INDIVIDUAL LEARNERS

UDACITY

SCHOOL OF ARTIFICIAL INTELLIGENCE

Deep Reinforcement Learning

Nanodegree Program Syllabus

Overview

The Deep Reinforcement Learning Nanodegree program is designed to enhance students' existing machine learning and deep learning skills with the addition of reinforcement learning theory and programming techniques. This program will grow students' deep learning and reinforcement learning expertise, give them the skills they need to understand the most recent advancements in deep reinforcement learning, and build and implement their own algorithms.

Built in collaboration with:

unity

NVIDIA DEEP LEARNING INSTITUTE

Program information

Estimated Time

4 months at 10-15hrs/week*

Skill Level

Advanced

Prerequisites

A well-prepared learner should have:

- Intermediate to advanced Python experience.
- Familiarity with object-oriented programming.
- Read and understand code.
- Understanding of probability and statistics.
- Intermediate knowledge of machine learning techniques.
- Ability to describe backpropagation and knowledge of neural network architectures (like a CNN for image classification).
- Experience with a deep learning framework like TensorFlow, Keras, or PyTorch.

Required Hardware/Software

Learners need access to a computer running a 64-bit operating system with at least 8GB of RAM, along with administrator account permissions sufficient to install programs including Anaconda with Python 3.6 and supporting packages.

*The length of this program is an estimation of total hours the average student may take to complete all required coursework, including lecture and project time. If you spend about 5-10 hours per week working through the program, you should finish within the time provided. Actual hours may vary.

Course 1

Foundations of Reinforcement Learning

Master the fundamentals of reinforcement learning by writing one's own implementations of many classical solution methods.

Lesson 1

Introduction to RL

- A friendly introduction to reinforcement learning.

Lesson 2

The RL Framework: The Problem

- Learn how to define Markov decision processes to solve real-world problems.

Lesson 3

The RL Framework: The Solution

- Learn about policies and value functions.
- Derive the Bellman equations.

Lesson 4

Dynamic Programming

- Write one's own implementations of iterative policy evaluation, policy improvement, policy iteration, and value iteration.

Lesson 5

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.

Lesson 6

Temporal-Difference Methods

- Learn the difference between the Sarsa, Q-Learning, and Expected Sarsa algorithms.

Lesson 7

Solve openai Gym's Taxi-V2 Task

- Design one's own algorithm to solve a classical problem from the research community.

Lesson 8

RL in Continuous Spaces

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

Course 2

Value-Based Methods

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

Course Project

Navigation

Leverage neural networks to train an agent to navigate a virtual world and collect as many yellow bananas as possible while avoiding blue bananas.

Lesson 1

Deep Learning in PyTorch

- Learn how to build and train neural networks and convolutional neural networks in PyTorch.

Lesson 2

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.

Lesson 3

Deep RL for Robotics

- Learn from experts at NVIDIA how to use value-based methods in real-world robotics.

Course 3

Policy-Based Methods

Learn the theory behind evolutionary algorithms and policy-gradient methods. Design one's own algorithm to train a simulated robotic arm to reach target locations.

Course Project

Continuous Control

Train a robotic arm to reach target locations. For an extra challenge, train a four-legged virtual creature to walk.

Lesson 1

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 2

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).

Lesson 3

Actro-Critic Methods

- Study cutting-edge algorithms such as Deep Deterministic Policy Gradients (DDPG).

Lesson 4

Deep RL for Financial Trading

- Learn from experts at NVIDIA how to use actor-critic methods to generate optimal financial trading strategies.

Course 4

Multi-Agent Reinforcement Learning

Learn how to apply reinforcement learning methods to applications that involve multiple interacting agents. These techniques are used in a variety of applications such as the coordination of autonomous vehicles.

Course Project

Collaboration & Competition

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

Lesson 1

Introduction 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.

Lesson 2

Case Study: Alphazera

- Master the skills behind DeepMind's AlphaZero.

Meet your instructors.

Alexis Cook

Curriculum Lead

Alexis is an applied mathematician with a master’s in computer science from Brown University and a master’s in applied mathematics from the University of Michigan. She was formerly a National Science Foundation Graduate Research Fellow.

Arpan Chakraborty

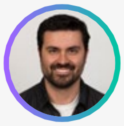
Computer Scientist

Arpan is a computer scientist with a PhD from North Carolina State University. He teaches at Georgia Tech (within the Master of Computer Science program), and is a coauthor of the book Practical Graph Mining with R.

Mat Leonard

Instructor

Mat is a former physicist, research neuroscientist, and data scientist. He completed his PhD and postdoctoral fellowship at the University of California, Berkeley.



Luis Serrano

Instructor

Luis was formerly a machine learning engineer at Google. He holds a PhD in mathematics from the University of Michigan and a postdoctoral fellowship at the University of Quebec at Montreal.

Cezanne Camacho

Curriculum Lead

Cezanne is an expert in computer vision with a master's in electrical engineering from Stanford University. As a former researcher in genomics and biomedical imaging, she's applied computer vision and deep learning to medical diagnostic applications.

Dana Sheahen

Electrical Engineer

Dana is an electrical engineer with a master's in computer science from Georgia Tech. Her work experience includes software development for embedded systems in the automotive group at Motorola, where she was awarded a patent for an onboard operating system.

Chhavi Yadav

Content Developer

Chhavi is a computer science graduate student at New York University where she researches machine learning algorithms. She is also an electronics engineer and has worked on wireless systems.

Juan Delgado

Computational Physicist

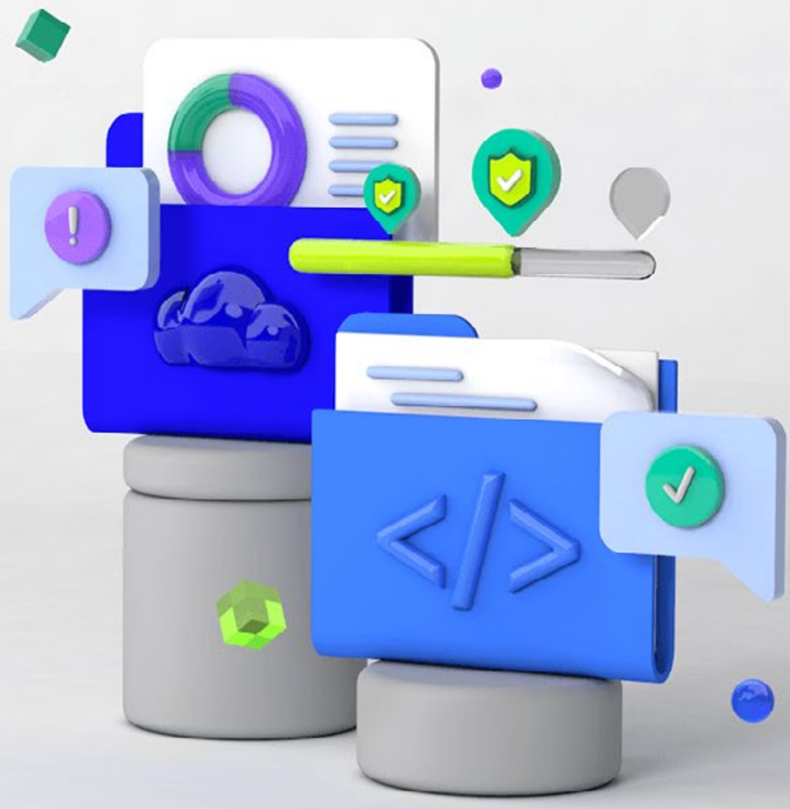
Juan is a computational physicist with a master's in astronomy. He is finishing his PhD in biophysics. He previously worked at NASA developing space instruments and writing software to analyze large amounts of scientific data using machine learning techniques.

Miguel Morales

Content Developer

Miguel is a software engineer at Lockheed Martin. He earned a master's in computer science at Georgia Tech and is an instructional associate for the Reinforcement Learning and Decision Making course. He's the author of Grokking Deep Reinforcement Learning.

Udacity’s learning experience



Hands-on Projects

Open-ended, experiential projects are designed to reflect actual workplace challenges. They aren’t just multiple choice questions or step-by-step guides, but instead require critical thinking.

Quizzes

Auto-graded quizzes strengthen comprehension. Learners can return to lessons at any time during the course to refresh concepts.

Knowledge

Find answers to your questions with Knowledge, our proprietary wiki. Search questions asked by other students, connect with technical mentors, and discover how to solve the challenges that you encounter.

Custom Study Plans

Create a personalized study plan that fits your individual needs. Utilize this plan to keep track of movement toward your overall goal.

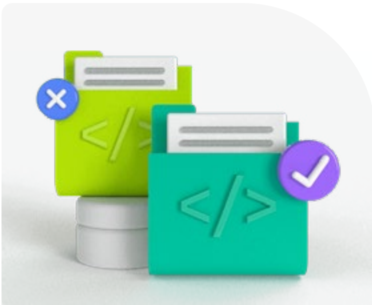
Workspaces

See your code in action. Check the output and quality of your code by running it on interactive workspaces that are integrated into the platform.

Progress Tracker

Take advantage of milestone reminders to stay on schedule and complete your program.

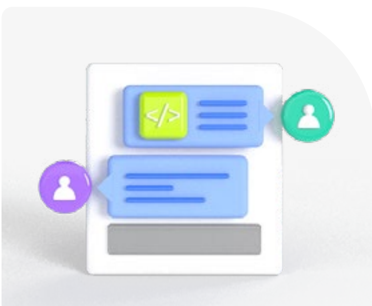
Our proven approach for building job-ready digital skills.



Experienced Project Reviewers

Verify skills mastery.

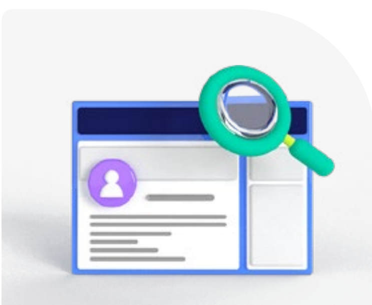
- Personalized project feedback and critique includes line-by-line code review from skilled practitioners with an average turnaround time of 1.1 hours.
- Project review cycle creates a feedback loop with multiple opportunities for improvement—until the concept is mastered.
- Project reviewers leverage industry best practices and provide pro tips.



Technical Mentor Support

24/7 support unblocks learning.

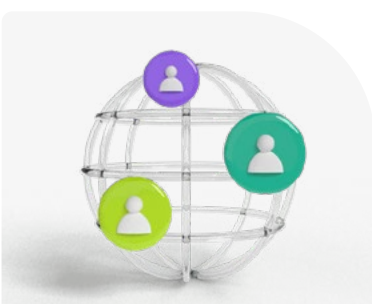
- Learning accelerates as skilled mentors identify areas of achievement and potential for growth.
- Unlimited access to mentors means help arrives when it's needed most.
- 2 hr or less average question response time assures that skills development stays on track.



Personal Career Services

Empower job-readiness.

- Access to a Github portfolio review that can give you an edge by highlighting your strengths, and demonstrating your value to employers.*
- Get help optimizing your LinkedIn and establishing your personal brand so your profile ranks higher in searches by recruiters and hiring managers.



Mentor Network

Highly vetted for effectiveness.

- Mentors must complete a 5-step hiring process to join Udacity's selective network.
- After passing an objective and situational assessment, mentors must demonstrate communication and behavioral fit for a mentorship role.
- Mentors work across more than 30 different industries and often complete a Nanodegree program themselves.

*Applies to select Nanodegree programs only.

UDACITY

Learn more at

www.udacity.com/online-learning-for-individuals →

12.22.22 | V1.0