Adaptive Machine Learning for Behavioral Trading Signals

Research Paper

Submitted by:

Auguste PIROMALLI
Nadiy ABDEL KARIM DIT ARAMOUNI

Double Degree Data-Finance
École Polytechnique & HEC Paris

Thesis Statement

This research project explores the potential integration of reinforcement learning (RL) and machine learning (ML) in identifying and exploiting behavioral inefficiencies in financial markets through technical and market-implied proxies. By modeling financial markets as dynamic environments influenced by investor psychology, the study follows an iterative process to design a decision-making framework that adapts to behavioral signals through structured proxies, with an emphasis on robust out-of-sample validation.

Research Advisor

Prof. Augustin LANDIER — HEC Paris

September 2, 2025

Note

The final explanatory notebook alongside the entire development codebase are available on the Github Repo. An HTML version of the notebook is also made available for convenience.

Abstract

This thesis explored the potential integration of machine learning, reinforcement learning and behavioral finance in designing adaptive trading strategies that respond to behavioral inefficiencies in financial markets.

The research followed an iterative development path, with early experiments exploring retail sentiment signals from Reddit, Twitter and financial news APIs. However, persistent issues with noise, time alignment and validation led to a pivot toward more structured behavioral proxies.

To this end, over 70 features spanning technical, macroeconomic, and behavioral dimensions were engineered across multiple horizons. These were funneled into a modular architecture supporting multiple model classes notably logistic regression, tree-based ensembles as well as reinforcement learning algorithms (A2C and PPO). A feature registry, lag and horizon ensemble techniques, and walk-forward validation ensured transparency, reproducibility, and robustness across regimes.

Despite extensive experimentation with reward shaping, regularization, and hybrid ML/RL ensembles, reinforcement learning agents consistently failed to generalize out-of-sample for this kind of data. In contrast, cross-sectional machine learning models, particularly ridge and elastic net ensembles, delivered more stable and interpretable results. As a result, the final contribution of this thesis emphasizes ML driven ensemble methods informed by behavioral proxies, while documenting the limitations of RL in this context.

Consequently, the final framework emphasizes ML driven ensemble methods informed by behavioral proxies, complemented by explicit risk-control overlays. The findings contribute to ongoing debates in behavioral asset pricing and computational finance by showing that machine learning models trained on structured behavioral signals yield more stable predictive power than reinforcement learning agents.

Contents

Introduction

Financial markets are dynamic and complex systems shaped not only by economic fundamentals, but also by the collective behavior of market participants. While traditional finance theory assumes rational agents and efficient markets, persistent anomalies such as momentum, reversals, and volatility clustering suggest the presence of systematic inefficiencies, often rooted in human psychology.

Behavioral finance addresses this gap by incorporating cognitive biases and emotional responses into models of investor behavior. These biases manifest indirectly through observable market proxies that may give rise to predictable patterns. For sophisticated market participants, these dynamics offer opportunities to detect and exploit inefficiencies that are not explained by fundamentals alone.

In parallel, machine learning (ML), and more recently reinforcement learning (RL), have opened new ways to model and interact with financial markets. Unlike traditional static approaches, these systems are designed to learn patterns from high dimensional data and adapt as market conditions evolve. ML techniques allow capturing complex, non linear relationships between market inputs and returns. RL offers a powerful framework for sequential decision making under uncertainty, aligning well with the dynamic nature of trading. Taken together, these methods can form the basis of adaptive systems that learn how to trade rather than simply what to predict.

This research project started by investigating both approaches in the context of behavioral finance. Early on, experiments with sentiment data from Reddit, Twitter, and financial news APIs were made aiming to build retail oriented features. While informative in theory, these sources proved noisy, difficult to align temporally, and hard to validate at scale. We therefore pivoted toward more robust, market derived behavioral proxies. These signals reflect trader expectations and activity, offering a clearer lens into behavioral shifts without relying on fragile NLP pipelines.

Rather than attempting to predict broad, ill defined "market regime changes", the system targeted shifts in the behavior of market participants, as revealed through asset level signals. The underlying hypothesis was that trader behavior is more amenable to modeling than the market itself and that by detecting shifts in sentiment, positioning, or uncertainty, models can adapt their strategies more effectively. The system was thus designed to adapt to observable shifts in trader behavior and market context i.e. regime awareness rather than prediction.

As a starting point, over 70 features were engineered across multiple domains (macro, technical, behavioral), structured into reusable feature scripts and processed through a feature registry. Modeling pipelines supporting multiple learning modes were developed. Initial attempts included logistic regression, tree ensembles and reinforcement learning agents (PPO, A2C). A flexible architecture was designed to handle feature ingestion, rolling window validation, and cross model comparisons, enabling systematic evaluation across market periods and hypothesis types.

This project's contribution also included the design of a unified ML and RL based trading system that treats the market as a dynamic environment shaped by investor psychology. The agent attempts not only to maximize returns but also to adjust to shifting behavioral regimes. This system integrated sentiment data, macroeconomic signals, and technical indicators into a modular architecture. However, a persistent lack of generalization in the use of RL agents on this type of data was faced. Numerous regularization attempts were made before eventually abandoning this approach.

Performance was evaluated by rolling window backtests using out of sample data. Benchmarks included standard behavioral strategies and non adaptive ML models. The use of risk adjusted metrics such as Sharpe ratio, alpha as well as additional risk control overlays ensure robust results across different market scenarios.

The key contribution of this work lies in demonstrating that asset-level behavioral signals, when processed through cross-sectional ML ensembles, yield stronger and more reliable predictive power than either raw sentiment inputs or standalone RL agents. While reinforcement learning struggled to generalize despite extensive reward shaping and regularization, ML based approaches with behavioral proxies delivered more consistent risk-adjusted performance. This progression highlights both the promise of behavioral finance as a source of signals and the importance of rigorous methodology in validating adaptive trading systems.

Literature Review

2.1 Traditional Behavioral Finance Models and Trading Implications

Behavioral finance challenges the notion that investors are perfectly rational, introducing models where some agents exhibit cognitive biases. Two key pillars are limits to arbitrage (i.e.: even rational traders may fail to correct mispricings) and psychology (i.e.: systematic deviations from rational behavior). Classic behavioral models such as Prospect Theory (Kahneman & Tversky, 1979) illustrate that investors value losses more heavily than gains. This loss aversion leads to the disposition effect by which investors hold losing positions too long and sell winning positions too quickly. Such behavior can create predictable price patterns. For example, reluctance to realize losses can slow price adjustment, contributing to momentum or reversal patterns in returns.

Common cognitive biases have direct trading implications. Overconfidence is a prominent bias: investors overestmate their knowledge or skill, leading to excessive trading volume. In rational models, very little trading would occur without new information, but real markets see much more churn. Behavioral research attributes this to overconfident traders who believe their information is strong enough to justify frequent trades when it is not. This excessive trading driven by overconfidence can reduce portfolio returns after accounting for transaction costs. Herding behavior is another bias where individuals follow the crowd. Herding can inflate asset bubbles or exacerbate crashes as many traders simultaneously buy or sell based on others' actions rather than fundamentals. Mental accounting, representativeness, and anchoring are additional biases affecting investment choices, often causing mispricing that astute strategies might exploit.

The implications of these behavioral models for trading strategies are significant. For instance, overreaction and underreaction in markets (due to biases like representativeness or conservatism) can lead to momentum and reversal anomalies. Momentum, the tendency of assets that have performed well to continue rising in the short term, has been partially attributed to underreaction (investors slowly incorporating news) followed by herding driven overreaction. Empirical studies show that momentum effects can indeed be driven by biases. Xu (2005) finds that trading behavior linked to psychological biases (e.g. trend chasing and overconfidence) contributes to momentum in stock returns. In a model with both momentum traders and overconfidence driven traders, aggregated trend chasing behavior was shown to cause persistent return momentum. The study's empirical tests confirmed that higher investor confidence (measured by consumer confidence indices) and greater cognitive limitations (proxied by information processing costs like R&D intensity) are associated with stronger momentum profits. These findings support the view that behavioral limitations of investors create price trends that certain strategies (e.g. momentum strategies) can exploit. In summary, traditional behavioral finance provides a framework for understanding why certain trading strategies such as momentum, contrarian, or value strategies can be profitable by capitalizing on predictable biases in investor behavior.

2.2 Behavioral Signals Trading: Sentiment, Volatility, and Momentum

Beyond formal models of biases, traders often use behavioral signals in the form of measurable proxies for market psychology in order to inform strategies. Three widely studied signals are sentiment, volatility, and momentum, each reflecting aspects of investor behavior or emotion. Investor sentiment refers to the overall mood or attitude of market participants (optimistic or pessimistic) and is commonly extracted from news, social media, and other textual data. High pessimism or fear in media can push prices below fundamental values, whereas extreme optimism can inflate prices. In a seminal study, Tetlock (2007) analyzed the tone of Wall Street Journal news and found that high media pessimism predicts downward pressure on stock prices, followed by a reversal toward fundamental values. Unusually high or low media sentiment also correlated with surges in trading volume, consistent with sentiment driven mispricing and subsequent correction. This and many subsequent studies demonstrate that news sentiment has predictive power for short term returns and volatility. The rise of social media has provided new sentiment sources: for example, research by Bollen et al. (2011) showed that aggregated Twitter mood states could predict changes in the Dow Jones Industrial Average. As a result, incorporating sentiment analysis into trading strategies (through natural language processing of news headlines, articles, or tweets) has become a popular approach to gauge the market's psychological state. Strategies that go long or short based on sentiment indicators often aim to anticipate price moves driven by waves of optimism or fear.

Volatility is another important behavioral signal. Volatility indices (like the CBOE VIX for the S&P 500) are sometimes refered to as “fear index” because they reflect the degree of investor anxiety about the future. A high VIX indicates that market participants expect significant price swings (often due to fear of losses or uncertainty), whereas a low VIX reflects complacency. Empirical research links volatility to sentiment driven activity. For instance, So and Lei (2015) show that increases in the VIX are associated with increases in trading volume, especially during high volatility periods. In other words, when the VIX spikes (high fear), noise traders become more active, adding liquidity and volatility to the market. High volatility periods tend to coincide with behavioral phenomena like panic selling or exuberant buying. Thus, some trading strategies use volatility measures as triggers. For example, scaling positions based on volatility (risk parity) or taking contrarian positions when volatility is extreme (assuming mean reversion once fear subsides). Additionally, the volatility anomaly (low volatility stocks outperforming what CAPM predicts) is sometimes attributed to behavioral biases (investors overpaying for volatile “lottery” stocks). Volatility forecasting with ML is an active area, but integrating volatility predictions into trading rules or RL agents remains a developing frontier.

Momentum as a signal lies at the intersection of behavioral finance and technical analysis. The momentum anomaly which says that assets that have performed well in the recent past tend to continue outperforming in the short run, is often seen as a product of behavioral underreaction or herd behavior. From a signals perspective, momentum indicators (like past 3 , 6 , or 12 month returns) are used to rank assets or time trades. Behavioral finance provides an explanation: investors underreact to new information initially (causing trends to persist), then eventually overreact (leading to reversals in longer horizons). As noted, evidence suggests momentum profits are stronger when behavioral biases or constraints are present. Practitioners incorporate momentum signals in trend following strategies or relative strength strategies, effectively betting that psychology driven trends will continue over the short term. Conversely, contrarian signals (e.g. extreme one day drops as overreactions) are used to exploit mean reversion when sentiment swings too far. In summary, sentiment, volatility, and momentum are key behaviorally informed signals: sentiment gauges the market’s emotional state, volatility gauges fear and uncertainty, and momentum captures collective behavioral trends. Successful trading strategies often blend these signals with fundamental or quantitative models to account for the human element in market movements.

2.3 ML Applications in Finance: Predictive Models and Strategy Optimization

Machine learning has become a cornerstone of modern quantitative finance, enabling complex pattern recognition and data driven strategy optimization beyond what traditional statistical methods can achieve. In the last decade, deep learning and other ML techniques have been applied to predictive modeling (e.g. forecasting asset returns, price movements) and automated trading strategy design. A recent systematic review by Salehpour and Samadzamini (2023) highlights that algorithmic trading has evolved from simple rule based systems to sophisticated ML driven systems, including deep neural networks and reinforcement learning agents. These models can adapt to evolving market conditions and capture non linear relationships in financial data, which are often missed by linear models.

One core application of ML in Finance is building models to predict financial variables. Traditional techniques include regression models, decision trees, and support vector machines for predicting stock returns or price direction. More recently, deep learning models (e.g. recurrent neural networks, LSTM, and Transformer architectures) are used to model time series patterns and even incorporate textual/news data alongside price data. These models aim to forecast short term price movements or volatility with greater accuracy than classical models. For example, neural networks can uncover intricate patterns from technical indicators or order book data to generate trading signals. Ensemble methods have also been popular. They consist of combining multiple models (possibly using different algorithms or data inputs such as price momentum and sentiment) to improve robustness and predictive power. The literature reports many cases where ML based predictors outperform naive benchmarks or traditional approaches, especially when non-linear market dynamics or high dimensional data (like high frequency data or alternative data) are involved.

However, the applications of Learning in Finance are not limited to predictive models. ML is also used for directly crafting or optimizing trading strategies. Techniques like genetic algorithms and evolutionary computation have a history in strategy discovery (searching for trading rules that maximize return or Sharpe ratio). More recently, deep reinforcement learning (discussed further in the next section) allows an agent to learn an optimal trading policy by trial and error. Supervised ML can also optimize strategies by calibrating parameters: for example, using historical data to train a model that outputs optimal portfolio weights or signals (this can be viewed as a regression/classification task where the target might be future returns or an optimal action). In portfolio management, ML helps in asset allocation by forecasting risk and return for many assets and finding the best mix. Studies consistently find that ML driven strategies can outperform static or heuristic strategies by identifying subtle patterns. ML algorithms excel at processing "alternative data" (such as news sentiment, social media feeds, satellite images, search trends) in addition to traditional market data, thus providing a more holistic view of market drivers. By leveraging such data, ML models often detect signals of investor behavior or economic changes faster. For example, sentiment features from news can be inputs to an ML model predicting stock jumps, or graph networks can model relationships between assets in a portfolio. Importantly, ML based automated trading systems (ATS) have shown improved performance in terms of higher returns or Sharpe ratios compared to conventional strategies. They adapt to regime changes more readily and can execute complex strategies (like high frequency trading strategies responding in milliseconds).

However, ML in finance also brings challenges. Overfitting, the fact that a model that performs excellently in backtest may fail in live trading if it learned noise, is a constant concern. Indeed, ML models can be too flexible, fitting idiosyncrasies of historical data that don't persist. The literature emphasizes techniques to curb overfitting and improve robustness: cross validation, regularization, and walk forward testing (repeatedly retraining and testing on moving time windows) are common practices. Instability is another issue; markets are non stationary, so an ML model's parameters may need frequent updating as the data distribution shifts.

Interpretability is limited for complex models. Understanding why a deep network makes a certain prediction is difficult, which is problematic in high stakes financial decisions. Researchers have begun exploring more transparent algorithms or adding explainability tools to black box models. Despite these challenges, the consensus is that ML creates valuable opportunities for trading by processing new data sources (like sentiment), engineering better features, and continuously adapting strategies. Indeed, future directions highlighted in surveys include hybrid systems (combining ML with domain insights), sentiment aware models, and further integration of advanced deep learning techniques into finance. Overall, machine learning has significantly enhanced the toolkit for traders and portfolio managers, enabling more predictive and adaptive strategies, albeit with careful attention needed to avoid pitfalls like overfitting and to ensure robustness.

2.4 RL in Trading: Recent Advancements and Key Algorithms (DQN, PPO, A2C)

Reinforcement learning has gained traction in trading strategy research as it directly addresses sequential decision making under uncertainty akin to how a trader continuously decides to buy, hold, or sell based on market evolution. In RL, an agent learns an optimal policy by interacting with an environment (in this case, the market) and receiving rewards (e.g. profits). Early applications of RL in finance date back over two decades (e.g. using Q-learning for trading), but the recent explosion of deep reinforcement learning (DRL) has led to major advancements. Modern DRL algorithms can learn complex trading policies from highdimensional inputs (like price histories, technical indicators, or even raw limit order book data) without being explicitly programmed with trading rules.

Deep Q-Network (DQN), introduced by Mnih et al. (2015), was a milestone in RL and has been adapted to trading tasks. DQN uses a deep neural network to approximate the Q-value function (state action value), enabling the agent to evaluate the long term reward of actions from raw inputs. Essentially, DQN extended traditional Q-learning to handle large state spaces by using neural networks as function approximators. In the trading domain, studies have applied DQN to problems like stock trading, cryptocurrency trading, and portfolio allocation. For example, an agent observes recent market states (prices, indicators) and Q-network outputs action values for buy/hold/sell. The agent learns to execute actions that maximize cumulative return. Research shows DQN based strategies can outperform certain heuristic or classical strategies under specific conditions. However, DQN can be prone to instability and overestimation issues, and it traditionally handles discrete action spaces (e.g. buy or sell one unit), which can be limiting for portfolio optimization or continuous position sizing.

Policy gradient and actor critic methods are another class of RL algorithms that have been fruitful in finance. Advantage Actor Critic (A2C) is a popular method that combines policy gradients with value function estimation in a synchronous framework. A2C (proposed by Mnih et al., 2016) parallelizes multiple training agents to stabilize and speed up learning. The actor network outputs actions (e.g. portfolio weight adjustments) while a critic network evaluates the value of states; training both together leads to an efficient policy search. A2C improved over earlier RL by reducing variance in gradient estimates and better handling continuous action spaces. An extension, A3C (Asynchronous A2C), uses asynchronous updates from multiple threads to decorrelate training data, but A2C is essentially the synchronized version with similar performance. In trading tasks, actor critic methods like A2C/A3C have shown strength in learning stable policies that account for riskadjusted rewards. For example, an A2C agent could be trained to maximize an objective like Sharpe ratio by appropriately shaping the reward function (penalizing volatility). Actor/critic algorithms are flexible: they can handle multi asset portfolio decisions or continuous position sizes, and they often converge faster than value only methods like DQN.

Among policy gradient methods, Proximal Policy Optimization (PPO) has become a go to algorithm for many trading applications due to its stability and ease of use. PPO (Schulman et al., 2017) is essentially an improved policy gradient method that uses a “proximal” update rule to avoid large policy updates that destabilize training. It clips the change in policy at each step, ensuring that the new policy does not deviate too far from the old one, a valuable property in volatile financial environments where overly aggressive policy updates could lead to erratic behavior. PPO has been applied in various financial RL studies and often outperforms earlier methods in terms of achieving higher rewards with less volatility in training. For instance, one study applied PPO (along with A2C and DDPG) to trading and found these advanced RL algorithms yielded robust performance across different market conditions. PPO’s balance of performance and stability makes it well suited to tasks like portfolio rebalancing or high frequency trading, where maintaining stable decisions is crucial.

The strength of RL in trading is its ability to learn adaptive, non linear strategies that maximize a chosen performance metric. Unlike supervised learning that predicts a next step price, RL directly optimizes for long term return or risk adjusted return by considering the sequential nature of trades. This means an RL agent can learn when to stay out of the market, when to enter aggressively, how to cut losses or let profits run, all in an integrated framework. Studies have demonstrated cases where RL agents surpassed benchmark strategies or human like strategies in profit and Sharpe ratio. RL is particularly powerful for complex decision problems such as optimal execution of large orders (minimizing market impact), or portfolio optimization where an agent allocates weights to maximize return while controlling risk. Moreover, RL agents can incorporate transaction costs, risk constraints, and other practical considerations into the training reward, leading to more realistic strategies.

However, RL applications in trading face notable challenges. Financial markets are noisy, non stationary, and partially observable environments. The RL assumption of a stable Markov Decision Process is violated by regime shifts, evolving market conditions, and unmodeled exogenous factors. As a result, an RL agent trained in one period may degrade in performance if market dynamics change. The literature identifies non stationarity as a critical issue. Few works explicitly address it, and there is a need for methods like change point detection or adaptive retraining to keep agents relevant. Another challenge is sample efficiency: training RL, especially deep RL, often requires a huge number of interactions. Historical market data is limited, and strategies that look good in training might be overfit to that data. Simulation or bootstrapping techniques are used to expand training data, but creating realistic market simulators is itself difficult. Some researchers have turned to model based RL or hierarchical RL to improve data efficiency.

Risk management is also a concern: vanilla RL optimizing pure profit might take on excessive risk. To address this, many studies incorporate risk measures (like volatility penalties or drawdown constraints) into the reward function. In fact, it is common to include the Sharpe ratio or similar metrics directly as part of the reward in portfolio management RL, to ensure the agent balances return and volatility. This highlights a strength of RL: the performance metric of interest (Sharpe, Sortino, etc.) can be plugged into the optimization objective. On the flip side, interpretability of RL trading agents is low. The learned policy (especially if represented by a neural network) is a black box. In finance where understanding the rationale for decisions is important for trust, this limits adoption. There are emerging efforts to make RL more explainable, but it remains a gap. Finally, a practical limitation is evaluation and reproducibility. A 2021 survey noted a lack of consistency in how DRL trading studies evaluate performance, which hampers progress. Different works use different datasets, time periods, and benchmarks, making it hard to compare results; a situation that calls for standardized benchmarks and open frameworks. In summary, RL (with algorithms like DQN, A2C, PPO, and others) has proven to be a promising approach for developing trading strategies that are adaptive and can theoretically learn to exploit market patterns. Successes have been reported in various contexts (single stock trading, multi asset portfolios, high frequency strategies), but challenges like non stationarity, sample efficiency, and robustness to unseen conditions remain active research areas. The next sections discuss how combining behavioral insights with such learning agents can potentially yield even better results, and how researchers measure performance of these advanced strategies.

2.5 Hybrid Approaches Incorporating Market Psychology

An emerging area of research integrates behavioral finance insights with machine learning and reinforcement learning to form hybrid trading systems. These systems aim to adapt to investor sentiment and psychological regimes in markets rather than relying solely on technical indicators.

One key approach involves incorporating sentiment analysis into learning agents. For example, Koratamaddi et al. (2021) developed a sentiment aware deep reinforcement learning trader that inputs both historical prices and news based sentiment into its decision process. The agent achieved improved Sharpe ratios and returns by adjusting exposure based on market mood, reducing risk during pessimistic periods and increasing positions during bullish sentiment phases.

Similarly, Ye et al. (2024) proposed a sentiment based ensemble approach that switches between different RL agents depending on the current sentiment regime. Their system dynamically combines qualitative sentiment with quantitative signals to improve responsiveness and robustness. This approach outperformed both static ensembles and individual agents by adapting faster to psychological shifts in the market.

Chen and Huang (2021) developed a multimodal RL system that jointly processes market data and news text. Their model interprets how news sentiment influences price and uses this to enhance decisions, resulting in significantly higher returns and better robustness across market sectors.

Other researchers have modeled investor attention using Google Trends or forum activity as inputs, while some have trained models to operate differently in regimes characterized by fear or greed. Multi-agent RL frameworks also simulate interactions between biased and rational traders, enabling agents to learn strategies that exploit predictable behaviors.

Hybrid systems have also been explored for risk management. For instance, RL agents trained with prospect theory utility functions align more closely with human like preferences, learning strategies that reflect loss aversion and non linear risk attitudes.

These approaches demonstrate that merging psychological indicators with adaptive algorithms can enhance trading robustness and performance. Hybrid agents that understand sentiment and behavior can anticipate emotional extremes that price based models might overlook.

2.6 Evaluation Methods in Trading Strategy Research

Evaluating ML and RL trading strategies requires rigorous testing to ensure that reported performance is not an artifact of overfitting. The most foundational technique is backtesting, in which the strategy is simulated on historical data. Walk forward testing, where models are retrained over rolling windows and evaluated on future periods, is a preferred method for assessing adaptability to market changes.

Risk adjusted performance metrics are widely used. The Sharpe ratio, which balances return with volatility, is the most common. Some studies incorporate the Sharpe ratio directly into the reward function of RL agents. Alpha, which measures return in excess of a benchmark, is also important for showing value beyond market exposure. The information ratio further refines this by dividing alpha by its volatility.

Other metrics include maximum drawdown, Sortino ratio, Calmar ratio, hit rate, and profit factor. These help gauge downside risk, consistency, and average profitability per trade. Some works also report the Ulcer Index and the average duration of drawdowns to assess risk perception from an investor's perspective.

To assess statistical robustness, researchers often use bootstrap p-values or reality checks to avoid data snooping bias. Performance is also tested across various market regimes, time periods, and asset classes. Transaction costs and turnover are included to reflect real world execution constraints.

Benchmarking is essential. New strategies are typically compared against passive strategies, traditional signals like moving average crossovers, or rule based momentum systems. Some literature proposes quantifying the "RL premium", i.e. the improvement in Sharpe or return relative to these baselines.

Ultimately, the most credible studies are those that combine strong performance with thorough out of sample validation, explainable mechanics, and resilience across regimes.

2.7 Research Gaps and Future Opportunities

Although behavioral ML and RL strategies have shown promise, key research gaps remain. Many current models use sentiment indicators but do not explicitly encode behavioral theories such as prospect theory. Future work could integrate these directly into agent objectives or constraints to better reflect human preferences and loss aversion.

Another major gap is explainability. As ML models grow more complex, their decision logic becomes harder to interpret. There is a need for explainable AI techniques that help understand which inputs are driving trading actions and whether these correspond to known behavioral patterns. This would help build trust in automated strategies.

Markets are non stationary, yet most learning systems assume a fixed data distribution. Developing agents that detect regime changes or use contextual policies to adapt to fear dominated versus greed dominated environments is a critical frontier. Meta learning and contextual RL are promising directions to address this challenge.

Robustness also requires attention. Strategies must be tested across diverse assets and stress tested under extreme conditions. There is a need for benchmark datasets and evaluation protocols to allow fair comparison across studies.

Hybrid human-AI systems represent another opportunity. Human traders may detect subtle contextual shifts that algorithms miss. Designing interactive systems where human feedback can shape or override AI behavior could yield safer and more adaptive agents.

Finally, new data sources such as Reddit sentiment, Google Trends, and even biometric stress measures offer ways to deepen psychological modeling. Applying behavioral ML to cryptocurrencies or derivatives, where sentiment plays a larger role, remains underexplored.

In conclusion, future research should prioritize integrating behavioral theory into algorithms, improving transparency, and building adaptive agents that reflect the psychology of financial markets. These advances will push the boundaries of quantitative finance and deepen our understanding of investor behavior.

Data Collection and Feature Engineering

Earlier work mixed market-derived features with text sentiment from Reddit, Twitter, and news APIs, in addition to options and macro fields. Due to noise, time alignment and reproducibility issues, these text features were dropped. The below section details the final data collection pipeline used.

3.1 End-to-end ingest and normalization

We build a daily, leak-safe cross-sectional panel for the 70+ S&P 500 constituents through a reproducible pipeline:

- **Recursive ingest and schema unification:** Recursively scan a root directory of per-ticker CSVs; parse dates to tz-naive daily stamps; coerce non-date fields to `float32`. Normalize headers to ASCII alphanumerics, remove zero-width and bidi characters, and deduplicate names.
- **Duplicate handling:** Sort by `(date, ticker)` and keep the last record for any duplicated key, then reindex.
- **Caching:** Persist the concatenated panel to Parquet so subsequent runs avoid CSV re-parsing.
- **Windowing and indexing:** Clip the analysis window to `[START_DATE, END_DATE]`. Set the canonical index name `date` and store `ticker` as a categorical to guarantee stable groupby behavior.

3.2 Targets and as-of discipline

- **Targets:** For each horizon $h \in$ `TARGET_HORIZONS`, define $y_{t,i}^{(h)} = \log P_{t+h,i} - \log P_{t,i}$.

- **As-of freeze:** All feature columns are shifted by ASOF_LAG_DAYS business days per ticker (globally 1) to prevent leakage. Targets use log differences for stationarity; backtests use simple returns.

3.3 Feature hygiene and burn-in

- **Constant-by-date filter:** Drop features that are cross-sectionally constant on more than CONST_THRESH of dates (default 80%), since they have no ranking power.
- **Known leakers:** Remove a short, hard-coded list of post-event or tainted columns discovered in prior runs (e.g., selected STARC outputs with _y, PMOSignal_PMO_35_20_10, Result_Hist_Vol_10_252_1, a specific Gator_*hist1, and MA_Neg_Vol_255_ma). Assertions ensure they do not re-enter.
- **Burn-in:** Discard observations until a ticker has at least MIN_HIST_DAYS = 252 days of history following its first appearance so that technical indicators are fully initialized.

3.4 Coverage checks and universe

- **Coverage:** Print per-ticker row counts and first/last dates, and a missingness profile on pre-target features.
- **Universe definition:** The working universe is the set of tickers present inside the window that also pass later eligibility filters.

3.5 Feature taxonomy and naming guide

Features are treated agnostically by the learner after as-of lagging, but names carry meaning for interpretation:

- **Moving averages:** _ma, ema_n are MAs of the base series; parameters are in the name.
- **Volatility and dispersion:** std_n, vol_n, hist_vol_a_b_lag.
- **Bands and ranges:** bb_n_k_{top,mid,bot}, keltner_n_k_{}, starc_n_k_m_{}; atr_n, tr.
- **Oscillators:** rsi_n, macd_f_s_sig, ppo_f_s_sig, stoch, cci_n, williamsR_n, pmo, gator_*.
- **Volume and breadth:** adv_n, dollar_vol_n, pvi, nvi.

Methodology

This section covers the methodology used within this research in details. A small summary of all exploratory work done will be provided followed by a detailed breakdown of the final production grade methodology.

4.1 Exploration attempts

At the outset of this project, the methodology was deliberately broad. The initial framework was designed as a highly modular research sandbox, capable of supporting both supervised machine learning (ML) and reinforcement learning (RL) pipelines in parallel. The codebase developed is made available on this link

- **Feature registry and modular pipelines:** A centralized registry orchestrated technical, macroeconomic, and sentiment-based features. Scripts allowed rapid switching between feature sets, preprocessing choices, and learning modes.

- **Reinforcement learning agents:** PPO and A2C agents (via `stable-baselines3`) were trained in a custom trading environment. Their state space included OHLCV data, engineered technical indicators, and behavioral proxies (e.g., abnormal volume, volatility spikes, short interest). Rewards were shaped with realized returns, transaction costs, and risk-adjusted penalties such as Sharpe ratio terms.

- **Exploratory sentiment inputs:** Early versions also integrated unstructured sentiment from Reddit, Twitter, and news APIs. Although innovative, these signals were difficult to align temporally and produced fragile results in backtests.

Despite extensive experimentation with reward shaping, regularization, and hybrid ML–RL ensembles, RL agents consistently failed to generalize out-of-sample. They were highly sensitive to training windows and prone to overfitting on sparse, noisy reward structures. These limitations, combined with reproducibility challenges from text-based sentiment, motivated a methodological pivot: away from RL and unstructured features, toward a leak-resistant, cross-sectional ML ensemble framework built on structured market-implied proxies.

4.2 Refined Framework Overview

We implement a conservative, leak-resistant cross-sectional learning framework that converts technical and market-implied behavioral proxies into tradable ranks. The key elements are leak guards, walk-forward splits with embargo, lag and horizon ensembles, optional risk overlays, and a portfolio builder that mirrors practice.

4.3 Walk-forward evaluation and leak guards

- **Expanding train:** Training starts at the first date and expands to `TRAIN_YEARS` (default 3).
- **Monthly test:** Test spans advance by `STEP_MONTHS` = 1.
- **Ceiling guard:** Enforce $t_{\text{train\_end\_eff}} \leq t_{\text{test\_start}} - (\texttt{ASOF\_LAG\_DAYS} + 1)$ business days.
- **Embargo:** Gap of `EMBARGO_DAYS` = max(`TARGET_HORIZONS`) between effective train end and test start. Splits are leak-safe by construction.

4.4 Preprocessing and estimators

- **Preprocess:** `SimpleImputer(median)` then `StandardScaler` for linear and robust models. Tree models run without scaling.
- **Estimators:** Ridge, Elastic Net, Huber, and HistGradientBoostingRegressor with deterministic defaults. Optional `RandomizedSearchCV` is available for exploration but not used in the production runner.
- **Objective:** Maximize cross-sectional ranking quality rather than absolute RMSE.

4.5 Lag ensemble (as-of robustness)

For each $L \in$ `LAG_ENS` (default $\{1, 3, 5, 10\}$ extra days beyond the global +1D shift), train the same model on identical walk-forward windows and produce $\hat{y}^{(h,L)}$. Blend per-lag predictions either:

- **Equally**, or
- **IC-weighted on train only:** Compute daily Spearman rank-IC on the first training window, clip negatives to zero, normalize to a simplex, then shrink toward equal by `IC_SHRINK_LAMBDA`.

Finally, per date, regress blended scores on prior 1D returns and use residuals. Guards: `MIN_XS_NAMES` and a small ridge term.

4.6 Horizon ensemble (multi-h integration)

Repeat the lag-ensemble for each $H \in$ `HORIZON_LIST` (e.g., 5D, 21D). Per date, z-score each horizon's scores to equalize scales, then blend across horizons with train-only horizon weights computed analogously to the lag case and shrunk by `HORIZON_IC_SHRINK_LAMBDA`. No test-period reweighting.

4.7 Optional overlays: capacity and beta

- **ADV eligibility:** A ticker is tradable if its lagged rolling mean of dollar volume over `ADV_LOOKBACK_DAYS` exceeds the per-date `ADV_PCT` percentile. Applied inside the position builder.
- **Beta-neutralization:** Estimate rolling betas to an equal-weight market over `BETA_LOOKBACK_DAY` shift by one day, regress scores on beta, and remove a fraction set by `BETA_NEUTRAL_STRENGTH`.

4.8 Portfolio construction and PnL

- **Rebalance:** Weekly Friday or month-end. Rank scores on eligible names.
- **Selection and weights:** Take top `TOP_QUANTILE` long and, if `LONG_SHORT=True`, bottom quantile short. Equal weight within sleeves. Target gross is `GROSS_LEVERAGE` with long-short splitting gross evenly.
- **Per-name cap:** Apply `PER_NAME_CAP`, then scale down only to restore sleeve totals.
- **Holding and costs:** Enter next day. Held weights are the rolling mean over `HOLDING_PERIOD` to approximate overlapping holds. Linear costs apply to daily L1 weight change with `TCOST_BPS`.
- **Returns and equity:** Daily simple return is the cross-sectional dot of held weights with next-day simple returns minus costs. Equity is compounded from 1.

4.9 Diagnostics

- **Rank IC:** Daily Spearman between scores and realized forward simple returns at the target horizon. Report mean, std, and a rough t-stat.

- **Lead/lag IC profile:** Across $[-15, +15]$ trading days.
- **Noise probes:** Shuffle scores across names within a day and permute the return calendar; both should collapse IC toward zero if the signal is genuine.
- **Entry timing:** IC of score lagged by one day.

Experiments and Results

This chapter presents the experimental setup, validation protocols, and performance analysis of the proposed trading strategies. The research followed an iterative trajectory with an initial attempt at training RL agents (PPO, A2C) in a custom environment using Sharpe-aware reward functions and engineered behavioral features as state inputs. While this approach was promising conceptually, agents consistently failed to generalize out-of-sample. These early challenges motivated a methodological pivot toward ML ensembles with rigorous "as-of" lags, horizon-ensembles, and diagnostic overlays. What follows therefore emphasizes the refined ML methodology, while noting prior RL results as a baseline reference.

5.1 Experimental Design

Assets and Period

Experiments were conducted on daily equity data for S&P 500 constituents between 2014 and 2024. This period spans a wide range of market regimes, including bull markets, corrections, and crisis episodes such as the COVID-19 drawdown and the 2022 technology selloff. Working with the full cross-section ensured breadth of coverage and provided a realistic setting for rank-based prediction and portfolio formation.

Feature Sets and Inputs

To enforce realism, all features were processed under an as-of freeze: information at date t was shifted by at least one business day (or more, for lag ensembles) before being passed to the model. Features that were constant across the cross-section or exhibited high missingness were dropped. Remaining features were standardized cross-sectionally by date, ensuring comparability across tickers.

Labels and Prediction Targets

Models were trained to predict forward returns at multiple horizons: 1-day, 5-day, and 21-day. Targets were log returns, while realized returns in backtests were simple returns to remain consistent with portfolio arithmetic. Labels were horizon-specific and lag-adjusted, ensuring no overlap between predictors and targets.

Validation Protocol

Evaluation used a deterministic walk-forward design. Training windows expanded over time (minimum three years), and models were retrained before each new monthly test span. Between the effective training end and test start, an embargo equal to the maximum horizon length was enforced to prevent leakage.

Backtests were constructed on the resulting prediction series, with portfolio-level returns and diagnostics derived from lag- and horizon-ensembles. This structure replicated live conditions, where only past information is available at decision time.

5.2 Models Evaluated

Machine Learning Benchmarks

We implemented a range of supervised ML models for cross-sectional prediction:

- Logistic Regression and Ridge Classifier
- Ridge and Elastic Net regression
- Random Forest and Gradient Boosted Trees
- LightGBM
- Histogram Gradient Boosted Trees (HGBT)

Regularized linear models especially Ridge regression proved most effective, balancing predictive power with robustness and interpretability.

5.3 Performance Metrics

Performance was evaluated along multiple dimensions:

- **Predictive diagnostics:** Rank information coefficient (IC), reported as mean, standard deviation, and t-statistic. Lead–lag IC profiles were also computed to test alignment of predictions with realized returns.

- **Portfolio metrics:** Annualized return, volatility, Sharpe ratio, maximum drawdown, turnover, and breadth (number of names held).

- **Risk overlays:** Diagnostics on beta neutrality and liquidity (ADV screens), to ensure robustness and tradability.

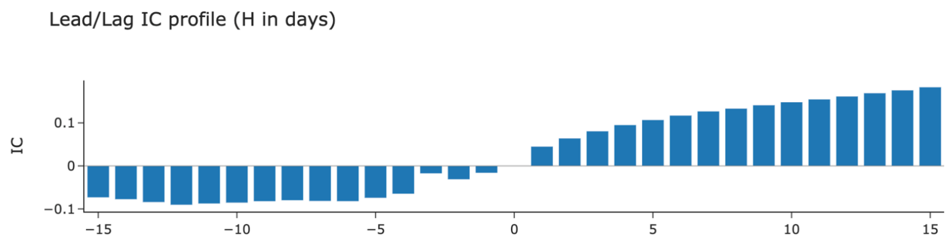

Figure 5.1: Lead–lag IC profile across horizons*

Figure 5.2: Rolling IC over time for a 5 day horizon.*

*The configurations used to generate these plots are available in Appendix C

5.4 Results

Machine Learning Performance

Cross-sectional ML models delivered consistent predictive power.

- One-day horizon models achieved the strongest ICs and the most robust risk-adjusted portfolio returns.
- Predictive power decayed at longer horizons (5-day, 21-day), reflecting increased noise in medium-term forecasts.

Lag (days)	IC Mean
1	0.091
3	0.087
5	0.086
10	0.070

Table 5.1: Lag vs. Mean Information Coefficient (IC)*

- Ridge models emerged as the most effective, balancing interpretability with stability.

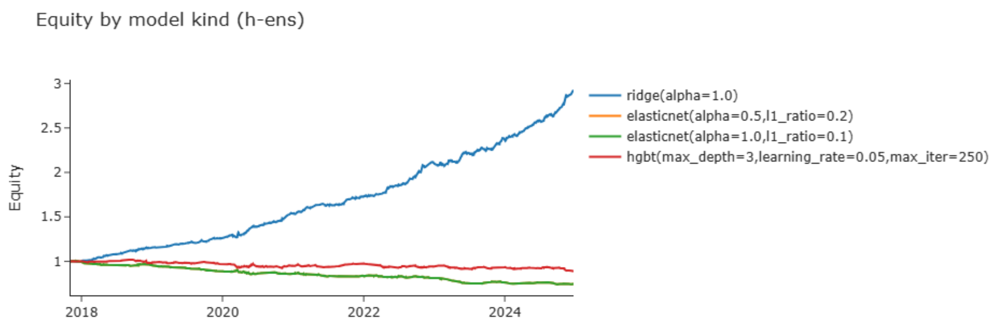

Figure 5.3: Model Performance Comparison*

*The configurations used to generate these plots are available in Appendix C

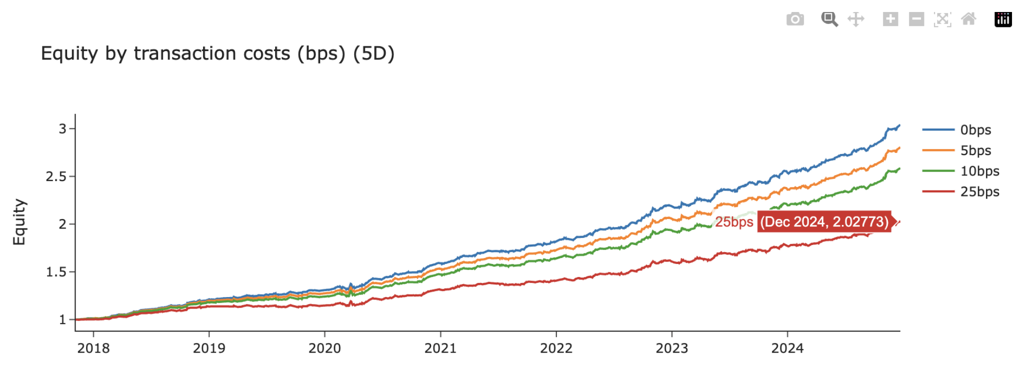

Figure 5.4: Equity curve v/s transaction costs for a 5 day horizon*

Ensemble and Risk-Control Enhancements

Lag- and horizon-ensembles improved performance by blending predictions across lags/horizons with IC-based weights. Risk overlays further enhanced robustness:

- Beta-neutralization reduced systematic exposures.
- ADV eligibility screens improved liquidity realism.
- Together, these overlays produced lower drawdowns and more stable Sharpe ratios across regimes.

5.5 Summary of Findings

Three key insights emerge:

1. **Standalone RL agents underperform.** Despite extensive effort, PPO and A2C failed to generalize, confirming the difficulty of DRL in live-like settings.
2. **Cross-sectional ML models capture behavioral signals effectively.** Ridge regression consistently extracted predictive content, particularly at short horizons.
3. **Ensembles and overlays maximize robustness.** Horizon-ensembles with IC-based weighting, combined with beta-neutralization and ADV screens, produced the most stable and interpretable performance profiles.

In conclusion, while RL represented an important prior step, the experimental evidence strongly favors ML ensembles guided by behavioral proxies. By combining rigorous walk-forward validation with structured overlays, the refined framework provides a credible and reproducible path toward adaptive trading strategies.

*The configurations used to generate these plots are available in Appendix C

Discussion

6.1 Interpretation and Limitations

The results of this thesis underscore the value of incorporating behavioral signals into machine learning based trading strategies.

The central hypothesis that changes in trader behavior, observable through asset-level signals, are more predictable and stable than broad macro regime shifts was supported empirically. Cross-sectional ML models successfully extracted predictive content from behavioral proxies. By conditioning decisions on these richer contextual signals, the models were able to capture aspects of investor psychology that are not visible in raw price series.

The analysis revealed several key findings:

1. **Cross-sectional ML models** (e.g., Elastic Net, HGBT) consistently produced strong predictive power at short horizons, particularly around volatility surges where behavioral effects are most pronounced.

2. **Reinforcement learning agents**, despite extensive tuning of reward functions and regularization, failed to generalize out-of-sample. While RL showed adaptability in-sample, the sparse and noisy reward structure of financial markets prevented stable policy learning.

3. **Horizon ensembles and risk-control overlays** (beta-neutralization, ADV eligibility) proved essential for stability, improving robustness across regimes and mitigating systematic exposures.

These findings highlight the importance of methodological adaptability: while RL was initially a promising avenue, empirical evidence necessitated a pivot toward ML based ensembles with explicit risk management.

Despite encouraging robustness in the final system, several limitations remain:

1. **Data Quality and Coverage** Early experiments with retail sentiment (Reddit, Twitter, news APIs) proved too noisy and poorly aligned with market timing. The final reliance on market-implied proxies, while cleaner, may exclude important nuances of crowd narratives and retail positioning.

2. **Evaluation Gaps** Although rolling backtests were conducted, some key metrics such as turnover decomposition, alpha vs. benchmark exposures, and detailed transaction cost modeling were not fully integrated. Formal ablation tests of feature groups and robustness under explicit regime splits also remain to be completed.

3. **Generality of Results** While walk-forward validation reduces look-ahead bias, markets remain inherently non-stationary. A model calibrated on one period (e.g., 2015–2020) may degrade in structurally different conditions (e.g., 2022). The absence of online or adaptive recalibration mechanisms limits continuous adaptation.

4. **Interpretability** Even with modular feature engineering, ML ensembles and, especially, RL agents act as partial black boxes. Explaining why signals align with specific behavioral hypotheses remains challenging, particularly for tree ensembles and high-dimensional z-scored feature blends.

Addressing these limitations suggests several promising directions for future work:

1. Adding explainability layers to ML models (e.g., SHAP, permutation importance) to better link predictive performance with behavioral theory.

2. Incorporating adaptive retraining triggers or meta-learning techniques to better handle structural breaks and concept drift.

3. Developing synthetic or agent-based market simulators to test strategy robustness under extreme conditions, including feedback effects from execution and liquidity constraints.

Overall, the discussion highlights both the potential and the constraints of behaviorally informed learning systems. While reinforcement learning struggled to generalize, cross-sectional ML models with behavioral proxies and explicit risk overlays proved effective, underscoring the value of rigorous methodology and careful signal design in adaptive trading research.

Conclusion and Future Work

This thesis investigated the integration of behavioral finance, machine learning (ML), and reinforcement learning (RL) in the design of adaptive trading strategies. Building on the foundations of behavioral finance, we examined how behavioral biases can give rise to persistent anomalies (momentum, reversals, etc.) that can manifest in technical data. To capture these dynamics, we developed a large set of features spanning technical, macroeconomic, and behavioral domains, systematically engineered through a modular feature registry and validated with rolling backtests.

The research followed an iterative trajectory. Early experiments attempted to integrate unstructured retail sentiment data from social media and news APIs, but these proved too noisy, difficult to align temporally, and hard to validate. We therefore pivoted to more structured, market-implied behavioral proxies. These proxies provided more stable and interpretable signals of investor psychology.

Subsequent experiments compared reinforcement learning agents with supervised and ensemble ML approaches. Despite substantial effort in shaping reward functions, applying regularization, and testing hybrid ML–RL configurations, RL agents consistently struggled to generalize out-of-sample. In contrast, cross-sectional ML ensembles, augmented with risk-control overlays such as beta-neutralization and ADV eligibility, demonstrated more reliable predictive power and superior risk-adjusted performance. This outcome underscores the importance of methodological adaptability in financial research: while RL remains conceptually appealing, the evidence in this study favored ML based methods as more robust in practice.

Several limitations of the present work remain. Markets are inherently non-stationary and partially observable, which challenges model stability and generalization. Although our framework incorporated walk-forward testing, it lacked online recalibration or adaptive retraining mechanisms to address structural breaks. Furthermore, interpretability remains limited: even behaviorally motivated ML ensembles act as partial black boxes, making it difficult to link predictions directly to behavioral theory.

Future research should therefore pursue several directions. First, the integration of explicit behavioral models (e.g., prospect theory) into learning objectives could more closely align predictive outputs with documented investor psychology. Second, methods to enhance interpretability, such as SHAP or causal feature attribution, could help bridge the gap between model outputs and behavioral finance hypotheses. Third, adaptive retraining and meta-learning approaches offer a promising avenue to improve robustness under regime shifts. Finally, expanding the feature space to include richer alternative data, such as real-time social media sentiment, Google Trends, or cross-asset flow measures, may provide new behavioral signals, particularly in emerging domains such as cryptocurrencies.

In conclusion, this work demonstrates that behaviorally informed ML ensembles, supported by explicit risk-control overlays, provide a credible and effective pathway for adaptive trading strategies. While reinforcement learning struggled to deliver generalizable results in this setting, the broader framework illustrates how combining psychological insights with rigorous ML methodology can advance the development of robust, adaptive strategies in financial markets.

Appendix A: Indicator Definitions and Behavioral Proxies

This appendix groups indicators by category. For each group, we first provide concise definitions (with formulas when relevant), then explain their behavioral interpretation as proxies for investor psychology.

1. Trend and Momentum

Feature	Definition
ROC	$\frac{P_t - P_{t-n}}{P_{t-n}} \times 100$
Momentum	$P_t - P_{t-n}$
MACD	$EMA_{12}(P) - EMA_{26}(P)$; Signal $= EMA_9(MACD)$
RSI	$100 - \frac{100}{1+RS}$, $RS = \frac{AvgGain}{AvgLoss}$
Stochastic	$\%K = \frac{C - L_n}{H_n - L_n} \times 100$, $\%D = SMA(\%K)$
Williams %R	$-100 \cdot \dfrac{H_n - C_t}{H_n - L_n}$
CCI	$\dfrac{TP - MA_n(TP)}{0.015\sigma_n(TP)}$, $TP = \frac{H+L+C}{3}$
TRIX	Triple-smoothed EMA ROC
Aroon Osc.	Time since highest high vs lowest low
Supertrend	ATR-based trailing filter

Behavioral Proxy: Trend and momentum indicators capture *underreaction* (slow adjustment to news), *herding* (trend-chasing), and prolonged runs often reflect *overconfidence*. Reversals indicate exhaustion of herding or panic exits.

2. Volatility and Risk

Feature	Definition
ATR	$EMA(\max\{\lvert H-L\rvert, \lvert H-C_{t-1}\rvert, \lvert L-C_{t-1}\rvert\})$
Bollinger Bands	$MA \pm k\sigma$
%b	$\frac{P-Lower}{Upper-Lower}$
Bandwidth	$\frac{Upper-Lower}{Middle}$
STARC Bands	$Upper = MA + k \cdot ATR,\ Lower = MA - k \cdot ATR$
ATR Bands	$Upper = MA + k \cdot ATR,\ Lower = MA - k \cdot ATR$
Hist. Volatility	$\sqrt{252} \cdot \sigma(\Delta \log P)$
Ulcer Index	$\sqrt{\frac{1}{n}\sum DD_i^2}$, DD = drawdown %
Mass Index	Range-expansion reversal detector
Fractals	Local 5-bar highs/lows
High–Low Channel	$Top = HH_n,\ Median = (HH_n + LL_n)/2,\ Bottom = LL_n$

Behavioral Proxy: Volatility measures proxy for *fear and uncertainty* (spikes, wide bands), *complacency* (narrow bands), and *loss aversion* (Ulcer Index, drawdown depth).

3. Volume and Flow

Feature	Definition
OBV	$OBV_{t-1} \pm V_t$ (add if $C_t > C_{t-1}$)
A/D Line	$AD = AD_{t-1} + \frac{(C-L)-(H-C)}{H-L} \cdot V$
Chaikin MF	$\frac{\sum((C-L)-(H-C))/(H-L) \cdot V}{\sum V}$
Chaikin Vol.	Change in $EMA(H-L)$
Klinger Osc.	Volume flow EMA-long vs EMA-short
MFI	RSI applied to Price · Volume
Relative Vol.	$\frac{V_t/MA_n(V)}{MA_m(V/MA_n(V))}$
NVI	$NVI_t = NVI_{t-1}(1+ROC_t)$ if $V_t < V_{t-1}$

Behavioral Proxy: Volume indicators measure *herding and conviction* (volume surges), *sentiment shifts* (divergences between price and volume), and *informed vs noise trading* (NVI on low-volume days).

4. Overbought / Oversold & Mean Reversion

Feature	Definition
RSI	$100-\frac{100}{1+RS}$
Stochastics	$\%K=\frac{C-L}{H-L}\times 100$
Williams %R	$-100\cdot\frac{H-C}{H-L}$
CCI	$\frac{TP-MA}{0.015\sigma}$
QStick	$SMA(C-O)$
PGO	$\frac{C-SMA(C)}{ATR}$
Rainbow Osc.	Weighted blend of EMAs
DPO	$P_{t-\lfloor(n/2+1)\rfloor}-SMA_n(P)$

Behavioral Proxy: Overbought/oversold oscillators highlight extremes in *optimism and pessimism*. They proxy for *crowd overshooting*, *capitulation* in selloffs, and contrarian mean-reversion opportunities.

5. Composite and Exotic

Feature	Definition
Vortex	$VI^{+}=\frac{\sum\lvert H_t-L_{t-1}\rvert}{ATR}, VI^{-}=\frac{\sum\lvert L_t-H_{t-1}\rvert}{ATR}$
Gator Osc.	Alligator MA differences
Schaff Cycle	Stochastic of MACD
Parabolic SAR	$PSAR=PSAR_{t-1}+AF\cdot(EP-PSAR_{t-1})$
Prime Bands	Price levels near primes

Behavioral Proxy: Composite indicators capture *regime changes*, *nonlinear herding*, and turning points. They serve as robustness checks when multiple exotic signals align, indicating phases of excessive optimism or fear.

Appendix B: Codebase and Github Repository

The codebase developed throughout this research project as well as the final explanatory notebook are made available on the link below:

https://github.com/gusteauw/rp-adaptive-ml-trading

Appendix C: Configurations

Configuration for Figures 5.1, 5.2, 5.3, and 5.5


```
{
  "run_id": "20250901-174107",
  "start_date": "2014-01-01",
  "end_date": "2024-12-31",
  "target_horizons": [5, 21],
  "max_h": 21,
  "train_years": 3,
  "step_months": 1,
  "embargo_days": 21,
  "rebalance_freq": "W-FRI",
  "long_short": true,
  "top_quantile": 0.3,
  "holding_period": 5,
  "tcost_bps": 10,
  "random_state": 42,
  "n_tickers": 56,
  "n_rows_df": 154657,
  "n_features": 72
}
```

Configuration for Figure 5.4


```
{
  "run_id": "20250902-150714",
  "start_date": "2014-01-01",
  "end_date": "2024-12-31",
  "target_horizons": [5, 21],
  "max_h": 21,
  "train_years": 3,
  "step_months": 1,
  "embargo_days": 21,
  "rebalance_freq": "W-FRI",
  "long_short": true,
  "top_quantile": 0.3,
  "holding_period": 10,
  "tcost_bps": 5,
  "random_state": 42,
  "n_tickers": 56,
  "n_rows_df": 154657,
  "n_features": 72
}
```

Bibliography

[1] Barberis, N., & Thaler, R. (2003). A Survey of Behavioral Finance. In G. Constantinides, M. Harris, & R. Stulz (Eds.), *Handbook of the Economics of Finance.*

[2] Kahneman, D., & Tversky, A. (1979). Prospect Theory: An Analysis of Decision under Risk. *Econometrica.*

[3] Xu, W. (2005). Behavioral bias-driven trading and return momentum (Doctoral dissertation, Purdue University).

[4] Tetlock, P. C. (2007). Giving Content to Investor Sentiment: The Role of Media in the Stock Market. *Journal of Finance*, 62(3), 1139–1168.

[5] So, S. M. S., & Lei, V. U. (2015). On the relationship between investor sentiment, VIX and trading volume. *Risk Governance & Control: Financial Markets & Institutions*, 5(4-1), 114–122.

[6] Salehpour, A., & Samadzamini, K. (2023). Machine Learning Applications in Algorithmic Trading: A Comprehensive Systematic Review. *International Journal of Education and Management Engineering*, 13(6), 41–53.

[7] Millea, A. (2021). Deep Reinforcement Learning for Trading — A Critical Survey. *Data*, 6(11), 119.

[8] Bai, Y., Gao, Y., Wan, R., Zhang, S., & Song, R. (2024). A Review of Reinforcement Learning in Financial Applications. *arXiv preprint.*

[9] Mnih, V., et al. (2015). Human-level control through deep reinforcement learning. *Nature*, 518, 529–533.

[10] Mnih, V., et al. (2016). Asynchronous Methods for Deep Reinforcement Learning (A3C/A2C). *Proceedings of ICML.*

[11] Schulman, J., et al. (2017). Proximal Policy Optimization Algorithms. *arXiv preprint*, 1707.06347.

[12] Koratamaddi, P., Wadhwani, K., Gupta, M., & Sanjeevi, S. (2021). Market sentiment-aware deep reinforcement learning approach for stock portfolio allocation. *Engineering Science and Technology*, 24(4), 1717–1727.

[13] Ye, A., Xu, J., Veedgav, V., Wang, Y., Yu, Y., Yan, D., & Xu, S. (2024). Learning the Market: Sentiment-Based Ensemble Trading Agents. *arXiv preprint*.

[14] Chen, Y.-F., & Huang, S.-H. (2021). Sentiment-influenced trading system based on multimodal deep reinforcement learning. *Applied Soft Computing*, 112, 107788.

[15] Quantified Strategies. (2023). Trading Performance: Strategy Metrics, Risk-Adjusted Metrics, and Backtest.