



NANODEGREE PROGRAM SYLLABUS

Machine Learning Engineer



Overview

This goal of the Machine Learning Engineer Nanodegree program is to help students learn the key skills they need to perform well as a machine learning engineer.

A graduate of this program will be able to:

- Test Python code and build a Python package of their own.
- Build predictive models using a variety of unsupervised and supervised machine learning techniques.
- Use Amazon SageMaker to deploy machine learning models to production environments, such as a web application or piece of hardware..
- A/B test two different deployed models and evaluate their performance.
- Utilize an API to deploy a model to a website such that it responds to user input, dynamically.
- Update a deployed model, in response to changes in the underlying data source.

This program is comprised of 4 courses and 4 projects. Each project you build will be an opportunity to demonstrate what you've learned in the lessons. **Your completed projects will become part of a career portfolio that will demonstrate to potential employers that you have skills in feature engineering, building machine learning algorithms, and model deployment.**

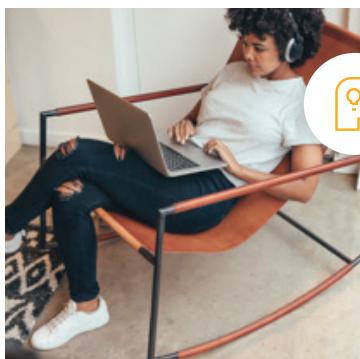
IN COLLABORATION WITH



Estimated Time:
3 Months at
10 hrs/week



Prerequisites:
Intermediate
Python &
Machine Learning
Algorithms



Flexible Learning:
Self-paced, so
you can learn on
the schedule that
works best for you



Need Help?
udacity.com/advisor
Discuss this program
with an enrollment
advisor.

Projects Overview

One of our main goals at Udacity is to help you **create a job-ready portfolio of completed projects**.

Building a project is one of the best ways to test the skills you've acquired and to demonstrate your newfound abilities to future employers or colleagues. Throughout this Nanodegree program, you'll have the opportunity to prove your skills by building the following projects:

- **Build a Python Package:** Write a Python package on your own using software engineering best practices for writing production level code. This project is optional and will not be graded.
- **Deploy a Sentiment Analysis Model:** Using SageMaker, deploy your own PyTorch sentiment analysis model, which is trained to recognize the sentiment of movie reviews (positive or negative).
- **Plagiarism Detector:** Engineer features that can help identify cases of plagiarism in text and deploy a trained plagiarism detection model using Amazon SageMaker.
- **Capstone Project & Proposal:** Complete a final project—choosing from a few, provided options or a project of your own design—that involves data exploration and machine learning.

In the sections below, you'll find detailed descriptions of each project along with the course material that presents the skills required to complete the project.

Course 1: Software Engineering Fundamentals

In this lesson, you'll write production-level code and practice object-oriented programming, which you can integrate into machine learning projects.

Course Project

Build a Python Package

This program is all about understanding how to build machine learning algorithms and prepare them for scalable, production systems. A first step towards building these systems is to gain an understanding of writing production level code, which you have the option of doing by writing a Python package of your own.

Key Skills Demonstrated:

- Object-oriented programming
- Clean and modular code
- Code documentation

LEARNING OUTCOMES

LESSON ONE

Software Engineering Practices

- Write clean, modular, and well-documented code
- Refactor code for efficiency
- Create unit tests to test programs
- Track actions and results of processes with logging
- Conduct and receive code reviews

LESSON TWO

Programming

- Understand when to use object-oriented programming
- Build and use classes
- Learn how large, modular Python packages and use object-oriented programming

LESSON THREE

Upload a Package to PyPI

- Portfolio Exercise: Build your own Python package.

Course 2: Machine Learning in Production

Learn how to deploy machine learning models to a production environment using Amazon SageMaker.

Course Project

Deploy a Sentiment Analysis Model

In this project, you will be provided a dataset reflecting data collected from an experiment. You'll use statistical techniques to answer questions about the data and report your conclusions and recommendations in a report.

Key Skills Demonstrated:

- Text analysis
- Model deployment via SageMaker
- APIs for web deployment

LEARNING OUTCOMES

LESSON ONE

Introduction to Deployment

- Gain familiarity with cloud and deployment terminology
- Understand the machine learning workflow in production
- Learn about workplace use cases of machine learning

LESSON TWO

Deploy a Model

- Deploy a model within SageMaker.
- Predict housing prices in Boston using XGBoost on SageMaker.
- Determine movie review sentiment using XGBoost on SageMaker.

LESSON THREE

Web Hosting

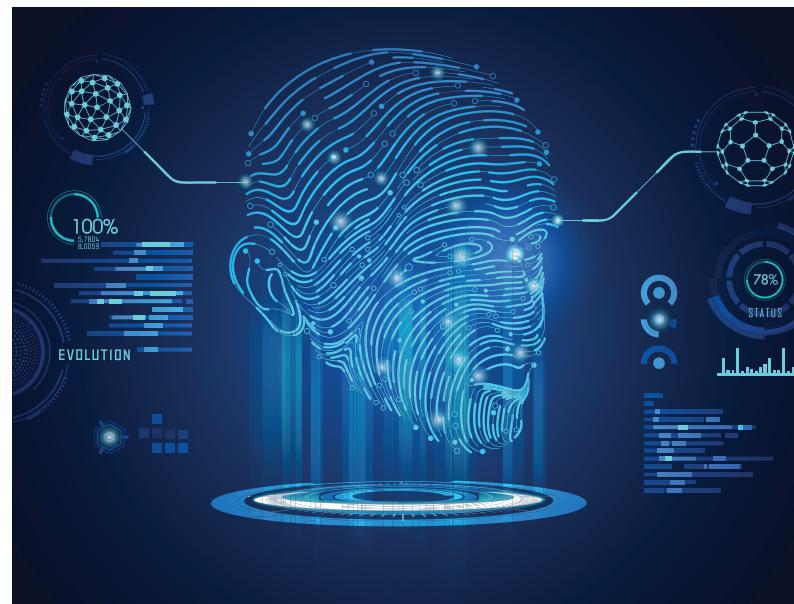
- Learn to provide access to an endpoint from a website
- Use API Gateway and Lambda to integrate ML models into a web app

LESSON FOUR**Model Monitoring**

- Learn how to monitor the behavior of your models over time.
- Tune hyperparameters of an XGBoost model using SageMaker's automatic hyperparameter tuning tools.
- Run an A/B test on SageMaker to compare the tuned model to the untuned model.

LESSON FIVE**Updating a Model**

- Update your model to account for changes in the data that were discovered during model monitoring
- Explore how to handle new phrases introduced to your model during your sentiment analysis



Course 3: Machine Learning Case Studies

Apply machine learning techniques to solve real-world tasks; explore data and deploy both built-in and custom-made Amazon SageMaker models.

Course Project

Deploy a Sentiment Analysis Model

Use your machine learning skills to compare two text sources and identify cases of plagiarism. In this project, you will extract relevant text features and train a model of your own design to do plagiarism detection. Then, you will deploy your trained model using Amazon SageMaker.

Key Skills Demonstrated:

- Feature engineering
- Model design and evaluation
- Model deployment via SageMaker

LEARNING OUTCOMES

LESSON ONE

Population Segmentation with SageMaker

- Learn the breadth of algorithms available using AWS SageMaker.
- Understand how you can use unsupervised algorithms to analyze data with SageMaker.
- Deploy an unsupervised model using SageMaker
- Draw insights about your data by extracting model attributes.

LESSON TWO

Detecting Credit Card Fraud

- Build and improve a linear model to identify cases of payment fraud.
- Handle cases of class imbalance in the training data.
- Tune a model in SageMaker to improve its performance according to a specific metric.

LESSON THREE

Deploying Custom Models

- Deploy a custom PyTorch model using SageMaker
- Write a custom training script to train a model of your own design.

LESSON FOUR

Time-Series Forecasting

- Process time-series data and format it for training a machine learning model.
- Use SageMaker's DeepAR algorithm for time-series forecasting
- Deploy a model and use it to predict future data points.

Course 4: Machine Learning Capstone

In this capstone lesson, you'll select a machine learning challenge and propose a possible solution.

Course Project

Capstone Proposal and Project

In this capstone project, you will leverage what you've learned throughout the program to build a machine learning project of your choosing. You will define the problem you want to solve, investigate and explore the data, identify and explore the data, then perform your analyses and develop a set of conclusions. You will present the analysis and your conclusions in a blog post and GitHub repository. This project will serve as a demonstration of your ability as a machine learning engineer, and will be an important piece of your job ready portfolio.

Key Skills Demonstrated:

- All of the machine learning workflow, from data exploration to model training and evaluation

LEARNING OUTCOMES

LESSON ONE

Elective 1: Starbucks

- Use purchasing habits to arrive at discount measures to obtain and retain customers.
- Identify groups of individuals that are most likely to be responsive to rebates

LESSON TWO

Elective 2: Arvato Financial Services

- Work through a real-world dataset and challenge provided by Arvato Financial Services, a Bertelsmann company.
- Top performers have a chance at an interview with Arvato or another Bertelsmann company!

LESSON THREE

Elective 3: Convolutional Neural Network

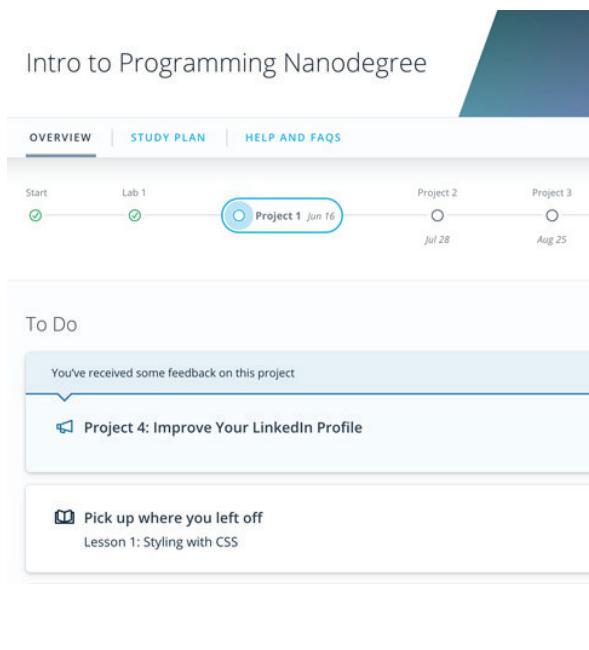
- Complete a project to identify dog breeds based on images.

LESSON FOUR

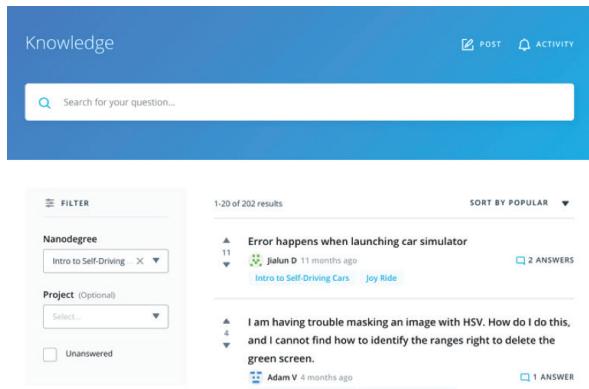
Elective 4: Your Choice

- Build a new project entirely of your own choosing.

Our Classroom Experience

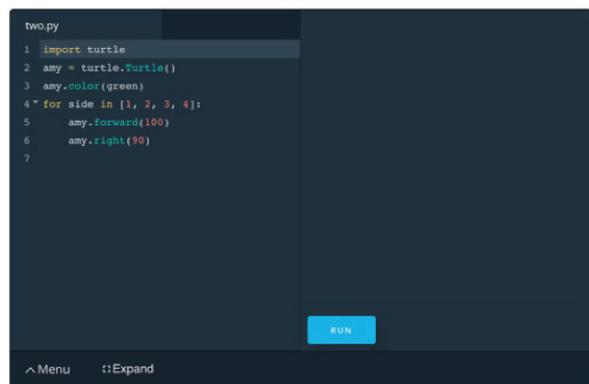


The screenshot shows the Udacity Intro to Programming Nanodegree dashboard. At the top, it says "Intro to Programming Nanodegree". Below that is a navigation bar with "OVERVIEW", "STUDY PLAN", and "HELP AND FAQS". The "STUDY PLAN" tab is selected. It shows a timeline from "Start" to "Aug 25" with milestones: "Lab 1" (Jun 16), "Project 1" (Jun 16), "Project 2" (Jul 28), and "Project 3" (Aug 25). A callout highlights "Project 1 Jun 16". Below the timeline is a "To Do" section with a message: "You've received some feedback on this project". It lists "Project 4: Improve Your LinkedIn Profile". Under "To Do", there's a "Pick up where you left off" section for "Lesson 1: Styling with CSS".



The screenshot shows the "Knowledge" section of the Udacity dashboard. At the top, it says "Knowledge". Below that is a search bar with the placeholder "Search for your question...". On the right, there are "POST" and "ACTIVITY" buttons. The main area displays a list of 202 results, sorted by popularity. The first two questions are:

- Error happens when launching car simulator** (11 answers) posted by Jialun D 11 months ago. Tags: Intro to Self-Driving Cars, Joy Ride.
- I am having trouble masking an image with HSV. How do I do this, and I cannot find how to identify the ranges right to delete the green screen.** (1 answer) posted by Adam V 4 months ago. Tags: Intro to Self-Driving Cars, Traffic Light Classifier.



The screenshot shows a workspace interface. On the left, a code editor window titled "two.py" contains the following Python code:

```

two.py
1 import turtle
2 amy = turtle.Turtle()
3 amy.color('green')
4 for side in [1, 2, 3, 4]:
5     amy.forward(100)
6     amy.right(90)
7

```

At the bottom of the code editor is a "RUN" button. At the very bottom of the workspace are "Menu" and "Expand" buttons.

REAL-WORLD PROJECTS

Build your skills through industry-relevant projects. Get personalized feedback from our network of 900+ project reviewers. Our simple interface makes it easy to submit your projects as often as you need and receive unlimited feedback on your work.

KNOWLEDGE

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

STUDENT HUB

Leverage the power of community through a simple, yet powerful chat interface built within the classroom. Use Student Hub to connect with your fellow students in your Executive Program.

WORKSPACES

See your code in action. Check the output and quality of your code by running them on workspaces that are a part of our classroom.

QUIZZES

Check your understanding of concepts learned in the program by answering simple and auto-graded quizzes. Easily go back to the lessons to brush up on concepts anytime you get an answer wrong.

CUSTOM STUDY PLANS

Preschedule your study times and save them to your personal calendar to create a custom study plan. Program regular reminders to keep track of your progress toward your goals and completion of your program.

PROGRESS TRACKER

Stay on track to complete your Nanodegree program with useful milestone reminders.

Learn with the Best



Cezanne Camacho

INSTRUCTOR

Cezanne is a machine learning educator with a Masters in Electrical Engineering from Stanford University. As a former researcher in genomics and biomedical imaging, she's applied machine learning to medical diagnostic applications.



Mat Leonard

INSTRUCTOR

Mat is a former physicist, research neuroscientist, and data scientist. He did 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.



Dan Romuald Mbanga

INSTRUCTOR

Dan leads Amazon AI's Business Development efforts for Machine Learning Services. Day to day, he works with customers—from startups to enterprises—to ensure they are successful at building and deploying models on Amazon SageMaker.vvv

Learn with the Best



Jennifer Staab
INSTRUCTOR

Jennifer has a PhD in Computer Science and a Masters in Biostatistics; she was a professor at Florida Polytechnic University. She previously worked at RTI International and United Therapeutics as a statistician and computer scientist.



Sean Carrell
INSTRUCTOR

Sean Carrell is a former research mathematician specializing in Algebraic Combinatorics. He completed his PhD and Postdoctoral Fellowship at the University of Waterloo, Canada.



Josh Bernhard
DATA SCIENTIST

Josh has been sharing his passion for data for nearly a decade at all levels of university, and as Lead Data Science Instructor at Galvanize. He's used data science for work ranging from cancer research to process automation.



Jay Alamar
INSTRUCTOR

Jay has a degree in computer science, loves visualizing machine learning concepts, and is the Investment Principal at STV, a \$500 million venture capital fund focused on high-technology startups.

All Our Nanodegree Programs Include:



EXPERIENCED PROJECT REVIEWERS

REVIEWER SERVICES

- Personalized feedback & line by line code reviews
- 1600+ Reviewers with a 4.85/5 average rating
- 3 hour average project review turnaround time
- Unlimited submissions and feedback loops
- Practical tips and industry best practices
- Additional suggested resources to improve



TECHNICAL MENTOR SUPPORT

MENTORSHIP SERVICES

- Questions answered quickly by our team of technical mentors
- 1000+ Mentors with a 4.7/5 average rating
- Support for all your technical questions



PERSONAL CAREER SERVICES

CAREER COACHING

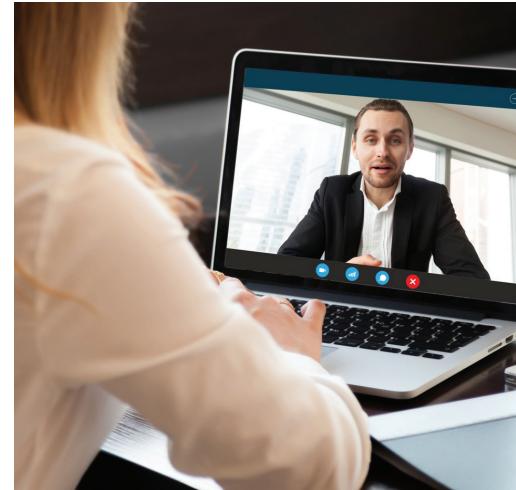
- Personal assistance in your job search
- Monthly 1-on-1 calls
- Personalized feedback and career guidance
- Interview preparation
- Resume services
- Github portfolio review
- LinkedIn profile optimization

Frequently Asked Questions

PROGRAM OVERVIEW

WHY SHOULD I ENROLL?

As more and more companies are looking to build machine learning products, there is a growing demand for engineers who are able to deploy machine learning models to global audiences. In this program, you'll learn how to create an end-to-end machine learning product. You'll deploy machine learning models to a production environment, such as a web application, and evaluate and update that model according to performance metrics. This program is designed to give you the advanced skills you need to become a machine learning engineer.



WHAT JOBS WILL THIS PROGRAM PREPARE ME FOR?

Students in the Machine Learning Engineer Nanodegree program will learn about machine learning algorithms and crucial deployment techniques, and will be equipped to fill roles at companies seeking machine learning engineers and specialists. These skills can also be applied in roles at companies that are looking for data scientists to introduce machine learning techniques into their organization.

HOW DO I KNOW IF THIS PROGRAM IS RIGHT FOR ME?

This program assumes that you are familiar with common supervised and unsupervised machine learning techniques. As such, it is geared towards people who are interested in building and deploying a machine learning product or application. Are you interested in deploying an application that is powered by machine learning? If so, then this program is right for you.

ENROLLMENT AND ADMISSION

DO I NEED TO APPLY? WHAT ARE THE ADMISSION CRITERIA?

No. This Nanodegree program accepts all applicants regardless of experience and specific background.

WHAT ARE THE PREREQUISITES FOR ENROLLMENT?

Intermediate Python programming knowledge, including:

- At least 40hrs of programming experience
- Familiarity with data structures like dictionaries and lists
- Experience with libraries like NumPy and pandas

Intermediate knowledge of machine learning algorithms, including:

- Supervised learning models, such as linear regression
- Unsupervised models, such as k-means clustering
- Deep learning models, such as neural networks

IF I DO NOT MEET THE REQUIREMENTS TO ENROLL, WHAT SHOULD I DO?

To succeed in this program, you are expected to know foundational machine

FAQs Continued

learning algorithms. If you'd like to learn more about common unsupervised and supervised techniques, it is suggested that you take the [Intro to Machine Learning Nanodegree](#) program.

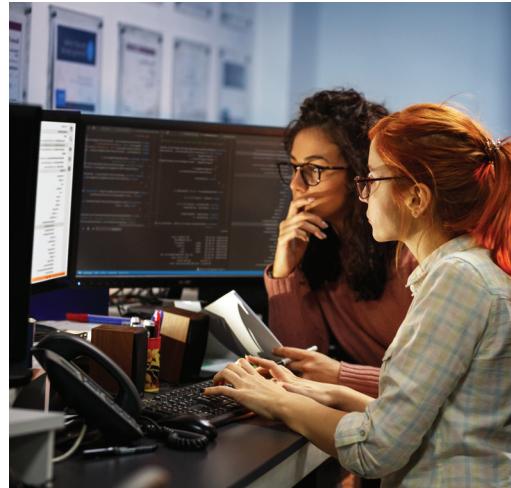
DO I HAVE TO TAKE THE INTRO TO MACHINE LEARNING NANODEGREE PROGRAM BEFORE ENROLLING IN THE MACHINE LEARNING ENGINEER NANODEGREE PROGRAM?

No. Each program is independent of the other. If you are interested in machine learning, you should look at the prerequisites for each program to help you decide where you should start your journey to becoming a machine learning engineer.

TUITION AND TERM OF PROGRAM

HOW IS THIS NANODEGREE PROGRAM STRUCTURED?

The Machine Learning Engineer Nanodegree program is comprised of content and curriculum to support four (4) projects. We estimate that students can complete the program in three (3) months, working 10 hours per week.



HOW LONG IS THIS NANODEGREE PROGRAM?

Access to this Nanodegree program runs for the length of time specified in your subscription plan. See the [Terms of Use](#) for other policies around the terms of access to our Nanodegree programs.

CAN I SWITCH MY START DATE? CAN I GET A REFUND?

Please see the Udacity Nanodegree program [FAQs](#) for policies on enrollment in our programs.

I HAVE GRADUATED FROM THE MACHINE LEARNING ENGINEER NANODEGREE PROGRAM, BUT I WANT TO KEEP LEARNING. WHERE SHOULD I GO FROM HERE?

Many of our graduates continue on to our [Artificial Intelligence](#) and [Self-Driving Car Engineer Nanodegree](#) programs.

SOFTWARE AND HARDWARE

WHAT SOFTWARE AND VERSIONS WILL I NEED IN THIS PROGRAM?

You will need 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.x and supporting packages.

Most modern Windows, OS X, and Linux laptops or desktop will work well; we do not recommend a tablet since they typically have less computing power. We will provide you with instructions to install the required software packages.