<TOP>

tags: Udacity, reinforcement learning, online course, 2023
👉 reinforcement learning cheat sheet
👉 Udacity Reinforcement Learning repo
👉 Unity ML-Agents GitHub repo
👉 an alternative curriculum

- **Project 1-3 (and other course practice) GitHub repo**
  - **P3, Unity Tennis (vector game)**
    - Algorithm: **MA-TD3** (Multi-Agent Twin Delayed DDPG), **MA-DDPG** (Multi-Agent Deep Deterministic Policy Gradient)
  - **P2, Unity Reacher-v2 (vector game)**
    - Algorithm: **DDPG** (Deep Deterministic Policy Gradient)
    - Demo post
  - **P1, Unity Banana Collector** (vector game)
    - Algorithm: **Double DQN** (Deep Q-Network), **Dueling DQN**
  - OpenAI Gym's Atari Pong (pixel game)
    - Algorithm: **PPO** (Proximal policy optimization)
    - Demo post
  - Tic-tac-toe (simple game)
    - Algorithm: **MCST** (Monte Carlo Tree Search)
  - <END>

- Certificate post
- Project documents (Google Docs)
- Environment: Windows 11, Miniconda 3, Python 3.10, VS Code, PowerShell
  OpenAI Baselines tf2, mujoco-py-1.50.1.68, pytorch-cuda=12.1, gym==0.14.0, bleach==1.5.0

Full Curriculum

Introduction to Deep Reinforcement Learning

Welcome to Deep Reinforcement Learning

Welcome to the Deep Reinforcement Learning Nanodegree program!

Getting Help

You are starting a challenging but rewarding journey! Take 5 minutes to read how to get help with projects and content.

Get Help with Your Account

What to do if you have questions about your account or general questions about the program.

Learning Plan

Obtain helpful resources to accelerate your learning in this first part of the Nanodegree program.

Introduction to RL

Reinforcement learning is a type of machine learning where the machine or software agent learns how to maximize its performance at a task.

The RL Framework: The Problem

Learn how to mathematically formulate tasks as Markov Decision Processes.

1. Introduction (YouTube)

2. The Setting, Revisited

3. Episodic vs. Continuing Tasks

4. Quiz: Test Your Intuition

5. Quiz: Episodic or Continuing?

6. The Reward Hypothesis

7. Goals and Rewards, Part 1

8. Goals and Rewards, Part 2

9. Quiz: Goals and Rewards

10. Cumulative Reward

11. Discounted Return

12. Quiz: Pole-Balancing

13. MDPs, Part 1

14. MDPs, Part 2

15. Quiz: One-Step Dynamics, Part 1

16. Quiz: One-Step Dynamics, Part 2

17. MDPs, Part 3

18. Finite MDPs

19. Summary (Evernote)

The RL Framework: The Solution

In reinforcement learning, agents learn to prioritize different decisions based on the rewards and punishments associated with different outcomes.

1. Introduction

2. Policies

3. Quiz: Interpret the Policy

4. Gridworld Example

5. State-Value Functions

6. Bellman Equations

7. Quiz: State-Value Functions

8. Optimality

9. Action-Value Functions

10. Quiz: Action-Value Functions

11. Optimal Policies

12. Quiz: Optimal Policies

13. Summary

Monte Carlo Methods (Evernote)

- Write your own implementation of Monte Carlo control to teach an agent to play Blackjack!

1. Review (YouTube)

2. Gridworld Example (YouTube)

3. Monte Carlo Methods (YouTube)
   equiprobable random policy

4. MC Prediction - Part 1 (YouTube)

5. MC Prediction - Part 2 (YouTube)
   Q-table

6. MC Prediction - Part 3 (Evernote, YouTube)
   prediction problem
   Monte Carlo (MC) prediction methods
   Option 1: Every-visit MC Prediction
   Option 2: First-visit MC Prediction (Algorithm 9)

7. OpenAI Gym: BlackJackEnv (Evernote)

8. Workspace - Introduction (Evernote, Colab notebook)

9. Coding Exercise (Evernote, YouTube)

10. Workspace (Colab notebook)

11. Greedy Policies (YouTube, Evernote)
    greedy(Q)

12. Epsilon-Greedy Policies (YouTube, Evernote)
    ε-greedy(Q)

13. MC Control (Evernote)

14. Exploration vs. Exploitation (Evernote)
    Exploration-Exploitation Dilemma
    Greedy in the Limit with Infinite Exploration (GLIE)
    the DQN (deep Q-network) algorithm

15. Incremental Mean (Evernote, YouTube)
    Algorithm 10

16. Constant-alpha (YouTube video 1, video 2, Evernote)
    Algorithm 11

17. Coding Exercise (Evernote, YouTube)

18. Workspace (Colab notebook)

19. Summary (Evernote)

Temporal-Difference Methods (Evernote)

- Learn about how to apply temporal-difference methods such as SARSA, Q-Learning, and Expected SARSA to solve both episodic and continuing tasks.
- Chapter 6 (especially 6.1-6.6) of the textbook

1. Introduction (YouTube)

2. Review: MC Control Methods (Evernote)

3. Quiz: MC Control Methods (Evernote, YouTube video 1, video 2)

4. TD Control: Sarsa (Evernote, YouTube video 1, video 2)
    Algorithm 13 **Sarsa(0)**

5. Quiz: Sarsa (Evernote)

6. TD Control: Q-Learning (YouTube, Evernote)
    Algorithm 14 **Sarsamax (Q-Learning)**

7. Quiz: Q-Learning (Evernote)

8. TD Control: Expected Sarsa (YouTube, Evernote)
    Algorithm 15 **Expected Sarsa**

9. Quiz: Expected Sarsa (Evernote)

10. TD Control: Theory and Practice (Evernote)
    Greedy in the Limit with Infinite Exploration (GLIE), ϵ-greedy

11. OpenAI Gym: CliffWalkingEnv (Evernote)

12. Workspace - Introduction (Evernote)

13. Coding Exercise (Evernote)

14. Workspace (Evernote)

15. Analyzing Performance (Evernote)

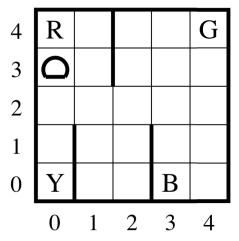
16. Quiz: Check Your Understanding (Evernote)

17. Summary (Evernote)

Solve OpenAI Gym's Taxi-v2 Task

With reinforcement learning now in your toolbox, you're ready to explore a mini project using OpenAI Gym!

1. Introduction (Evernote)
    OpenAI Gym's **'Taxi-v2'** environment
    MAXQ decomposition (**state abstraction,** Hierarchy Of Abstract Machines (HAM) framework, HAMQ, etc.)



2. Instructions (Evernote)

3. Mini Project (Evernote, Colab/GitHub notebook)

RL in Continuous Spaces (Evernote)

Learn how to adapt traditional algorithms to work with continuous spaces.

1. Introducing Arpan (Evernote)

2. Lesson Overview (YouTube)

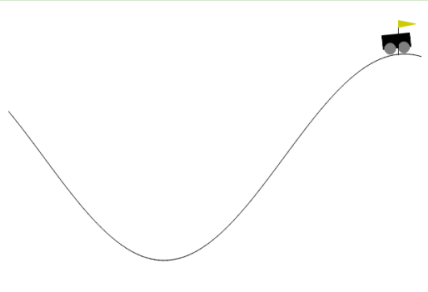
3. Discrete vs. Continuous Spaces (YouTube)

4. Quiz: Space Representations (Evernote)

5. Discretization (YouTube)

6. Exercise: Discretization (Evernote)

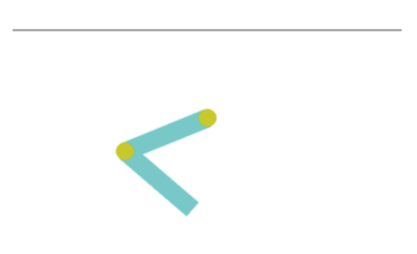
7. Workspace: Discretization (notebooks in Colab, GitHub)
   OpenAI Gym's **'MountainCar-v0'**



8. Tile Coding (YouTube)

9. Exercise: Tile Coding (Evernote)

10. Workspace: Tile Coding (notebook in GitHub, Colab)
    OpenAI Gym's **'Acrobot-v1'**



11. Coarse Coding (YouTube)

12. Function Approximation (YouTube)

13. Linear Function Approximation（YouTube）

14. Kernel Functions (YouTube)
    Radial Basis Functions

15. Non-Linear Function Approximation (YouTube)

16. Summary (Summary)

What's Next?

In the next parts of the Nanodegree program, you'll learn all about how to use neural networks as powerful function approximators in reinforcement learning.

1. Congratulations! (YouTube)

2. What can you do now? (Evernote)

Value-Based Methods

Study Plan

1. Study Plan (Evernote)

2. Deep RL for Robotics (Evernote, YouTube)

Deep Q-Networks

1. From RL to Deep RL (Evernote, YouTube video 1, video 2)
   The Deep Q-Learning algorithm represents the optimal action-value function q_* as a neural network (instead of a table).

2. Deep Q-Networks (Evernote, YouTube)

3. Experience Replay (Evernote, YouTube)

4. Fixed Q-Targets (Evernote, YouTube)

5. Deep Q-Learning Algorithm (Evernote, YouTube)

6. Coding Exercise (Evernote, notebooks on Colab, GitHub)
   OpenAI Gym's '**LunarLander-v2**'

7. Workspace

8. Deep Q-Learning Improvements (Evernote)

9. Double DQN (Evernote, YouTube)

10. Prioritized Experience Replay (Evernote, YouTube)

11. Dueling DQN (Evernote, YouTube)

12. **Rainbow** (Evernote)
    Learning from multi-step bootstrap targets(opens in a new tab)
    Distributional DQN
    Noisy DQN

13. Summary (YouTube)

Navigation

1. Unity ML-Agents (Evernote, GitHub repo ml-agents, repo Udacity DRL)

2. The Environment - Introduction (Evernote)

3. The Environment - Play (Evernote)

4. The Environment - Explore (Evernote)

Download the Unity Environment “**Banana**”

5. Project Instructions (Evernote)

6. Benchmark Implementation (Evernote)

7. Not sure where to start? (Evernote)

8. Collaborate! (Evernote)

9. Workspace (set up "drlnd" env)

10. (Optional) Challenge: Learning from Pixels (Evernote)
    Download the Unity Environment “VisualBanana”

11. Project Rubric (Evernote)

12. Submit Project (Evernote)

Opportunities in Deep Reinforcement Learning

1. Opportunities in DRL (Evernote)

2. Meet the Careers Team (Evernote, YouTube)

3. Access Your Career Portal (Evernote)

Policy-Based Methods

Study Plan (Evernote)

Introduction to Policy-Based Methods

1. Policy-Based Methods (YouTube)

2. Policy Function Approximation (YouTube)

3. More on the Policy (Evernote)
    OpenAI Gym's **MountainCarContinuous-v0**

4. Hill Climbing (YouTube, Evernote)

5. Hill Climbing Pseudocode (YouTube, Evernote)

6. Beyond Hill Climbing (YouTube, Evernote)
    Steepest Ascent Hill Climbing
    Simulated Annealing
    Adaptive Noise Scaling

7. More Black-Box Optimization (Evernote, YouTube)
    Cross-Entropy method
    Evolution strategies

8. Coding Exercise (Evernote)
    CEM.ipynb, Hill_Climbing.ipynb

9. Workspace

10. OpenAI Request for Research (Evernote)
    OpenAI Gym's **CartPole-v0** environment

11. Why Policy-Based Methods? (YouTube)

12. Summary (Evernote)

Policy Gradient Methods

1. What are Policy Gradient Methods? (YouTube, Evernote)

expected return $U(\theta) := \sum_{\tau} P(\tau;\theta) R(\tau)$

2. The Big Picture (YouTube, Evernote)

3. Connections to Supervised Learning (YouTube, Evernote)

4. Problem Setup (YouTube, Evernote)

5. REINFORCE (YouTube, Evernote)

6. (Optional) Derivation (Evernote)

7. Coding Exercise (Evernote)
   REINFORCE.ipynb (GitHub ,Colab)

8. Workspace

9. What's Next? (Evernote)

10. Summary (Evernote)

Proximal Policy Optimization

1. Instructor Introduction (YouTube)

2. Lesson Preview (YouTube, Evernote)
   Trust Region Policy Optimization (TRPO)
   arxiv 1707.06347

3. Beyond REINFORCE (Evernote)

4. Noise Reduction (YouTube, Evernote)
   Rewards Normalization

5. Credit Assignment (YouTube, Evernote)

6. Policy Gradient Quiz (Evernote)

7. pong with REINFORCE (code walkthrough) (YouTube, Evernote)
   OpenAI Gym's **Atari "Pong"** pixel game

8. pong with REINFORCE (workspace) (notebook in GitHub, Colab)

9. Importance Sampling (YouTube, Evernote)

10. PPO part 1- The Surrogate Function (YouTube, Evernote)

11. PPO part 2- Clipping Policy Updates (YouTube, Evernote)

12. PPO summary (YouTube, Evernote)

13. pong with PPO (code walkthrough) (YouTube, Evernote)

14. pong with PPO (workspace) (notebook in GitHub, Colab)

Actor-Critic Methods

1. Introduction

2. Motivation

3. Bias and Variance
    Monte Carlo estimation, unbiased but high variant

4. Two Ways for Estimating Expected Returns

5. Baselines and Critics

6. Policy-based, Value-Based, and Actor-Critic

7. A Basic Actor-Critic Agent

8. **A3C: Asynchronous Advantage Actor-Critic**, N-step (YouTube)
    N-step Bootstrapping (TD: 1-step, Monte Carlo: n-step)

9. A3C: Asynchronous Advantage Actor-Critic, Parallel Training (YouTube)

10. A3C: Asynchronous Advantage Actor-Critic, Off- vs On-policy (YouTube)
    SARSA, A3C: on-policy
    Q-learning, DQN: off-policy

11. **A2C: Advantage Actor-Critic**
    Q-Prop paper, arxiv 1611.02247

12. A2C Code Walk-through (YouTube)

13. GAE: Generalized Advantage Estimation (YouTube)

GAE paper, arxiv 1506.02438

14. DDPG: Deep Deterministic Policy Gradient, Continuous Actions (YouTube, Evernote)
    DDPG paper, arxiv 1509.02971
    Model-based Acceleration (NFA) paper, arxiv 1603.00748

15. DDPG: Deep Deterministic Policy Gradient, Soft Updates (YouTube)

16. DDPG Code Walk-through (YouTube)

17. Summary (YouTube, Evernote)

Deep RL for Finance (Optional)

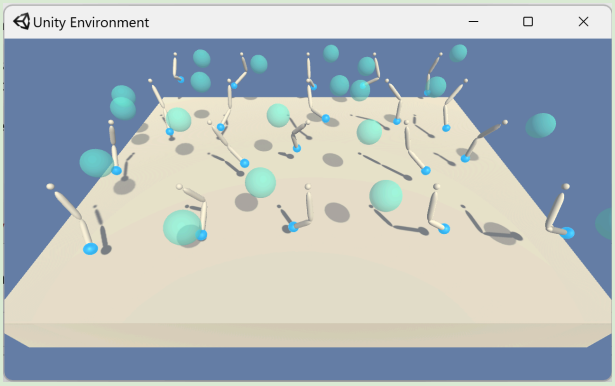
Continuous Control

1. Unity ML-Agents

2. The Environment - Introduction

3. The Environment - Real World

4. The Environment - Explore (YouTube, Evernote)
    Download the Unity Environment "**Reacher-v2**" (GitHub notebook)



5. Project Instructions

6. Benchmark Implementation (Evernote)
    Benchmarking Deep Reinforcement Learning for Continuous Control, arvix 1604.06778
    Trust Region Policy Optimization (TRPO)
    Truncated Natural Policy Gradient (TNPG)
    Distributed Distributional Deterministic Policy Gradients (D4PG)

7. Not sure where to start?

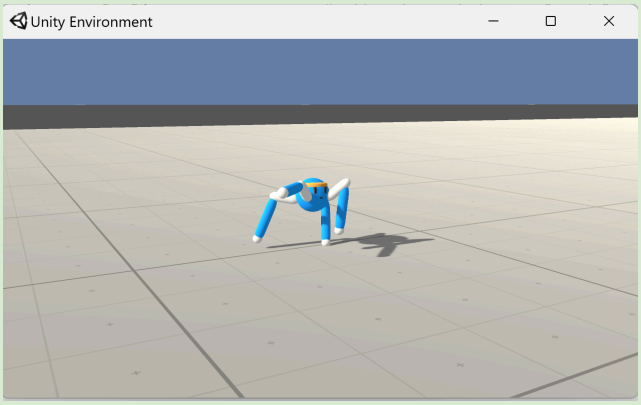
8. General Advice

9. Collaborate!

10. Workspace (Gist)

11. (Optional) Challenge: Crawl (Evernote)

    Download the Unity environment "**Crawl**" (GitHub notebook)

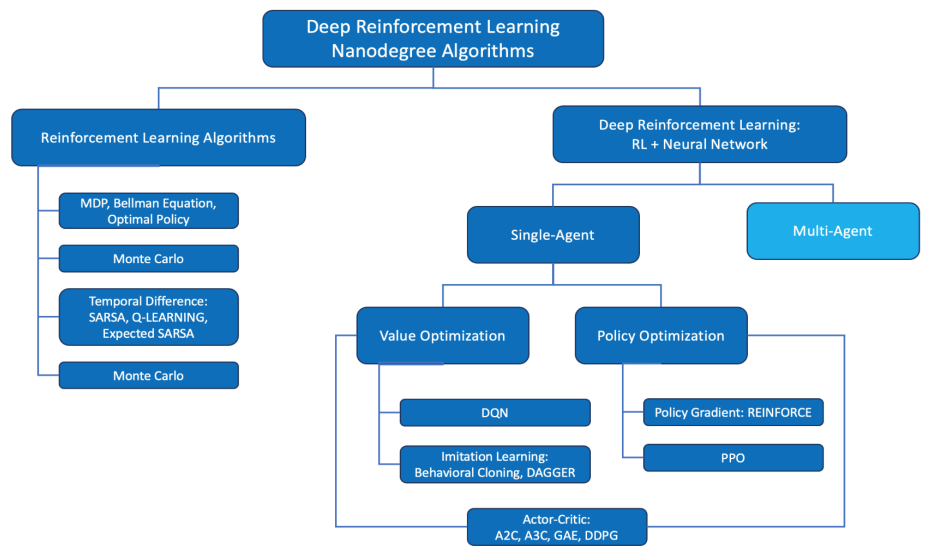


12. Project Rubric

13. Submit Project

Multi-Agent Reinforcement Learning

Study Plan (Evernote)



Introduction to Multi-Agent RL

1. Introducing Chhavi (YouTube)

2. Introduction to Multi-Agent Systems (YouTube)

3. Motivation for Multi-Agent Systems (YouTube)

4. Applications of Multi-Agent Systems (YouTube)

5. Benefits of Multi-Agent Systems (YouTube)

6. Markov Games (YouTube)

7. Markov Games (Evernote)

8. Approaches to MARL (YouTube)
    non-stationarity of the environment
    meta-agent approach
    joint action space

9. Cooperation, Competition, Mixed Environments (YouTube)

10. Research Topics (YouTube)
    OpenAI Five playing Dota 2, “team spirit” hyperparameter

11. Paper Description, Part 1 (Evernote, YouTube)
    OpenAI’s **Multi-agent Particle Environment (MPE)** (GitHub repo)
    “**Physical Deception**” environment
    OpenAI paper, “Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments”

12. Paper Description, Part 2 (YouTube)

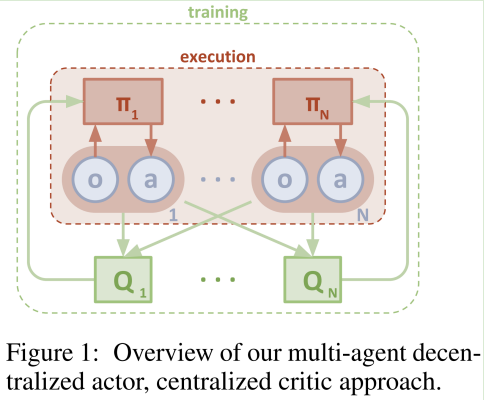


Figure 1: Overview of our multi-agent decentralized actor, centralized critic approach.

13. Summary (YouTube)

14. Lab Instructions (Evernote)
    “Physical Deception”

15. MADDPG - Lab

Case Study: AlphaZero

1. AlphaZero Preview (YouTube, Evernote)

2. Zero-Sum Game (YouTube)

3. **Monte Carlo Tree Search** 1 - Random Sampling (YouTube)

4. Monte Carlo Tree Search 2 - Expansion and Back-propagation (YouTube)

5. AlphaZero 1: Guided Tree Search (YouTube)

6. AlphaZero 2: Self-Play Training

7. TicTacToe using AlphaZero - walkthrough

8. TicTacToe using AlphaZero - Workspace

9. Advanced TicTacToe using AlphaZero

Collaboration and Competition

1. Unity ML-Agents

2. The Environment - Introduction

3. The Environment - Explore (YouTube, Evernote)

4. Project Instructions

5. Benchmark Implementation

6. Collaborate!

7. Workspace

8. (Optional) Challenge: Play Soccer

9. Project Rubric

10. Submit Project

<BOTTOM>

Udacity, reinforcement learning, online course, 2024