Reinforcement Learning
Syllabus Spring 2020 [Updated]

Course Title:

Reinforcement Learning

Course Number:

CSE 410 / 510 (Senior/Graduate)

Course Format:

Lecture are held online on Monday and Wednesday 11:00am - 12:20pm.

Course Catalog Description

This course is intended for students interested in artificial intelligence. Reinforcement learning is an area of machine learning where an agent learns how to behave in an environment by performing actions and assessing the results. Reinforcement learning is how Google DeepMind created the AlphaGo system that beat a high-ranking Go player and how AlphaStar become the first artificially intelligent system to defeat a top professional player in StarCraft II. We will study the fundamentals and practical applications of reinforcement learning and will cover the latest methods used to create agents that can solve a variety of complex tasks, with applications ranging from gaming to finance to robotics.
The course is comprised of assignments, short weekly quizzes, a final project and a final exam.

Course Outline

Reinforcement learning is an area of machine learning, where an agent or a system of agents learn to archive a goal by interacting with their environment. RL is often seen as the third area of machine learning, in addition to supervised and unsupervised areas, in which learning of an agent occurs as a result of its own actions and interaction with the environment.

In recent years there has been success in reinforcement learning research in both theoretical and applied fields. It was applied in a variety of fields such as robotics, pattern recognition, personalized medical treatment, drug discovery, speech recognition, computer vision, and natural language processing. This course primarily focuses on training students to frame reinforcement learning problems and to tackle algorithms from dynamic programming, Monte Carlo and temporal-difference learning. Students will progress towards larger state space environments using function approximation, deep Q-networks and state-of-the-art policy gradient algorithms. We will also go over the recent methods that are based on reinforcement learning, such as imitation learning, meta learning and more complex environment formulations.

Key Topics

1. RL task formulation (action space, state space, environment definition)
2. Tabular based solutions (dynamic programming, Monte Carlo, temporal-difference)

3. Function approximation solutions (Deep Q-networks)
4. Policy gradient from basic (REINFORCE) towards advanced topics (proximal policy optimization, deep deterministic policy gradient, etc.)
5. Model-based reinforcement learning
6. Imitation learning (behavioral cloning, inverse RL, generative adversarial imitation learning)
7. Meta-learning
8. Multi-agent learning, partial observable environments

Objectives / Goals:

By the end of this course, students should be able to do the following:

- Learn how to define RL tasks and the core principals behind the RL, including policies, value functions, deriving Bellman equations (as assets by the assignments, an exam and quizzes)
- Implement in code common algorithms following code standards and libraries used in RL (as assessed by the assignments and final project)
- Understand and work with tabular methods to solve classical control problems (as assessed by the assignments, quizzes and final exam)
- Understand and work with approximate solutions (deep Q network based algorithms) (as assessed by the assignments and final exam)
- Learn the policy gradient methods from vanilla to more complex cases (as assessed by the assignments, quizzes and final exam)
- Explore imitation learning tasks and solutions (as assessed by the quizzes and final exam)
- Recognize current advanced techniques and applications in RL (as assessed by the final project, quizzes and final exam)

Lecture Schedule

Foundations

1. Introduction and Basics of RL
2. Defining RL Framework and Markov Decision Process
3. Polices, Value Functions and Bellman Equations
4. Exploration vs. Exploitation
5. Code Standards and Libraries used in RL (Python/Keras/Tensorflow)

Tabular methods and Q-networks

6. Planning through the use of Dynamic Programming and Monte Carlo
7. Temporal-Difference learning methods (TD(0), SARSA, Q-Learning)
8. Deep Q-networks (DQN, DDQN, Dueling DQN, Prioritised Experience Replay)

Policy optimization

9. Introduction to policy-based methods
10. Vanilla Policy Gradient
11. REINFORCE algorithm and stochastic policy search
12. Actor-critic methods (A2C, A3C)
13. Advanced policy gradient (PPO, TRPO, DDPG)

Model based RL

14. Model-based RL approach

Recent Advances and Applications

15. Meta-learning
16. Multi-Agent Reinforcement Learning
17. Partially Observable Markov Decision Process
18. Ethics in RL
19. Applying RL for real-world problems

Assessment Overview *

40% - Assignments (15% + 15% + 10%)

20% - Final Project

10% - Weekly Quizzes

15% - Midterm 1

15% - Midterm 2

* Can be adjusted before the beginning of the semester

Weekly Quizzes

- Assigned every Monday 9:00am, due by Sunday 11:59pm
- Posted on UBlearns > Assignments
- Each quiz will contain 3-4 problems on topics covered that week
- At the end of a submission the system will give the final score
- 11 quizzes in total, only 10 quizzes with the highest scores will be counted toward the final grade
- Three attempts allowed, only the highest score will be kept
- Quizzes can be completed collaboratively. Every week you will be randomly matched with someone from our class, and you would have to connect online and discuss the quiz questions and answers. This interaction is not mandatory, so if you do not wish to participate in this collaboration, please let me know.

Late Day Policy

- Students can use up to 5 late days
- A late day extends the deadline by 24 hours
- If there is more then 5 days after the deadline, a penalty of 25% for one day will be applied to any work submitted after that time

Course Bibliography

This course is based on materials from the following textbooks, review articles and recent papers. Students will be assigned course materials to study prior to classes from the following sources along with others.

Books

- Richard S. Sutton and Andrew G. Barto, "Reinforcement learning: An introduction", Second Edition, MIT Press, 2019
- Li, Yuxi. "Deep reinforcement learning." arXiv preprint arXiv:1810.06339 (2018).
- Wiering, Marco, and Martijn Van Otterlo. "Reinforcement learning." Adaptation, learning, and optimization 12 (2012): 3..
- Russell, Stuart J., and Peter Norvig. "Artificial intelligence: a modern approach."Pearson Education Limited, 2016.
- Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. "Deep learning." MIT press, 2016.
- David Silver's course on Reinforcement Learning (link)

Selected Papers

- Mnih, Volodymyr, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves et al. "Human-level control through deep reinforcement learning." Nature 518, no. 7540 (2015): 529.
- Van Hasselt, Hado, Arthur Guez, and David Silver. "Deep reinforcement learning with double q-learning." In Thirtieth AAAI conference on artificial intelligence. 2016.
- Wang, Ziyu, Tom Schaul, Matteo Hessel, Hado Van Hasselt, Marc Lanctot, and Nando De Freitas. "Dueling network architectures for deep reinforcement learning." arXiv preprint arXiv:1511.06581 (2015).
- Schaul, Tom, John Quan, Ioannis Antonoglou, and David Silver. "Prioritized experience replay." arXiv preprint arXiv:1511.05952 (2015).
- Bojarski, Mariusz, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D. Jackel et al. "End to end learning for self-driving cars." arXiv preprint arXiv:1604.07316 (2016).
- Schulman, John, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. "Trust region policy optimization." In International conference on machine learning, pp. 1889-1897. 2015.
- Schulman, John, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. "Proximal policy optimization algorithms." arXiv preprint arXiv:1707.06347 (2017).
- Lillicrap, Timothy P., Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. "Continuous control with deep reinforcement learning." arXiv preprint arXiv:1509.02971 (2015).

Prerequisites

CSE 250 and one of the following: EAS 305 or MTH 411 or STA 301 or MTH 309.
CSE4/574 or CSE4/555 or CSE4/573 or CSE4/568 is recommended to be either completed or taken during the same semester.

Academic Integrity Policy

Academic integrity is a fundamental university value. No collaboration, cheating, and plagiarism is allowed in projects, quizzes or the exam. Those found violating academic integrity will get an immediate F in the course. Please refer to the Academic Integrity Policy for more details.

Auditing

Auditing of the course (i.e. attending lectures, but not turning in assignments, quizzes, project to grade) is welcome and permitted with prior approval. We give priority to students who are officially registered for the course, so auditors may only take a seat in the classroom if there is one available 10 minutes after the start of class. Auditors will not be given access to course materials such as assignments, quizzes and exam.

Rationale

Reinforcement learning is one of the key directions in the area of artificial intelligence. It is widely applied in robotics and complex systems that aim to build intelligent agents. Many leading computer science departments offer a reinforcement learning course to train their students to apply novel methods to a variety of tasks. An emerging application for reinforcement learning is in modeling and solving complex natural and social systems using the Markov property of the environment. This hands-on course trains students to apply reinforcement learning methods to solve tasks from a basic grid-world environment to complex large-scale systems. It is hoped that this course will be a core element in the area of artificial intelligence and it will satisfy a major breadth requirement in the area of Artificial Intelligence.