Hedging an Options Book with Reinforcement Learning

Petter Kolm
Courant Institute, NYU

petter.kolm@nyu.edu
https://www.linkedin.com/in/petterkolm

Frontiers in Quantitative Finance Seminar
University of Oxford
April 15, 2021

Our articles related to this talk

- Kolm and Ritter (2019), "Dynamic Replication and Hedging: A Reinforcement Learning Approach," *Journal of Financial Data Science*, 1 (1), 2019
- Kolm and Ritter (2020), "Modern Perspectives on Reinforcement Learning in Finance," *Journal of Machine Learning in Finance*, 1 (1), 2020. Also available here: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3449401
- Du, Jin, Kolm, Ritter, Wang, and Zhang (2020), "Deep Reinforcement Learning for Option Replication and Hedging," *Journal of Financial Data Science*, 2 (4), 2020

Background & motivation

Replication & hedging

- Replicating and hedging an option position is fundamental in finance
- The core idea of the seminal work by Black-Scholes-Merton (BSM):
  - In a complete and frictionless market there is a continuously rebalanced dynamic trading strategy in the stock and riskless security that perfectly replicates the option (Black and Scholes (1973), Merton (1973))
- In practice continuous trading of arbitrarily small amounts of stock is infinitely costly and the replicating portfolio is adjusted at discrete times
  - Perfect replication is impossible and an optimal hedging strategy will depend on the desired trade-off between replication error and trading costs

Related work I

- While number of articles consider hedging in discrete time or transaction costs alone, Leland (1985) was first to address discrete hedging under transaction costs
  - His work was subsequently followed by others (see, for example, Figlewski (1989), Boyle and Vorst (1992), Henrotte (1993), Grannan and Swindle (1996), Toft (1996), Whalley and Wilmott (1997), and Martellini (2000))
  - The majority of these studies consider proportional transaction costs
- More recently, several studies have considered option pricing and hedging subject to both permanent and temporary market impact in the spirit of Almgren and Chriss (1999), including Rogers and Singh (2010), Almgren and Li (2016), Bank, Soner, and Voß (2017), and Saito and Takahashi (2017)
- Halperin (2017) applies reinforcement learning to options but does not consider transaction costs

Related work II

- Buehler, Gonon, Teichmann, and Wood (2018) evaluate NN-based hedging under coherent risk measures subject to proportional transaction costs
- Cannelli, Nuti, Sala, and Szehr (2020) compare the risk-averse contextual k-armed bandit (R-CMAB) to DQN for the hedging of options in the BSM setting
- Cao, Chen, Hull, and Poulos (2020) explore DRL methods for option replication in BSM and stochastic volatility setups, comparing the performance of accounting P&L and cash flow approaches

What we do

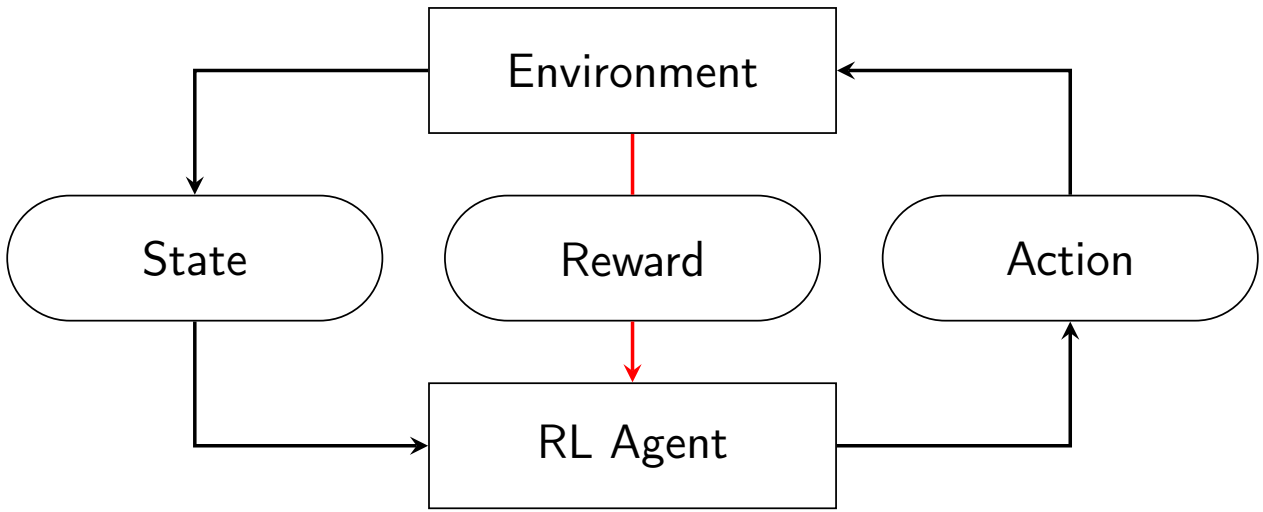
In these articles we:

- Show how to use reinforcement learning (RL) to optimally hedge an option (or other derivative securities) in a setting with
  - Discrete time rebalaning
  - Nonlinear transaction costs
  - Round-lotting
- The framework allows the user to "plug-in" any option pricing and simulation library, and train the system with no further modifications
  - Uses a continuous state space
  - Nonlinear regression techniques to the "sarsa targets"
  - State-of-the-art deep RL (DQN, DQN with Pop-Art, PPO)
  - The system learns how to optimally trade-off trading costs and hedging variance
- The approach extends in a straightforward way to arbitrary portfolios of derivative securities

Reinforcement learning

What is reinforcement learning I

- RL agent interacts with its environment. The "environment" is the part of the system outside of the agent's direct control
- At each time step t, the agent observes the current state of the environment s_t and chooses an action a_t from the action set
- This choice influences both the transition to the next state, as well as the reward R_t the agent receives



What is reinforcement learning II

- A (deterministic) policy $\pi : \mathcal{S} \to \mathcal{A}$ is a "rule" that chooses an action a_t conditional on the current state s_t
- RL is the search for policies which maximize the expected cumulative reward

$$\mathbb{E}[G_t] = \mathbb{E}[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots]$$

  where γ is discount factor (such that the infinite sum converges)
- Mathematically speaking, RL is a way to solve multi-period optimal control problems
- Standard texts on RL includes Sutton and Barto (2018) and Szepesvari (2010)

What is reinforcement learning III

- The action-value function expresses the value of starting in state s, taking an arbitrary action a, and then following policy π thereafter

$$Q^{\pi}(s, a) := \mathbb{E}_{\pi}[G_t \mid S_t = s, A_t = a] \tag{1}$$

where $\mathbb{E}_{\pi}$ denotes the expectation under the assumption that policy π is followed

- If we knew the Q-function corresponding to the optimal policy, Q^*, we would know the optimal policy itself, namely

$$\pi^*(s) = \arg\max_{a \in \mathcal{A}} Q^*(s, a) \tag{2}$$

This is called the *greedy policy*

What is reinforcement learning IV

- The optimal action-value function satisfies the *Bellman equation*

$$Q^*(s,a) = \mathbb{E}\left[R + \gamma \max_{a'} Q^*\left(s', a'\right) \middle| s, a\right] \tag{3}$$

- The basic idea of Q-learning is to turn the Bellman equation into the update

$$Q_{i+1}(s,a) = \mathbb{E}\left[R + \gamma \max_{a'} Q_i\left(s', a'\right) \mid s, a\right], \tag{4}$$

and iterate this scheme until convergence, $Q_i \to Q^*$

What is reinforcement learning V

- In deep Q-learning the action-value function is approximated with a deep neural network (DNN)

$$Q(s, a; \theta) \approx Q^*(s, a) \tag{5}$$

where θ represents the network parameters. The DNN is then trained by minimizing the sequence of losses

$$L_i(\theta_i) = \mathbb{E}_{(s,a,R,s')\sim U(D)} \left[L\Big(Q(s, a; \theta_i) - R - \gamma \max_{a'} Q(s', a'; \theta_i^-) \Big) \right]$$

where L is some loss function

Reinforcement learning for hedging

Automatic hedging in theory I

- We define automatic hedging to be the practice of using trained RL agents to handle hedging
- With no trading frictions and where continuous trading is possible, there may be a dynamic replicating portfolio which hedges the option position perfectly, meaning that the overall portfolio (option minus replication) has zero variance
- With frictions and where only discrete trading is possible the goal becomes to minimize variance and cost
  - We will use this to define the reward

Automatic hedging in theory II

- This suggest we can seek the agent's optimal portfolio as the solution to a mean-variance optimization problem with risk-aversion κ

$$\max \left(\mathbb{E}[w_T] - \frac{\kappa}{2} \mathbb{V}[w_T] \right) \tag{6}$$

where the final wealth w_T is the sum of individual wealth increments δw_t,

$$w_T = w_0 + \sum_{t=1}^{T} \delta w_t$$

We will let wealth increments include trading costs

Automatic hedging in theory III

- We choose the reward in each period to be

$$R_t := \delta w_t - \frac{\kappa}{2}(\delta w_t - \widehat{\mu})^2 \tag{7}$$

  where $\hat{\mu}$ is an estimate of a parameter representing the mean wealth increment over one period, $\mu := \mathbb{E}[\delta w_t]$.
- Thus, training reinforcement learners with this kind of reward function amounts to training automatic hedgers who tradeoff costs and hedging variance
- See Ritter (2017) for a general discussion of reward functions in trading

Automatic hedging in practice I

- Simplest possible example: A European call option with strike price K and expiry T on a non-dividend-paying stock
- We take the strike and maturity as fixed, exogenously-given constants. For simplicity, we assume the risk-free rate is zero
- The agent we train will learn to hedge this specific option with this strike and maturity. It is not being trained to hedge any option with any possible strike/maturity

- For European options, the state must minimally contain (1) the current price S_t of the underlying, (2) the time $\tau := T - t > 0$ remaining to expiry, and (3) our current position of n shares
- The state is thus naturally an element of

$$\mathcal{S} := \mathbb{R}_+^2 \times \mathbb{Z} = \{(S, \tau, n) \mid S > 0, \tau > 0, n \in \mathbb{Z}\}.$$

Automatic hedging in practice II

- The state *does not* need to contain the option Greeks, because they are (nonlinear) functions of the variables the agent has access to via the state
  - We expect the agent to learn such nonlinear functions on their own
- A key point: This has the advantage of not requiring any special, model-specific calculations that may not extend beyond BSM models

Simulation assumptions I

- We simulate a discrete BSM world where the stock price process is a geometric Brownian motion (GBM) with initial price S_0 and daily lognormal volatility of σ/day
- We consider an initially at-the-money European call option (struck at $K = S_0$) with T days to maturity
- We discretize time with D periods per day, hence each "episode" has $T \cdot D$ total periods
- We require trades (hence also holdings) to be integer numbers of shares
- We assume that our agent's job is to hedge one contract of this option
- In the specific examples below, the parameters are $\sigma = 0.01, S_0 = 100, T = 10$, and $D = 5$. We set the risk-aversion, $\kappa = 0.1$

Simulation assumptions II

- T-costs: For a trade size of n shares we define

$$\text{cost}(n) = \text{multiplier} \times \text{TickSize} \times (|n| + 0.01n^2)$$

where we take TickSize $= 0.1$

  - With multiplier $= 1$, the term TickSize $\times |n|$ represents the cost, relative to the midpoint, of crossing a bid-offer spread that is two ticks wide
  - The quadratic term is a simplistic model for market impact

Example: Baseline agent (discrete & no t-costs)

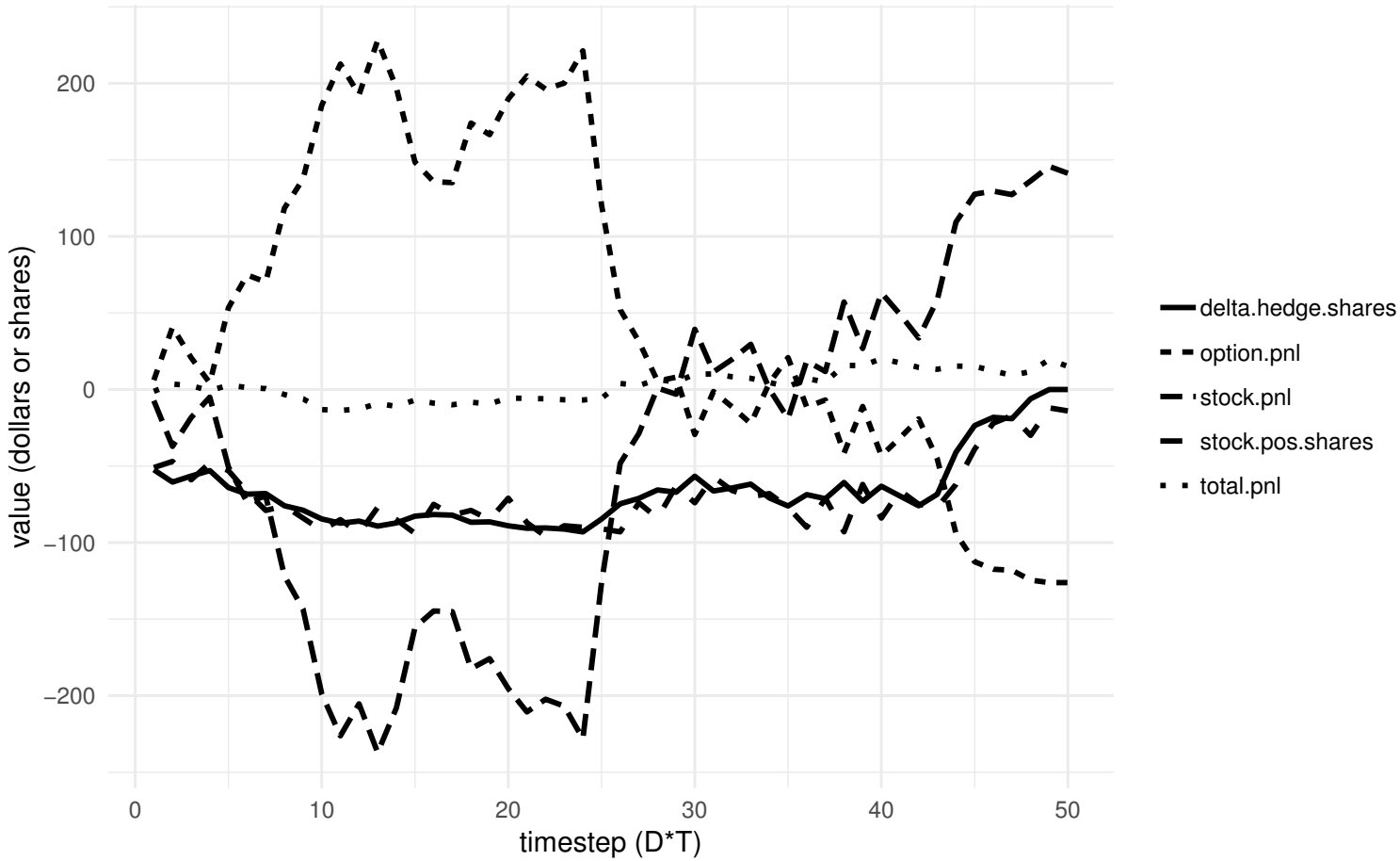


Figure 1: Stock & options P&L roughly cancel to give the (relatively low variance) total P&L. The agent's position tracks the delta

Example: Baseline agent (discrete & t-costs)

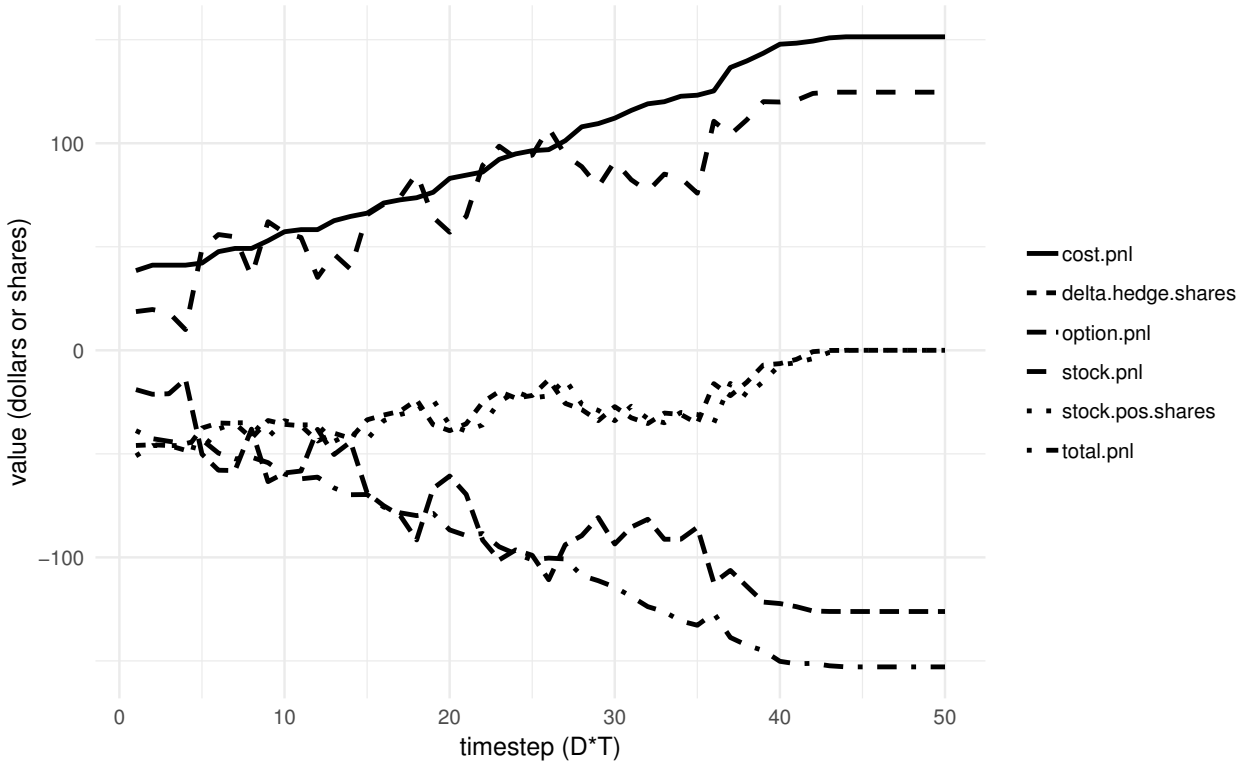
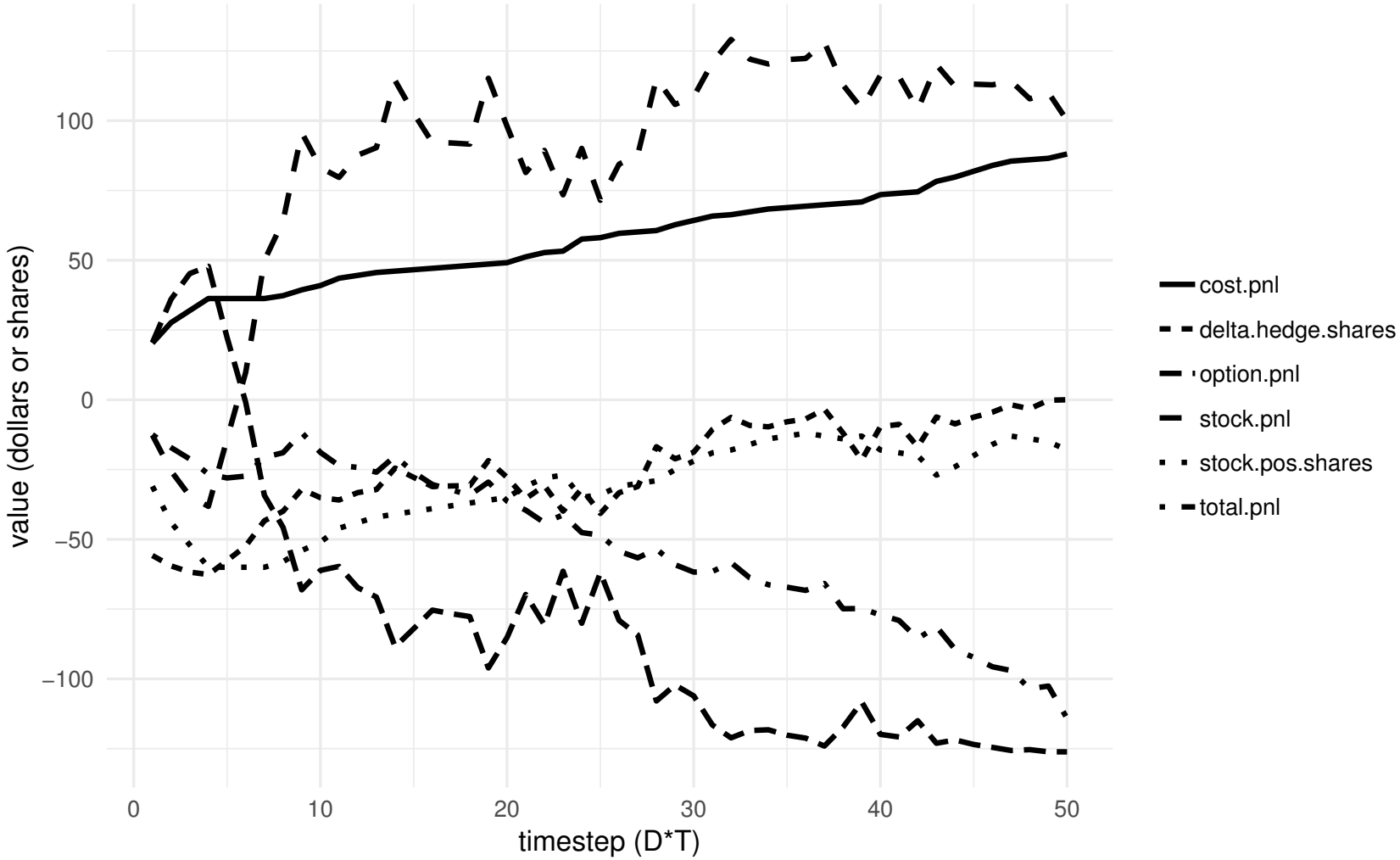


Figure 2: Stock & options P&L roughly cancel to give the (relatively low variance) total P&L. The agent trades so that the position in the next period will be the quantity $-100 \cdot \Delta$ rounded to shares

Example: T-cost aware agent (discrete & t-costs)



Kernel density estimates of total P&L

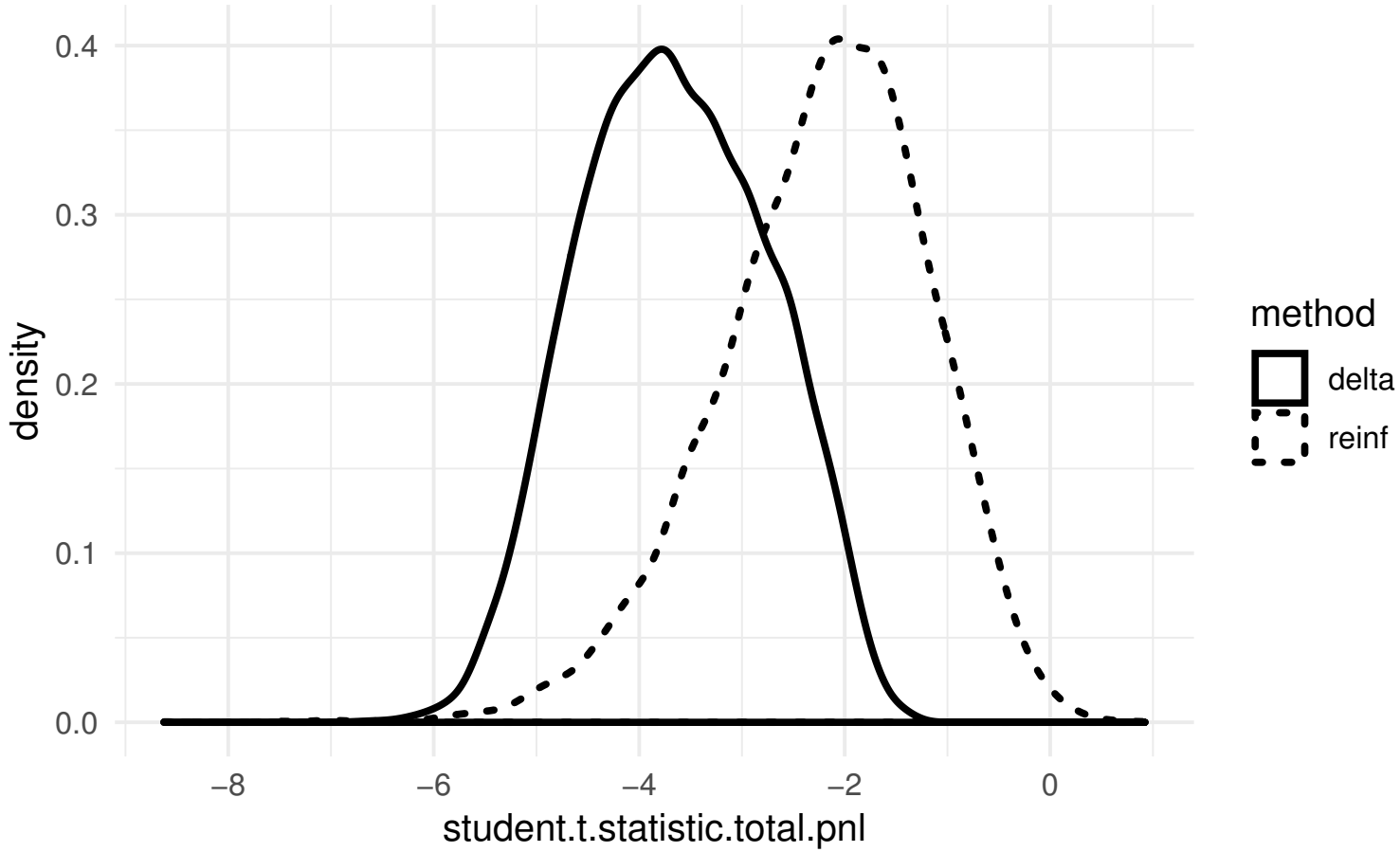


Figure 3: Kernel density estimates of the t-statistic of total P&L for each of our out-of-sample simulation runs, and for both policies represented above ("delta" and "reinf"). The "reinf" method is seen to outperform in the sense that the t-statistic is much more often close to zero and insignificant.

Extensions I

We have extended this approach in several different directions. Here is a summary of our findings:

- An agent can be trained at once for a whole range of strikes and maturities
- Deep Q-learning (DQN) and double deep Q-learning (DDQN) (Hasselt, 2010; Mnih, Kavukcuoglu, Silver, Rusu, Veness, Bellemare, Graves, Riedmiller, Fidjeland, and Ostrovski, 2015; Van Hasselt, Guez, and Silver, 2016) are "easy" to work with, but suffers from slow convergence
- DQN with Pop-Art (Hasselt, Guez, Hessel, Mnih, and Silver, 2016) improves training and overall performance due to its adaptive normalization

Extensions II

- Proximal policy optimization (PPO) and actor-critic policy-based reinforcement learning (Schulman, Wolski, Dhariwal, Radford, and Klimov, 2017; Wu, Mansimov, Grosse, Liao, and Ba, 2017)
  - Converge $\sim$ 2 magnitudes faster, and
  - Produce more robust policies than DQN

Pop-Art normalization stabilizes DQN

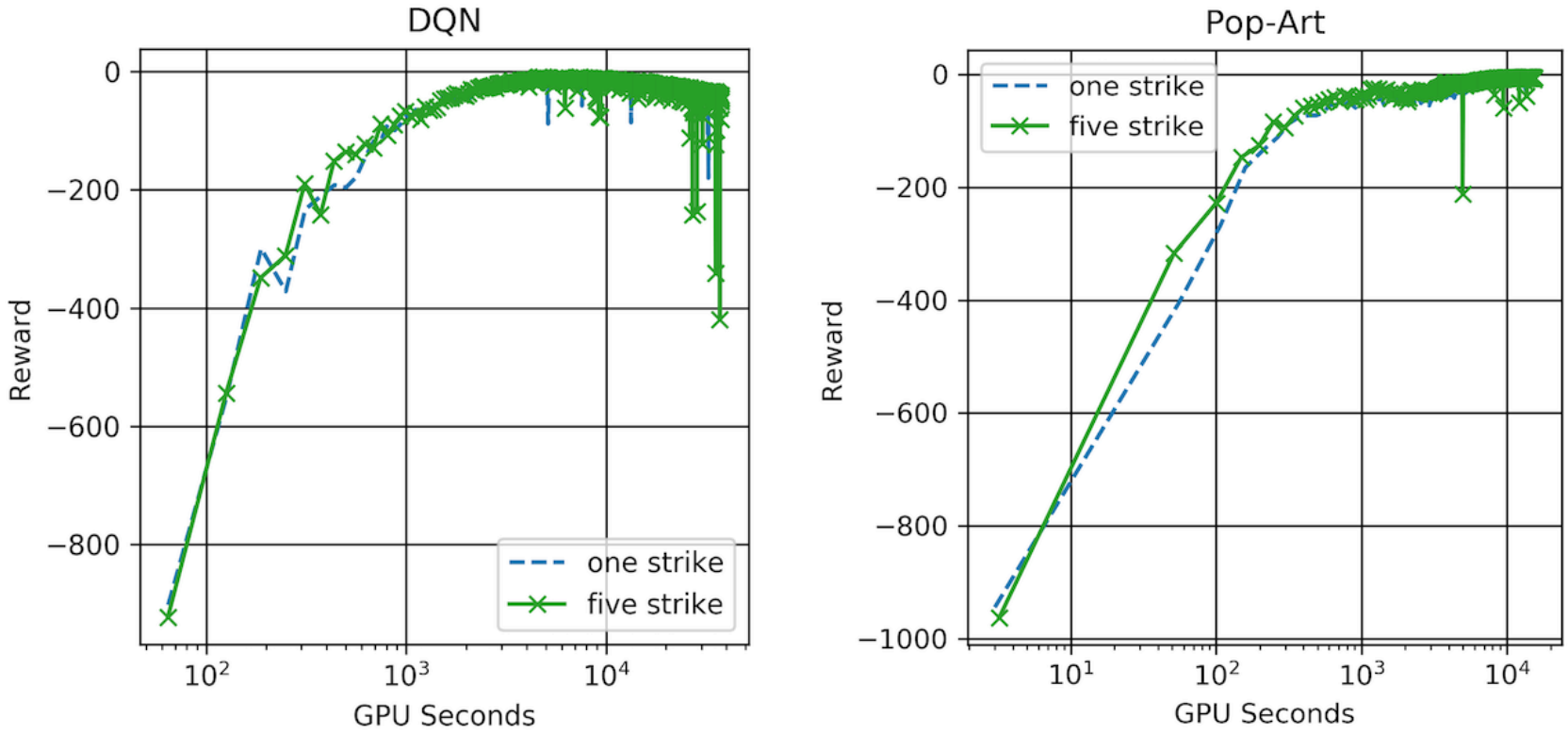


Figure 4: Left panel: It is well-known that DQN can diverge when the exploration rate becomes small. Right panel: Pop-Art remedies the divergence of DQN.

Proximal Policy Optimization (PPO) learns faster than DQN – By far

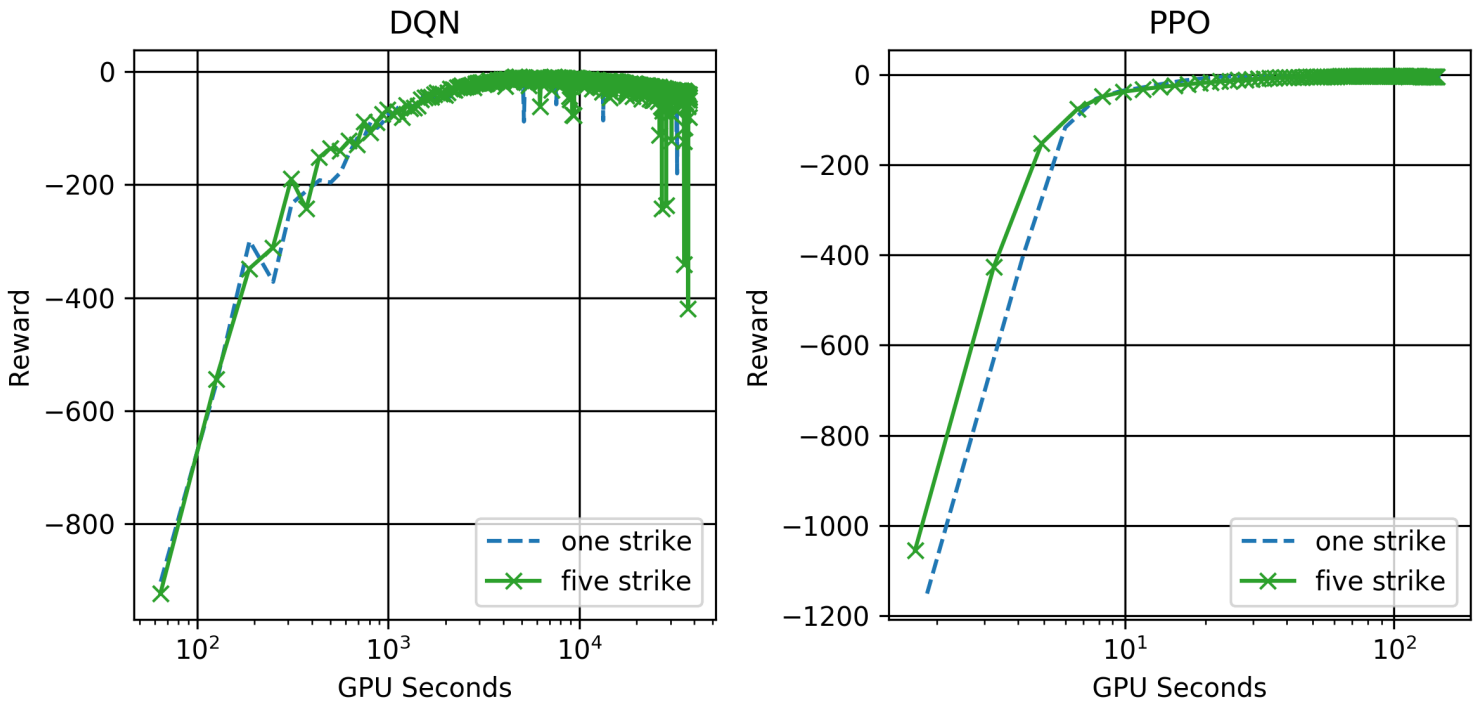


Figure 5: Left panel: Reward of DQN. Right panel: Reward of PPO.

Conclusions I

We have studied an RL-based framework that hedges options under realistic conditions of discrete trading, nonlinear t-costs and round lotting

- Our approach does not depend on the existence of perfect dynamic replication. The system learns to optimally trade off variance and cost, as best as possible using whatever securities it is given as potential candidates for inclusion in the replicating portfolio
- A key strength of the RL approach: It does not make any assumptions about the form of t-costs. RL learns the minimum variance hedge subject to whatever t-cost function one provides. All it needs is a good simulator, in which t-costs and options prices are simulated accurately

Conclusions II

- We have extended the approach in a number of different directions using state-of the-art deep RL such as DQN, DQN with Pop-Art and PPO

Contact

Petter Kolm

petter.kolm@nyu.edu

https://www.linkedin.com/in/petterkolm

Courant Institute, NYU

References I

Almgren, Robert and Neil Chriss (1999). "Value under liquidation". In: *Risk* 12.12, pp. 61–63.

Almgren, Robert and Tianhui Michael Li (2016). "Option hedging with smooth market impact". In: *Market Microstructure and Liquidity* 2.1, p. 1650002.

Bank, Peter, H Mete Soner, and Moritz Voß (2017). "Hedging with temporary price impact". In: *Mathematics and Financial Economics* 11.2, pp. 215–239.

Black, Fischer and Myron Scholes (1973). "The pricing of options and corporate liabilities". In: *Journal of Political Economy* 81.3, pp. 637–654.

Boyle, Phelim P and Ton Vorst (1992). "Option replication in discrete time with transaction costs". In: *The Journal of Finance* 47.1, pp. 271–293.

Buehler, Hans et al. (2018). "Deep hedging". In: *arXiv:1802.03042*.

Cannelli, Loris et al. (2020). "Hedging Using Reinforcement Learning: Contextual k-Armed Bandit versus Q-learning". In: *arXiv preprint arXiv:2007.01623*.

Cao, Jay et al. (2020). "Deep Hedging of Derivatives Using Reinforcement Learning". In: *Available at SSRN 3514586*.

Du, Jiayi et al. (2020). "Deep Reinforcement Learning for Option Replication and Hedging". In: *The Journal of Financial Data Science* 2.4.

Figlewski, Stephen (1989). "Options arbitrage in imperfect markets". In: *The Journal of Finance* 44.5, pp. 1289–1311.

References II

Grannan, Erik R and Glen H Swindle (1996). "Minimizing transaction costs of option hedging strategies". In: *Mathematical Finance* 6.4, pp. 341–364.

Halperin, Igor (2017). "QLBS: Q-Learner in the Black-Scholes (-Merton) Worlds". In: *arXiv:1712.04609*.

Hasselt, Hado P van et al. (2016). "Learning values across many orders of magnitude". In: *Advances in Neural Information Processing Systems*, pp. 4287–4295.

Hasselt, Hado V (2010). "Double Q-learning". In: *Advances in Neural Information Processing Systems*, pp. 2613–2621.

Henrotte, Philippe (1993). "Transaction costs and duplication strategies". In: *Graduate School of Business, Stanford University*.

Kolm, Petter and Gordon Ritter (2019). "Dynamic Replication and Hedging: A Reinforcement Learning Approach". In: *The Journal of Financial Data Science* 1.1, pp. 159–171.

— (2020). "Modern Perspectives on Reinforcement Learning in Finance". In: *Journal of Machine Learning in Finance* 1.1. URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3449401.

Leland, Hayne E (1985). "Option pricing and replication with transactions costs". In: *The Journal of Finance* 40.5, pp. 1283–1301.

Martellini, Lionel (2000). "Efficient option replication in the presence of transactions costs". In: *Review of Derivatives Research* 4.2, pp. 107–131.

Merton, Robert C (1973). "Theory of rational option pricing". In: *The Bell Journal of Economics and Management Science*, pp. 141–183.

References III


Mnih, Volodymyr et al. (2015). "Human-level control through deep reinforcement learning". In: *Nature* 518.7540, p. 529.


Ritter, Gordon (2017). "Machine Learning for Trading". In: *Risk* 30.10, pp. 84–89.


Rogers, Leonard CG and Surbjeet Singh (2010). "The cost of illiquidity and its effects on hedging". In: *Mathematical Finance* 20.4, pp. 597–615.


Saito, Taiga and Akihiko Takahashi (2017). "Derivatives pricing with market impact and limit order book". In: *Automatica* 86, pp. 154–165.


Schulman, John et al. (2017). "Proximal policy optimization algorithms". In: *arXiv preprint arXiv:1707.06347*.


Sutton, Richard S and Andrew G Barto (2018). *Reinforcement learning: An introduction*. Second edition, in progress. MIT press Cambridge.


Szepesvari, Csaba (2010). *Algorithms for Reinforcement Learning*. Morgan & Claypool Publishers.


Toft, Klaus Bjerre (1996). "On the mean-variance tradeoff in option replication with transactions costs". In: *Journal of Financial and Quantitative Analysis* 31.2, pp. 233–263.


Van Hasselt, Hado, Arthur Guez, and David Silver (2016). "Deep reinforcement learning with double q-learning". In: *Thirtieth AAAI conference on artificial intelligence*.


Whalley, A Elizabeth and Paul Wilmott (1997). "An asymptotic analysis of an optimal hedging model for option pricing with transaction costs". In: *Mathematical Finance* 7.3, pp. 307–324.

References IV

Wu, Yuhuai et al. (2017). "Scalable trust-region method for deep reinforcement learning using kronecker-factored approximation". In: *Advances in neural information processing systems*, pp. 5279–5288.