

Machine Learning Engineer (Advanced) Syllabus



Contact Info

While going through the program, if you have questions about anything, you can reach us at support@udacity.com. For help from Udacity Mentors and your peers visit the Udacity Classroom.

Nanodegree Program Info

Version: 6.0.0

Length of Program: 188 Days*

** 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.*

Part 1: Welcome to Machine Learning Engineer

Project: Titanic Survival Exploration

In this optional project, you will create decision functions that attempt to predict survival outcomes from the 1912 Titanic disaster based on each passenger's features, such as sex and age.

Supporting Lessons

Lesson	Summary
Program Orientation	How does a Nanodegree program work? This short lesson explains the process and best practices for being successful.
Welcome to Machine Learning	Welcome to the Machine Learning Nanodegree program!

Part 2: Machine Learning Foundations

Project: Predicting Boston Housing Prices

Part 3: Supervised Learning

Project: Finding Donors for CharityML

You've covered a wide variety of methods for performing supervised learning -- now it's time to put those into action!

Part 4: Unsupervised Learning

Project: Creating Customer Segments

Now that you've learned a lot about unsupervised learning, it's time to apply that to a project.

Part 5: Reinforcement Learning

Project: Q-learning Maze

We will use Q-learning algorithm to solve an typical MDP problem --- maze!

Supporting Lessons

Lesson	Summary
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.
The RL Framework: The Solution	In reinforcement learning, agents learn to prioritize different decisions based on the rewards and punishments associated with different outcomes.
Dynamic Programming	The dynamic programming setting is a useful first step towards tackling the reinforcement learning problem.
Monte Carlo Methods	Write your own implementation of Monte Carlo control to teach an agent to play Blackjack!
Temporal-Difference Methods	Learn about how to apply temporal-difference methods such as Sarsa, Q-Learning, and Expected Sarsa to solve both episodic and continuous tasks.
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!

Part 6: Deep Learning

Project: Dog Breed Classifier

In this project, you will learn how to build a pipeline to process real-world, user-supplied images. Given an image of a dog, your algorithm will identify an estimate of the canine's breed.

Part 7: Machine Learning Capstone

Project: Capstone Proposal

Before working on a machine learning problem, write up a proposal of your project to get valuable feedback!

Project: Capstone Project

Now you will put your Machine Learning skills to the test by solving a real world problem using the algorithms you have learned in the program so far.



Udacity

Generated Thu Oct 17 22:14:33 PDT 2019