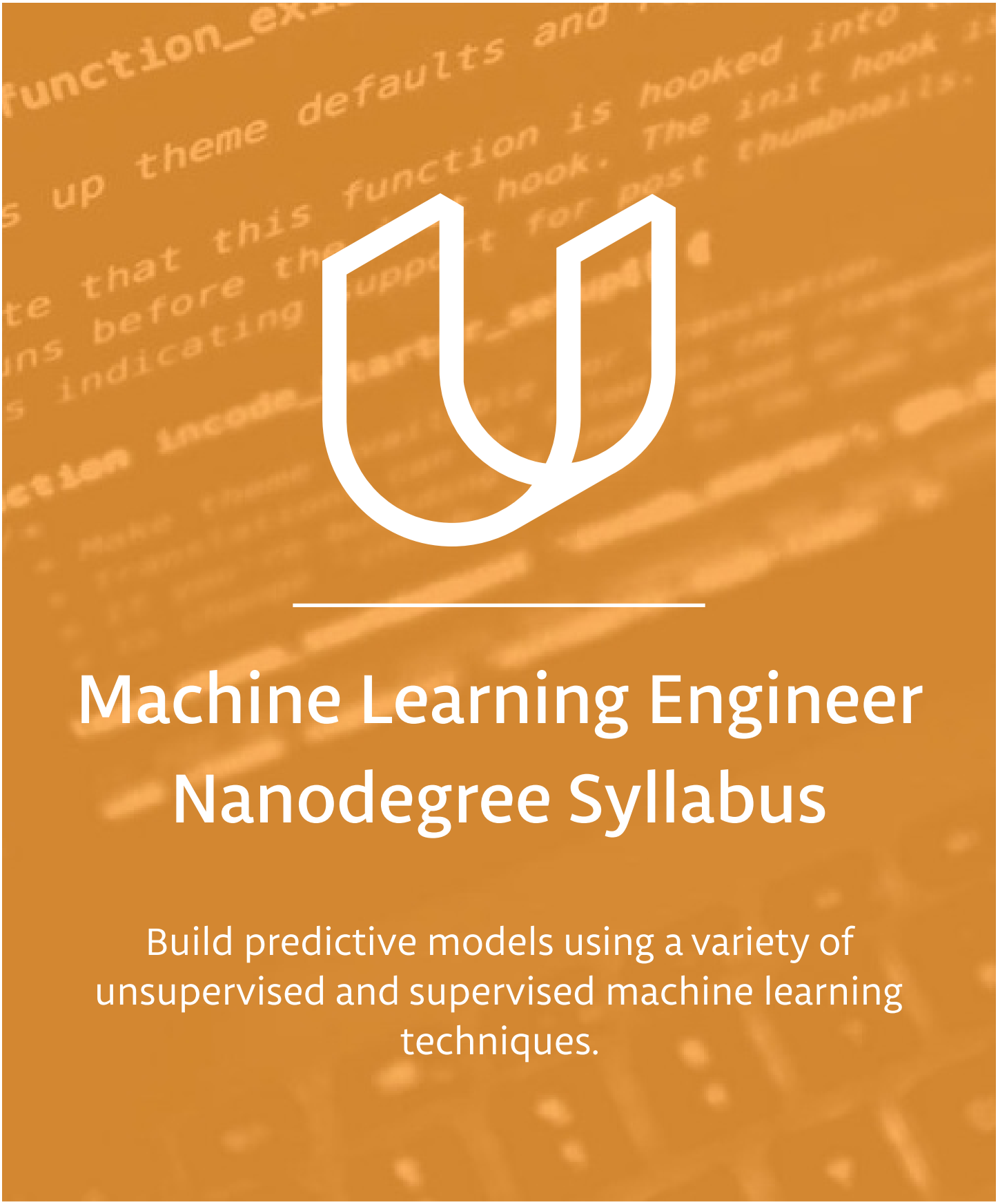

Machine Learning Engineer Nanodegree Syllabus

Build predictive models using a variety of unsupervised and supervised machine learning techniques.

UDACITY
FOR ENTERPRISE

Machine Learning Engineer

Table of Contents

Machine Learning Engineer

Before You Start

Prerequisites: To optimize your chances of success in this program, we recommend having experience with:

- 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
- To prepare for this program, we suggest you take the Intro to Programming Nanodegree Program.

Contact Info

While going through the program, if you have questions about anything, you can reach us at: machine-support@udacity.com.

Program 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.
- Understand cloud deployment terminology and best practices.
- 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.

Nanodegree Program Info

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 that you have skills in feature engineering, building machine learning algorithms, and model deployment.**

TECHNICAL REQUIREMENTS
Hardware Requirements: webcam, microphone
Software and Software Version Requirements: Text editor, Python 3.6, Jupyter Notebooks, Anaconda

LENGTH OF PROGRAM*: 4 months
FREQUENCY OF CLASSES: Self-paced
TEXTBOOKS REQUIRED: None

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 at AT&T. 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)

*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.

- **Plagiarism Detector**
  Using SageMaker, deploy your own PyTorch sentiment analysis model, which is trained to recognize the sentiment of movie reviews (positive or negative)
- **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.

Portfolio Exercise: 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.

Supporting Lesson Content: Software Engineering

Lesson Title	Learning Outcomes
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
PROGRAMMING	• Understand when to use object-oriented programming • Build and use classes • Learn how large, modular Python packages and use object-oriented programming
UPLOAD A PACKAGE TO PYPI	• Portfolio Exercise: Build your own Python package

Project: Deploy a Sentiment Analysis Model

In this project, you will use a recurrent neural network for the purpose of determining the sentiment of a movie review using a dataset of text from IMdB. You will create and deploy this model using Amazon SageMaker. After deploying your model, you will construct a simple web app which will interact with the deployed model and categorize any new, input review.

Supporting Lesson Content: Model Deployment

Lesson Title	Learning Outcomes
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
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
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
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
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

Project: Plagiarism Detector

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 perform plagiarism detection. Then, you will deploy your trained model using Amazon SageMaker.

Supporting Lesson Content: Machine Learning Case Studies

Lesson Title	Learning Outcomes
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.
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.
DEPLOYING CUSTOM MODELS	• Deploy a custom PyTorch model using SageMaker. • Write a custom training script to train a model of your own design.
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.

Project: Capstone & Proposal

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 portfolio.

Supporting Lesson Content: Project Options

Lesson Title	Learning Outcomes
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.
ELECTIVE 2: ARVATO FINANCIAL SERVICES	• Work through a real-world dataset and challenge provided by Arvato Financial Services, a Bertelsmann company.
ELECTIVE 3: CONVOLUTIONAL NEURAL NETWORK	• Complete a project to identify dog breeds based on images.
ELECTIVE 4: YOUR CHOICE	• Build a new project entirely of your own choosing.