Deep Reinforcement Learning Nanodegree Syllabus

Learn the deep reinforcement learning skills that are powering amazing advances in AI.

UDACITY
FOR ENTERPRISE

Table of Contents

Before You Start

Prerequisites: In order to succeed in this program, we recommend having significant experience with Python, and entry-level experience with probability and statistics, and deep learning architectures. Specifically, we expect you to be able to write a class in Python and to add comments to your code for others to read. Also, you should be familiar with the term "neural networks" and understand the differential math that drives backpropagation.

Educational Objectives: In this program, you'll learn the theory and practice driving recent advances in deep reinforcement learning. This program will cover the latest techniques used to create artificially intelligent agents that can solve a variety of complex tasks, with applications ranging from gaming to finance to robotics. With the practical skills you gain in this program, you'll be able to understand the most cutting-edge research papers, and build an impressive portfolio containing your own coding implementations.

Contact Info

While going through the program, if you have questions about anything, you can reach us at enterprise-support@udacity.com. For help from Udacity mentors and peers, please visit the Udacity classroom.

Nanodegree Program Info

Technical Requirements
Hardware Requirements: 64-bit operating system with 8GB of RAM, webcam, microphone
Software and Software Version Requirements: Text editor, Python 3.6, Jupyter Notebooks, Anaconda, PyTorch, Keras, OpenCV
LENGTH OF PROGRAM*: 5 months
FREQUENCY OF CLASSES: Self-paced
TEXTBOOKS REQUIRED: None

*This is a self-paced program and the length is an estimation of total hours the average student may take to complete all required coursework, including lecture and project time. Actual hours may vary.

Project 0: Foundations of Reinforcement Learning

Master the fundamentals of reinforcement learning by writing your own implementations of many classical solution methods.

Supporting Lesson Content: Foundations of Reinforcement Learning

Lesson Title	Learning Outcomes
INTRODUCTION TO REINFORCEMENT LEARNING	• A friendly introduction to reinforcement learning.
THE RL FRAMEWORK: THE PROBLEM	• Learn how to define Markov Decision Processes to solve real-world problems.
THE RL FRAMEWORK: THE SOLUTION	• Learn about policies and value functions. • Derive the Bellman equations.
DYNAMIC PROGRAMMING	• Write your own implementations of iterative policy evaluation, policy improvement, policy iteration, and value iteration.
MONTE CARLO METHODS	• Implement classic Monte Carlo prediction and control methods. • Learn about greedy and epsilon-greedy policies. • Explore solutions to the Exploration-Exploitation Dilemma.
TEMPORAL-DIFFERENCE METHODS	• Learn the difference between the Sarsa, Q-Learning, and Expected Sarsa algorithms.
SOLVE OPENAI GYM'S TAXI V2 TASK	• Design your own algorithm to solve a classical problem from the research community.
RL IN CONTINUOUS SPACES	• Learn how to adapt traditional algorithms to work with continuous spaces.

Project 1: Navigation

Leverage neural networks to train an agent that learns intelligent behaviors from sensory data.

Supporting Lesson Content: Value-based Methods

Lesson Title	Learning Outcomes
DEEP LEARNING IN PYTORCH	• Learn how to build and train neural networks and convolutional neural networks in PyTorch.
DEEP Q-LEARNING	• Extend value-based reinforcement learning methods to complex problems using deep neural networks. • Learn how to implement a Deep Q-Network (DQN), along with Double-DQN, Dueling-DQN, and Prioritized Replay.
DEEP RL FOR ROBOTICS	• Learn from experts at NVIDIA how to use value-based methods in real-world robotics.

Project 2: Continuous Control

Train a robotic arm to reach target locations, or train a four-legged virtual creature to walk.

Supporting Lesson Content: Policy-based methods

Lesson Title	Learning Outcomes
INTRODUCTION TO POLICY-BASED METHODS	• Learn the theory behind evolutionary algorithms, stochastic policy search, and the REINFORCE algorithm. • Learn how to apply the algorithms to solve a classical control problem.

Lesson Content: Policy-based methods (Continued)

Lesson Title	Learning Outcomes
IMPROVING POLICY GRADIENT METHODS	• Learn about techniques such as Generalized Advantage • Estimation (GAE) for lowering the variance of policy gradient methods. • Explore policy optimization methods such as Trust Region Policy Optimization (TRPO) and Proximal Policy Optimization (PPO).
ACTOR-CRITIC METHODS	• Study cutting-edge algorithms such as Deep Deterministic Policy Gradients (DDPG).
DEEP RL FOR FINANCIAL TRADING	• Learn from experts at NVIDIA how to use actor-critic methods to generate optimal financial trading strategies.

Project 3: Collaboration and Competition

Train a system of agents to demonstrate collaboration or cooperation on a complex task.

Supporting Lesson Content: Multi-Agent Reinforcement Learning

Lesson Title	Learning Outcomes
INTRODUCTION TO MULTI-AGENT RL	• Learn how to define Markov games to specify a reinforcement learning task with multiple agents. • Explore how to train agents in collaborative and competitive settings.
CASE STUDY: ALPHAZERO	• Master the skills behind DeepMind’s AlphaZero.

UDACITY
FOR ENTERPRISE

Learn more at www.udacity.com/enterprise