Fall 2024

DS 669-1J1: Reinforcement Learning

Daming Li

Course Syllabus

DS 669: Reinforcement Learning

Fall 2024

Instructor: Daming(David) Li, email: dli@njit.edu

Course Description and Format:

This course will prepare students for hands-on skills in deep reinforcement learning with Python and the advanced framework PyTorch. This course will combine theory and practice together, and guide students step by step to analyze, design, implement and present real-world deep reinforcement learning projects. This course will also promote the very latest reinforcement learning tools and techniques in industry as well as the new innovations in this emerging field to prepare students well for career development in data science.

Course Learning Outcomes:

By the end of this course, students will be able to:

1. Understand the concepts in deep reinforcement learning
2. Hands-on experience in deep reinforcement learning with Python/PyTorch
3. Learn to analyze, design, implement and present real-world deep reinforcement learning projects

Textbooks (helpful but not required):

- Maxim Lapan, Deep Reinforcement Learning Hands-On 2nd Edition https://www.amazon.com/dp/1838826998
- Richard S. Sutton & Andrew G. Barto, Reinforcement Learning An Introduction, 2nd Edition http://incompleteideas.net/book/the-book-2nd.html

Collaboration and Honor Code: Students may discuss problems together but must write up their own solutions. When writing up the solutions, students should write the names of people, if any, with whom they discussed the assignment. Note in particular that copying homework or programming assignments, in full or in part is forbidden. Students found cheating or plagiarizing will be immediately referred to the Dean of Students and the NJIT Committee on Professional Conduct and subject to Disciplinary Probation, a permanent marking on the record, possible dismissal, and an "F" grade in the course. All submitted assignments will be checked for similarities, and plagiarism and guilty students identified.

Grading:

The requirements of this course will consist of participating in lectures, homework, in class computing lab assignments, two exams and a project. The grading breakdown is the following:

- Homework, computing lab exercise (10%)

- Quiz (20%)
- Term Project (20%)
- Midterm (20%)
- Exam (30%)

Homework (10 %)

- Only use Python/PyTorch in homework
- Try to do it independently, discussions allowed, but copying is forbidden.
- 25% penalization per late day;
- Not accepted more than 3 days late

Lab exercise

- Have a lab session every week
- We will solve some simple problems
- Post your answers by replying on canvas
- Some answers may be selected for discussion by the end of lab session.
- Some problems may become part of homework

Quiz (20%)

- Focus on course materials.
- 4 Quizzes
- Every other week
- Only Python/PyTorch is allowed

Two Term Projects (20%)

- Use Python/PyTorch for your projects
- Use Jupyter
- Submit code and report to summarize what you have done and results you obtained.
- Prepare for presentation and demo.
- If it is a group project, the group size will be announced.
- More details to be announced on canvas
- Cheating/Copying is strictly prohibited.

Two Exams (50%)

- One midterm and one Final (20%+30%)
- In-class
- Final is cumulative
- Only Python/PyTorch

Tentative course topics (Subject to changes according to progress)

1. Introduction to Reinforcement Learning
2. Deep Learning with PyTorch
3. The Cross-Entropy Method
4. Exploration vs. Exploitation with multi-armed bandit
5. Markov Decision Process
6. Dynamic Programming
7. Temporal-Difference Learning and Sarsa
8. **Midterm Exam for all above**
9. Dyna-Q Algorithm
10. Deep Q-Networks
11. DQN Extensions
12. Policy Gradients
13. The Actor-Critic Method
14. Continuous Action Space
15. **Final Exam for all above**