Algorithmic Trading with Reinforcement Learning

First semester report

Leonardo Toffalini

Supervised by András Lukács

1 Introduction

During this semester we continued where we left off after my undergraduate thesis work. Given the length constraints of the present report, we only focus on defining the problem at hand and briefly mention some notable achievements.

Consider a modeled financial market wherein the price of a risky asset adheres to an adapted process S_t, where $t \in [0, T]$. The trader may trade at finite rates on the risky asset, though they incur a temporary nonlinear price impact as a consequence. The family of feasible strategies available to the trader is

$$S(T) \coloneqq \left\{ \phi : \phi \text{ is a } \mathbb{R}\text{-valued optional process and} \int_0^T |\phi| \, \mathrm{d}u < \infty \text{ a.s.} \right\}.$$

The trader's initial asset position is represented by $z = (z^0, z^1)$, where z^0 is the number of units of the riskless asset, and z^1 represents the number of units of risky assets.

The number of units of the risky asset at time $t \in [0, T]$, after following the strategy ϕ is given by

$$X_t^1(\phi) \coloneqq z^1 + \int_0^t \phi_u \, \mathrm{d}u.$$

The aggregate position in the riskless asset is defined in a comparable manner, albeit incorporating the effect of price impact. The trader incurs a superlinear penalty associated with the trading speed, as determined by parameters $\alpha > 1$ and $\lambda > 0$. The aggregate position in the riskless asset at time $t \in [0, T]$ is given by

$$X_t^0(\phi) \coloneqq z^0 - \int_0^t \phi_u S_u \, \mathrm{d}u - \int_0^t \lambda |\phi_u|^\alpha \, \mathrm{d}u.$$

Let $\mathcal{A}(T)$ be the family of feasible strategies starting with zero initial capital, and with the final position composed exclusively of the riskless asset, and a well-defined notion of expected terminal riskless asset position, that is

$$\mathcal{A}(T) \coloneqq \{ \phi \in S(T) : X_T^1 = 0, \quad \mathbb{E}[X_T^0(\phi)_-] < \infty \},$$

where $x_- = -\min(x, 0)$.

The objective of our problem is to identify the strategy $\phi \in \mathcal{A}(T)$ that realizes maximal expected profits of the riskless asset.

It can be shown that there exists an optimal strategy for any time horizon T that achieves maximal returns [1]. It can also be shown that a simple contrarian strategy in the anti persistent case, and a momentum strategy in the persistent case with linear liquidation after $T/2$ steps achieves asymptotically optimal returns, that is of order $T^{H(1+\kappa)+1}$ [1], when $\kappa \to 1/(\alpha - 1)$.

The goal of this project, and that of my undergraduate thesis, is to compete with the analytical results by learning a strategy that compares in its expected returns. We will learn such a strategy using reinforcement learning (RL).

2 Previous work

During my undergraduate thesis, we showed that with a standard PPO [2] algorithm we were able to outcompete the analytical strategy on expected returns for time horizons $T \leq 512$ by training a bespoke model for each tested time horizon, which is to say that we did not find a general strategy that worked for any time horizon. For time horizons greater than 512 the learned strategies were not able to outperform the simple analytical strategy on expected returns, which we attributed to the credit assignment problem becoming increasingly more difficult for longer horizons.

3 Current work

Building on the previous work, the overall reinforcement learning approach remains unchanged and continues to rely on PPO as the core algorithm.

The following modifications were introduced:

- The environment was reimplemented in C using the pufferlib framework [3], yielding an approximate three orders of magnitude increase in simulation speed, from about 1.5k steps per second (SPS) to roughly 1.5M SPS.
- The improved simulation efficiency made it feasible to perform large-scale hyperparameter search using a modified variant of CARBS [4].
- The liquidation mechanism was redesigned, replacing single-step forced liquidation with a user-configurable linear liquidation schedule.
- Thanks to the increased simulation speed, the agent can now evaluate a small set of short, plausible future scenarios to assess its performance under forced liquidation within a given time horizon for each step.
- The reward function was reformulated to depend on the anticipated liquidation cost rather than on the temporal difference in the riskless asset.

Bibliography

[1] P. Guasoni, Z. Nika, and M. Rásonyi, “Trading fractional Brownian motion,” SIAM journal on financial mathematics, vol. 10, no. 3, pp. 769–789, 2019.

[2] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” arXiv preprint arXiv:1707.06347, 2017.

[3] J. Suarez, “PufferLib 2.0: Reinforcement Learning at 1M steps/s,” Reinforcement Learning Journal, vol. 6, pp. 1378–1388, 2025.

[4] A. J. Fetterman et al., “Tune As You Scale: Hyperparameter Optimization For Compute Efficient Training,” arXiv e-prints, 2023.