Implementing behavioral biases into financial reinforcement learning with reward functions and overfitting

Mateusz Buczyński[a], Marcin Chlebus[a], Marcin Zajenkowski[b]

[a] *Faculty of Economic Sciences, University of Warsaw, Dluga 44/50, Warsaw, 00-241, PL*
[b] *Faculty of Psychology, University of Warsaw, Stawki 5/7, Warsaw, 00-183, PL*

Abstract

Reinforcement learning (RL) agents typically optimize objective reward functions to achieve task performance. However, in real-world decision-making, in financial contexts, agents often operate under bounded rationality and are influenced by cognitive biases. This paper explores how some behavioral biases (including varying attitude towards risk and loss) can be systematically embedded into the reward functions of RL agents, moving beyond the view of biases as noise to be eliminated. We also propose a framework for testing biases such as overconfidence and intelligence into financial RL testing process. We further analyze how these modifications affect learning dynamics, agent behavior, and economic outcomes. We find that reward function design strongly shapes RL agent behavior: risk-seeking agents achieve higher returns in favorable markets but incur elevated downside risk, while risk-averse agents provide stable yet less profitable outcomes. Additionally, we observe that agents trained on extended datasets without volatility-aware objectives exhibit overconfidence-like behaviors, overfitting to past trends and reducing adaptability.

Keywords: financial markets, deep learning, reinforcement learning, forecast comparison, reward functions

1. Introduction

Reinforcement learning has achieved large success in domains ranging from game playing through robotics to financial modeling. At its core, it relies on an agent interacting with an environment to maximize cumulative rewards, with the reward abstraction serving as the agent's primary source of feedback. However, the implementation of this reward function is both critical and non-trivial, often determining the agent's learning dynamics, convergence and ultimately - performance.

Traditionally, reward functions are constructed to reflect objective task performance (Sutton & Barto, 2020). Yet, in many real-world decision-making scenarios, agents - both artificial and human - operate under bounded rationality and are influenced by cognitive biases (Jara-Ettinger et al., 2016; Leibo et al., 2017). In economics and psychology, these biases are well-documented, often leading to systematic deviations from optimality, i. a. prospect theory (Tversky & Kahneman, 1974), time-inconsistent and endowment-biased preferences (Kopczewski & Bil, 2024; O'Donoghue & Rabin, 1999) or herding (Banerjee, 1992), among others, have been shown to influence human decision processes.

Our work explores how such behavioral biases can be explicitly embedded into the reward functions of reinforcement learning agents. Rather than treating biases as noise or irrationality to be eliminated, we investigate the potential of using them to guide learning, imagining that real agents - investors, also follow similar, biased reward systems. Beyond shaping reward functions, biases can be modeled through exposure to biased data, overtraining to simulate overconfidence, or modifying agent's action space to reflect heuristic decision patterns. This perspective opens new directions in financial reinforcement learning design: agents that simulate human-like behavior for modeling of human-driven systems.

Our contribution is two-fold. First, we demonstrate how incorporating behavioral biases, such as overconfidence, risk and loss aversion, into reinforcement learning reward functions influences learning parameters and the training dynamics in real environments. Second, we evaluate their impact on economic criteria and the agent's capacity to replicate human-like strategies in uncertain, noisy domains.

2. Literature review

Early applications of reinforcement learning (RL) in financial forecasting laid the foundation for modern algorithmic trading systems. Notably, Moody and Saffell (2001); Nevmyvaka et al. (2006) pioneered adaptive learning models that leveraged past market data to optimize trading strategies dynamically. These early studies framed financial trading as a sequential decision-making problem, where RL agents learned policies to maximize a predefined reward function, typically tied to profit or risk-adjusted returns. Building on these foundations, subsequent research expanded RL applications into more complex trading environments, e.g. Chan and Shelton (2001) developed a simulation model focusing on the trading dynamics of a single security. Their framework allows the market maker to achieve multiple objectives, such as maximizing profits and minimizing the bid-ask spread. Another paper by Nevmyvaka et al. (2006) applies RL to high-frequency trading, using limit order book data, emphasizing the need for adaptability in rapidly evolving financial markets. Avellaneda and Stoikov (2008) proposed a stochastic control framework for a market maker operating in a limit order book. The trader chooses bid and ask quotes dynamically to maximize expected utility of terminal wealth, modeled using exponential utility. In these early implementations, reward functions were primarily designed to optimize standard performance metrics, such as cumulative returns or Sharpe ratios, without incorporating behavioral considerations.

Reward function design has undergone significant refinement since then. Early approaches often relied on cumulative reward signals rooted in basic reinforcement learning setups (such as cumulative profit), as formalized by Sutton and Barto (2020), but such naive formulations often led to instability or unintended behavior in complex environments. As such, Nevmyvaka et al. (2006) proposed that incorporating execution costs and market impact into the reward structure improves real-world robustness, while Avellaneda and Stoikov (2008) introduced a utility-based reward framework where a market maker maximizes the expected exponential utility of terminal wealth, explicitly penalizing inventory risk. More recently, more researchers are developing topics like reward shaping and programmatic reward design that encode domain constraints directly into the learning process, while reward machines allow modular specification of complex objectives (Camacho et al., 2019; Icarte et al., 2022). An alternative to manually designing reward functions is called inverse RL, which seeks to infer the reward structure from observed expert behavior. Ng and Russell (2000) first formalized inverse RL as the problem of recovering an unknown reward function given expert demonstrations. Abbeel and Ng (2004) later developed practical algorithms that enabled RL agents to mimic expert behavior in robotics. More recently, Google introduced Receding Horizon Inverse Planning (RHIP) algorithm demonstrating scalability in inverse RL by efficiently estimating reward functions in large-scale applications like Google Maps, enabling more accurate modeling of user behavior Barnes et al. (2024). All these advancements reflect a broader trend - reward functions are no longer just performance measures but central tools for controlling agent behavior in sparse reward environments.

With the advent of deep reinforcement learning there has been increasing interest in reward function design, particularly in aforementioned environments. Vecerik et al. (2018) demonstrated that auxiliary rewards, such as predicting future states or leveraging expert demonstrations, significantly improved sample efficiency in deep RL. Similarly, Burda et al. (2018) introduced curiosity-driven rewards, an intrinsic motivation mechanism that encouraged exploration in sparse-reward environments. Despite these advances, financial applications of RL have remained largely reliant on conventional reward formulations. Historically, trading agents have been designed to optimize monetary reward signals, such as immediate profit or risk-adjusted returns. More recently, Deng et al. (2017) demonstrated that deep RL could learn effective trading policies purely from raw financial data, optimizing reward functions based on immediate returns without requiring intricate reward engineering. Several contemporary studies continue to adopt this approach. For instance, Otabek and Choi (2024) applied Deep Q-Networks to optimize Bitcoin trading strategies, using a simple financial reward function centered on maximizing trading profits. Similarly, Cao et al. (2024:) explored high-frequency trading strategies using Proximal Policy Optimization, employing a straightforward reward function based on the Sharpe ratio. Additionally, Goluža et al. (2024) introduced a novel RL-based framework that integrates imitation learning to mitigate market noise while prioritizing financial gains.

Despite the success of these approaches in financial applications, several limitations arise from their reliance on traditional reward formulations. Financial markets are not solely driven by rational decision-making, but are influenced by cognitive and emotional biases exhibited by market participants. Standard RL models typically assume agents behave as utility-maximizing rational actors, consistent with classical economic theory (Von Neumann & Morgenstern, 1944). However, such assumptions fail to capture the behavioral biases observed in real-world trading. Behavioral finance literature has extensively documented systematic deviations from rationality, including cognitive distortions that significantly impact financial decision-making (Barberis & Thaler, 2002; Kahneman & Tversky, 1979). These include: (a) **loss aversion** - a disproportionate sensitivity to losses relative to gains (Kahneman & Tversky, 1979); (b) **risk aversion** - a tendency to overweigh potential negative outcomes, leading to suboptimal avoidance of uncertainty (Pratt, 1964); (c) **overconfidence** - the overestimation of one's information accuracy or forecasting ability, often resulting in excessive trading (Fischhoff et al., 1977; Odean, 1998); These behavioral distortions suggest that effective modeling of financial agents requires reward functions that can account for bounded rationality and cognitive biases, potentially through hybrid RL-behavioral frameworks.

Prospect theory (Kahneman & Tversky, 1979), which laid ground for many other biases to be found, introduced a value function that is concave for gains, convex for losses, and steeper for losses, encapsulating the essence of loss aversion. Several researchers have incorporated prospect theory - based reward structures into RL models to simulate more human-like trading behavior. For instance, Prashanth, L.A. et al. (2016) modified reward functions to weight gains and losses asymmetrically, allowing RL agents to exhibit greater conservatism during market downturns. In other paper Ramasubramanian et al. (2021) introduced a cumulative prospect theory utility function instead of expected returns, showing that their approach is better aligned to mimick human investors. Other studies have explored risk-sensitive reinforcement learning, where agents are incentivized not just to maximize expected returns but to account for downside risk. Mihatsch and Neuneier (2002) introduced risk-sensitive utility functions that penalized volatility, while Eriksson and Dimitrakakis (2019) experimented with utility-based functions explicitly modeling risk preferences. Similarly, Chow et al. (2015) proposed Conditional Value-at-Risk-constrained reinforcement learning. Another promising direction involves multi-objective RL, where hybrid reward functions balance traditional performance metrics with behavioral bias corrections. Peschl et al. (2021) proposed a system for combining rewards from several different experts, they present their results on tasks that require an agent to act and choose between conflictive choices. In another study, Fulfillment Priority Logic was introduced, a framework that allows to define logical formulas representing intentions and priorities in multi-objective RL, which agents should follow. Interdisciplinary approaches that merge neuroeconomics and RL have also emerged. Studies such as Schultz et al. (1997) have examined how dopamine neurons encode reward prediction errors or newer paper by Wang et al. (2018) proposes that the prefrontal cortex functions as a meta-reinforcement learning system, dynamically adjusting reward-based learning strategies based on context and past experiences.

An additional psychological perspective relevant to reward function design in RL is provided by Reinforcement Sensitivity Theory (RST) (Corr & Cooper, 2016; Gray & McNaughton, 2007), which models individual differences in sensitivity to reward and punishment through two key neuropsychological systems: the Behavioral Inhibition System (BIS) and the Behavioral Activation System (BAS) (Carver & White, 1994). The BIS is sensitive to signals of punishment, uncertainty, and novelty, promoting avoidance behaviors and heightened sensitivity to potential losses, thus linking directly to phenomena such as loss aversion and risk aversion. Conversely, the BAS responds to cues of reward and goal attainment, fostering approach behaviors and correlating with overconfidence and risk-seeking tendencies. Empirical studies have demonstrated that individual differences in BIS/BAS activation predict variation in financial risk-taking behavior (Vermeersch et al., 2013), investment decision-making (Peterson, 2007), and susceptibility to biases such as overconfidence (Krupić, 2017; Visser et al., 2019). Integrating RST-inspired mechanisms into RL models could enable agents to exhibit heterogenous, human-like behavioral patterns under risk and uncertainty. For example, Kim and Lee (2011) showed that individuals with higher BAS and low BIS tend to exhibit lower loss aversion and higher willingness to take financial risks, while those with higher BIS demonstrate conservative gambling behaviors. These insights suggest that parametrizing reward functions or policy updates to reflect varying BIS/BAS profiles could enhance the behavioral realism of RL agents in financial contexts, but the application needs to be applied on an individual level

Recent studies have also begun to explore how RL models exhibit overconfidence bias, mostly on Large Language Models (LLM) example. Hayes et al. (2024) demonstrated that RL-based decision-making models develop relative value biases, reinforcing overconfidence in suboptimal strategies. Similarly, Leng et al. (2025); Li et al. (2024) showed that human-designed reward functions can inadvertently encourage overconfident behaviors, leading to premature convergence on flawed policies. In case of Leng et al. (2025) they also provide a way to reshape reward calculation so that the models are less overconfident. Yang et al. (2024) proposed knowledge transfer techniques to mitigate overconfidence in large language models (LLMs), suggesting that similar methods could be applied to RL environments. In the case of financial markets overconfidence can be viewed through the lens of overfitting. Overfitting occurs when an agent over-optimizes for a specific environment, leading to poor generalization in new conditions. We believe that in RL, excessive training in a stable market environment can reinforce rigid, overfitted policies, preventing adaptability in volatile conditions, which for a trained agent would resemble a human-like agent that is biased with overconfidence.

What we also notice is that one critical area of research is the divergence between expected and realized volatility in financial markets. Investors may overestimate short-term risks during turbulent periods (causing excessive caution) or underestimate risk during stable conditions (leading to overconfidence). This effect is particularly evident in the equity premium puzzle (Mehra & Prescott, 1985), where expected returns often exceed what is predicted by standard risk models, suggesting that investors demand higher compensation for perceived risk rather than actual risk. We believe that risk estimation functions in financial models should explicitly incorporate investor expectations, as the divergence between expected and realized rewards remains underexplored in financial reinforcement learning.

Building on aforementioned literature, we hypothesize that the choice and design of reward functions will play a large role in shaping the performance and behavior of RL agents in financial markets. Different formulations encode distinct behavioral assumptions and optimization goals. We expect that agents trained with varying reward structures will exhibit systematically different trading strategies and risk profiles. Moreover, by parametrizing reward functions to reflect individual differences in risk preferences and loss aversion, inspired by prospect theory and RST, we hypothesize that agents can be induced to mimic diverse human-like decision-making styles, from highly conservative to risk-seeking. This would allow RL agents to better model the heterogeneity observed among real-world investors. Our approach systematically implements and compares multiple reward formulations grounded in economic and behavioral theories, including prospect theory-based value functions, loss aversion penalties, risk-sensitive utility functions, and volatility-aware objectives. In contrast to earlier financial RL studies that often relied on shallow models or narrow market conditions, we utilize deep RL architectures and conduct experiments across an extensive dataset encompassing both volatile and stable market periods.

Additionally, we hypothesize that overconfidence can manifest in RL agents through overfitting and poor generalization. We expect that agents exposed to larger training datasets or subjected to extended training durations will become less adaptable and more resistant to new market signals. Finally, we hypothesize that reward functions incorporating expected volatility, rather than relying solely on realized returns, will improve agent performance in out-of-sample tests, particularly in turbulent markets. Rather than viewing overfitting as a proxy for intelligence (Fernando et al., 2017; Gigerenzer & Brighton, 2009; Lake et al., 2016), we argue that true intelligence is better reflected in an agent's capacity to adapt to new information and generalize across varying market regimes To investigate the emergence of overconfidence-like behavior, we analyze how increased training data exposure and extended training durations affect agent generalization and adaptability, testing whether agents trained on longer histories become overfit and resistant to new or contradictory market signals.

3. Technical approach

3.1. Reinforcement Learning

Reinforcement Learning (RL) is a machine learning paradigm where an agent learns to make sequential decisions by interacting with an environment. Through a trial-and-error process, the agent refines its policy based on feedback received in the form of rewards, with the goal of maximizing long-term cumulative rewards (Sutton & Barto, 2020).

Formally, RL problems are modeled as a Markov Decision Process (MDP), defined by the tuple (S, A, P, R, γ):

- S - the set of all possible states representing the environment;
- A - the set of all possible actions the agent can take;
- $P(s\prime|s, a)$ - the transition probability function, which defines the probability of reaching state $s\prime$ after taking action a in state s;
- $R(s, a)$ - the reward function, providing scalar feedback for taking action a in state s;
- $\gamma \in [0, 1]$ - the discount factor, which determines the relative importance of future rewards compared to immediate rewards.

At each discrete time step t, the agent observes the current state s_t, selects an action a_t according to its policy $\pi(a|s)$, and transitions to a new state s_{t+1} with probability $P(s_{t+1}|s_t, a_t)$. The agent then receives a reward $r_t = R(s_t, a_t)$, which it uses to improve future decision-making. The agent's objective is to maximize the expected cumulative discounted reward, known as the return G_t, which is expressed as:

$$G_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k}. \tag{1}$$

The optimal policy π^* is the policy that maximizes the expected return G_t from any given state. To achieve this, RL agents often rely on estimating an action-value function $Q(s, a)$, which represents the expected cumulative reward for taking action a in state s and following an optimal strategy thereafter. This function satisfies the Bellman equation:

$$Q^{(s,a)} = \mathbb{E}\left[R(s, a) + \gamma \max_{a'} Q^{(s',a')} \mid s, a\right]. \tag{2}$$

Traditional RL methods struggle with high-dimensional state spaces and complex decision environments.

Modern deep RL algorithms fall into two broad categories: (a) value-based methods estimate $Q^*(s, a)$ using a deep neural network, updating the network weights using temporal difference learning; (b) policy-based methods learn a direct mapping from states to actions, optimizing policies via gradient ascent in the policy space.

Hybrid approaches, such as actor - critic methods, combine both value and policy learning, offering a more stable and efficient framework for training RL agents. As RL research advances, the reward function remains a critical component, directly shaping the agent's learning behavior and influencing the effectiveness of learned policies.

3.1.1. Proximal Policy Optimization

Proximal Policy Optimization (PPO) (Schulman et al., 2017) is a policy optimization algorithm that improves stability in reinforcement learning by constraining the magnitude of policy updates. Instead of applying large and potentially destabilizing updates to the policy parameters, PPO introduces a clipped surrogate objective that ensures incremental improvements while preventing drastic policy shifts.

The foundation of PPO is the probability ratio, which measures the change in policy before and after an update:

$$r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}, \tag{3}$$

where $\pi_{\theta_{\text{old}}}(a_t|s_t)$ is the policy before the update, and $\pi_\theta(a_t|s_t)$ is the updated policy.

To ensure stable updates, PPO employs a clipped surrogate objective:

$$L^{\text{CLIP}}(\theta) = \mathbb{E}_t\left[\min\left(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1 - \epsilon, 1 + \epsilon)\hat{A}_t\right)\right], \tag{4}$$

where $\hat{A}_t$ is the estimated advantage function, and ϵ is a hyperparameter that controls the range within which updates are allowed. The clipping mechanism ensures that $r_t(\theta)$ remains within a predefined range, preventing excessively large policy updates that could degrade learning stability.

To estimate the advantage function $\hat{A}_t$, PPO uses Generalized Advantage Estimation (GAE), which smooths the advantage estimate across multiple time steps:

$$\hat{A}_t = \delta_t + (\gamma\lambda)\delta_{t+1} + \cdots + (\gamma\lambda)^{T-t}\delta_T, \tag{5}$$

where $\delta_t = r_t + \gamma V(s_{t+1}) - V(s_t)$ represents the temporal difference (TD) error, V(s) is the estimated value function, γ is the discount factor, and λ controls the extent to which bootstrapped estimates influence the advantage computation.

PPO alternates between collecting trajectories through interactions with the environment and performing multiple epochs of stochastic gradient descent (SGD) on minibatches of data. To further encourage exploration, PPO includes an entropy bonus, given by:

$$L^{\text{ENTROPY}}(\theta) = -\beta \sum_a \pi_\theta(a|s) \log \pi_\theta(a|s), \tag{6}$$

where β is a hyperparameter controlling the strength of the entropy regularization.

The final objective function combines:

1. the clipped policy loss to ensure stable updates,
2. the value function loss to improve state-value estimation, and
3. the entropy bonus to promote exploration:

$$L(\theta) = \mathbb{E}_t \left[L^{\text{CLIP}}(\theta) - c_1 L^{\text{VF}}(\theta) + c_2 L^{\text{ENTROPY}}(\theta) \right], \tag{7}$$

where $L^{\text{VF}}(\theta)$ is the squared-error loss for the value function, and c_1, c_2 are weighting coefficients.

During training, PPO iteratively samples trajectories from the environment, updates the policy using gradient ascent, and refines the agent's decision-making by constraining policy shifts, leading to a more stable and sample-efficient learning process.

3.1.2. Advantage Actor Critic

Advantage Actor-Critic (A2C) (Mnih et al., 2016) is an on-policy reinforcement learning algorithm that integrates policy-based learning (actor) with value-based learning (critic) to improve stability and efficiency. A2C is a synchronized and computationally efficient variant of Asynchronous Advantage Actor-Critic (A3C), designed for streamlined training.

A2C consists of two primary components:

(a) actor - learns a policy $\pi_\theta(a|s)$ that maps states s to actions a, aiming to maximize the expected cumulative reward;

(b) critic - learns a value function $V_\phi(s)$ that estimates the expected return from a given state s, serving as a baseline for policy updates.

The actor optimizes the policy using the policy gradient theorem, which states that the gradient of the expected return can be expressed as:

$$\nabla_\theta J(\theta) = \mathbb{E}_t \left[\nabla \theta \log \pi_\theta(a_t|s_t) \hat{A}_t \right], \tag{8}$$

where $\hat{A}_t = Q(s_t, a_t) - V(s_t)$ is the advantage function, which determines whether taking action a_t in state s_t leads to higher-than-expected rewards, $Q(s_t, a_t)$ is the action-value function, estimating the expected return of taking action a_t in state s_t and $V(s_t)$ is the state-value function, estimating the expected return from state s_t The advantage function $\hat{A}_t$ provides a learning signal to improve the policy by favoring actions that perform better than expected, while discouraging suboptimal ones.

The critic updates the value function $V_\phi(s)$ by minimizing the temporal difference (TD) error, defined as:

$$L_{\text{critic}}(\phi) = \mathbb{E}_t\left[\left(r_t + \gamma V\phi(s_{t+1}) - V_\phi(s_t)\right)^2\right], \tag{9}$$

where r_t is the reward received at time t, and γ is the discount factor determining the weight of future rewards. This loss function ensures that the critic accurately estimates the expected future return.

To encourage exploration, A2C incorporates an entropy regularization term, defined as:

$$\mathcal{H}(\pi_\theta) = -\sum_a \pi_\theta(a|s) \log \pi_\theta(a|s), \tag{10}$$

which maximizes policy entropy, preventing premature convergence to suboptimal deterministic policies.

The total A2C objective function integrates the actor loss, critic loss, and entropy bonus, balancing learning efficiency and exploration:

$$L(\theta, \phi) = -\mathbb{E}_t\left[\nabla\theta \log \pi_\theta(a_t|s_t)\hat{A}_t\right] + c_1\mathbb{E}_t\left[\left(r_t + \gamma V\phi(s_{t+1}) - V_\phi(s_t)\right)^2\right] - c_2\mathbb{E}_t\left[\mathcal{H}(\pi_\theta)\right], \tag{11}$$

where c_1 and c_2 are hyperparameters that balance the contributions of the critic loss and entropy regularization. The first term corresponds to the actor's objective, maximizing policy improvement. The second term represents the critic's loss, minimizing the TD error. The third term promotes policy exploration through entropy regularization.

A2C alternates between sampling environment interactions and performing gradient-based updates, enabling stable training while efficiently leveraging both policy-based and value-based learning. Its synchronous nature ensures that updates are computed deterministically, reducing variance and improving sample efficiency compared to its asynchronous counterpart, A3C.

3.1.3. Reward functions

The reward function plays a fundamental role in shaping an agent's behavior. It serves as the primary feedback mechanism that guides the learning process, determining how an agent evaluates the desirability of different actions in a given state (Ng et al., 1999). The goal of an RL agent is to maximize the expected cumulative reward over time, as expressed in equation 1.

An appropriately designed reward function ensures that the agent learns a policy that is aligned with the desired objectives. Conversely, a poorly defined reward function may lead to reward hacking, where the agent discovers unintended strategies that optimize for the specified reward but fail to achieve meaningful goals (Amodei et al., 2016).

The simplest form of a reward function assigns a numerical value to each state-action pair based on direct performance metrics:

$$R(s,a) = \begin{cases} +1, & \text{if the action leads to a favorable outcome,} \\ -1, & \text{if the action leads to an unfavorable outcome.} \end{cases} \tag{12}$$

Other form does it in a continuous manner:

$$R(s,a) = f(\cdot), \tag{13}$$

where f is a function of an underlying metric, such as profit or return.

Another technique includes reward shaping that modifies the original reward function to accelerate learning while preserving the optimal policy. One well-known technique is potential-based reward scaling (Ng et al., 1999):

$$F(s, s') = \gamma\Phi(s') - \Phi(s), \tag{14}$$

where $\Phi(s)$ is a potential function encoding additional domain knowledge. This technique ensures policy invariance while guiding the agent towards desirable behaviors.

In financial reinforcement learning, reward functions are usually tailored to optimize trading strategies or portfolio allocation. Some of the most commonly used reward formulations include:

- profit and loss (PnL) (Jiang et al., 2017; Théate & Ernst, 2021):

  $$R_t = P_t - P_{t-1}, \tag{15}$$

  where P_t represents the portfolio value at time t. This formulation encourages the agent to maximize absolute returns.

- risk-adjusted return:
  - Sharpe ratio (Rodinos et al., 2023):

    $$R_t = \frac{\mathbb{E}[r_t]}{\sigma[r_t]}, \tag{16}$$

    where r_t is the return at time t, and $\sigma[r_t]$ is the standard deviation of returns. The Sharpe ratio ensures that the agent maximizes returns while minimizing volatility.

- risk-aware return:
  - VaR-based (Ma & Yu, 2018):

    $$R_t = \mathbb{E}[r_t] - \lambda \cdot \mathrm{VaR}_\alpha(r_t, ..., r_{t-k}), \tag{17}$$

    where λ is a risk penalty factor, and VaR_α is the Value at Risk at confidence level α.
  - CVaR-based (Ni et al., 2024; Ying et al., 2022):

    $$R_t = \mathbb{E}[r_t] - \lambda \cdot \mathrm{CVaR}_\alpha(r_t, ..., r_{t-k}), \tag{18}$$

    where Conditional VaR (CVaR) accounts for extreme losses beyond the VaR threshold.

- behavioral finance-based (prospect theory) (Borkar & Chandak, 2021; Prashanth, L.A. et al., 2016; Von Neumann & Morgenstern, 1944):

  $$R_t = w_+(r_t) \cdot v_+(r_t) - w_-(r_t) \cdot v_-(r_t), \tag{19}$$

  where $v_+(r_t)$ and $v_-(r_t)$ are value functions for gains and losses and $w_+(r_t)$ and $w_-(r_t)$ are probability weighting functions.

4. Experiment setting

Our proposal is to examine loss aversion and risk aversion by designing reinforcement learning (RL) agents with reward functions that explicitly encode these biases. To incorporate it, we propose five distinct reward functions (including two novel ones - based on VaR expectations), each designed to penalize excessive risk-taking and emphasize preservation of capital. These reward functions adjust the agent's sensitivity to drawdowns and volatility, mimicking real-world investor behavior as described in Prospect Theory.

All of them include a standard reward $\mathbb{E}[r_t] = P_t - P_{t-1}$, however each adds factors on top of that:

- risk-loving expected VaR-based - the agent is incentivized to take on higher risk by positively weighting changes in VaR:
$$R = \mathbb{E}[r_t] + \mathbb{E}(\mathrm{VaR}_t) - \mathrm{VaR}_{t-1} \tag{20}$$
- risk-averse expected VaR-based - the agent is negatively incentivized to take on higher risk by negatively weighting changes in VaR:
$$R = \mathbb{E}[r_t] - \mathbb{E}(\mathrm{VaR}_t) - \mathrm{VaR}_{t-1} \tag{21}$$
  These reward functions reward for taking actions that increase or decrease the portfolio risk, instead only looking on the portfolio risk itself.
- loss-averse - the agent is penalized when gaining losses, where penalty factor is equal to 1.5:
$$R = \begin{cases} \mathbb{E}[r_t] & \text{if } \mathbb{E}[r_t] \geq 0 \\ 1.5 \cdot \mathbb{E}[r_t] & \text{if } \mathbb{E}[r_t] < 0 \end{cases} \tag{22}$$
- extreme loss-averse - the agent is extremely penalized when gaining losses, where penalty factor is equal to 2.5:
$$R = \begin{cases} \mathbb{E}[r_t] & \text{if } \mathbb{E}[r_t] \geq 0 \\ 2.5 \cdot \mathbb{E}[r_t] & \text{if } \mathbb{E}[r_t] < 0 \end{cases} \tag{23}$$
- prospect theory based - where agent scales gains and losses nonlinearly, making the agent sensitive to small fluctuations while reducing the impact of extreme returns:
$$R = \begin{cases} \mathbb{E}[r_t]^{0.88} & \text{if } \mathbb{E}[r_t] \geq 0 \\ 2.25 \cdot \mathbb{E}[r_t]^{0.88} & \text{if } \mathbb{E}[r_t] < 0 \end{cases} \tag{24}$$
  Specific parameters of this function were hypertuned on a single asset (S&P 500) in one training period (period I).
- benchmark - gains and losses are equally rewarding or penalizing the agent.

In contrast, we assume that overconfidence bias does not originate directly from the reward function, but rather from the agent's association with data. Specifically, we hypothesize that overfitting - where the agent excessively tailors its strategy to a limited dataset - can serve as a proxy for overconfidence, leading to excessive reliance on historical patterns. An overconfident agent may underestimate (or overestimate if trained on volatile market period) tail risks, assuming that market conditions will remain consistent with past observations. To test this hypothesis, we first evaluate the agent's performance after different numbers of training epochs, assessing whether prolonged training leads to reinforced overconfidence and overfitting or contributes to better generalization. By analyzing how the agent's behavior evolves with increasing training duration, we aim to determine whether excessive exposure to the same dataset results in rigid trading strategies that fail to adapt to new market conditions. Additionally, we investigate whether exposure to a broader dataset (in form of broader time horizon that the agent learns on) mitigates overfitting-induced overconfidence. We assume that intelligence of trading agents (and through that real-life investors) is linked to the ability to generalize across diverse datasets, rather than rigidly adhering to patterns observed in a single, limited environment.

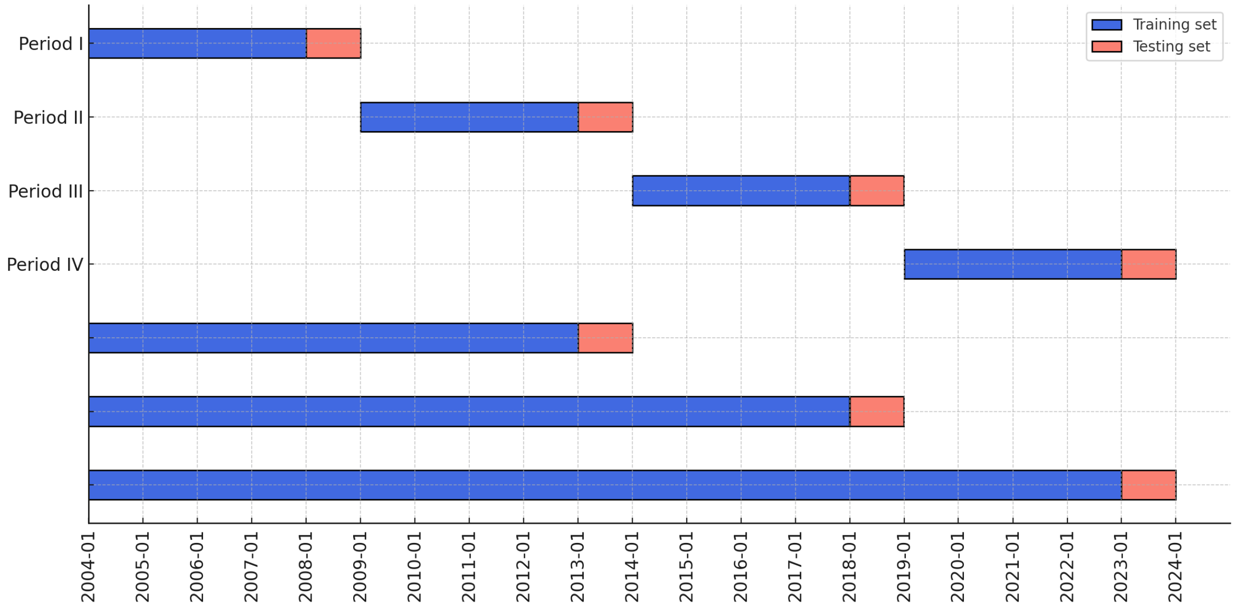
4.1. Technical details

We employ twelve different agents: two models (A2C and PPO) with six different reward functions. We trained and tested these agents in four different time periods:

1. period I: training: 2004-01-01 to 2007-12-31; testing: 2008-01-01 to 2008-12-31;
2. period II: training: 2009-01-01 to 2012-12-31; testing: 2013-01-01 to 2013-12-31;
3. period III: training: 2014-01-01 to 2017-12-31; testing: 2018-01-01 to 2018-12-31;
4. period IV: training: 2019-01-01 to 2022-12-31; testing: 2023-01-01 to 2023-12-31;

In addition to that we also trained agents in expanding window fashion to test for overconfidence when a broader dataset is introduced (this scenario is presented in the figure 1).

Figure 1: Schema of model training and testing in our approach.



The experiment was conducted on a dataset encompassing a diverse set of financial assets, grouped into four categories:

- Indexes: (a) WIG20 (b) S&P 500 (c) FTSE 250 (d) Nikkei 225 (e) NASDAQ 100 (f) Dow Jones Industrial Average (g) KOSPI (h) Shanghai Stock Exchange Composite (i) DAX (j) CAC 40
- Stocks: (a) Apple (b) Meta (only periods II, III and IV) (c) Amazon (d) Tesla (only period II, III and IV) (e) Google (f) Netflix
- Currencies: (a) EUR/PLN (b) GBP/PLN (c) USD/PLN (d) EUR/USD (e) EUR/GBP (f) USD/GBP (g) CHF/GBP (h) CHF/USD (i) EUR/CHF (j) CHF/PLN
- Goods: (a) Bitcoin (only periods III and IV) (b) Gold Futures (c) Brent Crude Oil Futures

The agent's performance was evaluated after several epochs: {5,000; 10,000; 25,000; 50,000; 75,000; 100,000; 150,000; 250,000}. Every model was trained ten times and the reported results were averaged.

To compare agents' results we used four different metrics:

- annual return: $\left(\frac{V_{\text{end}}}{V_{\text{start}}}\right)^{\frac{N}{252}-1}$, where V_{start} is endowment at the beginning of the test and V_{end} is final portfolio value, and N is number of days in testing sample - in this scenario typically equal to 252, full financial year assumed to be 252 days.

- sharpe ratio - $\frac{\frac{1}{N}\sum_{t=1}^{N}(r_t - r_f)}{\sigma(r_t - r_f)} \times \sqrt{252}$, where r_t is return at timestep t, r_f is a risk free rate of return, here assumed to be 0.

- number of trades - number of puts and calls made by an agent, regardless of quantity of shares;

- daily Value-at-Risk - is a VaR at 5% confidence level of all rates od return throughout testing period, calculated as a 5% percentile.

5. Results

In table 1 we demonstrate how different reward functions shape trading behavior, influencing risk, return, and trading frequency in distinct ways.

The risk-loving and benchmark agents consistently exhibit the highest annual returns, particularly in later periods. Risk-loving, achieving 16.38% in period IV, maintains the most aggressive approach, though this comes with significantly higher daily Value at Risk (VaR), peaking at -0.0333 in period I Despite this volatility, it also produces the highest Sharpe ratio (0.6486) in the final period, suggesting that risk-taking is rewarded in favorable market conditions. Benchmark, while also generating strong returns, takes a more balanced approach by executing a significantly higher number of trades, frequently adjusting positions to optimize outcomes.

Conversely, extreme loss-averse and prospect theory prioritize downside protection at the expense of profitability. Extreme loss averse struggles in early periods, with negative returns in period I (-1.86%), but improves over time, reaching 4.53% in period IV. Its low VaR (-0.0031 in period IV) confirms a highly risk-averse strategy. Similarly, prospect theory remains conservative, maintaining low volatility but failing to capitalize on market gains, reflected in its consistently low Sharpe ratio (≈0.27-0.28).

Loss-averse and risk-averse strike a middle ground, delivering moderate returns while keeping risk in check. Loss-averse improves its performance over time, reaching 3.99% in period IV, while risk-averse maintains relatively steady performance across periods, balancing returns and trading frequency. These models maintain higher trade counts, particularly risk-averse, which frequently executes over 270 trades per period, reinforcing its cautious but adaptive approach.

Across time periods, all models perform poorly in period I, reflecting the challenges of pre-2008 market conditions, with risk-loving and benchmark suffering the largest drawdowns. However, in later periods, particularly period II and IV, risk-seeking strategies clearly outperform, while loss-averse models lag behind. The final period highlights the growing gap between these approaches, with risk-loving and benchmark significantly outpacing others in both return and risk-adjusted performance.

Table 1: Performance metrics for different reward functions across various time periods.

Reward Function	2004-2007				2009-2012				2014-2017				2019-2022			
	Annual Return	**Daily VaR**	**No Trades**	**Sharpe**	**Annual Return**	**Daily VaR**	**No Trades**	**Sharpe**	**Annual Return**	**Daily VaR**	**No Trades**	**Sharpe**	**Annual Return**	**Daily VaR**	**No Trades**	**Sharpe**
extreme loss-averse	-0.0186	-0.0062	185.60	-0.0803	46.3739	0.0094	168.26	0.1897	42.1144	-0.0040	181.35	-0.0267	45.3285	0.0164	177.06	0.2808
loss-averse	-0.0551	-0.0114	234.83	-0.1505	58.6539	0.0213	213.38	0.2963	53.4222	-0.0107	221.86	-0.0542	55.4486	0.0399	221.70	0.3939
prospect theory	-0.0193	-0.0064	189.58	-0.0862	47.3674	0.0116	174.29	0.2108	43.6274	-0.0045	185.90	-0.0256	46.4662	0.0164	181.31	0.2726
risk-averse	-0.0349	-0.0066	273.17	-0.2165	68.2274	0.0099	270.72	0.2365	67.7399	-0.0072	271.41	-0.0483	67.8383	0.0367	278.93	0.4803
risk-loving	-0.1631	-0.0333	113.09	-0.2294	28.1650	0.1422	101.77	0.6254	25.6309	-0.0363	105.17	-0.1341	26.2456	0.1638	102.74	0.6486
benchmark	-0.1546	-0.0267	304.13	-0.2407	75.9259	0.1086	303.86	0.5829	76.1357	-0.0241	286.94	-0.1292	71.6938	0.1345	291.60	0.6565

Algorithm	**2004-2007**				**2009-2012**				**2014-2017**				**2019-2022**				**Grand Total**
	Annual Return	**Daily VaR**	**No Trades**	**Sharpe**	**Annual Return**	**Daily VaR**	**No Trades**	**Sharpe**	**Annual Return**	**Daily VaR**	**No Trades**	**Sharpe**	**Annual Return**	**Daily VaR**	**No Trades**	**Sharpe**	
A2C	-0.0838	-0.0175	237.48	-0.1916	0.0606	-0.0074	220.49	0.3946	-0.0154	-0.0086	216.02	-0.0660	0.0763	-0.0080	216.92	0.4736	55.72
5000	-0.0713	-0.0158	197.93	-0.1665	0.0542	-0.0065	185.93	0.3651	-0.0149	-0.0075	182.60	-0.0682	0.0815	-0.0079	192.40	0.4847	47.47
10000	-0.0801	-0.0161	190.13	-0.1633	0.0597	-0.0067	176.15	0.4006	-0.0166	-0.0075	175.36	-0.0928	0.0759	-0.0075	183.26	0.4550	45.34
25000	-0.0817	-0.0159	208.85	-0.1901	0.0550	-0.0066	186.47	0.3463	-0.0168	-0.0079	191.10	-0.0741	0.0708	-0.0072	192.06	0.4386	48.69
50000	-0.0896	-0.0175	241.42	-0.2237	0.0590	-0.0071	217.54	0.3714	-0.0152	-0.0083	212.93	-0.0504	0.0737	-0.0078	217.49	0.4706	55.62
75000	-0.0827	-0.0179	252.32	-0.2017	0.0589	-0.0074	232.65	0.3801	-0.0133	-0.0087	224.29	-0.0506	0.0736	-0.0081	222.10	0.4731	58.25
100000	-0.0835	-0.0186	261.05	-0.1921	0.0619	-0.0078	242.06	0.4065	-0.0141	-0.0091	233.13	-0.0514	0.0769	-0.0084	228.45	0.4733	60.33
150000	-0.0875	-0.0191	268.58	-0.1885	0.0635	-0.0082	255.23	0.4222	-0.0152	-0.0097	245.07	-0.0633	0.0778	-0.0086	241.44	0.4882	63.18
250000	-0.0941	-0.0195	279.57	-0.2067	0.0730	-0.0088	267.93	0.4642	-0.0172	-0.0103	263.69	-0.0771	0.0798	-0.0089	258.15	0.5054	66.88
PPO	-0.0647	-0.0127	195.98	-0.1430	0.0404	-0.0053	190.26	0.3193	-0.0136	-0.0068	201.52	-0.0733	0.0596	-0.0063	200.87	0.4373	49.32
5000	-0.0655	-0.0145	338.03	-0.2102	0.0401	-0.0067	338.76	0.4243	-0.0175	-0.0092	333.45	-0.1155	0.0746	-0.0086	331.17	0.5486	83.88
10000	-0.0632	-0.0140	311.33	-0.2178	0.0389	-0.0063	310.87	0.3746	-0.0159	-0.0082	308.00	-0.1117	0.0682	-0.0079	309.65	0.5248	77.53
25000	-0.0618	-0.0127	231.01	-0.1737	0.0379	-0.0052	223.20	0.3321	-0.0141	-0.0069	232.59	-0.0789	0.0566	-0.0064	233.14	0.4274	57.53
50000	-0.0635	-0.0123	174.96	-0.1326	0.0392	-0.0049	162.87	0.3023	-0.0126	-0.0063	179.47	-0.0569	0.0558	-0.0058	181.13	0.4143	43.68
75000	-0.0642	-0.0121	150.65	-0.1249	0.0402	-0.0049	140.24	0.2931	-0.0125	-0.0061	158.37	-0.0429	0.0557	-0.0055	159.37	0.3873	38.07
100000	-0.0657	-0.0121	136.06	-0.1083	0.0413	-0.0048	128.08	0.3046	-0.0124	-0.0060	148.31	-0.0513	0.0553	-0.0055	145.39	0.3834	34.90
150000	-0.0670	-0.0120	120.44	-0.0917	0.0411	-0.0048	114.43	0.2691	-0.0123	-0.0060	133.71	-0.0651	0.0556	-0.0054	130.94	0.3947	31.25
250000	-0.0669	-0.0120	105.40	-0.0844	0.0443	-0.0048	103.67	0.2541	-0.0113	-0.0058	118.28	-0.0645	0.0551	-0.0052	116.13	0.4179	27.75

Table 2: Performance comparison of A2C and PPO models across different training timesteps and time periods.

In table 2 we present results in regard to the RL model. Across all periods, the A2C model generally performs better with longer training horizons, achieving its best performance at 250,000 steps, where it reaches a Sharpe ratio of 0.505 and an annual return of 7.98% in the period IV. This trend suggests that A2C benefits from extended exploration and refinement, gradually learning a more stable policy. The improvement is particularly clear post-2008, as the model transitions from negative or marginal returns in earlier periods to positive returns with improving risk-adjusted performance.

The PPO model, on the other hand, shows a different learning curve. It achieves competitive or superior Sharpe ratios at much shorter timesteps, with its peak Sharpe value of 0.549 observed already at 5,000 steps. Although its absolute annual returns are generally lower than A2C's at higher steps, PPO consistently maintains lower risk exposure, as indicated by its more favorable daily VaR across nearly all periods and step counts. This suggests that PPO develops more risk-aware policies more quickly, favoring stability over aggressive return-seeking.

In terms of trading behavior, A2C models exhibit higher trading frequency across all timesteps, particularly as training progresses, with average trade counts reaching over 270 trades at 250,000 steps. PPO, in contrast, reduces trading frequency steadily with more training, suggesting that its policies converge toward more selective, long-term positioning.

When comparing performance across time periods, both models struggle in the pre-crisis (2004–2007) period, often posting negative returns and Sharpe ratios. However, they both adapt more effectively to post-crisis and post-pandemic periods, with the 2019–2022 window yielding the best overall results for both A2C and PPO, highlighting how recent market dynamics may be more predictable or model-friendly.

In summary, A2C benefits from longer training and achieves higher returns, particularly in the most recent data, while PPO converges faster to a stable policy and exhibits better risk control early on. These findings suggest that A2C may be preferable in environments where sufficient training time and compute are available, while PPO provides a more robust choice for quicker deployment with lower risk exposure.

Table 3: Performance metrics for selected currencies, goods, and stocks across four time periods. Metrics include annual return, daily VaR, number of trades, and Sharpe ratio.

Asset	2004–2007				2009–2012				2014–2017				2019–2022			
	Ann. Return	**Daily VaR**	**Trades**	**Sharpe**	**Ann. Return**	**Daily VaR**	**Trades**	**Sharpe**	**Ann. Return**	**Daily VaR**	**Trades**	**Sharpe**	**Ann. Return**	**Daily VaR**	**Trades**	**Sharpe**
DJI	-0.1348	-0.0199	92.46	-0.65	0.0981	-0.0054	90.46	1.31	-0.0298	-0.0117	92.82	-0.14	0.0648	-0.0070	91.97	0.93
FCHI	-0.1926	-0.0182	197.19	-0.56	0.0455	-0.0057	176.26	0.36	-0.0472	-0.0059	183.65	-0.37	0.0701	-0.0064	215.34	0.62
FTMC	-0.1909	-0.0158	252.15	-0.70	0.0929	-0.0050	205.67	0.80	-0.0660	-0.0056	195.72	-0.62	0.0117	-0.0066	200.48	0.13
GDAXI	-0.1926	-0.0169	230.42	-0.67	0.0883	-0.0057	206.89	0.69	-0.0703	-0.0067	195.63	-0.54	0.0873	-0.0056	217.03	0.79
KS11	-0.1732	-0.0183	225.75	-0.48	-0.0079	-0.0053	211.66	-0.15	-0.0605	-0.0063	188.09	-0.36	0.0691	-0.0063	197.48	0.52
N225	-0.1385	-0.0168	208.52	-0.29	0.1650	-0.0095	194.10	0.58	-0.0742	-0.0080	209.62	-0.56	0.1014	-0.0066	203.73	0.62
NDX	-0.1401	-0.0201	107.61	-0.67	0.1319	-0.0063	110.95	1.45	-0.0152	-0.0139	103.80	0.05	0.2495	-0.0109	96.90	1.88
SPX	-0.1317	-0.0191	104.25	-0.69	0.0987	-0.0052	105.13	1.27	-0.0347	-0.0090	100.16	-0.19	0.1062	-0.0074	94.55	1.31
SSEC	-0.2998	-0.0207	188.45	-0.98	-0.0285	-0.0073	196.85	-0.19	-0.0981	-0.0090	212.31	-0.53	-0.0203	-0.0049	183.55	-0.19
WIG20	-0.2446	-0.0174	198.44	-0.90	-0.0277	-0.0071	181.05	-0.17	-0.0294	-0.0072	178.20	-0.20	0.1065	-0.0069	178.78	0.60
CHF/GBP	0.1884	-0.0057	368.19	1.43	0.0004	-0.0027	339.29	0.01	0.0217	-0.0027	353.76	0.38	0.0195	-0.0024	368.57	0.42
CHF/PLN	0.1140	-0.0076	264.08	0.76	0.0033	-0.0034	271.69	0.14	0.0269	-0.0033	302.24	0.32	-0.0010	-0.0036	308.49	0.00
CHF/USD	0.0251	-0.0057	322.17	0.21	0.0113	-0.0036	318.03	0.11	-0.0049	-0.0025	324.93	-0.08	0.0401	-0.0034	344.86	0.50
EUR/CHF	-0.0441	-0.0041	297.69	-0.63	0.0033	-0.0017	300.01	-0.01	-0.0143	-0.0020	343.77	-0.31	-0.0229	-0.0021	340.50	-0.48
EUR/GBP	0.1314	-0.0043	342.85	1.30	0.0046	-0.0026	318.81	0.02	0.0077	-0.0025	338.80	0.18	-0.0053	-0.0018	338.13	-0.12
EUR/PLN	0.0677	-0.0062	257.95	0.50	0.0030	-0.0026	262.08	0.11	0.0113	-0.0022	303.96	0.24	-0.0266	-0.0029	278.80	-0.46
EUR/USD	-0.0150	-0.0140	290.18	-0.10	0.0145	-0.0031	299.72	0.13	-0.0186	-0.0028	315.36	-0.35	0.0110	-0.0031	335.69	0.14
GBP/PLN	-0.0401	-0.0075	247.78	-0.34	0.0003	-0.0041	253.95	0.09	0.0036	-0.0036	276.82	0.07	-0.0234	-0.0037	274.26	-0.30
USD/GBP	0.1395	-0.0048	358.53	1.28	-0.0108	-0.0028	343.02	-0.20	0.0280	-0.0032	361.62	0.45	-0.0149	-0.0033	347.19	-0.11
USD/PLN	0.0880	-0.0098	256.60	0.44	-0.0082	-0.0043	258.25	0.03	0.0347	-0.0041	296.00	0.42	-0.0289	-0.0048	294.32	-0.25
Gold Futures	0.0110	-0.0176	103.04	0.22	-0.1244	-0.0140	97.11	-0.85	-0.0096	-0.0044	69.27	-0.04	0.0488	-0.0074	83.21	0.51
Brent Crude	-0.3175	-0.0390	123.47	-1.07	0.0076	-0.0094	84.39	0.24	-0.0639	-0.0121	73.34	-0.39	0.0031	-0.0172	100.36	0.21
AAPL	-0.1495	-0.0250	148.79	-0.62	0.0537	-0.0123	131.15	0.42	-0.0268	-0.0149	107.68	-0.03	0.2291	-0.0115	92.21	1.54
AMZN	-0.0937	-0.0214	149.97	-0.41	0.1612	-0.0112	130.66	0.82	0.0882	-0.0196	107.27	0.45	0.2578	-0.0158	82.07	1.11
GOOG	-0.1939	-0.0260	142.85	-0.81	0.1667	-0.0089	120.51	1.18	-0.0184	-0.0144	99.98	-0.05	0.2195	-0.0160	84.89	1.08
NFLX	-0.0033	-0.0114	155.66	0.09	0.3704	-0.0155	132.18	1.08	0.0833	-0.0232	93.31	0.40	0.2143	-0.0187	77.84	0.86

Currency pairs (results in table 3) generally offered stable and profitable environments for the trading agent. Assets such as CHF/GBP, USD/GBP, and EUR/GBP achieved consistently strong Sharpe ratios, often exceeding 1.0, especially in early and late periods. Their daily VaR values remained low throughout, suggesting relatively limited downside volatility. This performance indicates that currencies are highly model-friendly, offering frequent trading opportunities with predictable risk-return profiles.

In contrast, goods such as Brent Crude Oil Futures and Gold Futures showed much more variable performance. Brent Crude exhibited significant drawdowns, particularly in the period I, with a Sharpe ratio of –1.07 and the highest VaR across all assets. Gold performed better in recent years, especially in period IV, where it posted a Sharpe ratio above 0.5, suggesting that commodities can offer strong returns, but only under certain market conditions. Their performance is heavily dependent on macroeconomic regimes and periods of volatility.

The stock indexes revealed a strong temporal dependency in model performance. Across most indexes, returns were negative and risk-adjusted performance was poor in the period I. However, the same indexes - such as the S&P 500, NASDAQ 100, and Dow Jones - rebounded significantly in later years, particularly period IV, where Sharpe ratios frequently exceeded 1.0, with NASDAQ reaching 1.88, the highest among all assets.

Finally, the individual stocks exhibited a wide range of behaviors. Early periods were generally unfavorable, with Amazon, Apple, and Google experiencing negative returns and low Sharpe ratios. However, by the period IV, these assets showed remarkable improvements. Apple and Netflix, in particular, reached Sharpe ratios of 1.54 and 0.86, respectively, indicating strong returns with manageable risk. These patterns reflect how high-growth tech stocks may become more learnable over time as trends solidify and volatility becomes more structured.

Overall, the findings suggest that asset class and market maturity play crucial roles in the success of reinforcement learning-based trading strategies. Currency markets emerge as consistently favorable for learning-based agents, offering smooth trends and low-risk trades. Indexes and tech stocks, while volatile, yield higher returns and stronger Sharpe ratios in more recent years. Commodities remain challenging, offering potential upside only in select conditions. These distinctions underscore the need to adapt model expectations and reward structures to the characteristics of each market.

Table 4: Comparison of performance between single-period and expanding-window training across models, asset classes, and reward functions.

Model / Group	**Single-Period Training**				**Expanding-Window Training**			
	Ann. Return	**Daily VaR**	**No Trades**	**Sharpe**	**Ann. Return**	**Daily VaR**	**No Trades**	**Sharpe**
A2C	0.0405	-0.0080	217.81	0.2674	0.0408	-0.0084	231.99	0.2663
Currencies	0.0024	-0.0035	343.56	0.0305	0.0021	-0.0036	350.77	0.0240
extreme loss-averse	0.0019	-0.0029	376.27	0.0312	0.0019	-0.0031	382.50	0.0133
loss-averse	0.0022	-0.0034	394.10	0.0294	0.0022	-0.0034	398.39	0.0251
prospect theory	0.0023	-0.0031	381.57	0.0295	0.0019	-0.0032	388.65	0.0224
risk-averse	0.0012	-0.0022	359.37	0.0255	0.0011	-0.0023	371.65	0.0214
risk-loving	0.0044	-0.0059	150.57	0.0429	0.0034	-0.0057	160.58	0.0395
benchmark	0.0022	-0.0035	399.47	0.0243	0.0023	-0.0037	402.84	0.0220
Goods	-0.0251	-0.0116	73.47	-0.0526	-0.0313	-0.0132	98.60	-0.1036
extreme loss-averse	-0.0018	-0.0040	46.35	0.0514	-0.0089	-0.0068	86.29	-0.0448
loss-averse	-0.0216	-0.0089	91.41	-0.0116	-0.0222	-0.0105	117.70	-0.0546
prospect theory	-0.0038	-0.0046	52.51	0.0170	-0.0144	-0.0079	87.80	-0.0520
risk-averse	-0.0131	-0.0097	149.21	-0.0408	-0.0143	-0.0098	144.93	-0.0728
risk-loving	-0.0693	-0.0238	10.29	-0.1773	-0.0685	-0.0232	13.95	-0.1937
benchmark	-0.0414	-0.0187	91.09	-0.1542	-0.0598	-0.0209	140.93	-0.2036
Indexes	0.0398	-0.0082	168.02	0.3555	0.0373	-0.0082	185.92	0.3427
extreme loss-averse	0.0157	-0.0042	137.08	0.2470	0.0145	-0.0052	172.01	0.2263
loss-averse	0.0441	-0.0088	220.64	0.3775	0.0363	-0.0081	233.49	0.3565
prospect theory	0.0156	-0.0043	140.50	0.2543	0.0123	-0.0049	177.56	0.2463
risk-averse	0.0125	-0.0046	231.78	0.2813	0.0111	-0.0047	235.56	0.2628
risk-loving	0.0826	-0.0149	13.71	0.4831	0.0797	-0.0147	20.31	0.4644
benchmark	0.0683	-0.0124	264.44	0.4896	0.0700	-0.0119	276.60	0.5001
Stocks	0.1704	-0.0171	100.08	0.7995	0.1824	-0.0185	116.91	0.8662
extreme loss-averse	0.0242	-0.0049	69.47	0.4348	0.0649	-0.0085	100.05	0.6211
loss-averse	0.0710	-0.0097	116.14	0.6596	0.0980	-0.0124	131.16	0.7415
prospect theory	0.0304	-0.0062	84.86	0.4663	0.0768	-0.0101	114.19	0.6717
risk-averse	0.0888	-0.0108	161.70	0.8082	0.0828	-0.0122	158.09	0.7980
risk-loving	0.4165	-0.0361	45.16	1.2197	0.4012	-0.0345	90.01	1.1761
benchmark	0.3912	-0.0347	123.18	1.2086	0.3707	-0.0335	107.98	1.1885
PPO	0.0288	-0.0061	197.55	0.2277	0.0291	-0.0063	210.05	0.2259
Currencies	0.0020	-0.0026	283.97	0.0414	0.0022	-0.0027	296.42	0.0311
extreme loss-averse	0.0009	-0.0012	232.33	0.0291	0.0015	-0.0014	249.13	0.0398
loss-averse	0.0020	-0.0018	285.24	0.0719	0.0020	-0.0020	298.52	0.0061
prospect theory	0.0011	-0.0013	242.63	0.0166	0.0017	-0.0016	257.09	0.0398
risk-averse	0.0009	-0.0011	381.08	0.0519	0.0008	-0.0013	389.02	0.0318
risk-loving	0.0040	-0.0061	190.90	0.0439	0.0042	-0.0060	205.01	0.0457
Goods	-0.0210	-0.0099	95.75	-0.0514	-0.0243	-0.0104	101.82	-0.0756
extreme loss-averse	-0.0041	-0.0031	51.56	0.0604	-0.0035	-0.0036	60.01	0.0596
loss-averse	-0.0062	-0.0040	63.95	0.0133	-0.0073	-0.0049	76.55	-0.0248
prospect theory	-0.0043	-0.0031	51.34	0.0497	-0.0048	-0.0035	58.28	0.0332
risk-averse	-0.0101	-0.0076	160.64	-0.1438	-0.0124	-0.0075	157.69	-0.1589
risk-loving	-0.0703	-0.0243	65.91	-0.2069	-0.0716	-0.0243	70.07	-0.2150
benchmark	-0.0310	-0.0172	181.10	-0.0812	-0.0464	-0.0185	188.30	-0.1476
Indexes	0.0253	-0.0061	166.56	0.2900	0.0269	-0.0064	179.85	0.2941
extreme loss-averse	0.0032	-0.0019	98.51	0.1577	0.0080	-0.0032	142.88	0.1897
loss-averse	0.0063	-0.0028	132.88	0.1981	0.0105	-0.0037	160.17	0.2058
prospect theory	0.0038	-0.0019	97.76	0.1782	0.0065	-0.0025	118.58	0.1908
risk-averse	0.0073	-0.0042	262.51	0.2576	0.0066	-0.0041	254.74	0.2527
risk-loving	0.0800	-0.0146	99.46	0.4902	0.0798	-0.0146	103.49	0.4908
benchmark	0.0514	-0.0111	308.26	0.4581	0.0500	-0.0102	299.24	0.4346
Stocks	0.1295	-0.0133	109.88	0.6777	0.1289	-0.0132	123.73	0.6934
extreme loss-averse	0.0189	-0.0029	53.33	0.2699	0.0234	-0.0036	58.68	0.2950
loss-averse	0.0250	-0.0040	70.77	0.4036	0.0410	-0.0054	80.41	0.4713
prospect theory	0.0187	-0.0029	53.59	0.2879	0.0217	-0.0034	57.26	0.3044
risk-averse	0.0386	-0.0062	154.49	0.6402	0.0375	-0.0065	153.75	0.6307
risk-loving	0.3942	-0.0351	122.10	1.2617	0.3587	-0.0320	184.68	1.2525
benchmark	0.2814	-0.0286	204.98	1.2028	0.2911	-0.0282	207.63	1.2068
Grand Total	0.0347	-0.0071	207.68	0.2476	0.0350	-0.0074	221.02	0.2461

In table 4 we present the overconfidence bias imitation results, comparing single- to multi-period training. Overall, performance is relatively stable between the two approaches, with only slight differences in annual return, risk exposure, and Sharpe ratio.

At a high level, A2C and PPO models show minimal differences in average outcomes. For instance, A2C's overall Sharpe ratio decreases slightly from 0.2674 (single) to 0.2663 (multi), while PPO's performance follows a similar trend. This suggests that expanding the training window does not necessarily improve model robustness or risk-adjusted returns, at least when evaluated over similar validation periods.

When broken down by asset class, patterns vary. Stocks benefit modestly from multi-period training, showing improved average Sharpe ratios and slightly higher trade counts. In contrast, currency pairs and goods exhibit slightly worse Sharpe ratios under multi-training, indicating potential overfitting or diminishing returns from added historical data in these markets. These effects may stem from higher stationarity in currency markets, where recent data is more informative than long-term history.

Across reward functions, the differences are again small but reveal tendencies in model behavior. Risk-neutral and risk-loving agents, which prioritize return over risk, tend to show slightly lower Sharpe ratios in the multi-training setup, possibly due to overconfidence from past market trends being incorporated into the policy. On the other hand, risk-averse agents (e.g., loss-averse, prospect theory) demonstrate more stable or even slightly improved risk-adjusted performance in some asset groups, suggesting that multi-period training may help them generalize more cautious policies.

An interesting observation arises in the stock index and individual stock categories, where multi-period training leads to more trades on average, especially for risk-averse agents. This implies that broader historical exposure may help these models identify more nuanced price movements, though not always translating into better Sharpe ratios.

In summary, while expanding the training window introduces a richer learning context, and should lower overconfidence, its impact on performance is subtle and highly dependent on the asset class and reward structure. In stable or stationary markets like currencies, single-period training may be sufficient. For more complex or volatile assets like stocks, a broader training window can offer slight advantages, particularly for more cautious strategies. However, the overall message is that more data does not guarantee better results, and tailoring training schemes to market characteristics remains essential.

6. Summary

This study investigated how behavioral biases can be explicitly embedded into RL agents operating in financial markets. While traditional RL relies on reward functions aligned with objective task performance, real-world investors operate under bounded rationality and cognitive biases - such as loss aversion, risk aversion, and overconfidence. Building on behavioral economics and psychology, we design RL agents whose reward functions incorporate these biases, aiming to replicate diverse, human-like trading styles. Rather than treating biases as irrational noise, we leverage them as mechanisms to guide agent learning, better modeling the heterogeneity observed among real-world investors and human-driven markets.

To test these ideas, we implement a systematic experimental framework using deep RL agents (A2C and PPO) trained with six distinct reward functions: risk-loving, risk-averse, loss-averse, extreme loss-averse, Prospect Theory-based, and an unbiased benchmark. We also explore overconfidence by manipulating training regimes - varying the number of training steps and using both fixed-period and expanding-window training. Experiments are conducted across four historical market periods (2004–2022), using a broad dataset of indexes, stocks, currencies, and commodities. The agents' performance is evaluated using standard financial metrics (annual return, Sharpe ratio, number of trades, Value-at-Risk), providing a comprehensive basis for analyzing how different bias-driven reward structures and training exposures affect learning, generalization, and trading behavior.

Building on behavioral and economic theories, we hypothesized that the choice and design of reward functions would substantially influence the performance and behavior of RL agents in financial markets. Specifically, we expected that parametrizing reward functions to reflect varying risk preferences and loss aversion would induce systematically distinct trading styles and risk profiles, from highly conservative to risk-seeking. Additionally, we hypothesized that overconfidence-like behaviors would emerge via overfitting when agents are exposed to extended training horizons or larger datasets, and that volatility-aware reward functions would enhance out-of-sample robustness, particularly under turbulent conditions.

Our results confirm that the choice of reward function dramatically shaped agent behavior and outcomes. Risk-loving and benchmark agents achieved the highest returns, but exhibited high risk and volatility (high daily VaR, frequent large drawdowns in 2004–2007). Conversely, extreme loss-averse and Prospect Theory-based agents consistently protected against downside risk but failed to capture large market gains, resulting in low Sharpe ratios and limited profitability. Balanced strategies such as loss-averse and risk-averse agents delivered moderate, stable returns and higher trade frequencies, confirming the capacity of parametrized reward functions to induce heterogeneous, human-like behaviors, consistent with our expectations.

The A2C model benefited from extended training, reaching its best Sharpe ratio (0.505) and annual return (7.98%) in 2019–2022 after 250k steps. PPO converged more quickly, achieving competitive risk-adjusted returns at shorter horizons, while maintaining lower VaR and more selective trading. This suggests that PPO naturally develops more risk-aware policies with limited training, whereas A2C relies on prolonged exploration to optimize more aggressive strategies.

In terms of the asset class, currency pairs consistently provided favorable environments for learning-based agents, with strong and stable Sharpe ratios and low downside risk, validating the hypothesis that RL agents can model stable and predictable markets effectively. In contrast, commodities exhibited volatile, regime-dependent performance; stock indexes and tech equities became highly learnable past 2008 crisis, particularly in the 2019–2022 period where NASDAQ-100 reached a Sharpe ratio of 1.88.

Comparing single-period and expanding-window training revealed that while overall performance remained similar, overfitting tendencies were observed in certain contexts. For instance, currency agents and goods traders slightly underperformed under multi-period training - suggesting diminished adaptability when exposed to extended histories, rejecting our overconfidence hypothesis. Stocks benefited modestly from multi-period training, indicating that complex, non-stationary assets may benefit from richer historical contexts, while stable markets favor more recent data. Risk-neutral and risk-loving agents exhibited slight performance degradation under multi-period training, supporting the view that reward functions encoding volatility awareness can mitigate overfitting risks.

Agents with volatility-aware components in their reward functions demonstrated more robust generalization across market regimes, particularly in stocks and commodities. These agents outperformed purely return-seeking strategies in out-of-sample evaluations, validating the hypothesis that volatility-sensitive objectives improve robustness in turbulent environments.

In conclusion, the results strongly confirm that reward function design affects both trading behavior and performance, enabling RL agents to mimic a wide range of investor archetypes. Moreover, overconfidence and overfitting risks are real and measurable, particularly under extended training horizons and with complex reward objectives. These insights extend prior work by demonstrating that deep RL agents - when equipped with well-designed reward functions - can model the behavioral diversity and adaptive challenges faced by real-world investors across heterogeneous markets and dynamic regimes.

References

Abbeel, P., & Ng, A. Y. Apprenticeship learning via inverse reinforcement learning. In: In *Proceedings of the twenty-first international conference on Machine learning*. ICML '04. New York, NY, USA: Association for Computing Machinery, 2004, July 4, 1. https://doi.org/10.1145/1015330.1015430

Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., & Mané, D. (2016, July 25). *Concrete Problems in AI Safety*. arXiv: 1606.06565 [cs]. https://doi.org/10.48550/arXiv.1606.06565

Avellaneda, M., & Stoikov, S. (2008). High-frequency trading in a limit order book. *Quantitative Finance*, *8*(3), 217–224. https://doi.org/10.1080/14697680701381228

Banerjee, A. V. (1992). A Simple Model of Herd Behavior*. *The Quarterly Journal of Economics*, *107*(3), 797–817. https://doi.org/10.2307/2118364

Barberis, N., & Thaler, R. (2002, September). *A Survey of Behavioral Finance*. 9222. https://doi.org/10.3386/w9222

Barnes, M., Abueg, M., Lange, O. F., Deeds, M., Trader, J., Molitor, D., Wulfmeier, M., & O'Banion, S. (2024, March 5). *Massively Scalable Inverse Reinforcement Learning in Google Maps*. arXiv: 2305.11290 [cs]. https://doi.org/10.48550/arXiv.2305.11290

Borkar, V. S., & Chandak, S. (2021). Prospect-theoretic Q-learning. *Systems & Control Letters*, *156*, 105009. https://doi.org/10.1016/j.sysconle.2021.105009

Burda, Y., Edwards, H., Pathak, D., Storkey, A., Darrell, T., & Efros, A. A. (2018, August 13). *Large-Scale Study of Curiosity-Driven Learning*. arXiv: 1808.04355 [cs]. https://doi.org/10.48550/arXiv.1808.04355

Camacho, A., Toro Icarte, R., Klassen, T. Q., Valenzano, R., & McIlraith, S. A. LTL and Beyond: Formal Languages for Reward Function Specification in Reinforcement Learning. In: In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*. Twenty-Eighth International Joint Conference on Artificial Intelligence {IJCAI-19}. Macao, China: International Joint Conferences on Artificial Intelligence Organization, 2019, August, 6065–6073. https://doi.org/10.24963/ijcai.2019/840

Cao, G., Zhang, Y., Lou, Q., & Wang, G. (2024:). Optimization of High-Frequency Trading Strategies Using Deep Reinforcement Learning. *Journal of Artificial Intelligence General science (JAIGS) ISSN:3006-4023*, *6*(1), 230–257. Retrieved March 30, 2025, from https://ideas.repec.org//a/das/njaigs/v6y2024i1p230-257id247.html

Carver, C. S., & White, T. L. (1994). Behavioral inhibition, behavioral activation, and affective responses to impending reward and punishment: The BIS/BAS Scales. *Journal of Personality and Social Psychology*, *67*(2), 319–333. https://doi.org/10.1037/0022-3514.67.2.319

Chan, N. T., & Shelton, C. (2001). An Electronic Market-Maker. Retrieved March 27, 2025, from https://dspace.mit.edu/handle/1721.1/7220 Accepted: 2004-10-20T20:50:09Z.

Chow, Y., Tamar, A., Mannor, S., & Pavone, M. (2015, June 6). *Risk-Sensitive and Robust Decision-Making: A CVaR Optimization Approach*. arXiv: 1506.02188 [cs]. https://doi.org/10.48550/arXiv.1506.02188

Corr, P. J., & Cooper, A. J. (2016). The Reinforcement Sensitivity Theory of Personality Questionnaire (RST-PQ): Development and validation. *Psychological Assessment*, *28*(11), 1427–1440. https://doi.org/10.1037/pas0000273

Deng, Y., Bao, F., Kong, Y., Ren, Z., & Dai, Q. (2017). Deep Direct Reinforcement Learning for Financial Signal Representation and Trading. *IEEE Transactions on Neural Networks and Learning Systems*, *28*(3), 653–664. https://doi.org/10.1109/TNNLS.2016.2522401

Eriksson, H., & Dimitrakakis, C. (2019, June 14). *Epistemic Risk-Sensitive Reinforcement Learning*. arXiv: 1906.06273 [cs]. https://doi.org/10.48550/arXiv.1906.06273

Fernando, C., Banarse, D., Blundell, C., Zwols, Y., Ha, D., Rusu, A. A., Pritzel, A., & Wierstra, D. (2017, January 30). *PathNet: Evolution Channels Gradient Descent in Super Neural Networks*. arXiv: 1701.08734 [cs]. https://doi.org/10.48550/arXiv.1701.08734

Fischhoff, B., Slovic, P., & Lichtenstein, S. (1977). Knowing with certainty: The appropriateness of extreme confidence. *Journal of Experimental Psychology: Human Perception and Performance*, *3*(4), 552–564. https://doi.org/10.1037/0096-1523.3.4.552

Gigerenzer, G., & Brighton, H. (2009). Homo heuristicus: Why biased minds make better inferences. *Topics in Cognitive Science*, *1*(1), 107–143. https://doi.org/10.1111/j.1756-8765.2008.01006.x

Goluža, S., Kovačević, T., Begušić, S., & Kostanjčar, Z. (2024, November 13). *Robot See, Robot Do: Imitation Reward for Noisy Financial Environments*. arXiv: 2411.08637 [cs]. https://doi.org/10.48550/arXiv.2411.08637

Gray, J. A., & McNaughton, N. (2007). *The Neuropsychology of Anxiety: An Enquiry into the Functions of the Septo-Hippocampal System*. Oxford University Press, USA.

Hayes, W. M., Yax, N., & Palminteri, S. (2024, May 19). *Large Language Models are Biased Reinforcement Learners*. arXiv: 2405.11422 [cs]. https://doi.org/10.48550/arXiv.2405.11422

Icarte, R. T., Klassen, T. Q., Valenzano, R., & McIlraith, S. A. (2022). Reward Machines: Exploiting Reward Function Structure in Reinforcement Learning. *Journal of Artificial Intelligence Research*, *73*, 173–208. https://doi.org/10.1613/jair.1.12440

Jara-Ettinger, J., Gweon, H., Schulz, L. E., & Tenenbaum, J. B. (2016). The Naïve Utility Calculus: Computational Principles Underlying Commonsense Psychology. *Trends in Cognitive Sciences*, *20*(8), 589–604. https://doi.org/10.1016/j.tics.2016.05.011

Jiang, Z., Xu, D., & Liang, J. (2017, July 16). *A Deep Reinforcement Learning Framework for the Financial Portfolio Management Problem*. arXiv: 1706.10059 [q-fin]. https://doi.org/10.48550/arXiv.1706.10059

Kahneman, D., & Tversky, A. (1979). Prospect Theory: An Analysis of Decision under Risk. *Econometrica*, *47*(2), 263–291. https://doi.org/10.2307/1914185

Kim, D.-Y., & Lee, J.-H. (2011). Effects of the BAS and BIS on decision-making in a gambling task. *Personality and Individual Differences*, *50*(7), 1131–1135. https://doi.org/10.1016/j.paid.2011.01.041

Kopczewski, T., & Bil, (2024). Exploring stock markets dynamics: A two-dimensional entropy approach in return/volume space. *Bank i Kredyt*, *Vol. 55*, 731–758. https://doi.org/10.5604/01.3001.0054.9083

Krupić, D. (2017). HIGH BAS AND LOW BIS IN OVERCONFIDENCE, AND THEIR IMPACT ON MOTIVATION AND SELF-EFFICACY AFTER POSITIVE AND NEGATIVE PERFORMANCE. *Primenjena psihologija*, *10*(3), 297–312. Retrieved June 7, 2025, from https://www.ceeol.com/search/article-detail?id=589086

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., & Gershman, S. J. (2016, November 2). *Building Machines That Learn and Think Like People*. arXiv: 1604.00289 [cs]. https://doi.org/10.48550/arXiv.1604.00289

Leibo, J. Z., Zambaldi, V., Lanctot, M., Marecki, J., & Graepel, T. (2017, February 10). *Multi-agent Reinforcement Learning in Sequential Social Dilemmas*. arXiv: 1702.03037 [cs]. https://doi.org/10.48550/arXiv.1702.03037

Leng, J., Huang, C., Zhu, B., & Huang, J. (2025, February 28). *Taming Overconfidence in LLMs: Reward Calibration in RLHF*. arXiv: 2410.09724 [cs]. https://doi.org/10.48550/arXiv.2410.09724

Li, Z., Ji, X., Chen, M., & Wang, M. Policy Evaluation for Reinforcement Learning from Human Feedback: A Sample Complexity Analysis. In: In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*. International Conference on Artificial Intelligence and Statistics. PMLR, 2024, April 18, 2737–2745. Retrieved March 31, 2025, from https://proceedings.mlr.press/v238/li24l.html

Ma, S., & Yu, J. Y. (2018, November 29). *Transition-based versus State-based Reward Functions for MDPs with Value-at-Risk*. arXiv: 1612.02088 [cs]. https://doi.org/10.48550/arXiv.1612.02088

Mehra, R., & Prescott, E. C. (1985). The equity premium: A puzzle. *Journal of Monetary Economics*, *15*(2), 145–161. https://doi.org/10.1016/0304-3932(85)90061-3

Mihatsch, O., & Neuneier, R. (2002). Risk-Sensitive Reinforcement Learning. *Machine Learning*, *49*(2/3), 267–290. https://doi.org/10.1023/A:1017940631555

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T. P., Harley, T., Silver, D., & Kavukcuoglu, K. (2016, June 16). *Asynchronous Methods for Deep Reinforcement Learning*. arXiv: 1602.01783 [cs]. https://doi.org/10.48550/arXiv.1602.01783

Moody, J., & Saffell, M. (2001). Learning to trade via direct reinforcement. *IEEE Transactions on Neural Networks*, *12*(4), 875–889. https://doi.org/10.1109/72.935097

Nevmyvaka, Y., Feng, Y., & Kearns, M. Reinforcement learning for optimized trade execution. In: In *Proceedings of the 23rd international conference on Machine learning - ICML '06*. The 23rd International Conference. Pittsburgh, Pennsylvania: ACM Press, 2006, 673–680. https://doi.org/10.1145/1143844.1143929

Ng, A. Y., Harada, D., & Russell, S. J. Policy Invariance Under Reward Transformations: Theory and Application to Reward Shaping.

In: In *Proceedings of the Sixteenth International Conference on Machine Learning*. ICML '99. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1999, June 27, 278–287.

Ng, A. Y., & Russell, S. J. Algorithms for Inverse Reinforcement Learning. In: In *Proceedings of the Seventeenth International Conference on Machine Learning*. ICML '00. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2000, June 29, 663–670.

Ni, X., Liu, G., & Lai, L. Risk-Sensitive Reward-Free Reinforcement Learning with CVaR. In: In *Proceedings of the 41st International Conference on Machine Learning*. International Conference on Machine Learning. PMLR, 2024, July 8, 37999–38017. Retrieved March 31, 2025, from https://proceedings.mlr.press/v235/ni24c.html

Odean, T. (1998). Volume, Volatility, Price, and Profit When All Traders Are Above Average. *The Journal of Finance*, *53*(6), 1887–1934. https://doi.org/10.1111/0022-1082.00078

O'Donoghue, T., & Rabin, M. (1999). Doing It Now or Later. *The American Economic Review*, *89*(1), 103–124. Retrieved March 26, 2025, from https://www.jstor.org/stable/116981

Otabek, S., & Choi, J. (2024). Multi-level deep Q-networks for Bitcoin trading strategies. *Scientific Reports*, *14*(1), 771. https://doi.org/10.1038/s41598-024-51408-w

Peschl, M., Zgonnikov, A., Oliehoek, F. A., & Siebert, L. C. (2021, December 30). *MORAL: Aligning AI with Human Norms through Multi-Objective Reinforced Active Learning*. arXiv: 2201.00012 [cs]. https://doi.org/10.48550/arXiv.2201.00012

Peterson, R. L. (2007). Affect and Financial Decision-Making: How Neuroscience Can Inform Market Participants. *Journal of Behavioral Finance*, *8*(2), 70–78. https://doi.org/10.1080/15427560701377448

Prashanth, L.A., Jie, C., Fu, M., Marcus, S., & Szepesvári, C. (2016, February 26). *Cumulative Prospect Theory Meets Reinforcement Learning: Prediction and Control*. arXiv: 1506.02632 [cs]. https://doi.org/10.48550/arXiv.1506.02632

Pratt, J. W. (1964). Risk Aversion in the Small and in the Large. *Econometrica*, *32*(1/2), 122–136. https://doi.org/10.2307/1913738

Ramasubramanian, B., Niu, L., Clark, A., & Poovendran, R. Reinforcement Learning Beyond Expectation. In: In *2021 60th IEEE Conference on Decision and Control (CDC)*. 2021 60th IEEE Conference on Decision and Control (CDC). 2021, December, 1528–1535. https://doi.org/10.1109/CDC45484.2021.9683261

Rodinos, G., Nousi, P., Passalis, N., & Tefas, A. A Sharpe Ratio Based Reward Scheme in Deep Reinforcement Learning for Financial Trading (I. Maglogiannis, L. Iliadis, J. MacIntyre, & M. Dominguez, Eds.). In: *Artificial Intelligence Applications and Innovations* (I. Maglogiannis, L. Iliadis, J. MacIntyre, & M. Dominguez, Eds.). Ed. by Maglogiannis, I., Iliadis, L., MacIntyre, J., & Dominguez, M. Cham: Springer Nature Switzerland, 2023, 15–23. https://doi.org/10.1007/978-3-031-34111-3_2

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017, August 28). *Proximal Policy Optimization Algorithms*. arXiv: 1707.06347 [cs]. https://doi.org/10.48550/arXiv.1707.06347

Schultz, W., Dayan, P., & Montague, P. R. (1997). A neural substrate of prediction and reward. *Science (New York, N.Y.)*, *275*(5306), 1593–1599. https://doi.org/10.1126/science.275.5306.1593

Sutton, R. S., & Barto, A. G. (2020). *Reinforcement Learning, second edition: An Introduction*. Bradford Books.

Théate, T., & Ernst, D. (2021). An application of deep reinforcement learning to algorithmic trading. *Expert Systems with Applications*, *173*, 114632. https://doi.org/10.1016/j.eswa.2021.114632

Tversky, A., & Kahneman, D. (1974). http://www.jstor.org Judgment under Uncertainty: Heuristics and Biases. *Science, New Series*, *185*, 1124–1131. http://www.jstor.org/stable/1738360

Vecerik, M., Hester, T., Scholz, J., Wang, F., Pietquin, O., Piot, B., Heess, N., Rothörl, T., Lampe, T., & Riedmiller, M. (2018, October 8). *Leveraging Demonstrations for Deep Reinforcement Learning on Robotics Problems with Sparse Rewards*. arXiv: 1707.08817 [cs]. https://doi.org/10.48550/arXiv.1707.08817

Vermeersch, H., T'Sjoen, G., Kaufman, J.-M., & Van Houtte, M. (2013). Social Science Theories on Adolescent Risk-Taking: The Relevance of Behavioral Inhibition and Activation. *Youth & Society*, *45*(1), 27–53. https://doi.org/10.1177/0044118X11409014

Visser, T. A. W., Bender, A. D., Bowden, V. K., Black, S. C., Greenwell-Barnden, J., Loft, S., & Lipp, O. V. (2019). Individual differences in higher-level cognitive abilities do not predict overconfidence in complex task performance. *Consciousness and Cognition*, *74*, 102777. https://doi.org/10.1016/j.concog.2019.102777

Von Neumann, J., & Morgenstern, O. (1944). *Theory of games and economic behavior*. Princeton University Press.

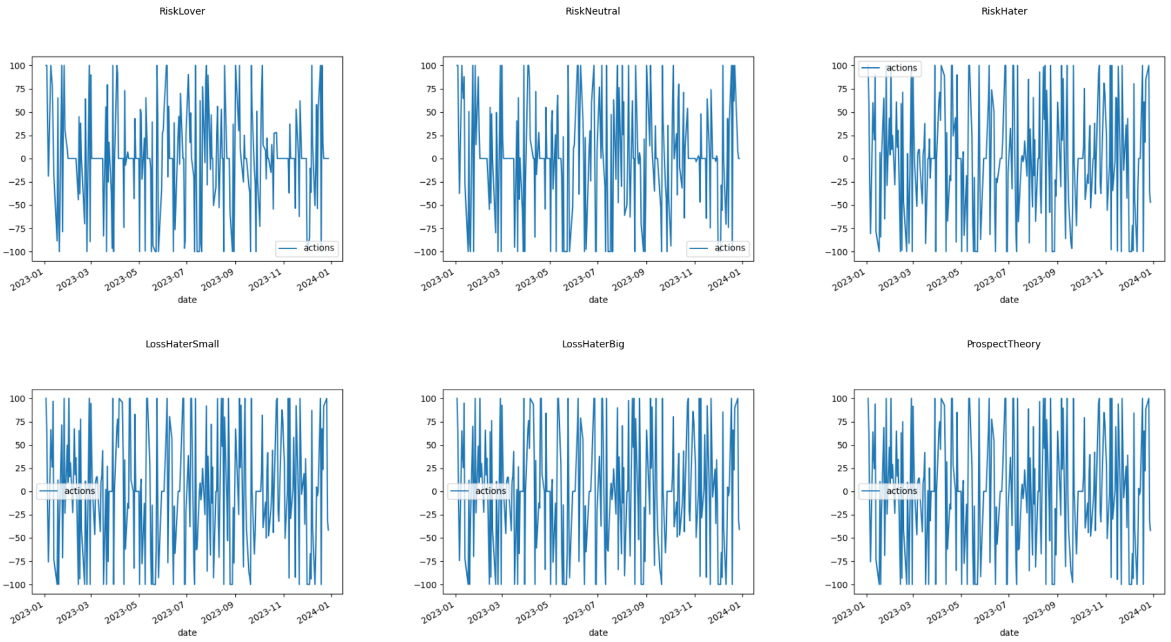


Figure 2: Actions taken by an agent while trading S&P 500 in 2023 testing sample for PPO model after training model for 5000 timesteps for all tested reward functions (one of the model iterations

Wang, J. X., Kurth-Nelson, Z., Kumaran, D., Tirumala, D., Soyer, H., Leibo, J. Z., Hassabis, D., & Botvinick, M. (2018). Prefrontal cortex as a meta-reinforcement learning system. *Nature Neuroscience*, *21*(6), 860–868. https://doi.org/10.1038/s41593-018-0147-8

Yang, H., Wang, Y., Xu, X., Zhang, H., & Bian, Y. (2024, May 27). *Can We Trust LLMs? Mitigate Overconfidence Bias in LLMs through Knowledge Transfer*. arXiv: 2405.16856 [cs]. https://doi.org/10.48550/arXiv.2405.16856

Ying, C., Zhou, X., Su, H., Yan, D., Chen, N., & Zhu, J. (2022, September 17). *Towards Safe Reinforcement Learning via Constraining Conditional Value-at-Risk*. arXiv: 2206.04436 [cs]. https://doi.org/10.48550/arXiv.2206.04436

7. Attachments

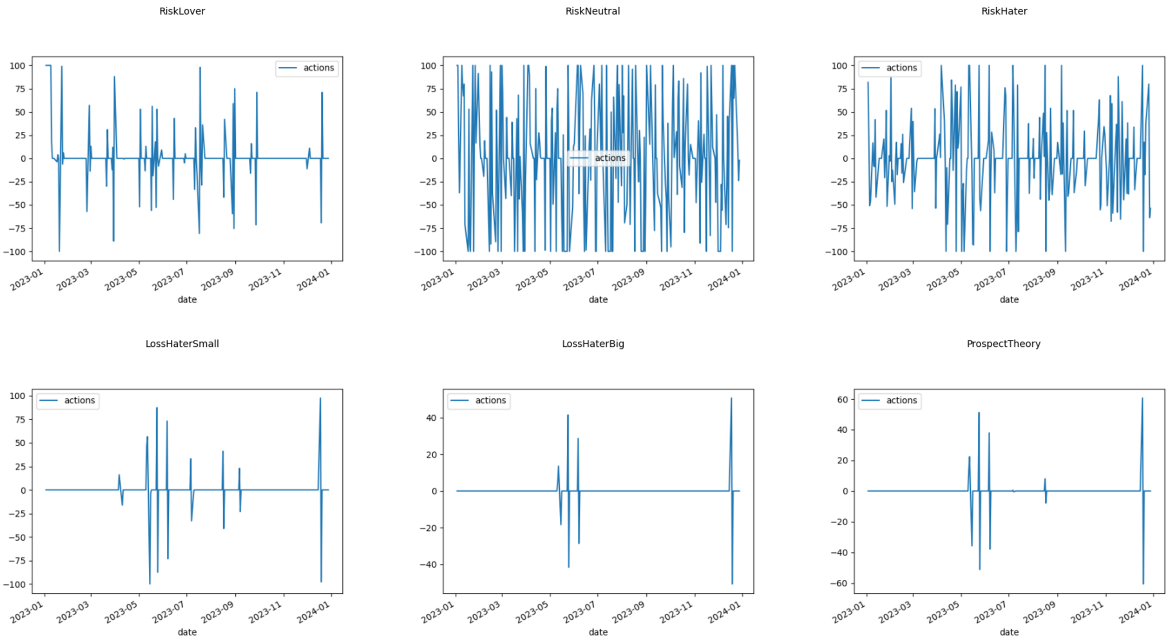


Figure 3: Actions taken by an agent while trading S&P 500 in 2023 testing sample for PPO model after training model for 75000 timesteps for all tested reward functions (one of the model iterations

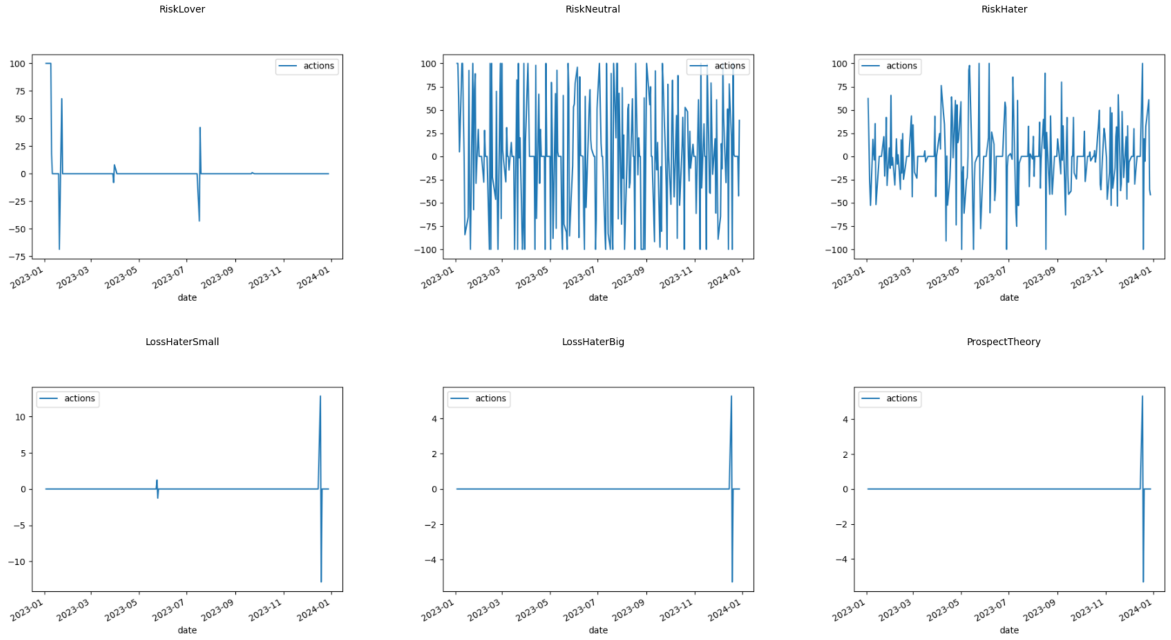


Figure 4: Actions taken by an agent while trading S&P 500 in 2023 testing sample for PPO model after training model for 250000 timesteps for all tested reward functions (one of the model iterations