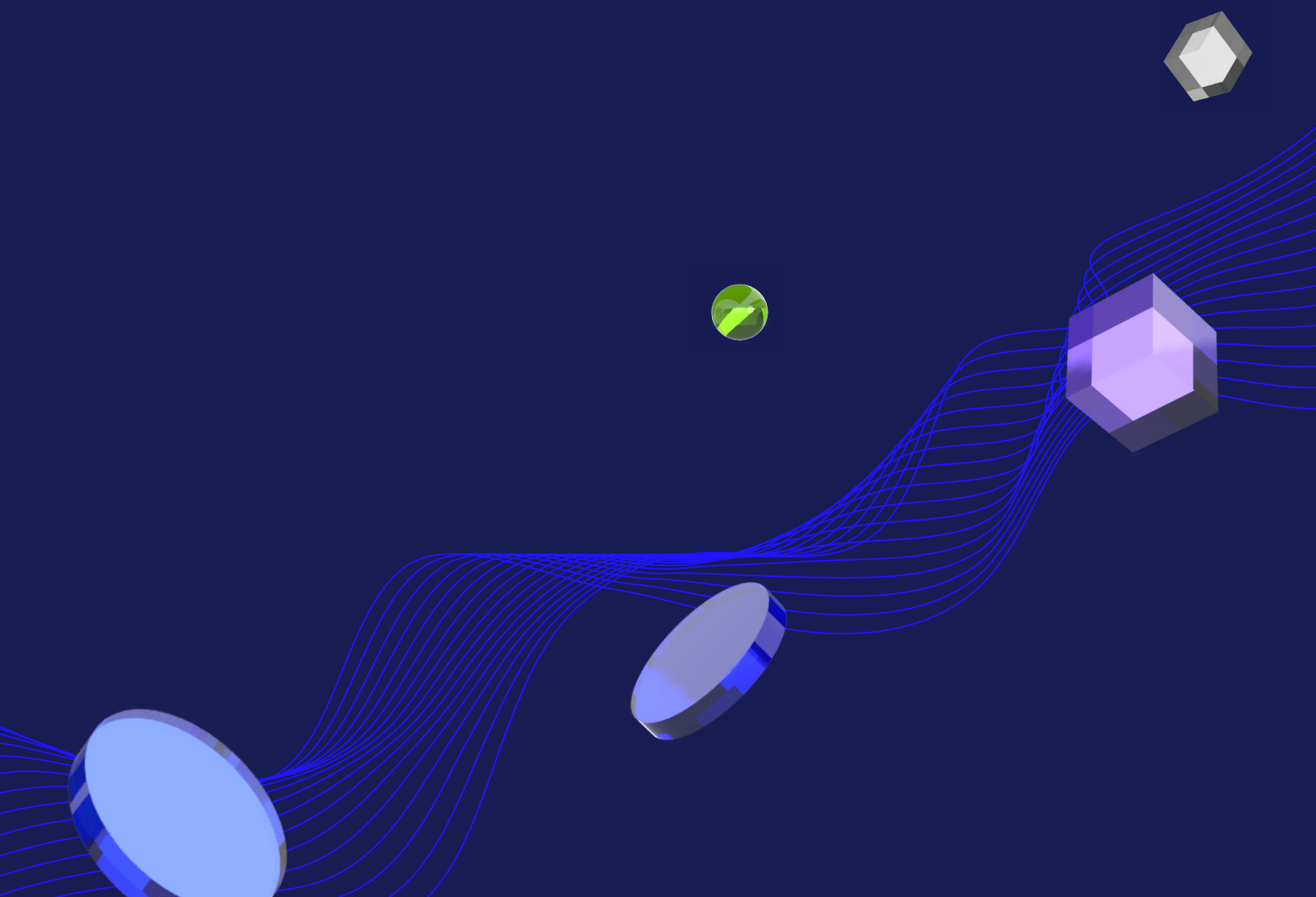
INDIVIDUAL LEARNERS

UDACITY

SCHOOL OF ARTIFICIAL INTELLIGENCE

Machine Learning DevOps Engineer

Nanodegree Program Syllabus



Overview

This program focuses on the software engineering fundamentals needed to successfully streamline the deployment of data and machine learning models in a production-level environment. Learners will build the DevOps skills required to automate the various aspects and stages of machine learning model building and monitoring over time.

Learning Objectives

A graduate of this program will be able to:

- Implement production-ready Python code/processes for deploying ML models outside of cloud-based environments facilitated by tools such as AWS SageMaker, Azure ML, etc.
- Engineer automated data workflows that perform continuous training (CT) and model validation within a CI/CD pipeline based on updated data versioning.
- Create multi-step pipelines that automatically retrain and deploy models after data updates.
- Track model summary statistics and monitor model online performance over time to prevent model-degradation.

Program information

Estimated Time

4 months at 10hrs/week*

Skill Level

Advanced

Prerequisites

A well-prepared learner should have prior experience with Python and machine learning.

Required Hardware/Software

Learners need access to a 64-bit computer, at least 8GB of RAM, and administrator account permissions sufficient to install programs including Anaconda with Python 3.x 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

Clean Code Principles

Develop skills that are essential for deploying production machine learning models. First, learners will put coding best practices on autopilot by learning how to use PyLint and AutoPEP8. Then they will further expand their Git and Github skills to work with teams. Finally, they will learn best practices associated with testing and logging used in production settings in order to ensure their models can stand the test of time.

Course Project

Predict Customer Churn with Clean Code

In this project, learners will implement their learnings to identify credit card customers that are most likely to churn. The completed project will include a Python package for a machine learning project that follows coding (PEP8) and engineering best practices for implementing software (modular, documented, and tested). The package will also have the flexibility of being run interactively or from the command-line interface (CLI). This project will give learners practice using their skills for testing, logging, and coding best practices from the lessons. It will also introduce them to a problem data scientists across companies face all the time: How do we identify (and later intervene with) customers who are likely to churn?

Lesson 1

Coding Best Practices

- Write clean, modular, and well-documented code.
- Refactor code for efficiency.
- Follow PEP8 standards.
- Automate use of PEP8 standards using PyLint and Auto PEP8.

Lesson 2

Working with Others Using Version Control

- Work independently using Git and Github.
- Work with teams using Git and Github.
- Create branches for isolating changes in Git and Github.
- Open pull requests for making changes to production code.
- Conduct and receive code reviews using best practices.

Lesson 3

Production Ready Code

- Correctly use try-except blocks to identify errors.
- Create unit tests to test programs.
- Track actions and results of processes with logging.
- Identify model drift and when automated or non-automated retraining should be used to make model updates.

Course 2

Building a Reproducible Model Workflow

This course empowers the learners to be more efficient, effective, and productive in modern, real-world ML projects by adopting best practices around reproducible workflows. In particular, it teaches the fundamentals of MLops and how to: a) create a clean, organized, reproducible, end-to-end machine learning pipeline from scratch using MLflow b) clean and validate the data using pytest c) track experiments, code, and results using GitHub and Weights & Biases d) select the best-performing model for production and e) deploy a model using MLflow. Along the way, it also touches on other technologies like Kubernetes, Kubeflow, and Great Expectations and how they relate to the content of the class.

Course Project

Build an ML Pipeline for Short-Term Rental Prices in NYC

Learners will write a machine learning pipeline to solve the following problem: A property management company is renting rooms and properties in New York for short periods on various rental platforms. They need to estimate the typical price for a given property based on the price of similar properties. The company receives new data in bulk every week, so the model needs to be retrained with the same cadence, necessitating a reusable pipeline. The students will write an end-to-end pipeline covering data fetching, validation, segregation, train and validation, test, and release. They will run it on an initial data sample, and then re-run it on a new data sample simulating a new data delivery.

Lesson 1

Machine Learning Pipelines

- Learn MLOps fundamentals.
- Version data and artifacts.
- Write a ML pipeline component.
- Link together ML components.

Lesson 2

Data Exploration & Preparation

- Execute and track the exploratory data analysis (EDA).
- Clean and preprocess the data.
- Segregate (split) datasets.

Lesson 3

Data Validation

- Use pytest with parameters for reproducible and automatic data tests.
- Perform deterministic and non-deterministic data tests.

Lesson 4

Training, Validation & Experiment Tracking

- Tame the chaos with experiment, code, and data tracking.
- Track experiments with W&B.
- Validate and choose best-performing model.
- Export model as an inference artifact.
- Test final inference artifact.

Lesson 5

Release & Deploy

- Release pipeline code.
- Options for deployment and how to deploy a model.

Course 3

Deploying a Scalable ML Pipeline in Production

This course teaches learners how to robustly deploy a machine learning model into production. En route to that goal they will learn how to put the finishing touches on a model by taking a fine grained approach to model performance, checking bias, and ultimately writing a model card. They will also learn how to version control their data and models using data version control (DVC). The last piece in preparation for deployment will be learning continuous integration and continuous deployment which will be accomplished using GitHub Actions and Heroku, respectively. Finally, learn how to write a fast, type-checked, and auto-documented API using FastAPI.

Course Project

Deploying a Machine Learning Model on Heroku with FastAPI

In this project, learners will deploy a machine learning model on Heroku. The learners will use Git and DVC to track their code, data, and model while developing a simple classification model on the Census Income dataset. After developing the model the learners will finalize the model for production by checking its performance on slices and writing a model card encapsulating key knowledge about the model. They will put together a continuous integration and continuous deployment framework and ensure their pipeline passes a series of unit tests before deployment. Lastly, an API will be written using FastAPI and will be tested locally. After successful deployment the API will be tested live using the requests module.

After completion, the learner will have a working API that is live in production, a set of tests, model card, and full CI/CD framework. On its own, this project can be used as a portfolio piece, but also any of the constituent pieces can be applied to other projects, e.g. continuous integration, to further flesh them out.

Lesson 1

Performance Testing & Preparing a Model for Production

- Analyze slices of data when training and testing models.
- Probe a model for bias using common frameworks such as Aequitas.
- Write model cards that explain the purpose, provenance, and pitfalls of a model.

Lesson 2

Data & Model Versioning

- Version control data/models/etc locally using DVC.
- Set up remote storage for use with DVC.
- Create pipelines and track experiments with DVC.

Lesson 3

CI/CD

- Follow software engineering principles by automating, testing, and versioning code.
- Set up continuous integration using GitHub Actions.
- Set up continuous deployment using Heroku.

Lesson 4

API Deployment with FastAPI

- Write an API for machine learning inference using FastAPI.
- Deploy a machine learning inference API to Heroku.
- Write unit tests for APIs using the requests module.

Course 4

Automated Model Scoring & Monitoring

This course will help learners automate the DevOps processes required to score and re-deploy ML models. After model deployment, learners will set up regular scoring processes, learn to reason carefully about model drift, and learn whether models need to be retrained and re-deployed. They will learn to diagnose operational issues with models, including data integrity and stability problems, timing problems, and dependency issues. Finally, they will learn to set up automated reporting with APIs.

Course Project

A Dynamic Risk Assessment System

In this project, learners will make predictions about attrition risk in a fabricated dataset. They'll set up automated processes to ingest data and score, re-train, and re-deploy ML models that predict attrition risk. They'll write scripts to automatically check for new data and check for model drift. They'll also set up API's that allow users to access model results, metrics, and diagnostics. After completing this project, learners will have a full end-to-end, automated ML project that performs risk assessments. This project can be a useful addition to learners' portfolios, and the concepts they apply in the project can be applied to business problems across a variety of industries.

Lesson 1

Model Training & Deployment

- Ingest data.
- Automatically train models.
- Deploy models to production.
- Keep records about processes.
- Automate processes using cron jobs.

Lesson 2

Model Scoring & Model Drift

- Automatically score ML models.
- Keep records of model scores.
- Check for model drift using several different model drift tests.
- Determine whether models need to be retrained and re-deployed.

Lesson 3

Diagnosing & Fixing Operational Problems

- Check data integrity and stability.
- Check for dependency issues.
- Check for timing issues.
- Resolve operational issues.

Lesson 4

Model Reporting & Monitoring with APIs

- Create API endpoints that enable users to access model results, metrics, and diagnostics.
- Set up APIs with multiple, complex endpoints.
- Call APIs and work with their results.

Meet your instructors.

Joshua Bernhard

Data Scientist at Thumbtack

Josh has been sharing his passion for data for nearly a decade at all levels of university and as a data science instructor for coding bootcamps. He's used data science for work ranging from cancer research to process automation.

Giacomo Vianello

Principal Data Scientist at Cape Analytics

Giacomo Vianello is an end-to-end data scientist with a passion for state-of-the-art but practical technical solutions. He is principal data scientist at Cape Analytics, where he develops AI systems to extract intelligence from geospatial imagery bringing, cutting-edge AI solutions to the insurance and real estate industries.

Justin Clifford Smith, PhD

Senior Data Scientist at Optum

Justin a senior data scientist at Optum where he works to make healthcare more efficient with natural language processing and machine learning. Previously he was a data scientist at the US Census Bureau. His doctorate is from the University of California, Irvine where he studied theoretical physics.

Bradford Tuckfield

Data Scientist & Writer

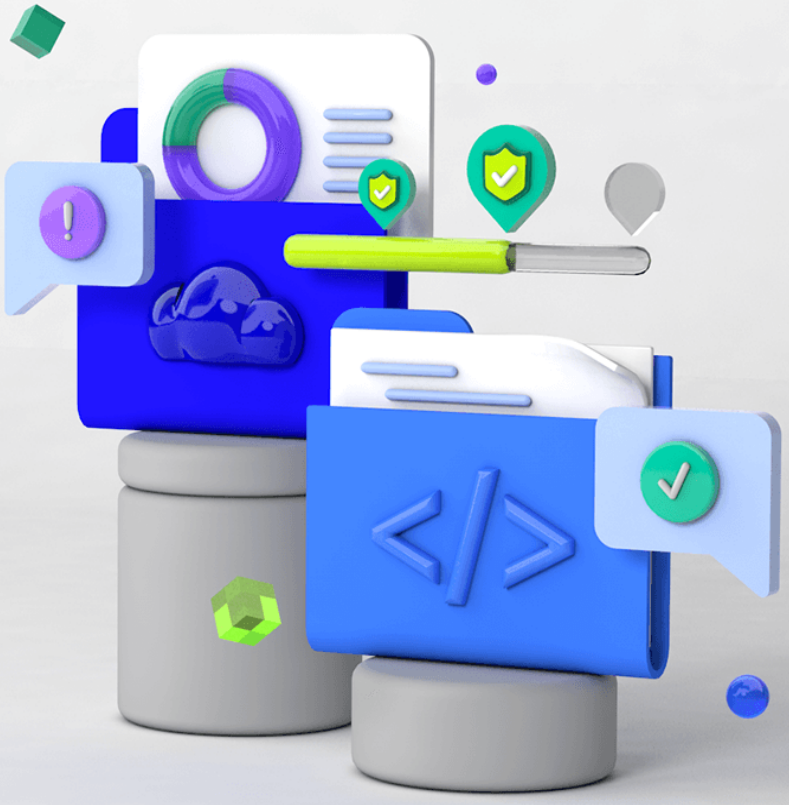
Bradford Tuckfield is a data scientist and writer. He has worked on applications of data science in a variety of industries. He's the author of Dive Into Algorithms, forthcoming with No Starch Press.

Ulrika Jägare

Head of AI/ML Strategy Execution at Ericsson

Ulrika has been with Ericsson for 21 years in various leadership roles, out of which 11 years in the data and AI space. Ulrika holds a master of science degree from University of Lund in Sweden and is also author of seven published books in data science.

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.

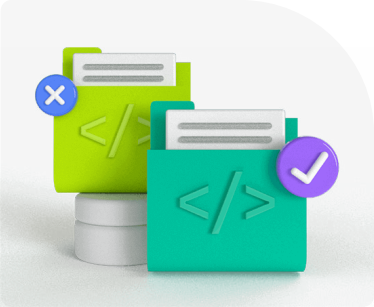
Workspaces

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

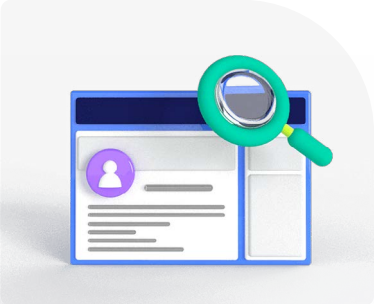
Verify skills mastery.

- 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 →

01.06.23 | V1.0