CME241: Reinforcement Learning with Applications in Finance

Final Project Details • Prof. Ashwin Rao

Final Project

For the final project of CME241, we are encouraging students to explore a topic of their own interest within reinforcement learning and its financial applications. We hope that this includes both an opportunity to explore the real-world examples discussed in the class as well as code up and experiment with simulations in the RL-book codebase.

We encourage you to start early and course staff will do their best to help you along the way. We will start with a project proposals containing a general purpose idea of what you plan to read / implement. The goal is to make sure you headed in a reasonable direction. Near the end of the class, we will have project presentations that are intended as summaries of what you have done. Finally, we will request code and an updated presentation turned-in for grading purposes.

Logistics

Teams:

1. Teams of 2 or 3
2. Special permission may be granted for solo projects with appropriate reasoning (email amil@stanford.edu). Otherwise, we will encourage you to group up.

Key Dates:

1. Project Proposal (February 22 - 11:59pm, <1 page, basic problem definition + plan)
2. Project Presentation (March 17 - Final Class - 3:00pm - 5:00pm, 10 min + 2 for questions)
3. Code + Updated Presentation Turn-In (March 21 - 8:30am)

Evaluation

As we are evaluating projects, we will be looking at the following criteria:

1. Did the students choose a topic relevant to reinforcement learning and financial applications?
2. Did the students identify an interesting, well-defined problem to explore?
3. Did the students well-explain the mathematical model underpinning the problem?
4. How well-founded were the simulations and experiments surrounding the project?

Potential Ideas

Inventory Management

In class, we have considered inventory management problems, such as multiple retail locations with the option of sharing inventory overnight. Extend these kinds of scenarios, either by scale or by more complex interactions. Consider when does dynamic programming fail; can the issues be solved with appropriate reinforcement learning algorithms? This problem could be also mapped to other MDPs such as sports, games (tic-tac-toe, small checkers), or other topics of students interest.

Asset Allocation

As will be covered in class, asset allocation is a difficult yet critical problem in financial math. Start

by designing a discrete time-asset asset allocation problem (of a single risky asset with no consumption) and code up several strategies to find solutions (Monte-Carlo and Actor-Critic-Eligibility-Traces Policy Gradients). As the book has some of these methods already implemented, consider extensions such as more assets and added variables such as consumption.

Options Pricing

Reinforcement learning algorithms can be used for pricing American Options, as an alternative to Binomial Tree implementations. Consider first implementing algorithms such as LSPI and Deep Q-Learning for pricing American Options. As the book has some of these methods implemented, extended this idea by either considering additional algorithms, alternative pricing strategies, or even comparing to historical data.

Deep Reinforcement Learning

OpenAI's GYM provides a stepping stone into deep reinforcement learning, with provided environments in which to test your algorithms. For this kind of final project, students could look to implement a few of the deep reinforcement learning algorithms discussed in class and evaluate how well-these can solve common control tasks such as CartPole or any of the Atari games.

Multi-Agent Reinforcement Learning

In this class, we mostly discussed the role of a single agent interacting with the environment and making decisions. How do the algorithms change when there are multiple agents that are able to control actions? Multi-agent reinforcement learning has become really popular in the past two years leading through breakthroughs in games such as hide and seek. For this kind of final project, we would hope that you would explore relevant discussion to multi-agent reinforcement learning, describe the differences that arise from having multiple characters controlling behavior (and why this may come up in finance), and design simulations surrounding this problem.

Your Own

Feel free to come talk to us during office hours!