DS-GA 3001.005
Reinforcement Learning

Professor: Jeremy Curuksu

Center for Data Science

New York University

jeremy.cur@nyu.edu

Syllabus

Overview:

Reinforcement Learning (RL) is the *science of learning to make decisions from interactions with an environment*". The goal of this course is to understand the entire range of reinforcement learning methods and applications, and teach the tools necessary to formulate RL problems and develop RL solutions. The lectures cover fundamental and more advanced concepts. The labs focus on case studies implementing RL agents in simulation environments. The homework cover both. In contrast to supervised and unsupervised learning, by interacting with an environment RL methods can learn optimal decisions from limited data and/or feedback (labels), including "sequential" decisions in order to reach a preset, potentially delayed goal. Advances in RL applications such as video games, board games, robotics, autonomous driving, and language models, have recently been made by combining RL with function approximation methods. These include deep learning and tree search which enable generalizing to very large state-action spaces e.g., an RL agent developed in 2016 defeated the world human champion at the boardgame Go which has over 10^{170} possible positions, by learning patterns accross the nearly infinite space of possible positions. More recently RL was combined with Tranformers and reward functions parameterized based on human preferences to design language models (ChatGPT, etc) that can emulate human conversations and artistic creations. The course progressively builds on mathematical concepts leading up to these recent developments, practices implementation of key RL algorithms in Python, and reviews industrial applications.

Prerequisites:

Intro to Data Science (DS-GA 1001), Machine Learning (DS-GA 1003), or similar

Topics covered:

- **Introduction to Reinforcement Learning Problems and Solutions**: The first lecture intoduces key RL concepts: agent, environment, states, actions, goals, rewards, policy *vs.* value optimization, Bandit *vs.* sequential decision making, exploration *vs.* exploitation, and the OpenAI Gym library

- **Multi-armed Bandit**: Action values methods, ϵ-greedy, Upper Confidence Bound, policy-gradient, Bayesian methods, contextual Bandit

- **Markov Decision Process and Dynamic Programming**: MDP model for sequential decision problems, state-value function, action-value function, Bellman equations, iterative algorithms using asynchronous dynamic programming

- **Model-free Reinforcement Learning**: Sampling of experiences by Monte Carlo and Temporal Difference, bias-variance tradeoff, n-step TD/TD(λ), off-policy learning, Q-learning, importance sampling, tracking *vs.* convergence

- **Value Function Approximation (Deep RL)**: Generalization of RL to large state-action spaces, semi-gradient descent, state-space representatoin and feature engineering, approximation with deep neural network, DQN, DDQN

- **Policy Function Approximation (Actor-Critic)**: Parametric policy function, Policy Gradient Theorem, Reinforce, Actor-Critic, A3C, PPO, DDPG

- **Planning from a Model of the Environment (AlphaZero)**: Generative models, trajectory sampling, integrated planning and learning, experience replay, Dyna, Monte Carlo Tree Search, DeepMind AlphaGo and AlphaZero

- **Reinforcement Learning from Human Preferences**: Fine tuning language models by RL from human feedback (RLHF), superalignment by RL from AI feedback (RLAIF), Constitutional AI

- **Examples of Industrial Applications**: RL in video gaming, autonomous driving, robotics, finance, retail, energy, natural language processing, chatbots

- **Advanced Topics and RL Development Frameworks**: OpenAI Gym, Google DeepMind Acme, Transformer RL, Amazon SageMaker RL, Meta AI ReAgent, multi-agent RL, RL in neuroscience and psychology

Labs and Project:

The labs focus on concrete case studies though examples implementing RL solutions in Python, with optional exercices. Each session covers a key RL topic covered during the lecture. Case studies will range from classical control use cases (Black Jack, Cart Pole, Moon Lander, Frozen lake, Multi-Armed Bandits, Grid Lock) and simple video games (Breakout, Space Invader, Flappy Bird, Super Mario Bros) to more custom applications such as financial trading, robotics and natural language processing with large language models.

The project consists in proposing a reinforcement learning problem and developing a solution to this problem. A complete project description and list of recommended RL problems will be posted on Brightspace within the first week of the semester.

Day, Time and Place

DS-GA 3001.005 Lecture

Wednesdays from 4:55pm-6:35pm EST

Location: 194 Mercer St, Room 204

DS-GA 3001.006 Lab

Thursdays from 7:10pm-8:00pm EST

Location: Bobst Library, Room LL138

Teaching Assistants

- Anjel Patel, ap8589@nyu.edu
- Anudeep Tubati, at5373@nyu.edu

Lecture Notes, Textbook & Documentation

- The material provided during the Lectures and Labs is sufficient for this course, but is best when complemented by reading the following material
- Recommended textbook: "Reinforcement Learning: An introduction" (2018) by R. Sutton and A. Barto (free online version also available: Sutton & Barto)

- The main open source RL library used in this course (in addition to standard Python and Tensorflow libraries) is OpenAI Gym (maintained as Gymnasium since 2022), which has a freely accessible, online documentation, and which we recommend you consult regularly during your practices. We will also introduce specialized RL platforms such as Google Deepmind's ACME and HuggingFace's Transformer RL.

Grading Policy

- Homework during the whole semester: 50% of the final grade
- Final Project at the end of the semester: 50% of the final grade