to partially exploit return predictability. However, across all configurations, the case with AI traders lags behind the rational benchmark. Panel B quantifies this gap, showing that it is most pronounced at high levels of $J\omega$. These results confirm that while AI agents make markets more efficient, they remain limited in their ability to eliminate predictability due to learning frictions.

These results highlight an additional key limitation of the partial equilibrium approach to AI, which may suggest that widespread adoption of AI will lead to highly efficient markets. In contrast, our results show that when AI agents learn and interact in equilibrium, frictions emerge that can significantly constrain market-wide efficiency improvements.

Table A9 shows the results for an alternative measure of market efficiency based on the R^2 of predictive OLS regressions of future returns on observable state variables. In other words, in contrast to Table 8 which reflects the true data generating process and is therefore computed "inside the model," Table A9 reflects the perspective of an econometrician who observes the public signals $\mathcal{I}_{n,t}$ but does not know the true return process and approximates the equilibrium function g_n with a linear specification. The results are qualitatively consistent across both approaches.

Table 8: Average market efficiency (as Δ % from $\mathcal{ME}(J=0)$)

Panel A: $\Delta\mathcal{ME}(J, J\omega)$ ($\mathcal{ME}(J=0)=0.867$)						
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB
J=1	1.265	2.041	3.320	6.528	3.165	7.656
	(0.425)	(0.605)	(1.530)	(3.038)	(2.254)	(4.419)
J=2	1.179	2.025	4.426	7.215	5.002	9.209
	(0.401)	(0.609)	(1.823)	(3.080)	(2.478)	(5.077)
J=5	1.106	2.008	4.725	7.529	6.547	10.098
	(0.405)	(0.613)	(1.893)	(3.056)	(3.871)	(5.350)

Panel B: $\Delta\mathcal{ME}^{AI}(J, J\omega)$ - $\Delta\mathcal{ME}^{RB}(J, J\omega)$			
	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$
J=1	-0.776	-3.208	-4.491
J=2	-0.846	-2.789	-4.207
J=5	-0.902	-2.804	-3.551

Notes: This table reports average market efficiency across stocks. Market efficiency is defined as the share of return variance that is unpredictable given the public information set $I_{n,t}$, as in Eq. (21). Panel A shows the percentage deviation from baseline market efficiency with only the representative investor, i.e., $\mathcal{ME}(J=0)$, for both the AI case and the rational benchmark (RB). Panel B reports the difference between the rational benchmark and AI traders' deviations from baseline. For each $(J, J\omega)$ pair, we compute the average out-of-sample market efficiency for each stock and simulation, then take the average across simulations and stocks. Standard deviations across stocks are reported in parentheses.

Market Liquidity. Table 9 (also Table A10 for all 10 stocks individually) shows that the comparative statics of liquidity with respect to AI competition and size are only partially in line with the predictions of Proposition 3. Furthermore, liquidity is an order of magnitude lower relative to the benchmark. The explanation for this result is that in the rational benchmark, traders understand the temporary nature of the shock and anticipate the price reversal in the following period, so they buy aggressively to profit from the anticipated reversal. By contrast, the AI interprets the initial price drop as a sign of reduced latent demand, which is only partly reversed by mean reversion. Thus, the AI perceives the price drop as a weaker signal of price reversal and it does not fully exploit such dips, resulting in lower liquidity provision.

In other words, the AI sees "price down" $\rightarrow$ "latent demand down," which is not a guaranteed short-run arbitrage opportunity. Consequently, it fails to supply liquidity in situations where an investor with structural knowledge of the shock would act aggressively, leading to under-provision of liquidity. This illustrates how structural knowledge—not just pattern recognition—remains crucial for market-stabilizing behavior.

For the case $(J, J\omega) = (5, 10\%)$–where learning frictions are most severe and the deviation from the rational benchmark is largest—AI trading results in even lower liquidity than in the absence of AI traders (the $J=0$ case).

Table 9: Average liquidity, $\mathcal{L}$

Panel A: $\mathcal{L}(J, J\omega)$						
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB
J=1	0.536	2.943	1.038	8.815	0.420	9.715
	(0.218)	(1.735)	(0.393)	(3.162)	(0.492)	(3.052)
J=2	0.496	2.902	1.628	10.532	0.324	12.736
	(0.201)	(1.702)	(0.652)	(4.276)	(0.760)	(4.187)
J=5	0.459	2.870	0.930	11.652	-0.503	15.147
	(0.209)	(1.663)	(0.513)	(5.303)	(0.646)	(5.257)

Panel B: $\mathcal{L}^{AI}(J, J\omega)$ - $\mathcal{L}^{RB}(J, J\omega)$			
	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$
J=1	-2.407	-7.777	-9.295
J=2	-2.406	-8.904	-12.412
J=5	-2.411	-10.722	-15.65

Notes: This table reports average market liquidity across stocks. Liquidity is measured as the price impact of a 1% supply shock, as defined in Eq. (22). AI represents the market liquidity level with AI traders. RB denotes the average liquidity level of the rational benchmark. Entries are multiplied by 100. For each $(J, J\omega)$ pair, we compute the average out-of-sample market liquidity for each stock and simulation, then take the average across simulations and stocks. Panel B reports the difference between the market liquidity level of the AI traders' and the rational benchmark. Standard deviations across stocks are reported in parentheses.

Volatility. Finally, we examine the impact of AI trading on stock return volatility. Panel A of Table 10 (also Table A11 for all 10 stocks individually) reports the percentage deviation of stock return volatility relative to a baseline market populated by the representative investor. The results for the rational benchmark align with the prediction on market efficiency in Proposition 2: the greater the capital allocated to the agents, and the more agents compete (for large enough $J\omega$), the larger the reduction in volatility. This is intuitive because rational agents trade on mean reversion, thereby aligning current prices to future prices and reducing return volatility. The market with AI traders exhibit a similar property, but quantitative disparities with the rational benchmark emerge, especially when multiple AIs compete or when their managed capital is larger. These results further corroborate the learning frictions emerging when AI agents interact with each other and the environment in an equilibrium framework.

Table 10: Average volatility (as $\Delta\%$ from $\sigma(R)_{J=0}$)

Panel A: $\Delta\sigma(\mathcal{R})(J, J\omega)$						
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB
J=1	-0.626	-1.003	-1.607	-3.083	-1.509	-3.562
	(0.208)	(0.293)	(0.720)	(1.366)	(1.055)	(1.928)
J=2	-0.583	-0.996	-2.129	-3.394	-2.386	-4.234
	(0.196)	(0.295)	(0.846)	(1.374)	(1.140)	(2.170)
J=5	-0.548	-0.988	-2.268	-3.535	-3.073	-4.614
	(0.198)	(0.297)	(0.871)	(1.358)	(1.718)	(2.263)

Panel B: $\Delta\sigma(R)^{AI}(J, J\omega)$ - $\Delta\sigma(R)^{RB}(J, J\omega)$			
	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$
J=1	0.377	1.476	2.053
J=2	0.413	1.265	1.848
J=5	0.44	1.267	1.541

Notes: This table reports average return volatility across stocks. Panel A shows the percentage deviation from baseline return volatility with only the representative investor, i.e., $J = 0$, for both the AI case and the rational benchmark (RB). Panel B reports the difference between the rational benchmark and AI traders' deviations from baseline. For each $(J, J\omega)$ pair, we compute the average out-of-sample return volatility for each stock and simulation, then take the average across simulations and stocks. Standard deviations across stocks are reported in parentheses.

5 Discussion

5.1 Learning Externalities in Practice

Our model highlights learning externalities in multi-agent reinforcement learning (MARL), which arise when agents experiment and adapt policies online in a shared environment. While this setup may overstate the extent of real-time (online) learning in financial markets, it serves as a useful benchmark for the potential impact of algorithmic experimentation in dynamic, multi-agent environments because of several reasons.

First, some degree of ongoing adaptation is likely unavoidable. As financial market dynamics evolve over time, strategies trained purely on historical data may fail to generalize out of sample. As a result, models are recalibrated or retrained frequently. Although this adaptation may occur in discrete intervals rather than continuously, it still introduces a form of strategic experimentation: when one agent updates its policy and redeploys, it alters the environment faced by others. In this sense, staggered or asynchronous learning can generate externalities similar to those in fully online MARL.

Even in the extreme case where algorithmic learning is entirely offline, the training data reflects the influence of previously deployed algorithms. As financial institutions test and iterate AI strategies in live markets, their actions shape asset prices. In this broader sense, the market

functions as a "meta-experiment": the behavior of deployed models influences the data on which future models are trained. This feedback loop creates indirect learning externalities that, while subtler than in MARL, are conceptually similar.

5.2 Policy Implications

These considerations have implications for the ongoing policy debate on algorithmic trading. Regulators treat algorithmic trading as a multi-dimensional risk that spans market integrity, operational resilience and systemic coordination failures. To address such concerns, Article 7 of MiFID II RTS-6 obliges investment firms to test algorithms "in controlled environments" before deployment.

Our simulations reveal that algorithmic exploration creates a negative learning externality that impairs learning and market functioning. Therefore, our policy implication is that such a controlled environment should replicate general-equilibrium (GE) feedback while holding rival agents' behavior constant. In practice this means certifying each model inside a single-agent GE sandbox where prices endogenously respond to its own orders; only once the agent has shown impact-aware stability should it enter multi-agent stress tests. This sequencing contrasts with recent proposals which prioritize multi-agent sandboxes to capture interaction effects, overlooking the impact of learning externalities on market outcomes.[14]

Our findings also suggest a broader policy consideration. In our setup, AI agents act as contrarian investors who, in principle, provide liquidity by absorbing transient supply shocks. However, they do so less effectively than fully rational agents because they fail to anticipate the structural nature of temporary price reversals. This underprovision of liquidity implies that learning frictions—rather than collusion or manipulation—can be a source of market fragility. In extreme cases, our simulations show that this may lead to amplification of shocks.

Thus, the model-free nature of algorithmic learning introduces a new channel of systemic risk which arises from endogenous non-stationarity in the learning environment. This suggests that regulators should expand their risk assessment frameworks to include such algorithm-induced informational externalities. Testing protocols should emphasize how agents behave in the presence

[14]See, for example, Jonathan Hall, of the Bank of England's Financial Policy Committee, suggesting that "[a]ny deep trading algorithms will need to be trained extensively, tested in multi-agent sandbox environments" (https://www.bankofengland.co.uk/speech/2024/may/jon-hall-speech-at-the-university-of-exeter).

of structural supply or demand shocks and whether their responses exacerbate or attenuate market volatility.

In sum, safe deployment of AI in trading requires not only robustness to adversarial inputs or algorithmic failures but also learning-aware certification procedures that account for general-equilibrium effects, feedback loops, and potential amplification mechanisms. Future work should explore the design of regulatory sandboxes that can isolate and measure these risks in realistic settings.

6 Conclusion

This paper studies how AI-driven investors, modeled using deep reinforcement learning, trade in an empirically calibrated demand-based asset pricing model with price impact and return predictability. Our work helps fill two gaps in the literature. First, we are the first to study AI-based trader behavior in an equilibrium environment with empirically plausible return predictability and price impact. Second, while recent studies often employ tabular Q-learning schemes with discrete action sets, we adopt a continuous-control DRL method. This allows the trading agent to learn flexibly and avoid coarse discretization of portfolio weights.

Our experimental results deliver several key insights. In line with the theoretical benchmark, the AI agents learn to exploit return predictability arising from mean-reverting fundamentals and latent demand. In particular, the AI policies qualitatively reproduce the benchmark comparative statics, indicating that they learn to decode latent demand shocks from prices and internalize the price impact of their trades.

Yet quantitative differences with the benchmark emerge. When multiple AI traders interact, their actions introduce additional noise that distorts each agent's learning process. This "negative learning externality" degrades performance relative to the benchmark—particularly when trader size increases and competition intensifies. Relatedly, while AI-driven trading consistently improves market efficiency by reducing predictable return components—albeit less than the benchmark—AI agents provide less liquidity than rational speculators would. Their model-free algorithms do not fully anticipate short-lived arbitrage opportunities from transient supply shocks, leading to more pronounced price drops.

These results have important practical implications for how AI-driven investors transform market outcomes. They do reduce return predictability from public signals, thus enhancing semi-strong form market efficiency. At the same time, they provide less liquidity than predicted by rational models, illustrating how structural knowledge—not just pattern recognition—remains crucial for market-stabilizing behavior. This tension highlights the importance of modeling the interplay between traders' learning processes and market dynamics, rather than treating AI strategies as atomistic or frictionless.

References

Abada, I. and X. Lambin (2023): "Artificial Intelligence: Can Seemingly Collusive Outcomes Be Avoided?" *Management Science*, 69, 5042–5065.

Albrecht, S. V., F. Christianos, and L. Schäfer (2024): *Multi-Agent Reinforcement Learning: Foundations and Modern Approaches*, MIT Press.

Bai, J., T. Philippon, and A. Savov (2016): "Have financial markets become more informative?" *Journal of Financial Economics*, 122, 625–654.

Barberis, N. and L. Jin (2023): "Model-free and Model-based Learning as Joint Drivers of Investor Behavior," Available at https://www.law.nyu.edu/sites/default/files/DGJ_AI_Trading_Collusion_Market%20Manipulation_Penn_NYU_Law.pdf.

Calvano, E., G. Calzolari, V. Denicolò, and S. Pastorello (2020): "Artificial Intelligence, Algorithmic Pricing, and Collusion," *American Economic Review*, 110, 3267–3297.

Cartea, Á., S. Jaimungal, and L. Sánchez-Betancourt (2021): "Deep reinforcement learning for algorithmic trading," *Available at SSRN 3812473*.

Colliard, J.-E., T. Focault, and S. Lovo (2023): "Algorithmic Pricing and Liquidity in Securities Markets," Working Paper.

Dou, W. W., I. Goldstein, and Y. Ji (2023): "AI-Powered Trading, Algorithmic Collusion, and Price Efficiency," Available at https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4452704.

Dugast, J. and T. Foucault (2018): "Data abundance and asset price informativeness," *Journal of Financial Economics*, 130, 367–391.

Farboodi, M., A. Matray, L. Veldkamp, and V. Venkateswaran (2022): "Where has all the data gone?" *The Review of Financial Studies*, 35, 3101–3138.

Federal Reserve Board (2022): *Financial Stability Report*, November 2022, Washington, D.C.: Federal Reserve Board.

Fuchs, W., S. Fukuda, and D. Neuhann (2024): "Demand-System Asset Pricing: Theoretical Foundations," Working paper.

Gu, S., B. Kelly, and D. Xiu (2020): "Empirical asset pricing via machine learning," *The Review of Financial Studies*, 33, 2223–2273.

Hambly, B., R. Xu, and H. Yang (2023): "Recent advances in reinforcement learning in finance," *Mathematical Finance*, 33, 437–503.

Heaton, J. B., N. G. Polson, and J. H. Witte (2017): "Deep learning for finance: deep portfolios," *Applied Stochastic Models in Business and Industry*, 33, 3–12.

Jiang, Z., D. Xu, and J. Liang (2017): "A Deep Reinforcement Learning Framework for the Financial Portfolio Management Problem," *arXiv preprint arXiv:1706.10059*.

Johnson, J. P., A. Rhodes, and M. Wildenbeest (2023): "Platform Design When Sellers Use Pricing Algorithms," *Econometrica*, 91, 1841–1879.

Kelly, B., S. Malamud, and K. Zhou (2024): "The virtue of complexity in return prediction," *The Journal of Finance*, 79, 459–503.

Kelly, B. and D. Xiu (2023): "Financial machine learning," *Foundations and Trends® in Finance*, 13, 205–363.

Kilinc, O. and G. Montana (2018): "Multi-agent Deep Reinforcement Learning with Extremely Noisy Observations," .

Koijen, R. S. J. and M. Yogo (2019): "A Demand System Approach to Asset Pricing," Forthcoming at *Journal of Political Economy*.

Lillicrap, T. P., J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra (2015): "Continuous Control with Deep Reinforcement Learning," ArXiv:1509.02971 [cs.LG].

Lopez-Lira, A. (2025): "Can Large Language Models Trade? Testing Financial Theories with LLM Agents in Market Simulations," *arXiv preprint arXiv:2504.10789*.

Lowe, R., Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch (2017): "Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments," in *Advances in Neural Information Processing Systems*, vol. 30, 6382–6393.

Nagel, S. (2021): *Machine learning in asset pricing*, Princeton University Press.

SEC (2020): *The Market Events of May 6, 2010*, Washington, D.C.: U.S. Securities and Exchange Commission (SEC).

van der Beck, P. (2022): "Short- versus Long-run Demand Elasticities in Asset Pricing," Swiss Finance Institute Research Paper 22-67, Swiss Finance Institute.

Vives, X. (2019): "Digital Disruption in Banking," *Annual Review of Financial Economics*, 11, 243–272.

Wang, H. and X. Y. Zhou (2020): "Continuous-time mean–variance portfolio selection: A reinforcement learning framework," *Mathematical Finance*, 30, 1273–1308.

Watkins, C. J. C. H. (1989): "Learning from Delayed Rewards," Ph.D. thesis, King's College, Cambridge.

Yang, H. (2024): "AI Coordination and Self-Fulfilling Financial Crises," Available at https://drive.google.com/file/d/1SCb68JxyXN3Zq35dXrU3ZHs8Ruoxrce-/view.

Yang, H., X.-Y. Liu, and Q. Wu (2018): "A practical machine learning approach for dynamic stock recommendation," in *2018 17th IEEE international conference on trust, security and privacy in computing and communications/12th IEEE international conference on big data science and engineering (TrustCom/BigDataSE)*, IEEE, 1693–1697.

Yang, H., X.-Y. Liu, S. Zhong, and A. Walid (2020): "Deep Reinforcement Learning for Automated Stock Trading: An Ensemble Strategy," Tech. rep.

Zhang, Z., S. Zohren, and S. Roberts (2020): "Deep reinforcement learning for trading," *The Journal of Financial Data Science*, 2, 25–40.

A Appendix

A.1 Model solution

Derivation of Eq. (7). Using the portfolio formula for the representative investor in Eq. (1) together with the representative investor's budget constraint $\sum_{m=1}^{N} w_{m,t} + w_{0,t} + \gamma_t = 1$, we obtain

$$w_{n,t} = \frac{\delta_{n,t}}{1 + \sum_{m=1}^{N} \delta_{m,t}}.$$

The market clearing condition Eq. (6) for each stock can be equivalently written as

$$A_t w_{n,t} = P_{n,t} \tilde{S}_{n,t}. \tag{A1}$$

Using Eq. (1) into the market clearing condition Eq. (A1) we obtain

$$\delta_{n,t} A_t = \left(1 + \sum_{m=1}^{N} \delta_{m,t}\right) P_{n,t} \tilde{S}_{n,t}.$$

Summing the last equation over n and rearranging, we obtain

$$\delta_{n,t} = \frac{P_{n,t} \tilde{S}_{n,t}}{A_t - \sum_{m=1}^{N} P_{m,t} \tilde{S}_{m,t}}.$$

Using Eq. (1) to substitute for $\delta_{n,t}$ in the last equation and taking logs, we obtain

$$\beta_0 (p_{n,t} + s_n) + \sum_{k=1}^{K-1} \beta_k x_{k,n,t} + \beta_K + \epsilon_{n,t} = p_{n,t} + \tilde{s}_{n,t} - \log\left(A_t - \sum_{m=1}^{N} P_{m,t} \tilde{S}_{m,t}\right). \tag{A2}$$

The budget constraint for trader j is

$$A_t^j = \sum_{m=1}^{N} S_{m,t}^j P_{m,t} + S_{0,t}^j. \tag{A3}$$

Combining Eq. (A3) with the traders' wealth dynamics in Eq. (5) we obtain

$$\sum_{m=1}^{N} S_{m,t}^j P_{m,t} + S_{0,t}^j = \sum_{m=1}^{N} S_{m,t-1}^j \left(P_{m,t} + D_{m,t}\right) + S_{0,t-1}^j R_f. \tag{A4}$$

Using the representative investor's wealth dynamics Eq. (4) and Eq. (A4) into Eq. (A2) and rearranging, we obtain Eq. (7) in the text. □

Derivation of Eq. (8). We define the aggregate dividend $D_{M,t}$ as

$$D_{M,t} = \sum_{m=1}^{N} S_m D_{m,t}.$$

Under Assumption 1, the argument of the logarithm in the r.h.s. of Eq. (7) reduces to $D_{M,t} + S_{0,t-1}R_f$. With Assumption 2, the representative investor's budget constraint reads

$$A_t = \sum_{m=1}^{N} \tilde{S}_{m,t} P_{m,t} + S_{0,t} + \lambda(D_{M,t} + S_{0,t-1}R_f),$$

which, together with Eq. (4) and Assumption 1, imply

$$S_{0,t} = (1-\lambda)(D_{M,t} + S_{0,t-1}R_f).$$

Using the previous equation recursively, we obtain

$$D_{M,t} + S_{0,t-1}R_f = \sum_{\tau=0}^{t} D_{M,t-\tau}(1-\lambda)^\tau R_f^\tau. \tag{A5}$$

Assumptions 3 and 4 imply

$$\sum_{\tau=0}^{t} D_{M,t-\tau}(1-\lambda)^\tau R_f^\tau = D_{M,t}\frac{(1+g)\left[1-\left(\frac{R_f(1-\lambda)}{1+g}\right)^{t-1}\right]}{1+g-R_f(1-\lambda)} \approx D_{M,t}\frac{(1+g)}{1+g-R_f(1-\lambda)},$$

where the approximation is exact for $t \to \infty$. Taken together, Assumptions 1-4 imply that the price equation Eq. (7) simplifies to

$$p_{n,t} = \frac{\beta_0 s_n - \tilde{s}_{n,t} + \sum_{k=1}^{K-1}\beta_k x_{k,n,t} + \beta_K + \epsilon_{n,t} + \log(D_{M,t}) + \phi}{1-\beta_0}, \tag{A6}$$

where

$$\phi = \log\left(\frac{(1+g)}{1+g-R_f(1-\lambda)}\right).$$

Using the definition $\alpha^a_{n,t} = \frac{S^a_{n,t}}{S_n}$ into Eq. (A6) and simplifying we obtain Eq. (8) in the text. □

A.2 Proofs of propositions

Note: In the following proofs we denote $E_t(\cdot) = E(\cdot \mid \mathcal{I}_{n,t})$ and $Var_t(\cdot) = Var(\cdot \mid \mathcal{I}_{n,t})$.

Proof of Proposition 1

Part-(i): Existence and uniqueness We begin by showing that speculators' objective functions

are concave in portfolio shares. Using Eqs. (15)-(16) we obtain

$$\frac{\partial^2 E_t(R^j_{p,t})}{(\partial\theta^j_{n,t})^2} = E_t\Bigg\{ -\frac{P_{n,t+1}}{P_{n,t}}\Bigg[\left(\frac{1}{P_{n,t}}\frac{\partial P_{n,t}}{\partial\theta^j_{n,t}}\right)\left(1-\frac{\theta^j_{n,t}}{P_{n,t}}\frac{\partial P_{n,t}}{\partial\theta^j_{n,t}}\right) + \frac{A^s_t\left[(1-\beta_0)\,S_n\,P_{n,t}\left(1-\frac{\theta^j_{n,t}}{P_{n,t}}\frac{\partial P_{n,t}}{\partial\theta^j_{n,t}}\right)+\beta_0\sum_{i\neq j}^{J}\theta^i_{n,t}A^s_t\right]}{\left[(1-\beta_0)\,S_n\,P_{n,t}+\beta_0\sum_{i=1}^{J}\theta^i_{n,t}A^s_t\right]^2}\Bigg]\Bigg\}. \tag{A7}$$

Since $\frac{\theta^j_{n,t}}{P_{n,t}}\frac{\partial P_{n,t}}{\partial\theta^j_{n,t}} < 1$ (see Eq. (A9) below), then Eq. (A7) is indeed strictly negative.

Next, combine Eqs. (15)-(16) as follows:

$$\frac{\partial E_t(R^j_{p,t})}{\partial\theta^j_{n,t}} = E_t\left[\frac{P_{n,t+1}}{P_{n,t}}\left(1-\frac{\theta^j_{n,t}A^s_t}{(1-\beta_0)S_nP_{n,t}+\beta_0\sum_{i=1}^{J}\theta^i_{n,t}A^s_t}\right)\right] + E_t(DY_{n,t+1}) - R_f := \delta^j \tag{A8}$$

Eq. (A8) implies that if an equilibrium exists, it must be unique. To see this, assume $\theta^j_{n,t} > \theta^i_{n,t}$, in which case Eq. (A8) implies $\delta^j < \delta^i$. In case both $\theta^j_{n,t}, \theta^i_{n,t} \in (0,1)$, optimality requires $\delta^j = \delta^i = 0$, a contradiction. In case $\theta^j_{n,t} = 1, \theta^i_{n,t} \in [0,1)$, optimality requires $\delta^j \geqslant 0, \delta^i \leqslant 0$, again a contradiction.

Using the definitions $\theta^j_{n,t} = \frac{S^j_{n,t}P_{n,t}}{A^s_t}$, $\alpha^j_{n,t} = \frac{S^j_{n,t}}{S_n}$ and $\alpha^S_{n,t} = \sum_{i=1}^{J}\alpha^i_{n,t}$, we can rewrite Eq. (16) as

$$\frac{\theta^j_{n,t}}{P_{n,t}}\frac{\partial P_{n,t}}{\partial\theta^j_{n,t}} = \frac{\theta^j_{n,t}A^j_t}{(1-\beta_0)S_nP_{n,t}+\beta_0\sum_{i=1}^{J}\theta^i_{n,t}A^i_t} = \frac{\alpha^j_{n,t}}{(1-\beta_0)+\beta_0\alpha^S_{n,t}}. \tag{A9}$$

We proceed to show that a symmetric equilibrium exists and is unique. We will first solve for an equilibrium in terms of $\alpha^S_{n,t}$. Given $\alpha^S_{n,t}$, and using the definitions $\theta^j_{n,t} = \frac{P_{n,t}S^j_{n,t}}{A^s_t}$ and $A^s_t = \omega P^*_{n,t}S_n$, and the fact that $P^*_{n,t}/P_{n,t} = \left(1-\alpha^S_{n,t}\right)^{\frac{1}{1-\beta_0}}$, we can uniquely determine the equilibrium portfolio share of each speculator as

$$\theta_{n,t} = \frac{\alpha^S_{n,t}}{J\omega\left(1-\alpha^S_{n,t}\right)^{\frac{1}{1-\beta_0}}}. \tag{A10}$$

Since each $\theta_{n,t} \in [0,1]$, the symmetric equilibrium must satisfy $\alpha^S_{n,t} \in [0,\bar{\alpha}]$ where $\bar{\alpha}$ solves

$$1 = \frac{\bar{\alpha}}{J\omega\left(1-\bar{\alpha}\right)^{\frac{1}{1-\beta_0}}}. \tag{A11}$$

Using Eqs. (2)-(3) and the price functions in Eqs. (13)-(14) we can compute the expected capital gain as

$$E_t\left(\frac{P_{n,t+1}}{P_{n,t}}\right) = \exp\left(-\frac{\sum_{k=1}^{K-1}\beta_k(1-\rho_{k,n})x_{k,n,t}+(1-\rho_{\epsilon_n})E_t\left(\epsilon_{n,t}\right)}{1-\beta_0}\right)\Phi_n\left(1-\alpha^S_{n,t}\right)^{\frac{1}{(1-\beta_0)}}, \tag{A12}$$

where

$$\Phi_n = \exp\left(\frac{\sum_{k=1}^{K-1}\beta_k c_{n,k} + c_{n,\epsilon} + \log(1+g)}{1-\beta_0} + \frac{\sum_{k=1}^{K-1}\beta_k^2\sigma_{\eta_{k,n}}^2 + \sigma_{\xi n}^2}{2(1-\beta_0)^2}\right) \tag{A13}$$

and, as already derived in the main text,

$$E_t(\epsilon_{n,t}) = (1-\beta_0)me_{n,t}^* - \left(\sum_{k=1}^{K-1}\beta_k x_{k,n,t} + \beta_K + \phi\right) = \epsilon_{n,t}. \tag{A14}$$

We define $z_{n,t}$ to be the following linear combination of $me_{n,t}^*, x_{1,n,t}, \ldots, x_{K-1,n,t}$:

$$\begin{aligned} z_{n,t} = &\frac{\sum_{k=1}^{K-1}\beta_k c_{n,k} + c_{n,\epsilon} + \log(1+g) + (1-\rho_{\epsilon_n})(\beta_K+\phi)}{1-\beta_0} + \frac{\sum_{k=1}^{K-1}\beta_k^2\sigma_{\eta_{k,n}}^2 + \sigma_{\xi n}^2}{2(1-\beta_0)^2} \\ &+ \frac{\sum_{k=1}^{K-1}\beta_k(\rho_{k,n}-\rho_{\epsilon_n})}{1-\beta_0}x_{k,n,t} - (1-\rho_{\epsilon_n})me_{n,t}^*. \end{aligned} \tag{A15}$$

Eqs. (A12)-(A14) imply that we can write the expected capital gain in Eq. (A12) as

$$E_t\left(\frac{P_{n,t+1}}{P_{n,t}}\right) = \exp(z_{n,t})\left(1-\alpha_{n,t}^S\right)^{\frac{1}{1-\beta_0}}. \tag{A16}$$

It follows that speculators' expected return can be expressed as a function of $z_{n,t}$ alone.

Consider the case where all J speculators choose the same strategy $\theta_{n,t} = \frac{x}{J\omega(1-x)^{\frac{1}{1-\beta_0}}}$, so that $\alpha_{n,t}^S = x$ by Eq. (A10). Using Eqs. (A8)-(A9) and Eq. (A16), we can write the the first derivative of a speculator's objective function as

$$\frac{\partial E_t(R_{p,t}^j)}{\partial\theta_{n,t}^j}|_{\{\theta_{n,t}^i=\theta_{n,t}\forall i\}} = G(x, z_{n,t}),$$

where we define

$$G(x, z_{n,t}) = \exp(z_{n,t})(1-x)^{\frac{1}{(1-\beta_0)}}\left(1 - \frac{x/J}{(1-\beta_0)+\beta_0 x}\right) + \overline{DY}_n - R_f. \tag{A17}$$

With this formulation, a symmetric equilibrium of the trading game is a fraction $\alpha_{n,t}^S \in [0, \bar{\alpha}]$ with $\bar{\alpha} \in (0,1)$ such that either (i) $\alpha_{n,t}^S = 0$ and $G(0, z_{n,t}) \leqslant 0$, or (ii) $\alpha_{n,t}^S \in (0,1)$ and $G\left(\alpha_{n,t}^S, z_{n,t}\right) = 0$, or (iii) $\alpha_{n,t}^S = \bar{\alpha}$ and $G(\bar{\alpha}, z_{n,t}) \geqslant 0$.

Eq. (A17) implies that G is strictly decreasing in the first argument and and strictly increasing in the second argument, and, furthermore, that $\lim_{z\downarrow-\infty} G(x,z) < 0$ and $\lim_{z\uparrow\infty} G(x,z) = \infty$.[15] This implies that there exist values $z_L < z_H$ such that $G(0, z_L) = G(\bar{\alpha}, z_H) = 0$. It is immediate, therefore, that an equilibrium exists and is unique for any $z_{n,t}$, and is such that $\alpha_{n,t}^S = 0$ for all $z_{n,t} \leqslant z_L$ and $\alpha_{n,t}^S = \bar{\alpha}$ for all $z_{n,t} \geqslant z_H$, whereas for all $z_{n,t} \in (z_L, z_H)$, equilibrium $\alpha_{n,t}^S$ is interior

[15] This argument implicitly assumes $\overline{DY}_n - R_f < 0$.

and solves

$$G\left(\alpha_{n,t}^{S}, z_{n,t}\right) = 0. \tag{A18}$$

□

Part-(ii): Linear sufficiency. Since $G\left(\alpha_{n,t}^{S}; z_{n,t}\right)$ is strictly decreasing in $\alpha_{n,t}^{S}$ and strictly increasing in $z_{n,t}$, implicit differentiation of Eq. (A18) implies that $\alpha_{n,t}^{S}$ is strictly increasing in $z_{n,t}$ in an interior equilibrium.

The equilibrium portfolio share $\theta_{n,t}$ corresponding to each $\alpha_{n,t}^{S}$ is determined by Eq. (A10), and is strictly increasing in $\alpha_{n,t}^{S}$. Since $\alpha_{n,t}^{S}$ is only a function of $z_{n,t}$, it follows that $\theta_{n,t}$ is only a function of $z_{n,t}$ and is weakly increasing.

Finally, using the price equations in Eqs. (13)–(14) and the definition of $z_{n,t}$ in Eq. (A15) we can write the capital gain as

$$\frac{P_{n,t+1}}{P_{n,t}} = \exp\left(z_{n,t} + \frac{\sum_{k=1}^{K-1}\beta_k\eta_{k,n,t+1} + \xi_{n,t+1}}{(1-\beta_0)} - \frac{\sum_{k=1}^{K-1}\beta_k^2\sigma_{\eta_{k,n}}^2 + \sigma_{\xi n}^2}{2(1-\beta_0)^2}\right)\left(1-\alpha_{n,t}^{S}\right)^{\frac{1}{1-\beta_0}}. \tag{A19}$$

Since $\alpha_{n,t}^{S}$ is only a function of $z_{n,t}$ and the dividend yield is i.i.d., $z_{n,t}$ is indeed a sufficient statistic for $\mathcal{I}_{n,t}$ with respect to $R_{n,t+1}$.

Part-(iii): comparative statics Eq. (A15) implies that $z_{n,t}$ is decreasing in $me_{n,t}^{*}$ and increasing in $x_{k,n,t}$ if and only if $\beta_k(\rho_{n,k} - \rho_\epsilon) > 0$. Since $\theta_{n,t}$ is only a function of $z_{n,t}$ and is weakly increasing, then $\theta_{n,t}$ is weakly decreasing in $me_{n,t}^{*}$ and is weakly increasing in $x_{k,n,t}$ if and only if $\beta_k(\rho_{n,k}-\rho_\epsilon) > 0$.

Since $G\left(\alpha_{n,t}^{S}; z_{n,t}\right)$ is independent of ω, then so is $\alpha_{n,t}^{S}$. Therefore, the equilibrium portfolio share is decreasing in ω by Eq. (A10). Finally, since $G\left(\alpha_{n,t}^{S}; z_{n,t}\right)$ is increasing in J, implicit differentiation of Eq. (A18) implies that $\alpha_{n,t}^{S}$ is also increasing in J. Then, for fixed ωJ, Eq. (A10) implies that the equilibrium portfolio share is increasing in J. □

Proof of Proposition 2. Taking the logarithm of Eqs. (A19) and using the definition of $z_{n,t}$ in Eq. (A15) we can write

$$p_{n,t+1} - p_{n,t} = g_n(\mathcal{I}_{n,t}) + e_{n,t+1}, \tag{A20}$$

where

$$g_n(\mathcal{I}_{n,t}) = -\frac{\sum_{k=1}^{K-1}\beta_k^2\sigma_{\eta_{k,n}}^2 + \sigma_{\xi n}^2}{2(1-\beta_0)^2} + z_{n,t} + \frac{\log\left(1-\alpha_{n,t}^{S}\right)}{1-\beta_0}; \quad e_{n,t+1} = \frac{\sum_{k=1}^{K-1}\beta_k\eta_{k,n,t+1} + \xi_{n,t+1}}{(1-\beta_0)}. \tag{A21}$$

Therefore,

$$\mathcal{ME} = \frac{\text{Var}(e_{n,t+1})}{\text{Var}\left(z_{n,t} + \frac{\log\left(1-\alpha_{n,t}^{S}\right)}{1-\beta_0}\right) + \text{Var}(e_{n,t+1})}. \tag{A22}$$

Since $e_{n,t+1}$ is exogenous and independent of ω and J, we will show that the first term in the denominator of Eq. (A22) is decreasing in ω and is decreasing in J for ωJ sufficiently high.

For an arbitrary parameter y (later either $y = J$ or $J = \omega$) set

$$f(z;y) = z + \frac{\log\big(1-\alpha(z;y)\big)}{1-\beta_0},$$

where $\alpha(z;y)$ denotes $\alpha^S_{n,t}$ valued at $z_{n,t} = z$ and indexed by the parameter y. We note that the equilibrium condition Eq. (A18) and Eq. (A17) imply that $f(z;y)$ is strictly increasing in z. Omitting subscripts, we denote $V(y)$ the first variance term in the denominator of Eq. (A22):

$$V(y) = \int_{-\infty}^{\infty} f(z;y)^2\, dP(z) - \Big(\int_{-\infty}^{\infty} f(z;y)\, dP(z)\Big)^2,$$

where P denotes the unconditional distribution of the random variable z. We have:

$$\frac{dV(y)}{dy} = 2\int_{-\infty}^{\infty} f(z;y)\,\frac{\partial f(z;y)}{\partial y}\, dP(z) - 2\Big(\int_{-\infty}^{\infty} f(z;y)\, dP(z)\Big)\Big(\int_{-\infty}^{\infty} \frac{\partial f(z;y)}{\partial y}\, dP(z)\Big),$$

where

$$\frac{\partial f(z;y)}{\partial y} = -\frac{1}{(1-\beta_0)\big(1-\alpha(z;y)\big)}\,\frac{\partial \alpha(z;y)}{\partial y}.$$

The proof of Proposition 1 implies $\alpha(z;y) = 0$ for $z \leqslant z_L$ and $\alpha(z;y) = \bar{\alpha}(y)$ for $z \geqslant z_H$. Hence,

$$\begin{aligned}\frac{dV(y)}{dy} = &-\frac{2}{1-\beta_0}\Big[\int_{z_L}^{z_H} f(z;y)\frac{1}{1-\alpha(z;y)}\frac{\partial\alpha(z;y)}{\partial y}\, dP(z) + \int_{z_H}^{\infty} f(z;y)\frac{1}{1-\bar{\alpha}(y)}\frac{d\bar{\alpha}(y)}{dy}\, dP(z)\Big]\\ &+\frac{2}{1-\beta_0}\Big(\int_{-\infty}^{\infty} f(z;y)\, dP(z)\Big)\Big[\int_{z_L}^{z_H}\frac{1}{1-\alpha(z;y)}\frac{\partial\alpha(z;y)}{\partial y}\, dP(z) + \int_{z_H}^{\infty}\frac{1}{1-\bar{\alpha}(y)}\frac{d\bar{\alpha}(y)}{dy}\, dP(z)\Big].\end{aligned}$$

Case $y = \omega$. By Eq. (A17) and Eq. (A11),

$$\frac{\partial\alpha(z;\omega)}{\partial\omega} = 0 \quad (z_L < z < z_H), \qquad \frac{d\bar{\alpha}(\omega)}{d\omega} > 0.$$

Let $\mu_\omega := \mathbb{E}[f(z;\omega)]$. Then

$$\frac{dV(\omega)}{d\omega} = \frac{2}{(1-\beta_0)\big(1-\bar{\alpha}(\omega)\big)}\frac{d\bar{\alpha}(\omega)}{d\omega}\,\mathbb{E}\big[(\mu_\omega - f(z;\omega))\mathbf{1}_{\{z\geqslant z_H\}}\big].$$

Since $f(\cdot,\omega)$ is strictly increasing, $\mathbb{E}[f(z;\omega)\mid z\geqslant z_H] > \mu_\omega \Rightarrow \mathbb{E}[(\mu_\omega - f(z;\omega))\mathbf{1}_{\{z\geqslant z_H\}}] < 0$, and therefore

$$\frac{dV(\omega)}{d\omega} < 0. \quad \square$$

Case $y = J$. Fix $J\omega$. By Eq. (A17) and Eq. (A11),

$$\frac{\partial\alpha(z;J)}{\partial J} > 0 \quad (z_L < z < z_H), \qquad \frac{d\bar{\alpha}(J)}{dJ} = 0. \tag{A23}$$

Put $\mu_J := \mathbb{E}[f(z;J)]$. The derivative reduces to

$$\frac{dV(J)}{dJ} = 2\int_{z_L}^{z_H} \big(\mu_J - f(z;J)\big)h(z;J)dP(z),$$

where we define

$$h(z;J) = \frac{1}{(1-\beta_0)(1-\alpha(z;J))}\,\frac{\partial\alpha(z;J)}{\partial J}. \tag{A24}$$

Since z_L, z_H are defined such that $G(0,z_L) = G(\bar{\alpha},z_H) = 0$, then Eqs. (A11) and (A17) imply that z_L is independent of ω whereas z_H is increasing in ω with $\lim_{\omega\uparrow\infty} z_H = \infty$. Thus,

$$\lim_{\omega\uparrow\infty}\frac{dV(J)}{dJ} = \mathbb{E}[(\mu_J - f(z;J))h(z;J)\mathbf{1}_{\{z\geqslant z_L\}}]. \tag{A25}$$

We now prove that under the standing assumptions that $h(z;J)$ is non-decreasing in z, the limit in (A25) is strictly negative:

$$\lim_{\omega\uparrow\infty}\frac{dV(J)}{dJ} = \mathbb{E}\big[(\mu_J - f(z;J))\,h(z;J)\,\mathbf{1}_{\{z\geqslant z_L\}}\big] \;< 0.$$

To this end, introduce the conditional expectation $\mathbb{E}_L[\cdot] := \mathbb{E}[\cdot \mid z \geqslant z_L]$ and set $p := \mathbb{P}(z \geqslant z_L)$. Then,

$$\mathbb{E}\big[(\mu_J - f(z;J))h(z;J)\mathbf{1}_{\{z\geqslant z_L\}}\big] = p\Big[\mu_J\,\mathbb{E}_L\big[h(z;J)\big] - \mathbb{E}_L\big[f(z;J)h(z;J)\big]\Big]. \tag{A26}$$

Because both $f(\cdot;J)$ and $h(\cdot;J)$ are non-decreasing,

$$\mathrm{Cov}_L\big(f(z;J),h(z;J)\big) = \mathbb{E}_L[f(z;J)\,h(z;J)] - \mathbb{E}_L[f(z;J)]\,\mathbb{E}_L[h(z;J)] \;\geqslant 0,$$

so that

$$\mathbb{E}_L\big[f(z;J)h(z;J)\big] \;\geqslant\; \mathbb{E}_L\big[f(z;J)\big]\,\mathbb{E}_L\big[h(z;J)\big]. \tag{A27}$$

Inserting (A27) into (A26) we obtain

$$\mathbb{E}\big[(\mu_J - f(z;J))h(z;J)\mathbf{1}_{\{z\geqslant z_L\}}\big] \;\leqslant\; p\,\mathbb{E}_L\big[h(z;J)\big]\,\big[\mu_J - \mathbb{E}_L\big[f(z;J)\big]\big] < 0,$$

where the last inequality follows from the fact that $p > 0$ (the unconditional distribution of $z$has support above z_L), $\mathbb{E}_L\big[h(z;J)\big] > 0$ (by Eq. (A23)) and, since $f(z;J)$ is strictly increasing in z, $\mathbb{E}_L\big[f(z;J)\big] > \mu_J$. To complete the proof it remains to show $\frac{1}{(1-\beta_0)(1-\alpha(z;J))}\,\frac{\partial\alpha(z;J)}{\partial J}$ in non-decreasing in z, which, using implicit differentiation of Eq. (A17), can be verified to be true.

□

Proof of Proposition 3 The supply shock implies that $p_{n,t}$ equals

$$p_{n,t} = -(s_n + \sigma) + \frac{-\log\left(1-\alpha^S_{n,t}\right) + \sum_{k=1}^{K-1}\beta_k x_{k,n,t} + \beta_K + \epsilon_{n,t} + \log\left(D_{M,t}\right) + \phi}{1-\beta_0}, \tag{A28}$$

And therefore the expected capital gain changes to

$$E_t\left(\frac{P_{n,t+1}}{P_{n,t}}\right) = \exp\left(z_{n,t}+\sigma\right)\left(1-\alpha^S_{n,t}\right)^{\frac{1}{(1-\beta_0)}}. \tag{A29}$$

The equilibrium condition is as in the proof of Proposition 1. Denote the resulting equilibrium value $\alpha^S(z_{n,t}+\sigma)$. Therefore, the definition of liquidity implies

$$\mathcal{L} = 1 + E\left(\frac{\partial p_{n,t}}{\partial\sigma}\bigg|_{\sigma=0}\right) = E\left(\frac{\partial}{\partial\sigma}\big[-\log\big(1-\alpha^S(z_{n,t}+\sigma)\big)\big]\bigg|_{\sigma=0}\right)\frac{1}{1-\beta_0}. \tag{A30}$$

The proof for the comparative statics of $\mathcal{L}$ with respect to ω and J follows similar steps as in the proof of Proposition 2 and is omitted for brevity.

For the case with AI trading, the supply shock changes the state variable $me^*_{n,t}$ to $me^*_{n,t} - \sigma$. Thus, defining the aggregate share of supply held by the AI traders as $\alpha^{AI}\big(me^*_{n,t},\{\boldsymbol{x}_{n,t}\}\big) = \sum_{j=1}^{J}\alpha^j\big(\mathcal{I}_{n,t}\big)$, the definition of liquidity implies

$$\mathcal{L} = E\left(\frac{\partial}{\partial\sigma}\left[-\log\left(1-\alpha^{AI}\big(me^*_{n,t}-\sigma,\{\boldsymbol{x}_{n,t}\}\big)\right)\right]\bigg|_{\sigma=0}\right)\frac{1}{1-\beta_0}. \tag{A31}$$

Numerically, we approximate the derivatives

$$\frac{\partial}{\partial\sigma}\big[-\log\big(1-\alpha^S(z_{n,t}+\sigma)\big)\big]\bigg|_{\sigma=0} \approx -\frac{1}{\sigma}\log\left(\frac{1-\alpha^S(z_{n,t}+\sigma)}{1-\alpha^S(z_{n,t})}\right), \tag{A32}$$

and

$$\frac{\partial}{\partial\sigma}\left[-\log\left(1-\alpha^{AI}\big(me^*_{n,t}-\sigma,\{\boldsymbol{x}_{n,t}\}\big)\right)\right]\bigg|_{\sigma=0} \approx -\frac{1}{\sigma}\log\left(\frac{1-\alpha^{AI}\big(me^*_{n,t}-\sigma,\{\boldsymbol{x}_{n,t}\}\big)}{1-\alpha^{AI}\big(me^*_{n,t},\{\boldsymbol{x}_{n,t}\}\big)}\right) \tag{A33}$$

for a numerical value of $\sigma = \log(1.01)$, corresponding to a 1% increase in the supply of the asset.

A.3 Simulating stock characteristics and latent demand

The stock characteristics log book equity, profitability, investment, dividend-to-book equity, market beta and latent demand are simulated from an AR(1) process such that:

$$x_{k,n,t+1} = c_{k,n} + \rho_{k,n}x_{k,n,t} + \eta_{k,n,t+1}, \tag{A34}$$

$$\epsilon_{n,t+1} = c_{\epsilon,n} + \rho_{\epsilon,n}\epsilon_{n,t} + \xi_{n,t+1}, \tag{A35}$$

where $x_{k,n}$ is the $k-th$ the stock's characteristics, $c_{k,n}$ is the intercept, ρ_k is the autoregressive coefficient and $\eta_{k,n,t+1}$ is the error term and similarly for the latent demand. The AR(1) process is estimated by simulated method of moments, by imposing the following moment conditions on the data:

$$\mathbb{E}_t[x_{k,n,t+1}] = \frac{c_{k,n}}{1-\rho_{k,n}}$$
$$Var_t[x_{k,t,t+1}] = \frac{\sigma^2_{k,n^\varepsilon}}{1-\rho^2_{k,n}}$$
$$\frac{\mathbb{E}_t[y_{k,n,t+1}y_{k,n,t}]}{Var_t[y_{k,n,t+1}]} = \rho_{k,n}$$

By exploiting empirical moments, we recover the parameters of the AR(1) process. Stock characteristics are then simulated accordingly using the empirical mean as initial condition.

We construct the dividend yield, $\frac{D_{n,t}}{P_{n,t-1}}$, by sampling with replacement from the data. Finally, we simulate returns by solving Eq. 7, given the simulated characteristics, dividends and the consumption rate of the representative investor.

Table A1: Autoregressive coefficients

Characteristic/Asset	**IBM**	**AXP**	**ABM**	**AEE**	**WEYS**	**GIS**	**KO**	**L**	**SJM**	**ARW**
Latent demand	0.871	0.788	0.440	0.858	0.823	0.290	0.834	0.861	0.914	0.880
Log book equity	0.970	0.952	0.932	0.633	0.907	0.559	0.851	0.532	0.977	0.800
Profutability	0.923	0.661	0.673	-0.025	0.315	-0.138	0.184	0.911	0.128	0.582
Investment	0.817	0.763	0.698	0.770	0.340	0.903	0.769	0.842	0.930	0.564
Dividend to book equity	0.965	0.336	0.818	-0.015	0.755	0.074	0.710	0.830	0.986	0.864
Market beta	0.886	0.919	0.842	0.951	0.944	0.855	0.898	0.799	0.906	0.832

Notes: This table reports the estimate coefficient of stock characteristics for each stock. The autoregressive coefficient is estimated using simulated method of moments described in Section A.3

Table A2 reports the summary statistics, mean and standard deviation of the stock characteristics, dividend, latent demand and returns of the data and the simulation.

Table A2: Simulated characteristics and returns

Log book equity	**IBM**	**AXP**	**ABM**	**AEE**	**WEYS**	**GIS**	**KO**	**L**	**SJM**	**ARW**
Data (mean)	6.42073	10.9827	9.12387	9.25862	7.08271	8.20547	8.97407	4.7242	9.80731	9.67367
Simulation (mean)	6.34819	10.9951	9.07388	9.25912	7.0522	8.18637	8.97068	4.72302	9.8181	9.65167
Data (std. dev.)	0.92798	0.93983	0.67001	0.53372	1.04751	0.56105	0.93541	0.30192	1.6421	0.70726
Simulation (std. dev.)	0.82138	0.89576	0.6584	0.53391	0.99783	0.56504	0.90323	0.303	1.37342	0.70275
t-stat	1.11986	-0.17463	0.96454	-0.01203	0.38829	0.42961	0.04766	0.04946	-0.09955	0.39788
Profitability	**IBM**	**AXP**	**ABM**	**AEE**	**WEYS**	**GIS**	**KO**	**L**	**SJM**	**ARW**
Data (mean)	0.21889	0.20844	0.30465	0.11783	0.15098	0.26824	0.117	0.06332	0.38611	0.18734
Simulation (mean)	0.22162	0.20721	0.30431	0.11821	0.1515	0.26896	0.11642	0.06688	0.38497	0.18809
Data (std. dev.)	0.09555	0.08512	0.09705	0.04408	0.05707	0.11134	0.08928	0.09923	0.11744	0.14144
Simulation (std. dev.)	0.0923	0.08418	0.09763	0.04363	0.0564	0.11123	0.08914	0.09333	0.11818	0.14133
t-stat	-0.37636	0.1857	0.04494	-0.11016	-0.11818	-0.08276	0.08237	-0.48475	0.12209	-0.06699
Investment	**IBM**	**AXP**	**ABM**	**AEE**	**WEYS**	**GIS**	**KO**	**L**	**SJM**	**ARW**
Data (mean)	0.07682	0.05851	0.0643	0.04703	0.08052	0.05172	0.06906	0.0015	0.14528	0.05576
Simulation (mean)	0.07964	0.06095	0.06401	0.04604	0.0822	0.05487	0.06589	-0.0037	0.15288	0.05884
Data (std. dev.)	0.11684	0.12934	0.11971	0.08153	0.18329	0.1003	0.21201	0.12631	0.12132	0.07414
Simulation (std. dev.)	0.11848	0.12482	0.11894	0.08002	0.17955	0.09629	0.21051	0.12452	0.11936	0.07497
t-stat	-0.3021	-0.2478	0.03097	0.15715	-0.11863	-0.41607	0.19138	0.53064	-0.80959	-0.52221
Dividend-to-book equity	**IBM**	**AXP**	**ABM**	**AEE**	**WEYS**	**GIS**	**KO**	**L**	**SJM**	**ARW**
Data (mean)	0.02268	0.04472	0.06078	0.03469	0.03105	0.01457	0.03598	0.00519	0.04535	0.03676
Simulation (mean)	0.02248	0.04507	0.06082	0.03467	0.0315	0.01456	0.04121	0.01358	0.04537	0.04012
Data (std. dev.)	0.00984	0.01128	0.02259	0.01242	0.01843	0.00625	0.03402	0.01869	0.01963	0.02838
Simulation (std. dev.)	0.00904	0.01134	0.02233	0.01239	0.01723	0.00608	0.02947	0.01264	0.01587	0.02478
t-stat	0.28184	-0.39266	-0.0233	0.02207	-0.33005	0.03591	-2.2517	-8.3563	-0.01684	-1.7186
Market beta	**IBM**	**AXP**	**ABM**	**AEE**	**WEYS**	**GIS**	**KO**	**L**	**SJM**	**ARW**
Data (mean)	-0.00115	0.00373	0.00301	0.00657	0.00056	2e-05	-0.00242	0.00272	0.00359	0.00118
Simulation (mean)	-0.00035	0.00407	0.00267	0.00719	0.00099	-0.0013	-0.00243	0.00272	0.0037	0.00143
Data (std. dev.)	0.01495	0.01105	0.01112	0.01267	0.01478	0.0132	0.01631	0.0164	0.01067	0.0089
Simulation (std. dev.)	0.01525	0.01091	0.01089	0.01174	0.01328	0.01287	0.01634	0.01643	0.01059	0.0089
t-stat	-0.66117	-0.3934	0.40746	-0.66577	-0.40642	1.3024	0.00912	0.00492	-0.12959	-0.36119
Dividend yield	**IBM**	**AXP**	**ABM**	**AEE**	**WEYS**	**GIS**	**KO**	**L**	**SJM**	**ARW**
Data (mean)	0.0009	0	0	0.01526	0	0.00068	0.00568	0.0008	0.00181	0.01158
Simulation (mean)	0.00086	0	0	0.01511	0	0.00074	0.00561	0.00112	0.0018	0.01195
Data (std. dev.)	0.00222	0	0	0.00718	0	0.00191	0.00683	0.01025	0.00212	0.00865
Simulation (std. dev.)	0.00217	0	0	0.00707	0	0.00196	0.00646	0.01206	0.00207	0.00878
t-stat	0.19258	0	0	0.25434	0	-0.37526	0.14317	-0.39294	0.0408	-0.54033
Latent demand	**IBM**	**AXP**	**ABM**	**AEE**	**WEYS**	**GIS**	**KO**	**L**	**SJM**	**ARW**
Data (mean)	-0.20346	0.04419	-0.07862	-0.18407	-0.2019	-0.68296	-0.33884	-0.24875	0.16588	-0.25759
Simulation (mean)	-0.19883	0.04132	-0.07849	-0.16776	-0.19298	-0.68338	-0.32888	-0.21586	0.16073	-0.23081
Data (std. dev.)	0.23989	0.13805	0.1569	0.16367	0.21682	0.40716	0.19511	0.42572	0.21157	0.24939
Simulation (std. dev.)	0.16791	0.09728	0.1099	0.12794	0.15707	0.28656	0.13923	0.30138	0.14371	0.17841
t-stat	-0.34786	0.37264	-0.01472	-1.6129	-0.7171	0.01858	-0.9023	-1.3765	0.45114	-1.8936
Returns (%)	**IBM**	**AXP**	**ABM**	**AEE**	**WEYS**	**GIS**	**KO**	**L**	**SJM**	**ARW**
Data (mean)	0.91451	1.08485	0.09177	0.75742	-0.21003	2.04382	0.88848	1.14193	2.00448	0.99955
Simulation (mean)	0.48439	1.00017	0.47566	1.34435	0.39247	0.76183	1.06302	1.18892	0.39994	0.5064
Data (std. dev.)	20.9169	12.7746	20.9741	12.5096	30.0809	21.2705	24.0883	14.1679	13.4409	14.4851
Simulation (std. dev.)	26.5417	25.2129	30.3453	59.572	35.9562	28.8055	39.2624	22.485	37.6887	25.9773
t-stat	0.20622	0.04287	-0.16119	-0.12596	-0.21318	0.56698	-0.05667	-0.02664	0.54416	0.24232

Notes: This table reports the summary statistics (mean and standard deviation) of stock characteristics, dividend, latent demand and returns of the data and simulation. Stock characteristics and latent demand are modeled as stock-specific AR(1) processes fitted on data. Statistics are based on 100 simulated processes. Dividend yields are sampled with replacement from the data. Returns are from prices computed according to Eq. 7

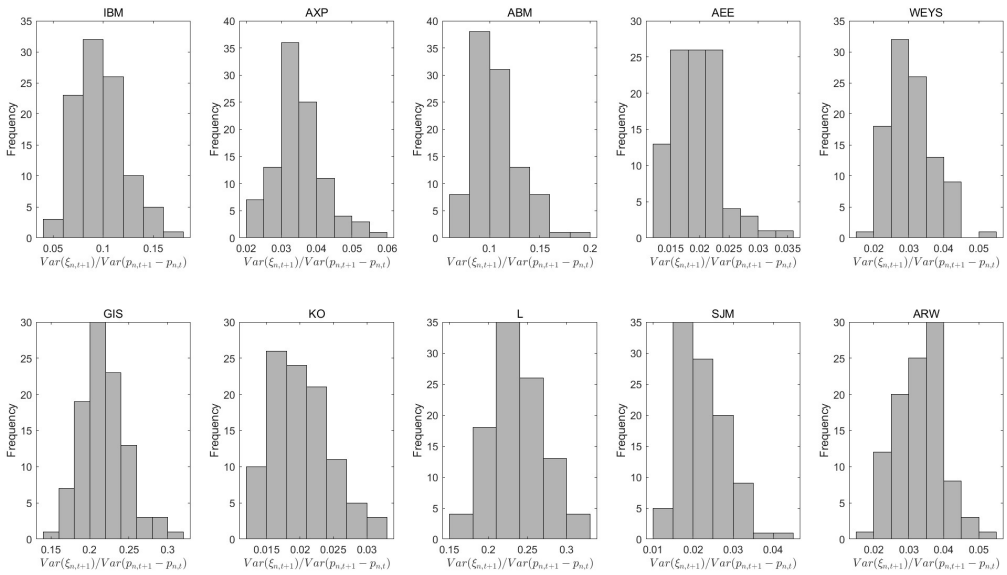
A.4 Price variance decomposition

In this section we study the contribution of latent demand on the variance of the capital gain. To assess the role of the unobserved latent demand we compute:

$$\frac{\text{Var}(\xi_{n,t+1})}{\text{Var}(p_{n,t+1} - p_{n,t})}$$

where $\xi_{n,t+1}$ is the innovation in the latent demand and $p_{n,t+1} - p_{n,t}$ refers to the capital gain. Figure A1 reports the distribution of the latent demand to the capital gain variance across 100 simulated time-series for the ten stocks.

Figure A1: Contribution of latent demand on the variance of the capital gain.



Notes: This figure reports the distribution the contribution of the variance of the latent demand on the variance of the capital gain ($\frac{\text{Var}(\xi_{n,t+1})}{\text{Var}(p_{n,t+1}-p_{n,t})}$) across 100 simulated time-series for the ten selected stocks.

The latent demand explains on average from around 2% (The Coca Cola Company, KO) to 24% (Loews Corporation, L). The mean and standard deviation across the 100 simulated time-series is shown in Table A3

Table A3: Summary statistics of the variance contibution of the latent demand to the variance of the capital gain

	IBM	**AXP**	**ABM**	**AEE**	**WEYS**	**GIS**	**KO**	**L**	**SJM**	**ARW**
Mean (%)	9.777	3.503	10.689	1.940	3.100	21.792	2.035	23.655	2.263	3.304
Std. Dev. (%)	2.304	0.703	2.261	0.405	0.619	2.870	0.440	3.333	0.572	0.653

Notes: This table reports the mean and standard deviation the contribution of the variance of the latent demand on the variance of the capital gain ($\frac{\text{Var}(\xi_{n,t+1})}{\text{Var}(p_{n,t+1}-p_{n,t})}$) across 100 simulated time-series for the ten selected stocks.

B Additional results

B.1 Portfolio distances

Table A4: Distance between AIs and benchmark across stocks, $\mathcal{D}_{\{AI,b\}}$

Panel A: IBM

	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}
J=1	32.139 (9.821)	0.436	46.199 (6.266)	0.146	68.894 (21.749)	0.049
J=2	24.906 (6.972)	0.527	50.615 (8.970)	0.305	80.268 (22.501)	0.158
J=5	24.64 (5.824)	0.596	71.118 (17.901)	0.500	163.422 (25.352)	0.395

Panel B: AXP

	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}
J=1	44.813 (11.353)	0.400	65.194 (19.218)	0.100	98.207 (8.780)	0.002
J=2	45.912 (13.842)	0.580	55.041 (10.841)	0.216	69.922 (14.844)	0.058
J=5	42.382 (4.669)	0.633	67.197 (18.759)	0.478	102.235 (36.595)	0.314

Panel C: ABM

	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}
J=1	36.423 (6.570)	0.461	57.567 (8.709)	0.328	111.17 (60.214)	0.247
J=2	34.141 (9.727)	0.556	35.342 (4.141)	0.371	47.681 (14.167)	0.199
J=5	33.297 (4.796)	0.609	35.144 (7.162)	0.519	44.247 (5.851)	0.428

Panel D: AEE

	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}
J=1	54.957 (5.373)	0.446	70.075 (1.996)	0.178	149.988 (102.827)	0.288
J=2	54.155 (5.876)	0.605	65.927 (8.946)	0.364	76.236 (3.659)	0.189
J=5	53.581 (4.013)	0.679	69.604 (10.592)	0.559	89.904 (12.581)	0.422

Panel E: WEYS

	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}
J=1	41.720 (8.009)	0.460	66.666 (33.939)	0.351	107.706 (79.112)	0.269
J=2	36.654 (5.401)	0.539	40.419 (7.603)	0.377	51.167 (6.903)	0.244
J=5	37.213 (3.863)	0.617	42.27 (8.893)	0.529	61.873 (14.392)	0.448

Panel F: GIS

	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}
J=1	49.215 (7.255)	0.550	56.381 (6.624)	0.464	59.467 (9.179)	0.411
J=2	48.037 (3.325)	0.558	52.889 (4.218)	0.507	54.631 (4.210)	0.453
J=5	48.133 (2.851)	0.569	48.945 (2.946)	0.548	49.635 (2.785)	0.527

Panel G: KO

	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}
J=1	35.515 (7.864)	0.485	66.445 (37.080)	0.496	45.999 (10.206)	0.12
J=2	34.367 (7.948)	0.630	29.09 (8.554)	0.396	41.671 (6.040)	0.265
J=5	33.366 (4.307)	0.651	36.555 (12.575)	0.560	70.152 (23.476)	0.483

Panel H: L

	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}
J=1	40.458 (7.795)	0.499	56.754 (18.070)	0.375	92.412 (51.436)	0.292
J=2	36.007 (5.276)	0.545	38.002 (4.744)	0.425	45.954 (6.537)	0.316
J=5	34.825 (2.851)	0.560	35.683 (4.045)	0.527	52.02 (7.148)	0.470

Panel I: SJM

	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}
J=1	53.86 (7.804)	0.446	134.167 (98.111)	0.220	325.935 (305.219)	0.252
J=2	48.621 (5.836)	0.630	76.213 (15.122)	0.213	82.626 (12.685)	0.032
J=5	44.309 (2.881)	0.749	106.348 (19.932)	0.525	164.743 (47.128)	0.325

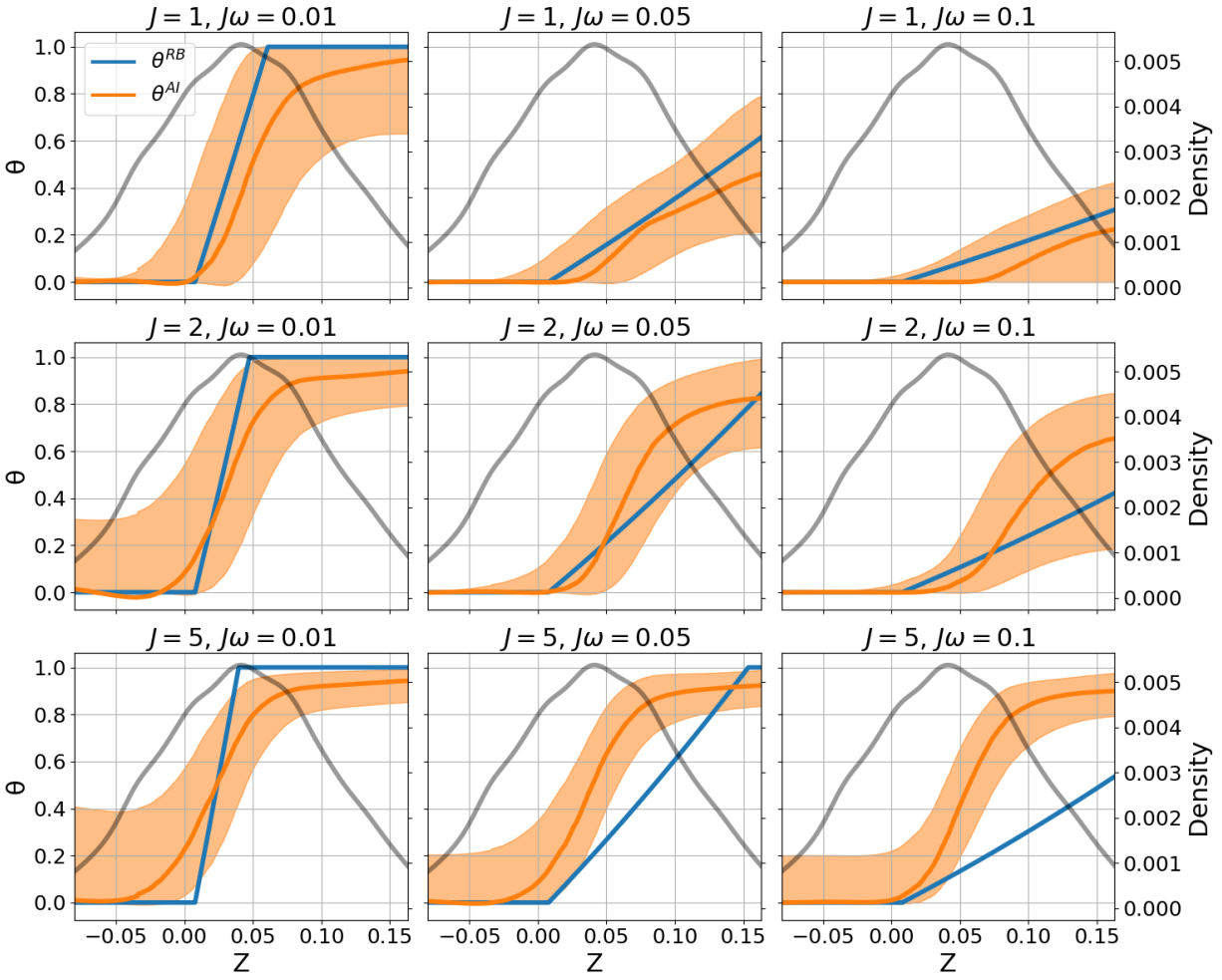
Panel J: ARW

	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}	$\mathcal{D}_{\{AI,b\}}$	θ^{AI}
J=1	33.031 (9.788)	0.521	70.931 (45.237)	0.356	52.084 (17.844)	0.091
J=2	32.557 (7.806)	0.632	34.021 (16.469)	0.400	51.154 (22.185)	0.229
J=5	32.764 (5.872)	0.706	44.495 (17.363)	0.583	90.615 (25.623)	0.478

Notes: This table reports the distance between the AI traders and the rational benchmark, $\mathcal{D}_{\{AI.b\}}(\omega, J)$, scaled by the average portfolio weight of the rational benchmark, θ^b. Values are expressed in percentage. Standard deviation across simulations is reported in parenthesis.

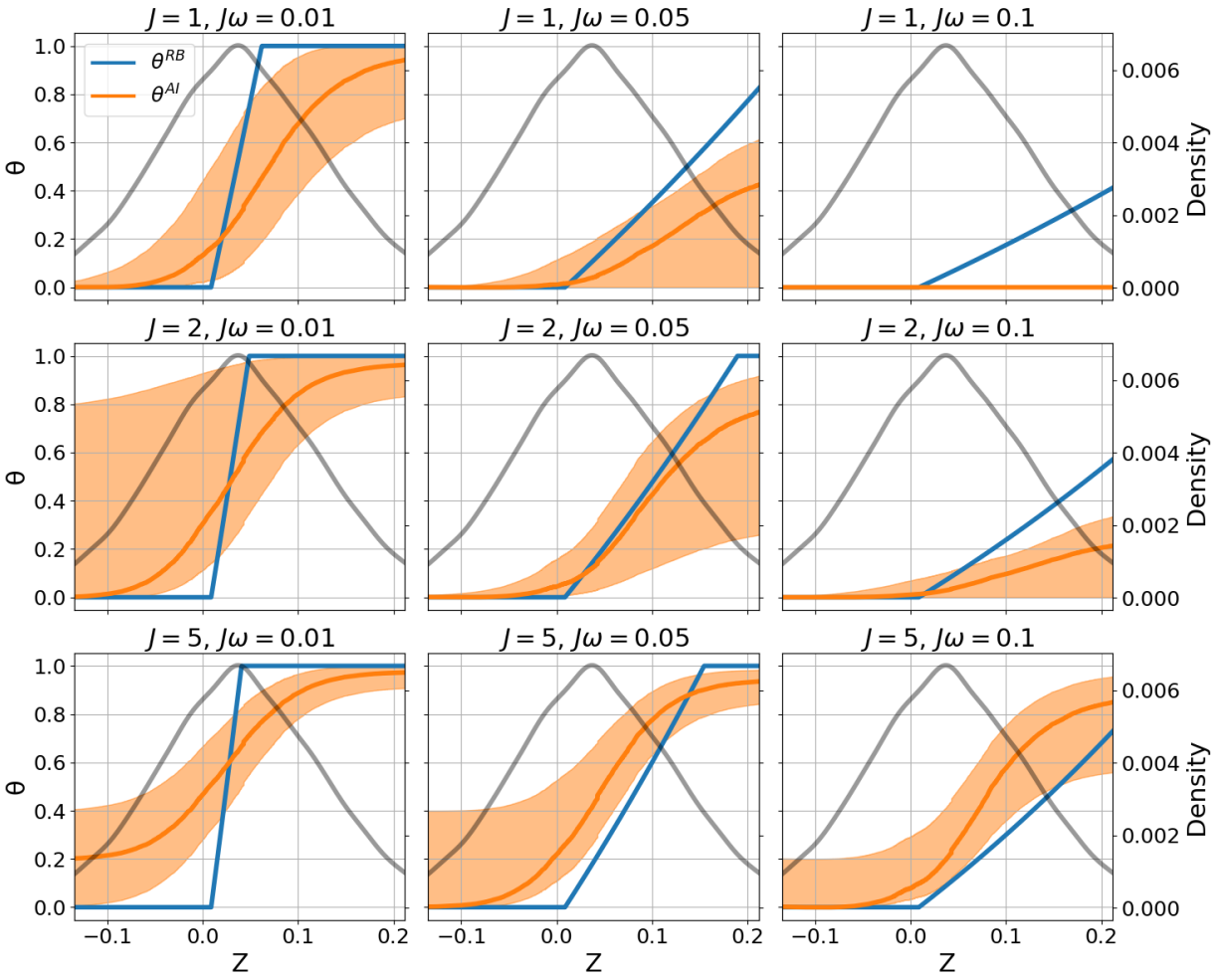
B.2 Portfolio comparative statics in $z_{n,t}$

Figure A2: Portfolio policy: IBM



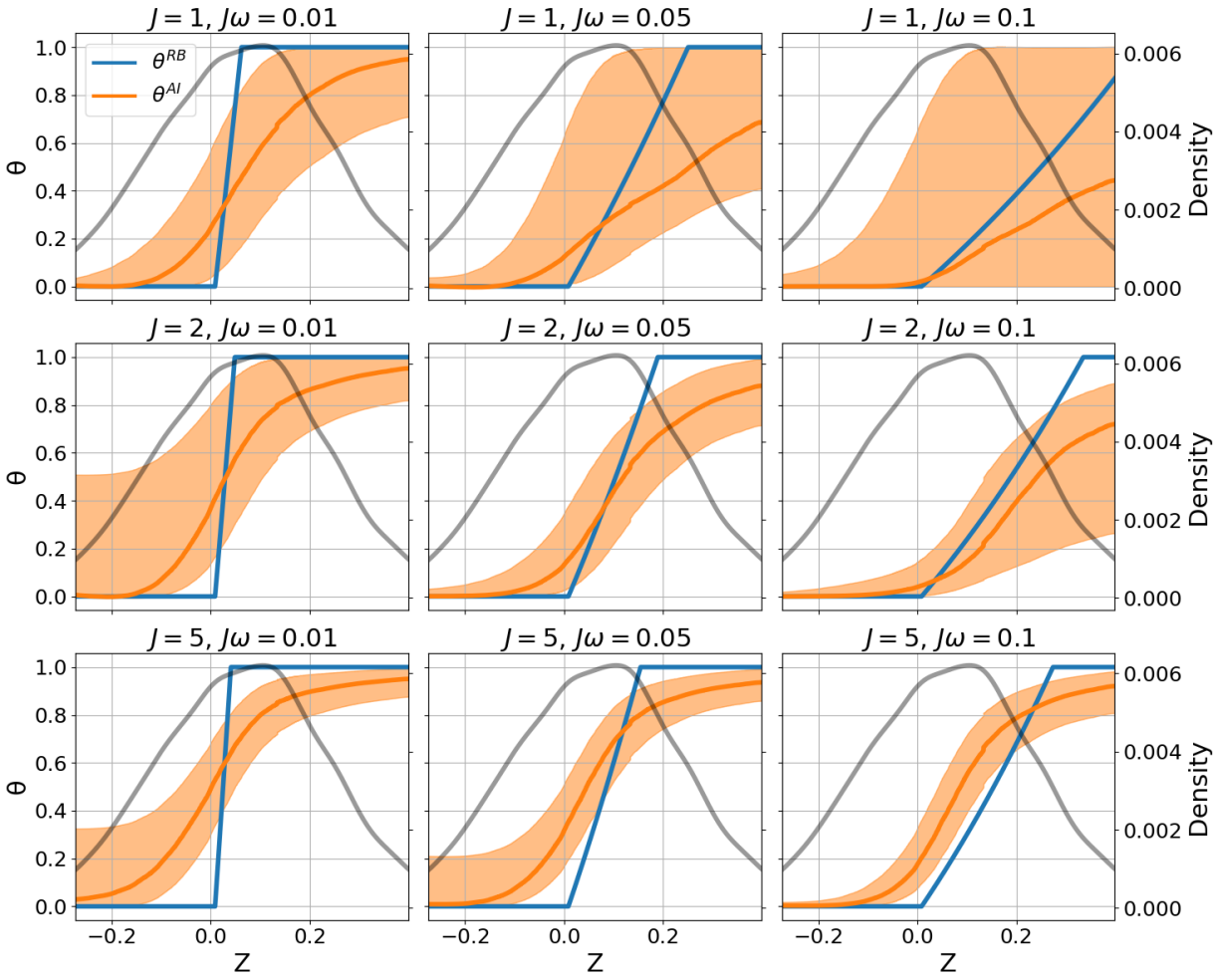
Notes: This figure plots the portfolio weight of the rational benchmark (blue line), the average portfolio weight of the AI traders θ^{AI} (orange line) and 5th-95th percentiles interval across 50 simulations (orange shaded area) and the empirical probability density function of the sufficient statistic $z_{n,t}$ (black line) as function of $z_{n,t}$ for each $(J, J\omega)$ pair. The stock is IBM.

Figure A3: Average across AIs - Stock: AXP



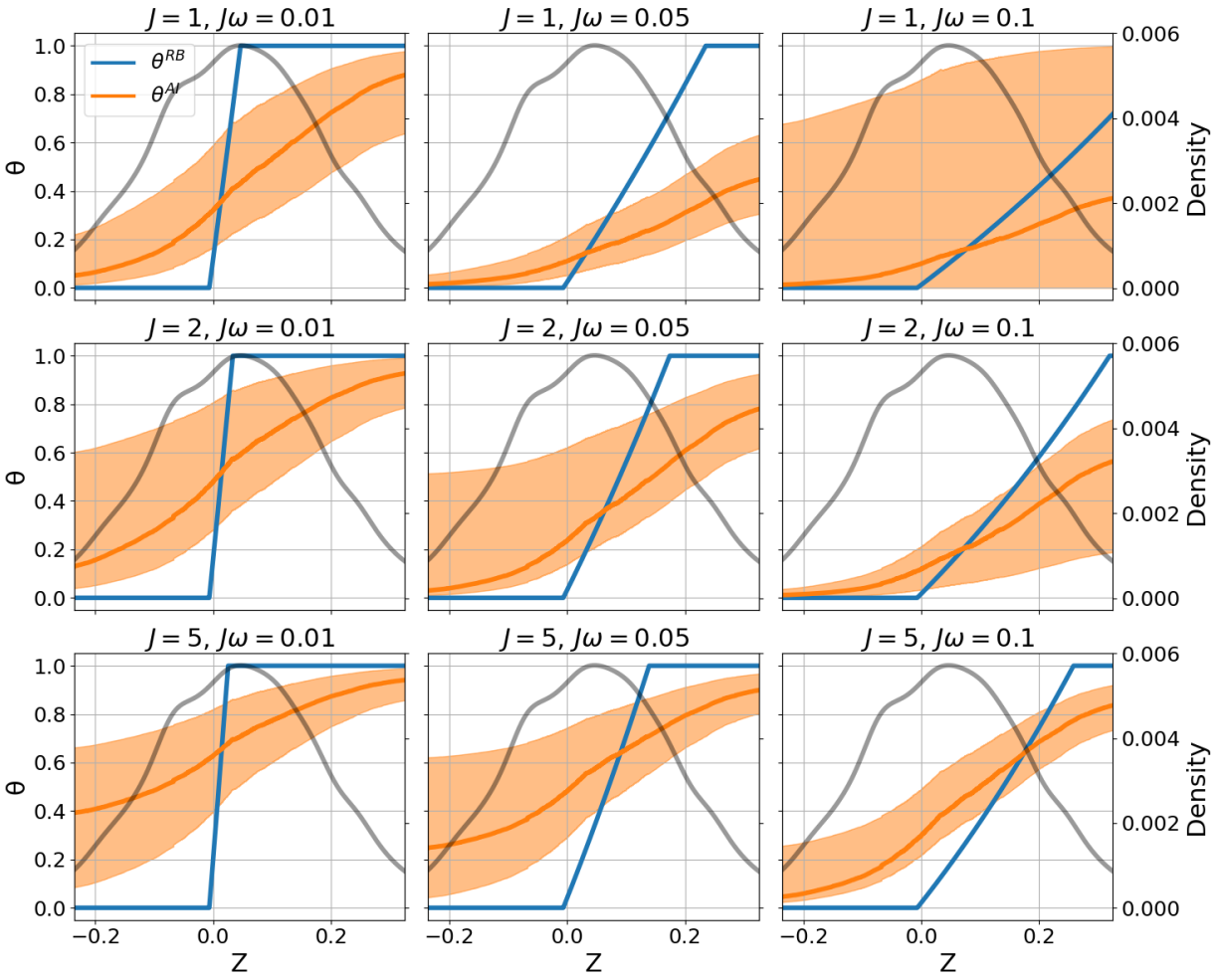
Notes: This figure plots the portfolio weight of the rational benchmark (blue line), the average portfolio weight of the AI traders θ^{AI} (orange line) and 5th-95th percentiles interval across 50 simulations (orange shaded area) and the empirical probability density function of the sufficient statistic $z_{n,t}$ (black line) as function of $z_{n,t}$ for each $(J, J\omega)$ pair. The stock is AXP.

Figure A4: Average across AIs - Stock: ABM



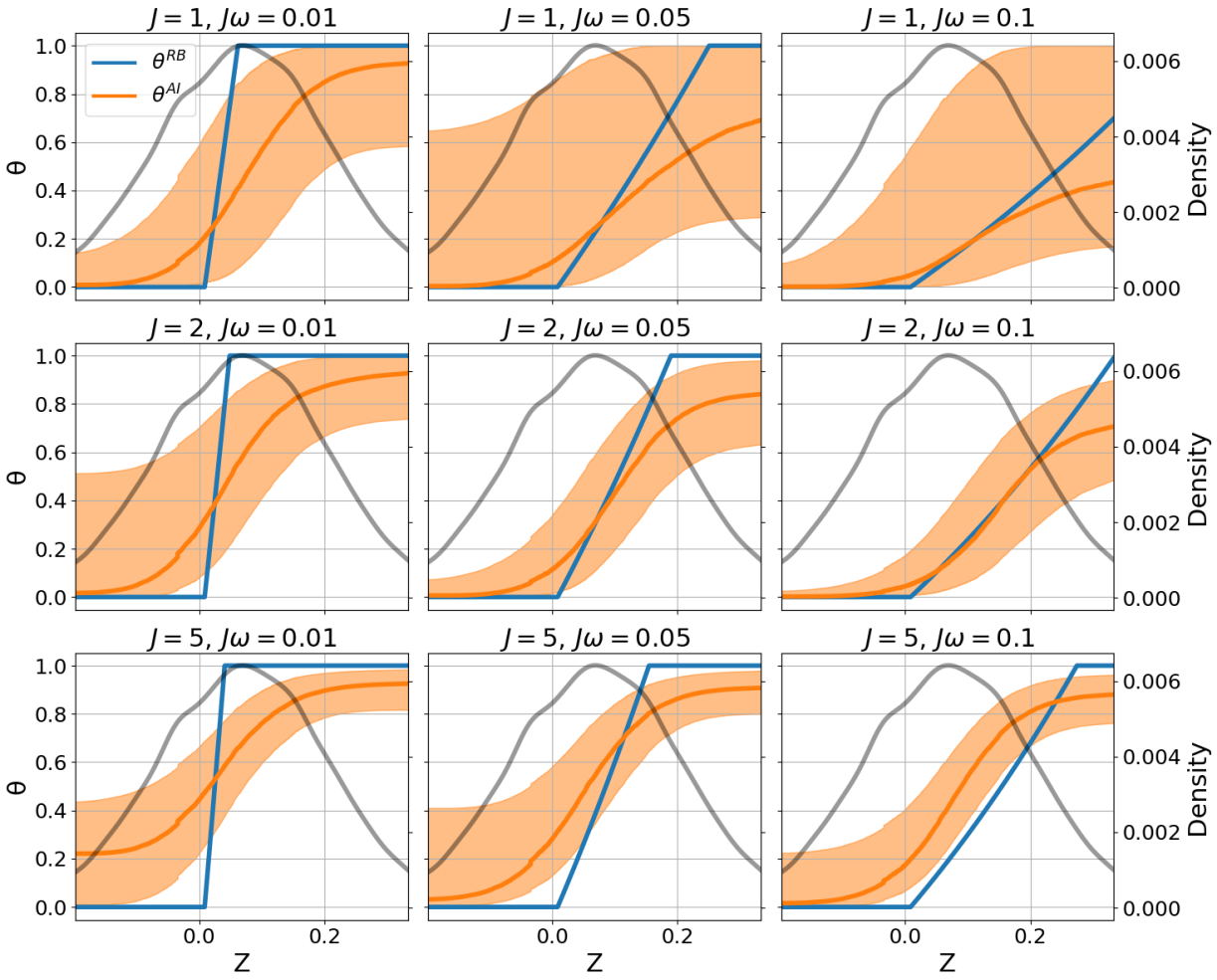
Notes: This figure reports the portfolio weight of the rational benchmark (blue line), the average portfolio weight of the AI traders (orange line) and 5th-95th percentiles interval across 50 simulations (orange shaded area) and the empirical probability density function of the expected return $z_{n,t}$ (black line) as function of the expected return $z_{n,t}$ for each $(J, J\omega)$. The stock is ABM.

Figure A5: Average across AIs - Stock: AEE



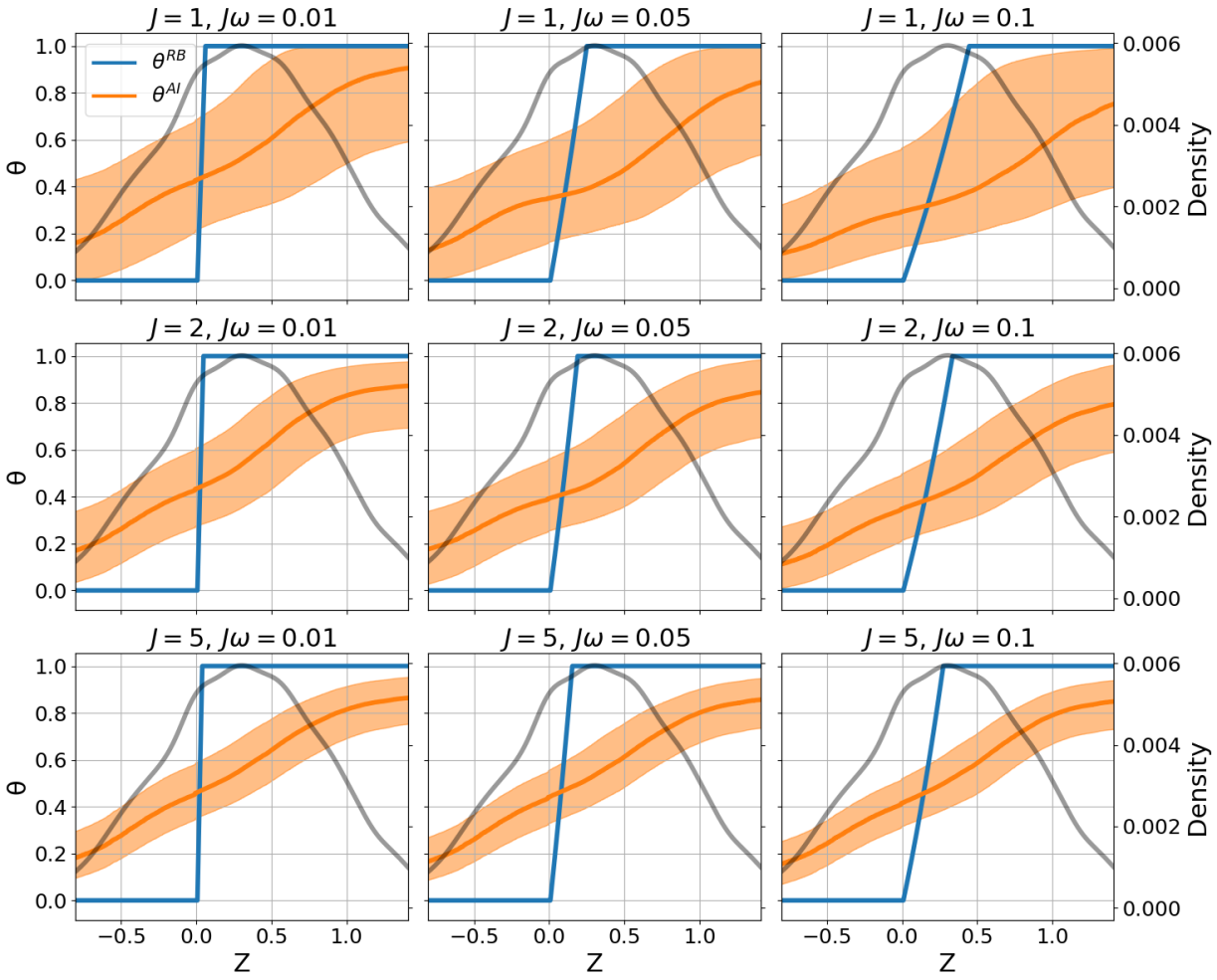
Notes: This figure plots the portfolio weight of the rational benchmark (blue line), the average portfolio weight of the AI traders θ^{AI} (orange line) and 5th-95th percentiles interval across 50 simulations (orange shaded area) and the empirical probability density function of the sufficient statistic $z_{n,t}$ (black line) as function of $z_{n,t}$ for each $(J, J\omega)$ pair.The stock is AEE.

Figure A6: Average across AIs - Stock: WEYS



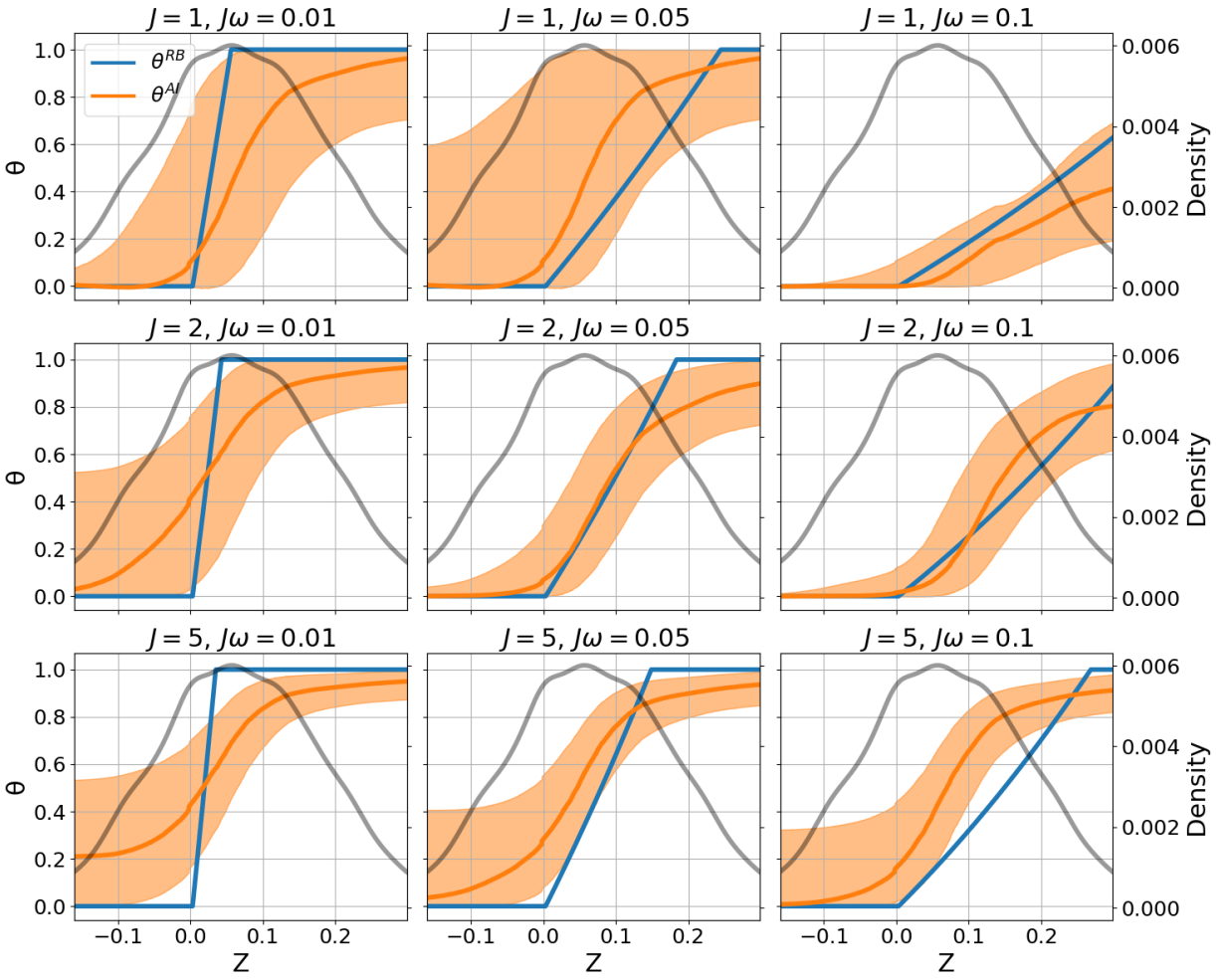
Notes: This figure plots the portfolio weight of the rational benchmark (blue line), the average portfolio weight of the AI traders θ^{AI} (orange line) and 5th-95th percentiles interval across 50 simulations (orange shaded area) and the empirical probability density function of the sufficient statistic $z_{n,t}$ (black line) as function of $z_{n,t}$ for each $(J, J\omega)$ pair. The stock is WEYS.

Figure A7: Average across AIs - Stock: GIS



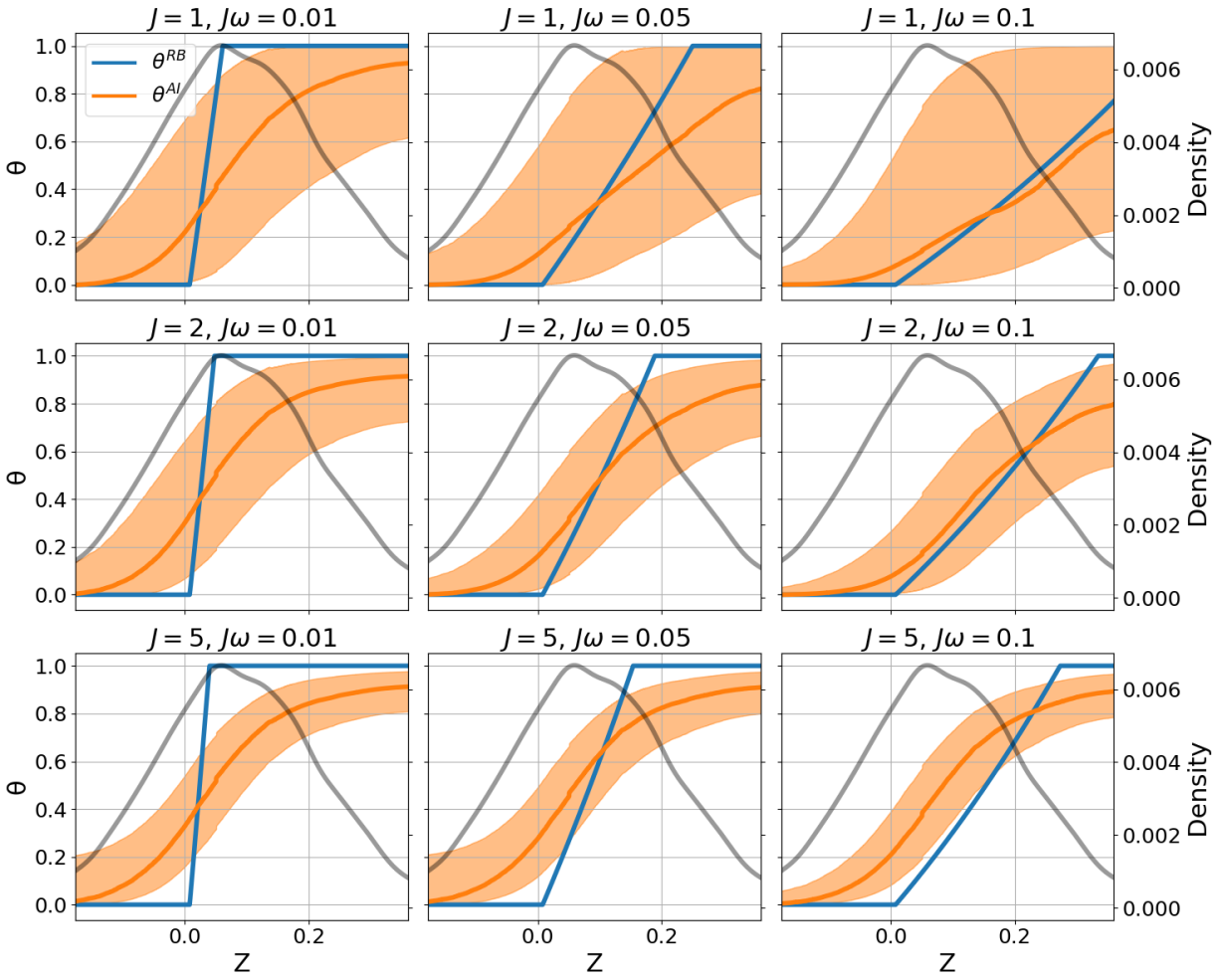
Notes: This figure plots the portfolio weight of the rational benchmark (blue line), the average portfolio weight of the AI traders θ^{AI} (orange line) and 5th-95th percentiles interval across 50 simulations (orange shaded area) and the empirical probability density function of the sufficient statistic $z_{n,t}$ (black line) as function of $z_{n,t}$ for each $(J, J\omega)$ pair. The stock is GIS.

Figure A8: Average across AIs - Stock: KO



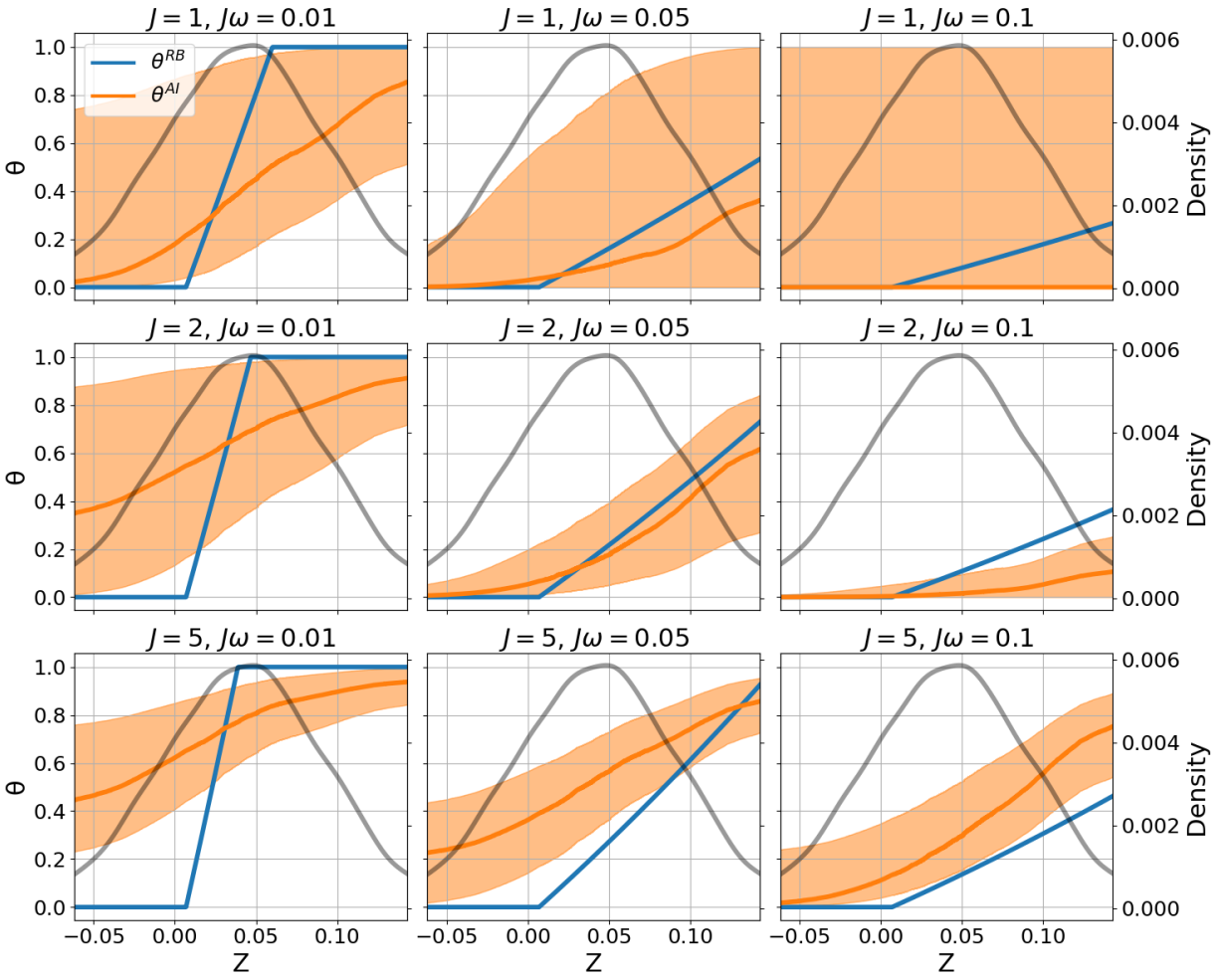
Notes: This figure plots the portfolio weight of the rational benchmark (blue line), the average portfolio weight of the AI traders θ^{AI} (orange line) and 5th-95th percentiles interval across 50 simulations (orange shaded area) and the empirical probability density function of the sufficient statistic $z_{n,t}$ (black line) as function of $z_{n,t}$ for each $(J, J\omega)$ pair. The stock is KO.

Figure A9: Average across AIs - Stock: L



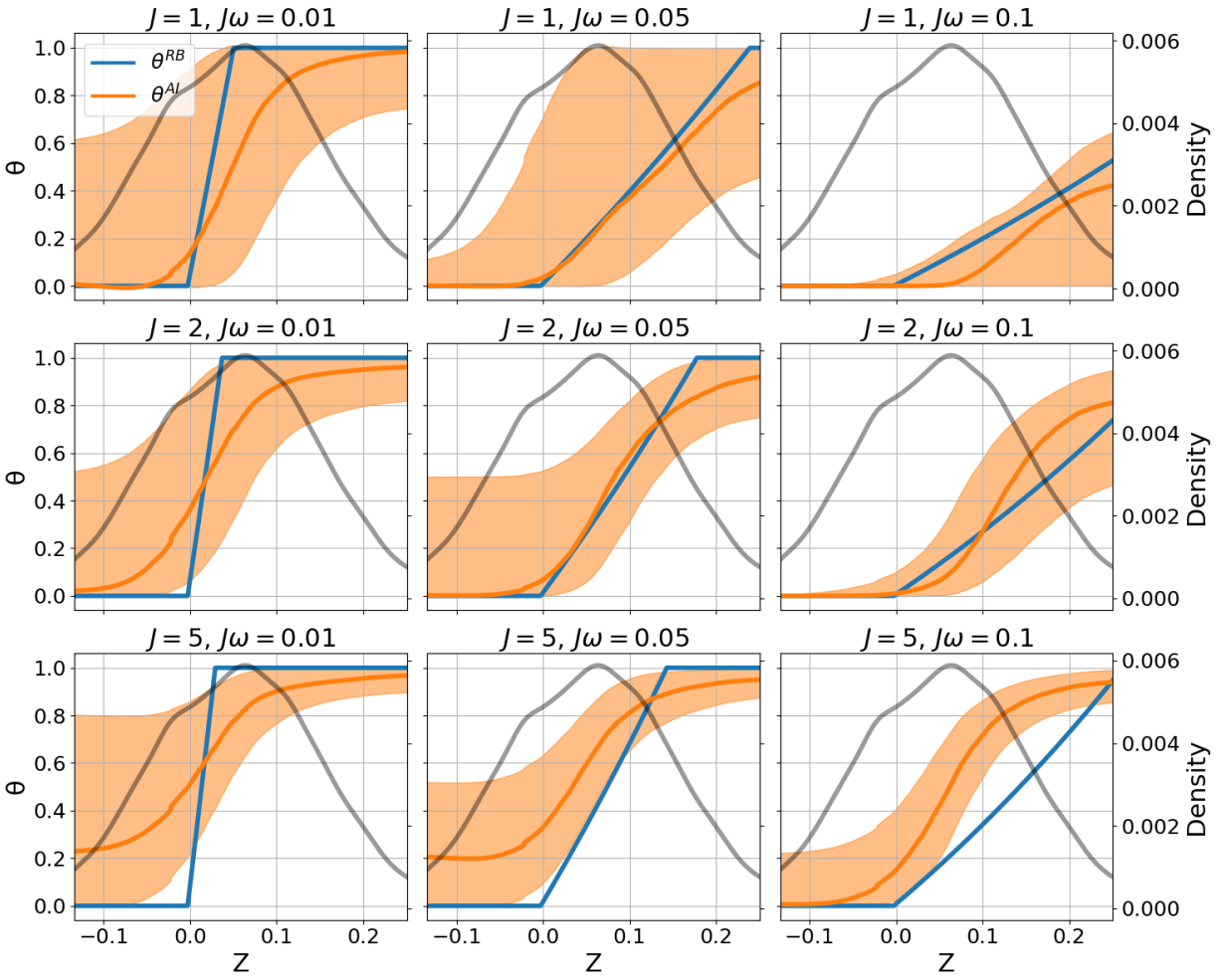
Notes: This figure plots the portfolio weight of the rational benchmark (blue line), the average portfolio weight of the AI traders θ^{AI} (orange line) and 5th-95th percentiles interval across 50 simulations (orange shaded area) and the empirical probability density function of the sufficient statistic $z_{n,t}$ (black line) as function of $z_{n,t}$ for each $(J, J\omega)$ pair. The stock is L.

Figure A10: Average across AIs - Stock: SJM



Notes: This figure plots the portfolio weight of the rational benchmark (blue line), the average portfolio weight of the AI traders θ^{AI} (orange line) and 5th-95th percentiles interval across 50 simulations (orange shaded area) and the empirical probability density function of the sufficient statistic $z_{n,t}$ (black line) as function of $z_{n,t}$ for each $(J, J\omega)$ pair. The stock is SJM.

Figure A11: Average across AIs - Stock: ARW



Notes: This figure plots the portfolio weight of the rational benchmark (blue line), the average portfolio weight of the AI traders θ^{AI} (orange line) and 5th-95th percentiles interval across 50 simulations (orange shaded area) and the empirical probability density function of the sufficient statistic $z_{n,t}$ (black line) as function of $z_{n,t}$ for each $(J, J\omega)$ pair. The stock is ARW.

B.3 Standard deviation of θ^{AI}

Table A5: Standard deviation of $\theta^{AI}(z_{n,t})$ and bias

	Panel A: IBM						**Panel F: GIS**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias
J=1	0.436	0.442	0.356	0.225	0.175	0.091	0.410	0.447	0.403	0.464	0.399	0.401
J=2	0.427	0.427	0.418	0.168	0.344	0.167	0.419	0.419	0.414	-0.205	0.413	-0.126
J=5	0.400	0.400	0.422	0.222	0.433	0.287	0.421	0.421	0.417	-0.191	0.412	-0.130
	Panel B: AXP						**Panel G: KO**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias
J=1	0.430	0.408	0.262	0.173	0.020	0.014	0.379	0.376	0.370	0.388	0.165	0.075
J=2	0.352	0.352	0.339	0.048	0.145	-0.008	0.315	0.315	0.368	0.006	0.319	0.052
J=5	0.347	0.347	0.402	0.175	0.380	0.192	0.317	0.317	0.364	0.072	0.358	0.159
	Panel C: ABM						**Panel H: L**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias
J=1	0.417	0.368	0.361	0.263	0.237	0.237	0.418	0.459	0.394	0.348	0.361	0.280
J=2	0.397	0.397	0.395	-0.050	0.313	0.011	0.432	0.432	0.419	0.009	0.397	0.098
J=5	0.387	0.387	0.411	0.000	0.418	0.080	0.427	0.427	0.418	-0.026	0.422	0.119
	Panel D: AEE						**Panel I: SJM**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias
J=1	0.386	0.342	0.306	0.173	0.254	0.254	0.441	0.555	0.326	0.353	0.181	0.253
J=2	0.342	0.342	0.380	-0.002	0.313	0.020	0.339	0.339	0.433	0.283	0.150	0.049
J=5	0.296	0.296	0.347	0.089	0.384	0.105	0.265	0.265	0.383	0.378	0.424	0.392
	Panel E: WEYS						**Panel J: ARW**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias	std(θ^{AI})	bias
J=1	0.423	0.467	0.377	0.400	0.352	0.316	0.402	0.612	0.391	0.482	0.271	0.183
J=2	0.409	0.409	0.411	0.044	0.399	0.123	0.346	0.346	0.408	0.315	0.413	0.292
J=5	0.369	0.369	0.410	0.089	0.412	0.219	0.286	0.286	0.354	0.283	0.400	0.371

Notes: This table reports the standard deviation of $\theta^{AI}_{n_s}$ across 100 random realizations of state variables that leave $z_{n,t}$ unchanged, averaged across $z_{n,t}$ values and simulations.

Table A6: Portfolio returns difference, $\Delta R_{J-1,J}$

	Panel A: IBM			**Panel F: GIS**		
	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$
J=2	0.874	1.566	1.060	-0.221	2.072	2.660
	(0.637)	(0.366)	(0.762)	(1.523)	(1.604)	(1.340)
J=5	1.309	3.948	6.316	0.646	3.451	6.596
	(0.466)	(0.842)	(1.102)	(1.374)	(1.478)	(1.768)
	Panel B: AXP			**Panel G: KO**		
	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$
J=2	1.340	0.274	-0.255	1.455	2.005	1.996
	(1.226)	(0.597)	(0.213)	(1.848)	(0.660)	(0.505)
J=5	1.243	2.478	3.566	1.700	4.769	6.868
	(0.810)	(0.980)	(1.835)	(1.116)	(1.351)	(2.276)
	Panel C: ABM			**Panel H: L**		
	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$
J=2	0.983	0.898	-0.157	0.581	2.005	2.317
	(1.951)	(0.680)	(0.887)	(0.424)	(0.611)	(0.823)
J=5	1.503	2.672	2.766	0.548	3.836	6.455
	(1.068)	(1.925)	(0.947)	(0.702)	(0.809)	(1.084)
	Panel D: AEE			**Panel I: SJM**		
	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$
J=2	1.130	1.829	0.798	1.083	0.641	-0.070
	(1.398)	(0.945)	(0.800)	(0.815)	(0.566)	(0.110)
J=5	1.468	4.072	5.179	1.598	3.439	4.021
	(1.351)	(1.838)	(1.348)	(0.584)	(1.005)	(1.564)
	Panel E: WEYS			**Panel J: ARW**		
	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$
J=2	0.655	1.674	1.469	1.383	1.959	1.572
	(1.300)	(0.947)	(0.702)	(1.279)	(1.088)	(1.144)
J=5	1.485	4.177	5.760	1.734	4.541	6.503
	(1.022)	(1.291)	(1.519)	(1.513)	(1.809)	(1.584)

Notes: This table reports the average percentage difference in portfolio returns between two settings: (i) a single AI trader trained while competing with $J-1$ rational speculators, and (ii) J AI traders trained jointly. For each stock and simulation, the return difference is computed and then averaged across simulations. Values are expressed as percentages. Standard deviation across simulations is reported in parenthesis.

B.4 Portfolio returns

Table A7: Portfolio returns

	Panel A: IBM						**Panel F: GIS**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	-0.016 (0.283)	0.022	-0.095 (0.042)	0.015	-0.155 (0.091)	0.012	-0.575 (0.086)	0.081	-0.782 (0.175)	0.052	-1.207 (0.580)	0.038
J=2	0.043 (0.216)	0.020	-0.588 (0.215)	0.012	-0.685 (0.451)	0.011	-0.531 (0.063)	0.080	-0.772 (0.149)	0.047	-1.304 (0.221)	0.030
J=5	0.008 (0.234)	0.018	-2.474 (0.826)	0.008	-4.167 (0.742)	0.009	-0.544 (0.041)	0.079	-0.881 (0.138)	0.043	-2.029 (0.293)	0.022
	Panel B: AXP						**Panel G: KO**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	-0.168 (0.268)	0.032	-0.227 (0.159)	0.018	-0.374 (0.051)	0.014	-0.037 (0.263)	0.045	-0.861 (1.029)	0.029	-0.122 (0.079)	0.022
J=2	-0.312 (0.383)	0.031	-0.201 (0.163)	0.015	-0.093 (0.136)	0.013	-0.146 (0.279)	0.044	-0.043 (0.258)	0.022	-0.440 (0.178)	0.018
J=5	-0.270 (0.137)	0.030	-1.510 (0.748)	0.011	-2.308 (1.301)	0.010	-0.107 (0.176)	0.043	-0.685 (0.670)	0.016	-2.869 (1.330)	0.011
	Panel C: ABM						**Panel H: L**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	-0.105 (0.034)	0.069	-0.300 (0.210)	0.043	-0.974 (1.181)	0.031	0.067 (0.225)	0.043	-0.380 (0.811)	0.029	-1.045 (1.221)	0.025
J=2	-0.153 (0.241)	0.068	-0.029 (0.053)	0.039	-0.068 (0.065)	0.027	0.157 (0.094)	0.040	0.146 (0.231)	0.019	-0.473 (0.372)	0.015
J=5	-0.160 (0.092)	0.068	-0.249 (0.205)	0.034	-0.811 (0.257)	0.020	0.224 (0.057)	0.039	0.014 (0.514)	0.010	-5.307 (1.700)	0.004
	Panel D: AEE						**Panel I: SJM**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	-0.327 (0.170)	0.055	-0.314 (0.025)	0.031	-1.883 (2.428)	0.022	-0.277 (0.336)	0.022	-1.145 (1.740)	0.014	-3.671 (5.169)	0.012
J=2	-0.421 (0.200)	0.054	-0.542 (0.323)	0.027	-0.516 (0.270)	0.018	-0.411 (0.383)	0.020	-0.538 (0.363)	0.013	-0.169 (0.083)	0.011
J=5	-0.455 (0.147)	0.053	-1.222 (0.589)	0.023	-2.843 (0.858)	0.012	-0.540 (0.247)	0.019	-2.798 (0.864)	0.010	-2.989 (1.206)	0.010
	Panel E: WEYS						**Panel J: ARW**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	-0.116 (0.222)	0.045	-0.635 (1.262)	0.027	-1.281 (1.927)	0.022	-0.050 (0.317)	0.037	-0.726 (1.215)	0.022	-0.149 (0.132)	0.018
J=2	-0.096 (0.187)	0.043	-0.075 (0.310)	0.021	-0.357 (0.288)	0.016	-0.125 (0.329)	0.035	-0.264 (0.588)	0.016	-0.598 (0.608)	0.014
J=5	-0.157 (0.165)	0.042	-0.715 (0.644)	0.015	-3.138 (1.233)	0.008	-0.200 (0.298)	0.034	-1.592 (1.416)	0.010	-4.526 (1.445)	0.008

Notes: This table reports *i*) the percentage deviation of log portfolio returns between the AI traders and the rational benchmark, defined as $\Delta_{AI,b} = \log(R^{AI})/\log(R^b) - 1$ and *ii*) the log portfolio return of the rational benchmark, $\log(R^b)$. Values denote the average and standard deviation (in parenthesis) across simulations.

B.5 Market outcomes

Table A8: Market efficiency (as Δ % from $\mathcal{ME}(J=0)$)

	Panel A: IBM ($\mathcal{ME}(J=0)=0.942$)						**Panel F: GIS** ($\mathcal{ME}(J=0)=0.665$)					
	$J\omega=1\%$		$J\omega=5\%$		$J\omega=10\%$		$J\omega=1\%$		$J\omega=5\%$		$J\omega=10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	1.377 (0.229)	1.739	2.517 (0.270)	3.382	1.778 (1.156)	3.386	1.227 (0.270)	1.561	4.606 (1.203)	5.340	3.779 (0.714)	5.758
J=2	1.455 (0.196)	1.729	3.973 (0.225)	4.133	3.269 (0.507)	4.176	1.015 (0.362)	1.554	4.894 (0.488)	5.859	6.585 (0.324)	7.133
J=5	1.367 (0.208)	1.706	4.514 (0.314)	4.575	3.552 (0.495)	4.742	0.990 (0.226)	1.544	4.652 (0.768)	6.040	7.284 (0.920)	7.989
	Panel B: AXP ($ME(J=0)=0.980$)						**Panel G: KO** ($ME(J=0)=0.967$)					
	$J\omega=1\%$		$J\omega=5\%$		$J\omega=10\%$		$J\omega=1\%$		$J\omega=5\%$		$J\omega=10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	1.698 (0.381)	2.529	2.475 (1.363)	5.871	0.137 (0.666)	5.943	0.810 (0.165)	1.132	2.744 (0.675)	4.477	3.550 (1.012)	5.185
J=2	1.520 (0.604)	2.537	4.372 (1.304)	7.081	2.675 (1.356)	7.384	0.833 (0.097)	1.111	3.420 (0.289)	4.759	4.901 (0.372)	6.197
J=5	1.540 (0.236)	2.535	6.029 (0.679)	7.762	6.046 (0.521)	8.411	0.837 (0.065)	1.096	3.467 (0.255)	4.801	5.587 (0.219)	6.737
	Panel C: ABM ($ME(J=0)=0.962$)						**Panel H: L** ($ME(J=0)=0.958$)					
	$J\omega=1\%$		$J\omega=5\%$		$J\omega=10\%$		$J\omega=1\%$		$J\omega=5\%$		$J\omega=10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	2.097 (0.179)	2.759	6.874 (2.026)	10.649	7.625 (6.131)	13.288	0.788 (0.232)	1.631	0.896 (0.668)	2.658	-0.824 (1.577)	2.658
J=2	2.030 (0.508)	2.753	8.744 (0.518)	11.346	10.382 (3.070)	16.093	0.653 (0.292)	1.611	1.431 (0.371)	3.282	0.619 (0.442)	3.281
J=5	1.964 (0.259)	2.744	9.284 (0.924)	11.562	16.243 (0.606)	17.495	0.557 (0.164)	1.575	1.928 (0.284)	3.733	0.135 (1.114)	3.745
	Panel D: AEE ($ME(J=0)=0.985$)						**Panel I: SJM** ($ME(J=0)=0.995$)					
	$J\omega=1\%$		$J\omega=5\%$		$J\omega=10\%$		$J\omega=1\%$		$J\omega=5\%$		$J\omega=10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	1.532 (0.340)	2.878	3.581 (0.318)	10.067	4.214 (2.640)	11.441	1.243 (0.333)	1.560	3.687 (1.077)	4.786	3.258 (1.141)	5.027
J=2	1.294 (0.474)	2.864	5.278 (0.819)	11.110	5.499 (1.050)	14.017	1.042 (0.372)	1.527	4.550 (0.811)	5.416	5.459 (0.752)	6.147
J=5	1.151 (0.346)	2.847	5.126 (1.064)	11.531	7.099 (0.485)	15.528	0.897 (0.308)	1.504	4.317 (0.810)	5.714	6.597 (0.478)	6.831
	Panel E: WEYS ($ME(J=0)=0.975$)						**Panel J: ARW** ($ME(J=0)=0.979$)					
	$J\omega=1\%$		$J\omega=5\%$		$J\omega=10\%$		$J\omega=1\%$		$J\omega=5\%$		$J\omega=10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	1.249 (0.255)	1.804	3.625 (1.231)	6.167	4.647 (1.713)	6.950	0.631 (0.271)	2.816	2.192 (0.950)	11.882	3.483 (2.041)	16.927
J=2	1.251 (0.246)	1.775	4.947 (0.514)	6.813	6.219 (0.542)	8.445	0.701 (0.178)	2.791	2.649 (0.849)	12.348	4.410 (1.316)	19.213
J=5	1.114 (0.204)	1.757	5.062 (0.684)	7.061	7.830 (0.712)	9.257	0.645 (0.115)	2.775	2.869 (0.524)	12.507	5.095 (0.925)	20.245

Notes: This table reports average market efficiency across stocks. Market efficiency is defined as the share of return variance that is unpredictable given the public information set $I_{n,t}$, as in Eq. (21). Panel A shows the percentage deviation from baseline market efficiency with only the representative investor, i.e., $ME(J=0)$, for both the AI case and the rational benchmark (RB). Panel B reports the difference between the rational benchmark and AI traders' deviations from baseline. For each $(n, J, J\omega)$ triple, we compute the average out-of-sample market efficiency for each simulation, then take the average across simulations. Standard deviation across simulations is reported in parenthesis.

Table A9: Average empirical market efficiency (as $\Delta\%$ from $\mathcal{ME}(J=0)$)

Panel A: $\Delta\mathcal{ME}(J, J\omega)$						
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB
J=1	0.283	0.337	0.787	1.131	0.812	1.278
	(0.236)	(0.270)	(0.552)	(0.601)	(0.683)	(0.646)
J=2	0.238	0.330	1.109	1.289	1.380	1.606
	(0.251)	(0.270)	(0.806)	(0.703)	(1.043)	(0.787)
J=5	0.207	0.324	1.102	1.371	1.827	1.833
	(0.244)	(0.268)	(0.876)	(0.773)	(1.106)	(0.875)

Panel B: $\Delta\mathcal{ME}^{AI}(J, J\omega)$ - $\Delta\mathcal{ME}^{RB}(J, J\omega)$			
	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$
J=1	-0.054	-0.344	-0.466
J=2	-0.092	-0.18	-0.226
J=5	-0.117	-0.269	-0.006

Notes: This table reports the average market efficiency across stocks, $\mathcal{ME}$, defined as the $1 - R^2$ from regressing excess returns $R^e_{n,t+1}$ on $me^*_{n,t}$ and $x_{k,n,t}$. Entries in Panel A report the average percentage deviations of AI trader or rational agents from the baseline market efficiency with representative investor only $\Delta\mathcal{ME}(J, J\omega)$, i.e. from $\mathcal{ME}(J=0)$. Standard deviation across stocks is reported in parenthesis. Panel B reports the difference in the average percentage deviation between the AI traders and rational agents of $\Delta\mathcal{ME}(J, J\omega)$.

Table A10: Liquidity, $\mathcal{L}$

	Panel A: IBM						**Panel F: GIS**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	0.594 (0.103)	5.011	1.201 (0.238)	11.187	0.852 (0.559)	11.196	0.245 (0.113)	0.501	0.611 (0.476)	2.172	-0.595 (0.961)	3.068
J=2	0.615 (0.091)	5.021	1.887 (0.401)	14.721	0.580 (0.454)	15.018	0.281 (0.080)	0.503	0.930 (0.327)	2.267	-1.649 (0.878)	3.608
J=5	0.572 (0.094)	5.014	1.167 (0.815)	17.474	-0.045 (0.507)	18.696	0.247 (0.052)	0.519	1.042 (0.253)	2.289	-2.297 (0.804)	3.925
	Panel B: AXP						**Panel G: KO**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	0.960 (0.236)	3.652	1.422 (0.812)	9.360	0.075 (0.364)	9.507	0.404 (0.103)	2.670	0.727 (0.548)	9.922	1.059 (0.362)	10.980
J=2	0.860 (0.355)	3.682	2.713 (0.985)	11.880	1.390 (0.730)	12.758	0.325 (0.128)	2.722	1.201 (0.322)	11.188	0.065 (0.345)	14.540
J=5	0.864 (0.146)	3.532	1.750 (1.526)	13.496	0.282 (1.339)	15.760	0.313 (0.073)	2.733	0.435 (0.637)	11.771	-0.554 (0.569)	17.176
	Panel C: ABM						**Panel H: L**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	0.878 (0.078)	0.961	1.882 (0.804)	3.936	0.618 (1.362)	4.982	0.311 (0.065)	2.348	0.653 (0.301)	10.014	0.255 (0.464)	12.290
J=2	0.842 (0.215)	0.965	2.798 (0.632)	4.303	0.928 (1.086)	6.231	0.319 (0.036)	2.197	0.955 (0.297)	11.032	0.086 (0.346)	15.673
J=5	0.815 (0.111)	0.968	1.512 (0.879)	4.423	-0.401 (0.504)	6.980	0.320 (0.032)	2.179	0.813 (0.340)	11.371	-0.359 (0.309)	17.719
	Panel D: AEE						**Panel I: SJM**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	0.536 (0.135)	2.175	1.312 (0.223)	8.523	0.548 (0.766)	9.993	0.545 (0.172)	6.667	0.738 (0.606)	13.052	-0.159 (0.586)	13.052
J=2	0.444 (0.160)	2.110	1.454 (0.752)	9.551	0.635 (0.609)	12.935	0.439 (0.204)	6.468	1.828 (0.507)	17.433	0.624 (0.436)	17.494
J=5	0.394 (0.123)	2.076	0.059 (0.996)	9.999	-0.434 (0.687)	14.904	0.347 (0.111)	6.323	1.277 (0.833)	21.280	-0.215 (1.213)	21.796
	Panel E: WEYS						**Panel J: ARW**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	0.508 (0.101)	2.250	1.056 (0.535)	8.832	0.702 (0.802)	10.207	0.383 (0.116)	3.194	0.778 (0.526)	11.147	0.840 (0.394)	11.874
J=2	0.495 (0.117)	2.241	1.541 (0.434)	9.845	0.270 (0.349)	13.261	0.336 (0.130)	3.114	0.968 (0.503)	13.104	0.315 (0.317)	15.837
J=5	0.438 (0.088)	2.252	0.960 (0.825)	10.227	-0.458 (0.675)	15.407	0.283 (0.103)	3.105	0.289 (0.726)	14.185	-0.548 (0.585)	19.107

Notes: This table reports average market liquidity across stocks. Liquidity is measured as the price impact of a 1% supply shock, as defined in Eq. (22). AI represents the market liquidity level with AI traders. RB denotes the average liquidity level of the rational benchmark. Entries are multiplied by 100. For each $(n, J, J\omega)$ triple, we compute the average out-of-sample market liquidity for each simulation, then take the average across simulations. Standard deviation across simulations is reported in parenthesis.

Table A11: Volatility, $\sigma(R)$

	Panel A: IBM						**Panel F: GIS**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	-0.679 (0.113)	-0.856	-1.233 (0.130)	-1.647	-0.870 (0.566)	-1.651	-0.315 (0.134)	-1.379	-1.076 (0.460)	-5.459	-1.684 (0.953)	-7.522
J=2	-0.718 (0.097)	-0.852	-1.927 (0.106)	-2.001	-1.593 (0.242)	-2.024	-0.350 (0.088)	-1.369	-1.297 (0.408)	-5.657	-2.130 (0.616)	-8.413
J=5	-0.674 (0.102)	-0.840	-2.181 (0.148)	-2.209	-1.727 (0.235)	-2.288	-0.322 (0.057)	-1.359	-1.405 (0.252)	-5.724	-2.452 (0.428)	-8.807
	Panel B: AXP						**Panel G: KO**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	-0.836 (0.188)	-1.238	-1.207 (0.664)	-2.810	-0.065 (0.325)	-2.845	-0.608 (0.134)	-0.771	-2.222 (0.579)	-2.568	-1.836 (0.343)	-2.761
J=2	-0.749 (0.296)	-1.242	-2.110 (0.618)	-3.363	-1.303 (0.659)	-3.499	-0.504 (0.178)	-0.769	-2.360 (0.231)	-2.806	-3.139 (0.148)	-3.387
J=5	-0.760 (0.115)	-1.242	-2.882 (0.312)	-3.667	-2.890 (0.239)	-3.955	-0.492 (0.111)	-0.764	-2.247 (0.360)	-2.890	-3.452 (0.419)	-3.770
	Panel C: ABM						**Panel H: L**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	-1.032 (0.087)	-1.351	-3.256 (0.913)	-4.933	-3.490 (2.746)	-6.046	-0.403 (0.082)	-0.561	-1.343 (0.326)	-2.167	-1.726 (0.478)	-2.496
J=2	-0.999 (0.249)	-1.348	-4.103 (0.229)	-5.231	-4.791 (1.348)	-7.188	-0.414 (0.048)	-0.551	-1.668 (0.137)	-2.298	-2.364 (0.174)	-2.962
J=5	-0.967 (0.126)	-1.343	-4.339 (0.407)	-5.323	-7.248 (0.243)	-7.746	-0.416 (0.032)	-0.544	-1.690 (0.121)	-2.318	-2.682 (0.101)	-3.208
	Panel D: AEE						**Panel I: SJM**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	-0.757 (0.168)	-1.408	-1.743 (0.151)	-4.682	-2.019 (1.249)	-5.273	-0.391 (0.115)	-0.806	-0.444 (0.329)	-1.303	0.424 (0.812)	-1.303
J=2	-0.640 (0.233)	-1.401	-2.537 (0.385)	-5.132	-2.638 (0.488)	-6.348	-0.324 (0.145)	-0.795	-0.707 (0.182)	-1.601	-0.307 (0.219)	-1.601
J=5	-0.570 (0.170)	-1.395	-2.465 (0.497)	-5.311	-3.370 (0.219)	-6.964	-0.277 (0.081)	-0.779	-0.950 (0.138)	-1.816	-0.063 (0.557)	-1.820
	Panel E: WEYS						**Panel J: ARW**					
	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$		$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB	AI	RB
J=1	-0.619 (0.126)	-0.889	-1.760 (0.590)	-2.948	-2.237 (0.797)	-3.305	-0.615 (0.165)	-0.772	-1.790 (0.516)	-2.312	-1.586 (0.554)	-2.423
J=2	-0.620 (0.121)	-0.876	-2.385 (0.242)	-3.243	-2.971 (0.248)	-3.974	-0.517 (0.184)	-0.754	-2.198 (0.385)	-2.603	-2.621 (0.351)	-2.940
J=5	-0.553 (0.101)	-0.868	-2.438 (0.320)	-3.355	-3.698 (0.319)	-4.332	-0.446 (0.152)	-0.743	-2.089 (0.383)	-2.740	-3.143 (0.219)	-3.249

Notes: This table reports average return volatility across stocks. Panel A shows the percentage deviation from baseline return volatility with only the representative investor, i.e., $\mathcal{ME}(J = 0)$, for both the AI case and the rational benchmark (RB). Panel B reports the difference between the rational benchmark and AI traders' deviations from baseline. For each $(n, J, J\omega)$ triple, we compute the average out-of-sample return volatility for each simulation, then take the average across simulations. Standard deviations across simulations are reported in parentheses.