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

Ivan Gufler[1] Francesco Sangiorgi[2] Emanuele Tarantino[3]

[1]Luiss [2]Frankfurt School of Finance & Management [3]Luiss and European Commission[1]

IE University, May 6, 2025

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

Motivation & Research Question

- Financial markets exhibit return predictability arising from:
  - Public signals (e.g., fundamentals, firm characteristics).
  - Latent demand shocks (e.g., investor sentiment, large trades).
- Reinforcement learning (RL) traders do not assume a known structure–they learn from *experience*.

Research Questions:

- Can AI-driven investors detect and exploit return predictability?
- How do they influence market efficiency, liquidity, and price formation?

Goal: Understand how AI strategies learn from and reshape asset prices.

Algorithmic Behavioral Finance

- Behavioral finance focuses on human biases.
- As AI-based trading grows, the decision-makers are algorithms.
- Goldstein, Spatt, and Ye (2021): *"Just as insights into human behavior from the psychology literature spawned the field of behavioral finance, so can insights into algorithmic behavior (or the psychology of machines) spawn an analogous blossoming of research in algorithmic behavioral finance."*

What We Do

- **Combine deep reinforcement learning (DRL)** with a **demand-based asset pricing approach** (Koijen and Yogo, 2019):
    - Prices reflect both fundamentals and latent demand from a representative investor.
    - AI traders learn endogenously and have price impact.
- **Compare AI outcomes to a rational expectations (RE) benchmark**:
    - Do AI traders discover predictable signals?
    - Do AI traders internalize their price impact and that of competing AIs?
    - How does AI trading shape market efficiency and liquidity?

Preview of Results

- **Qualitative Alignment with Rational Benchmark:**
  - AI portfolios largely mirror the comparative statics from the RE case.
  - Agents learn to decode price information and internalize price impact.
- **Negative Learning Externality with Many AI Traders:**
  - Quantitatively, AI traders deviate from RE as competition increases.
  - Multiple AIs generate additional noise, degrading each other's learning.
  - Performance relative to RE also decreases with size ($\approx$ decreasing returns).
- **Market Efficiency and Liquidity:**
  - AI trading increases market efficiency making returns less predictable.
    - But not as much as RE traders would.
  - AI traders provide less liquidity compared to RE traders.
    - AIs do not react to unexpected changes in the environment as RE would.

Literature

- **Reinforcement Learning in Finance and Economics:** Calvano et al. (2020); Colliard et al. (2023); Abada and Lambin (2023); Johnson et al. (2023); Dou et al. (2023); Yang (2024).

  Our paper differs in (i) focus (predictability vs. coordination/collusion), (ii) approach (embedding RL in empirically plausible settings), and (iii) methods (Deep RL with continuous action spaces vs. tabular-Q learning).
- **Behavioral Biases in Model-Free Algorithms:** Barberis and Jin (2023)
- **Demand-Based Asset Pricing:** Koijen and Yogo (2019)

Roadmap

1. Market Environment
2. Rational Expectation Benchmark
3. Empirical Design
4. DRL Implementation
5. Experiments
6. Conclusion

Market Environment: Setup

- Time: $t = 0, 1, 2, \ldots$
- N risky assets with supply S_n (exogenous) and price $P_{n,t}$ (endogenous)
  - Risky assets pay dividends $D_{n,t}$; we assume $D_{n,t}/P_{n,t-1}$ is exogenous.
- A riskless asset with exogenous gross return R_f.
- Two types of agents:
  - **J Traders** ($j = 1, \ldots, J$), holding $S^j_{n,t}$ shares of asset n at time t.
  - **Representative Investor** (marginal investor, or "the market"), holding the residual supply $S_n - \sum_{j=1}^{J} S^j_{n,t}$.

Representative Investor's Demand

Log-exponential form as in (Koijen and Yogo (2019)):

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

- $w_{n,t}, w_{0,t}$: portfolio weights in risky vs. risk-free.
- γ_t: fraction of assets consumed by the representative investor.
- $p_{n,t} = \log(P_{n,t})$, $s_n = \log(S_n)$.
- $x_{k,n,t}$: firm characteristics (e.g., book-to-market, profitability...).
- $\epsilon_{n,t}$: *latent demand*.
- $\{\beta_0, \beta_1, \cdots\}$ shape the demand elasticity of the representative investor.

Equilibrium Prices

- Solve for $\{P_{n,t}\}$ taking $\{S^j_{n,t}\}$ as given and addition assumptions.
- We obtain the following **equilibrium price**

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

  - $\alpha^S_{n,t} = \frac{\sum_{j=1}^{J} S^j_{n,t}}{S_n}$: fraction of supply demanded by J traders.
  - $D_{M,t} = \sum_{n=1}^{N} S_n D_{n,t}$: aggregate dividend (growing at rate g).
  - ϕ: constant capturing g, R_f, λ (consumption rate).

Equilibrium Prices

- Solve for $\{P_{n,t}\}$ taking $\{S^j_{n,t}\}$ as given and addition assumptions.
- We obtain the following **equilibrium price**

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

  - $\alpha^S_{n,t} = \frac{\sum_{j=1}^J S^j_{n,t}}{S_n}$: fraction of supply demanded by J traders.
  - $D_{M,t} = \sum_{n=1}^N S_n D_{n,t}$: aggregate dividend (growing at rate g).
  - ϕ: constant capturing g, R_f, λ (consumption rate).
- Larger $\alpha^S_{n,t} \implies$ higher $p_{n,t}$; price impact depends on β_0.
- β_0 closer to 1 means demand is less elastic $\implies$ stronger price impact.

Learning & Return Predictability

- Focus on one-period portfolio choice, thus $\alpha^S_{t+1} = 0$ (exit in t+1).
- In equilibrium, the (log) capital gain equals

$$p_{n,t+1} - p_{n,t} = \frac{\ln\left(1 - \alpha^S_{n,t}\right) + \sum_{k=1}^{K-1} \beta_k \Delta x_{k,n,t+1} + \Delta \epsilon_{n,t+1} + \log(1+g)}{1 - \beta_0}$$

- Stock characteristics and latent demand follow an AR(1) process:

$$x_{k,n,t+1} = c_{k,n} + \rho_{k,n} x_{k,n,t} + \eta_{k,n,t+1},$$
$$\epsilon_{n,t+1} = c_{\epsilon,n} + \rho_{\epsilon,n} \epsilon_{n,t} + \xi_{n,t+1}.$$

Learning & Return Predictability

- Focus on one-period portfolio choice, thus $\alpha^S_{t+1} = 0$ (exit in t+1).
- In equilibrium, the (log) capital gain equals

$$p_{n,t+1} - p_{n,t} = \frac{\ln\left(1-\alpha^S_{n,t}\right) + \sum_{k=1}^{K-1} \beta_k \Delta x_{k,n,t+1} + \Delta\epsilon_{n,t+1} + \log(1+g)}{1-\beta_0}$$

- Stock characteristics and latent demand follow an AR(1) process:

$$x_{k,n,t+1} = c_{k,n} + \rho_{k,n} x_{k,n,t} + \eta_{k,n,t+1},$$
$$\epsilon_{n,t+1} = c_{\epsilon,n} + \rho_{\epsilon,n}\epsilon_{n,t} + \xi_{n,t+1}.$$

Return predictability arises from mean reversion in $\{x_{k,n,t}\}, \epsilon_{n,t}$.

- Stock characteristics are public information, the latent demand is not. However...

Decoding Prices

In RE models:

- The equilibrium price is a function of public signals and latent demand:

$$p_{n,t} = f_n(\text{public info}, \alpha^S_{n,t}) + \frac{\epsilon_{n,t}}{1-\beta_0}$$

- RE agents know this function and the price formation process.
- They can invert it to extract $\epsilon_{n,t}$ from $p_{n,t}$.
  $\Rightarrow$ In equilibrium, $p_{n,t}$ fully reveals the latent demand $\epsilon_{n,t}$.
- AI investors have no knowledge of the data generating process.
- Can they learn it?

Why AI Agents Can't Use Price Directly

In Reinforcement Learning:

- Agents must act based on an observed state at time t.
- But the equilibrium price $p_{n,t}$ depends on agents' actions.
- Using $p_{n,t}$ as input creates circular logic.

Solution:

- Replace $p_{n,t}$ with a "pre-trade price" $p^*_{n,t}$, based only on public signals and representative investor demand.
- Exogenous to AI actions, but still reveals $\epsilon_{n,t}$.
- Allows AI to use it as part of their state avoiding circularity.

One-Period RE Benchmark: Setup

RE benchmark: useful to evaluate the AIs' portfolio policies, both qualitatively and quantitatively.

- J risk-neutral, rational, strategic speculators enter at t, exit at $t+1$.
- Trade in risky asset n and the riskless asset.
- Initial wealth equal to a share ω of the mkt. cap. of asset n.
- Each chooses portfolio share $\theta^j_{n,t} \in [0,1]$ in the risky asset.
- Next period, $\theta^j_{n,t+1} = 0$ (liquidation).

One-Period RE Benchmark: Information

- Define $p^*_{n,t}$ the pre-trade price

$$p^*_{n,t} = -s_n + \frac{\sum_{k=1}^{K-1} \beta_k x_{k,n,t} + \beta_K + \epsilon_{n,t} + \ln(D_{M,t}) + \phi}{1-\beta_0},$$

and define "adjusted market equity" as

$$me^*_{n,t} = p^*_{n,t} + s_n - \frac{\log(D_{M,t})}{1-\beta_0}.$$

- Before trading, speculators share the same information

$$\mathcal{I}_t = \{me^*_{n,t}, \{x_{n,k,t}\}_{k=1}^{K-1}\}.$$

- $\mathcal{I}_t$ fully reveals $\epsilon_{n,t}$:

$$E(\epsilon_{n,t}|\mathcal{I}_t) = (1-\beta_0)me^*_{n,t} - \left(\sum_{k=1}^{K-1} \beta_k x_{k,n,t} + \beta_K + \phi\right) = \epsilon_{n,t}.$$

One-Period RE Benchmark: Information (cont'd)

- We define $z_{n,t}$ as

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

for some coefficients $\gamma_0, \ldots \gamma_K$.

- $z_{n,t}$ is a sufficient statistic for $\mathcal{I}_{n,t}$ with respect to $R_{n,t+1}$.
- Rational traders should condition their portfolio choice on $z_{n,t}$ alone.

REE: Portfolio Choice

A Rational Expectation Equilibrium (REE) is a set of portfolio shares $\{\theta^j_{n,t}\}$ that are individually optimal for each speculator given their information and given the price formation process.

Proposition

An equilibrium exists, is unique, and is symmetric. In equilibrium:

(i) *θ is depends only on the sufficient statistic $z_{n,t}$ (linear sufficiency), and is increasing*

(ii) *θ is decreasing in ω (size effect).*

(iii) *Fixing ωJ, θ is increasing in J (competition effect).*

REE: Market Efficiency

- With a representative investor, returns are predictable from public info.
- Rational speculators trade on these signals, reducing predictability.

Market Efficiency:

$$R^e_{n,t+1} = g_n(\mathcal{I}_t) + e_{n,t+1}, \qquad \mathcal{ME} = \frac{\mathrm{Var}(e_{n,t+1})}{\mathrm{Var}(g_n(\mathcal{I}_t)) + \mathrm{Var}(e_{n,t+1})}$$

- $\mathcal{ME}$ closer to 1 $\rightarrow$ returns are unpredictable $\rightarrow$ greater efficiency.

Proposition

Market efficiency $\mathcal{ME}$ increases with speculator size ω and, for fixed total size ωJ, with the number of speculators J, when ωJ is large enough.

- The expected return function g_n is computed inside the model.
- Alternative: linear g_n, i.e., a predictive regression "outside the model."

REE: Market Liquidity

- Consider a temporary supply shock $s_n \to s_n + \sigma$.
- Without speculators, the price drops one-for-one: $\partial p_{n,t}/\partial\sigma = -1$.
- Rational speculators anticipate mean reversion and buy, reducing the price impact.

Liquidity:

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

- Higher $\mathcal{L}$ more liquidity (smaller price response to shocks).

Proposition

Liquidity $\mathcal{L}$ increases with speculator size ω and, for fixed total size ωJ, with the number of speculators J, when ωJ is large enough.

Algorithmic Learning vs. Rational Benchmark

- **Rational:** Perfectly extracts $\epsilon_{n,t}$ from prices, trades on mispricing internalizing price impact.
- **AI (RL):**
  - Learn from experience, without model knowledge.
  - Observe public signals but not the underlying structure.
  - May under- or overreact to price impact and predictability.
- How close do AIs come to RE agents in:
  - Portfolio policies and realized returns?
  - Market outcomes: efficiency and liquidity?

Demand Estimation: Koijen & Yogo (2019)

Objective: Estimate how investors' portfolio weights respond to stock characteristics and price, while allowing for a latent demand component.

- Following Koijen and Yogo (2019), for investor i and quarter t:

$$\frac{w_{i,n,t}}{w_{i,0,t}} = \exp\left(\beta_{i,t}^{me}\,\text{me}_{n,t} + \beta_{i,t}^{be}\,\text{be}_{n,t} + \cdots + \beta_{i,t}^{mkt}\,\text{mkt}_{n,t} + \beta_{i,t}^{0} + \epsilon_{i,n,t}\right).$$

- Characteristics: market equity, book equity, investment growth, dividend/book, profitability, market beta.
- $\epsilon_{i,n,t}$: investor-specific latent demand.
- Data sources:
  - SEC 13F filings (investor holdings).
  - CRSP/Compustat for firm characteristics (1982:Q2 – 2021:Q4).

Representative Investor Demand calibration

- Aggregate investor-level demand coefficients into a **representative investor**:

$$\beta_k = \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{I}\frac{\text{AUM}_{i,t}}{\sum_{i=1}^{I}\text{AUM}_{i,t}}\beta_{i,t}^{k}.$$

- Similarly, we compute the representative investor's latent demand as

$$\epsilon_{n,t} = \sum_{i=1}^{I}\frac{\text{AUM}_{i,t}}{\sum_{i=1}^{I}\text{AUM}_{i,t}}\epsilon_{i,n,t}.$$

.

Simulation of Characteristics & Dividends

- For each stock n, we fit AR(1) processes for characteristic $(\text{be}_n, \text{prof}_n, \text{inv}_n, \text{div}_n, \text{mkt}_n)$ and for the latent demand $\epsilon_{n,t}$.
- Dividends:
  - Sample dividend yield from data.
  - $D_{n,t+1} = (\text{div. yield}) \times P_{n,t}$.
- Calibrate λ and g to match average observed returns.
- Set R_f equal to the mean over the sample period.
- Use estimated processes and parameters to simulate equilibrium prices.

Deep Deterministic Policy Gradient (DDPG)

- **DDPG** is a reinforcement learning algorithm for **continuous action spaces**.
- Standard Q-learning deals with discrete actions; DDPG uses:
  - **Actor network**: selects actions (portfolio weights).
  - **Critic network**: evaluates the quality of these actions.
- The agent learns an optimal trading policy to maximize cumulative rewards (portfolio returns).

How DDPG Learns Over Time

Step-by-step:

1. Observe market state s_t (prices, characteristics, holdings).
2. Actor selects action $a_t = \mu_{\theta^\mu}(s_t)$ (portfolio weights).
3. Environment returns reward r_t and next state s_{t+1}.
4. Critic updates value function:

$$y = r_t + \gamma \, Q_{\theta^{Q'}}\left(s_{t+1}, \mu_{\theta^{\mu'}}(s_{t+1})\right).$$

5. Actor improves its policy via policy gradient:

$$\nabla_{\theta^\mu} J = \mathbb{E}\Big[\nabla_a Q_{\theta^Q}(s, a)\big|_{a=\mu_{\theta^\mu}(s)} \nabla_{\theta^\mu} \mu_{\theta^\mu}(s)\Big].$$

6. Use **experience replay** (random mini-batches) and **target networks** (slow update) to stabilize training.

Maintaining Stability in DDPG

- **Experience Replay**: store past transitions (s_t, a_t, r_t, s_{t+1}) in a buffer; sample randomly to break correlation.
- **Target Networks**:

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

  ensuring the critic does not overfit to recent data.
- Final output: a *policy* that adjusts portfolio weights dynamically in response to evolving market conditions.

Key Takeaways on DDPG

- Designed for continuous action spaces $\implies$ suitable for portfolio allocation problems.
- **Actor-Critic** method allows simultaneous learning of:
  - A policy function (actor).
  - A value function (critic).
- Stabilization via experience replay and target networks is critical in highly stochastic financial environments.
- Allows the agent to learn trading strategies endogenously, without prior structural knowledge of price formation.

Investigation Strategy

Training Setup:

- 50 simulations of 500 episodes each, each episode has 95 time periods.
- Each AI agent invests in one risky asset and a risk-free bond.
  - Set of 10 stocks chosen to have sufficient cross-sectional variation: $\mathcal{N} = \{\text{IBM}, \ldots, \text{ARW}\}$. stock list
- State space is $\mathcal{I}_t = \{me^*_{n,t}, \{x_{n,k,t}\}_{k=1}^{K-1}\}$, action space is $\theta_n \in [0,1]$.
- For each stock $n \in \mathcal{N}$ we consider:
  - $J \in \{1,2,5\}$ AI traders.
  - AI traders' aggregate size: $\omega J \in \{1\%, 5\%, 10\%\}$ of stock's mkt cap.

Comparisons:

- Compare average AI weight $\theta^{AI} = \frac{\sum_{j=1}^{J} \theta^j}{J}$ vs. RE benchmark θ^b.
- Compare portfolio returns for average AI vs benchmark
- Evaluate market efficiency $\mathcal{ME}$ and liquidity $\mathcal{L}$ against RE benchmark.

Portfolio choice (i): θ vs. the sufficient statistic z_n (IBM)

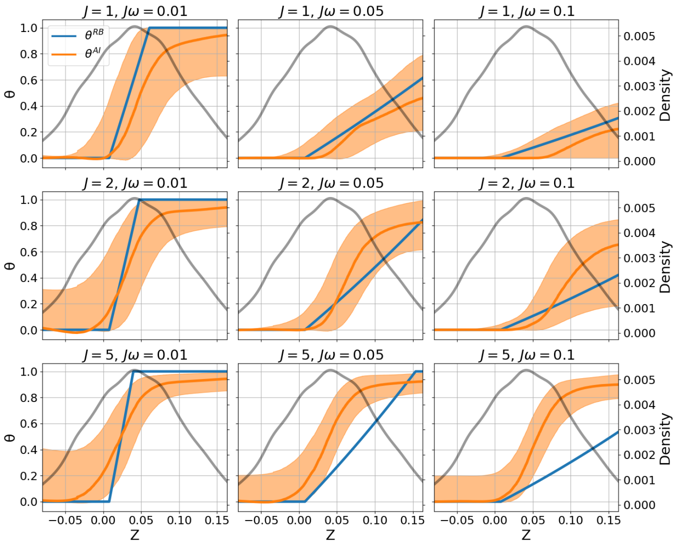


Figure: θ^{AI} (red line) vs. θ^{b} (blue line) against z_n

Portfolio choice (i): θ vs. the sufficient statistic z_n (AXP)

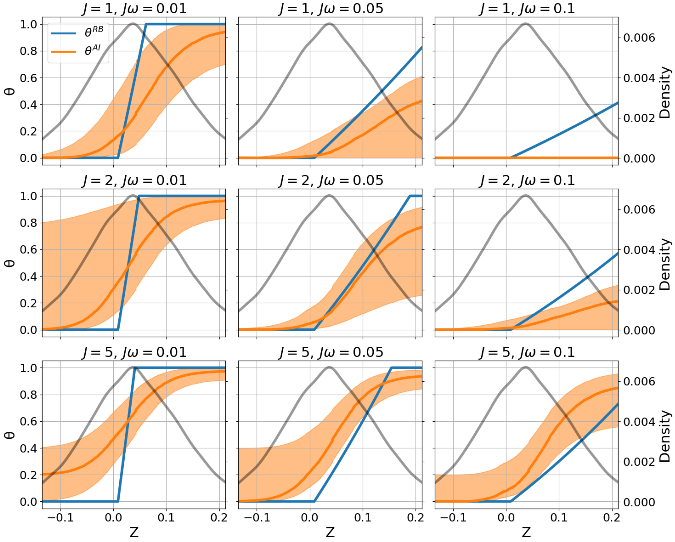


Figure: θ^{AI} (red line) vs. θ^{b} (blue line) against z_n

Portfolio choice (i): θ vs. the sufficient statistic z_n (KO)

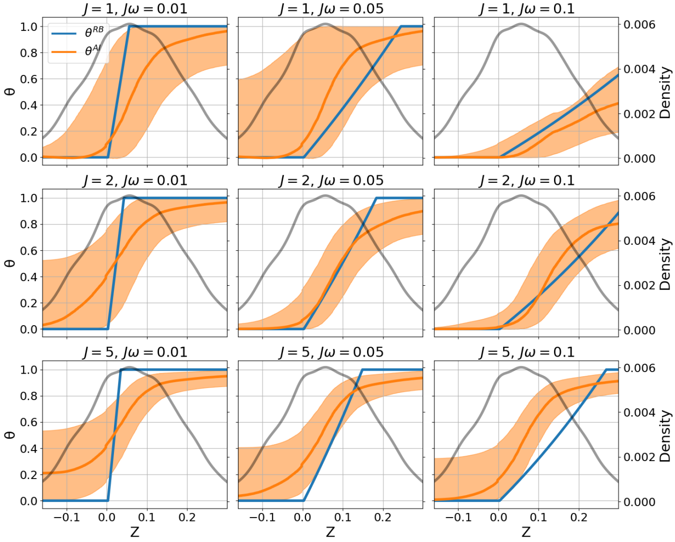


Figure: θ^{AI} (red line) vs. θ^{b} (blue line) against z_n

Portfolio choice (i): θ vs. the sufficient statistic z_n

Key Points:

- For all 10 stocks and all J, ω combinations, θ^{AI} increases in $z_{n,t}$ as in Proposition 1.
- A higher $z_{n,t}$ (either because of mean reversion in latent demand or stock characteristics) signals higher expected capital gains.

Portfolio choice (ii): linear sufficiency

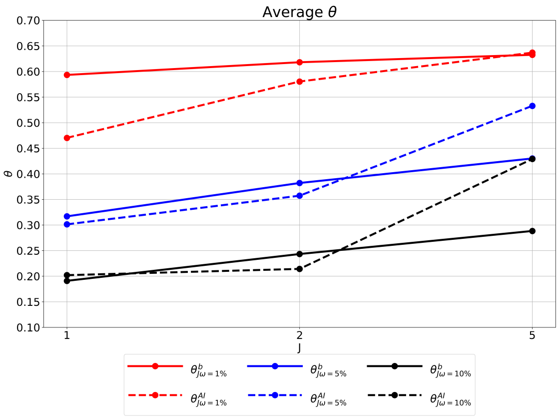
Average standard deviation of $\theta^{AI}_{n,t}$ across 100 perturbations of $\mathcal{I}_{n,t}$ holding $z_{n,t}$ fixed

	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$
	std(θ^{AI})	std(θ^{AI})	std(θ^{AI})
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)

Standard deviations across stocks in parentheses.

- Portfolio choice is unaffected by such perturbations in RB. ($z_{n,t}$ is a sufficient statistic for $\mathcal{I}_{n,t}$.)
- Instead, θ^{AI} shows substantial variation across perturbations.

Portfolio choice (iii): effects of size and competition



- Larger size (ω) leads to more cautious trading.
- Greater competition (higher J) leads to more aggressive trading.
- These patterns are qualitatively consistent with Proposition 1.

Portfolio choice (iii): effects of size and competition (cont'd)

Regression of average portfolio weight θ

	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	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	-0.290
	(0.020)	(0.014)	(0.038)	(0.029)	(0.040)	(0.030)

Effects of size (ω) and competition (J) on average portfolio weight θ for AI and the rational benchmark (RB). Baseline: $J = 1$, $J\omega = 1\%$. Regressions include stock fixed effects.

- AI reduces trading less than RB as size (ω) increases and overreacts to competition (larger J).
- This suggests qualitative alignment but quantitative deviation from RB.

Average Returns: AI vs. Benchmark

Relative log returns: AI vs. Rational Benchmark, $\Delta_{AI,b} = \log(R^{AI})/\log(R^{b}) - 1$

	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$
	$\Delta_{AI,b}$	$\Delta_{AI,b}$	$\Delta_{AI,b}$
J=1	-0.113	-0.590	-1.297
	(0.102)	(0.412)	(1.403)
J=2	-0.163	-0.394	-0.581
	(0.170)	(0.297)	(0.403)
J=5	-0.193	-1.763	-4.250
	(0.202)	(1.333)	(1.838)

Values are averages across stocks; standard deviations in parentheses.

- $J = 1$ and low price impact: AI achieves near-rational performance (5.00% vs 5.65%).
- AI returns decline faster than RB as size (ω) and competition (J) rise.
- The underperformance is especially pronounced when the number of agents rises from $J = 1$ to $J = 5$.

Summary: Portfolio Policies and Performance

Qualitative Match:

- AI policies are monotonic in $z_{n,t}$ and respond to ω and J, consistent with Proposition 1.

Quantitative Gaps:

- AI trades too aggressively as J increases and does not scale back enough as ω grows.
- Performance gaps with the rational benchmark widen substantially with higher J and $J\omega$.

What Drives the Gap?

- The environment becomes non-stationary due to other AIs' exploration.
- This **negative learning externality**—well-known in MARL—disrupts learning when J is large.

Negative Learning Externality

Experiment:

- Train one AI with $J-1$ rational traders.
- Compare to training J AIs together.
- Compute:

$$\Delta R_{J-1,J} = R^{AI}_{(\text{RB co-traders})} - R^{AI}_{(\text{AI co-traders})}$$

Average portfolio return difference: $\Delta R_{J-1,J}$ (in %)

	$J\omega = 1\%$	$J\omega = 5\%$	$J\omega = 10\%$
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)

Interpretation:

- Joint training creates a non-stationary environment, degrading learning.
- AIs learn better when rewards are not distorted by others' exploration.

Market Efficiency $\mathcal{ME}$

Average market efficiency (as Δ % 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	1.269	1.872	3.402	6.036	3.265	7.134
	(0.434)	(0.610)	(1.510)	(2.564)	(2.296)	(3.418)
J=2	1.163	1.853	4.497	6.665	5.170	8.493
	(0.405)	(0.614)	(1.854)	(2.666)	(2.469)	(3.990)
J=5	1.086	1.835	4.671	6.950	6.525	9.314
	(0.400)	(0.615)	(1.919)	(2.691)	(3.971)	(4.302)

Panel B: $\Delta\mathcal{ME}^{AI}(J, J\omega)$ - $\Delta\mathcal{ME}^{b}(J, J\omega)$

	$J\omega=1\%$	$J\omega=5\%$	$J\omega=10\%$
J=1	-0.603	-2.634	-3.869
J=2	-0.69	-2.168	-3.323
J=5	-0.749	-2.279	-2.789

Panel A shows deviations from the no-AI baseline; Panel B compares AI to the rational benchmark. Parentheses report standard deviations across stocks.

- AI traders improve market efficiency compared to the no-AI baseline.
- Efficiency increases with size (ω) and competition (J), but AI lags the rational benchmark, especially at high $J\omega$.

Market Liquidity $\mathcal{L}$

Average Liquidity and Gap vs. Rational Benchmark

	$J\omega = 1\%$		$J\omega = 5\%$		$J\omega = 10\%$	
	$\Delta_{AI,b}$	RB	$\Delta_{AI,b}$	RB	$\Delta_{AI,b}$	RB
J=1	-64.11	0.029	-81.61	0.088	-96.93	0.097
	(39.22)	(0.017)	(17.58)	(0.035)	(9.16)	(0.034)
J=2	-64.92	0.028	-76.29	0.106	-100.71	0.127
	(41.13)	(0.017)	(22.57)	(0.047)	(17.48)	(0.047)
J=5	-66.87	0.028	-83.59	0.117	-108.80	0.152
	(38.69)	(0.017)	(26.45)	(0.057)	(20.09)	(0.059)

$\Delta_{AI,b}$ *is the average percentage deviation in liquidity between AI and RB. Liquidity is computed under a 1% supply shock. Standard deviations across stocks in parentheses.*

- AI traders provide much less liquidity than rational agents.
- RE speculators buy into supply shocks, anticipating reversals.
- AIs misread price drops as latent demand shifts, missing arbitrage.
- This suggests AIs may underreact to shocks and regime changes.

Conclusion

- AI traders learn return predictability and impact efficiency and liquidity.
- Qualitative alignment with theory:
  - Portfolio comparative statics align with rational benchmark (Prop. 1).
  - More AI capital reduces predictability.
  - Liquidity improves but stays below the rational benchmark.
- Learning externality:
  - Multiple AIs distort learning, limiting price impact internalization.
  - This lowers AI profits and weakens market efficiency and liquidity.
- Future work: dynamic rebalancing (additional state variables), multiple assets (cross sectional predictability).

Appendix: stock selection

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 Distribut

Table: Company Information and Business Sectors

Back to Main

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

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.

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.

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

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 preprint arXiv:1509.02971*.

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.

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.

Vyetrenko, S. and S. Xu (2019): "Risk-Sensitive Compact Decision Trees for Autonomous Execution in Presence of Simulated Market Response," *ArXiv*, abs/1906.02312.

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

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

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.

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.

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

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 preprint arXiv:1509.02971*.

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.

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.

Vyetrenko, S. and S. Xu (2019): "Risk-Sensitive Compact Decision Trees for Autonomous Execution in Presence of Simulated Market Response," *ArXiv*, abs/1906.02312.

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