AI Programming with Python Nanodegree Syllabus

Programming Skills, Linear Algebra, Neural Networks

Welcome to the AI Programming with Python Nanodegree program!

Before You Start

Educational Objectives: In this program, you'll learn all the foundational skills necessary to start using AI techniques in your current role, prepare for a full-time career in an AI-powered industry, or get started in the amazing world of artificial intelligence.

Length of Program: The program is comprised of 1 term, lasting 3 months. We expect students to work 10 hours/week on average. Estimated time commitment is 120 hours for the term.

Frequency of Classes: This is a unique, termed program that requires students to keep pace with their peers throughout the duration of the program.

Textbooks required: None

Instructional Tools Available: Video lectures, personalized project reviews, and a dedicated mentor

Contact Info

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

Nanodegree Program Info

This program focuses on the fundamental building blocks you will need to learn in order to become an AI practitioner. Specifically, you will learn programming skills, linear algebra, and even dive into neural networks and deep learning.

One of our main goals at Udacity is to help you create a job-ready portfolio. Building a project is one of the best ways both to test the skills you've acquired and to demonstrate your newfound abilities to prospective employers. At the end of this Nanodegree program, you'll have the opportunity to prove your skills by building an image classifier.

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

Introduction to Python:

Lesson Title	Learning Outcomes
WHY PYTHON PROGRAMMING	Learn why we program. Prepare for the course ahead with a detailed topic overview. Understand how programming in Python is unique.
DATA TYPES AND OPERATORS	Understand how data types and operators are the building blocks for programming in Python. Use the following data types: integers, floats, booleans, strings, lists, tuples, sets, dictionaries. Use the following operators: arithmetic, assignment, comparison, logical, membership, identity.
CONTROL FLOW	Implement decision-making in your code with conditionals. Repeat code with for and while loops. Exit a loop with break, and skip an iteration of a loop with continue. Use helpful built-in functions like zip and enumerate. Construct lists in a natural way with list comprehensions.
FUNCTIONS	Write your own functions to encapsulate a series of commands. Understand variable scope, i.e., which parts of a program variables can be referenced from. Make functions easier to use with proper documentation. Use lambda expressions, iterators, and generators.
SCRIPTING	Write and run scripts locally on your computer. Work with raw input from users. Read and write files, handle errors, and import local scripts. Use modules from the Python standard library and from third-party

	libraries. Use online resources to help solve problems.
LAB	Learn how to use a pre-trained image classifier to write a script that identifies dog breeds.

Numpy, Pandas, and Matplotlib:

Lesson Title	Learning Outcomes
ANACONDA	Learn how to use Anaconda to manage packages and environments for use with Python.
JUPYTER NOTEBOOKS	Learn how to use Jupyter Notebooks to create documents combining code, text, images, and more.
NUMPY BASICS	Learn the value of NumPy and how to use it to manipulate data for AI problems. Mini-Project: Use NumPy to mean normalize an ndarray and separate it into several smaller ndarrays.
PANDAS BASICS	Learn to use Pandas to load and process data for machine learning problems. Mini-Project: Use Pandas to plot and get statistics from stock data.
MATPLOTLIB BASICS	Learn how to use Matplotlib to choose appropriate plots for one and two variables based on the types of data you have.

Linear Algebra Essentials:

Lesson Title	Learning Outcomes
INTRODUCTION	Learn the basics of the beautiful world of Linear Algebra and learn why it is such an important mathematical tool.
VECTORS	Learn about the basic building block of Linear Algebra.

LINEAR COMBINATION	Learn how to scale and add vectors and how to visualize them in 2 and 3 dimensions.
LINEAR TRANSFORMATION AND MATRICES	Learn what a linear transformation is and how is it directly related to matrices. Learn how to apply the math and visualize the concept.
LINEAR ALGEBRA IN NEURAL NETWORKS	Learn about the world of Neural Networks and see how it related directly to Linear Algebra.
LABS	
● VECTORS LAB	Learn how to graph 2D and 3D vectors.
● LINEAR COMBINATION LAB	Learn how to computationally determine a vector's span and solve a simple system of equations.
● LINEAR MAPPING LAB	Learn how to solve problems computationally using vectors and matrices.

Neural Networks:

Lesson Title	Learning Outcomes
INTRODUCTION TO NEURAL NETWORKS	Acquire a solid foundation in deep learning and neural networks. Implement gradient descent and backpropagation in Python.
TRAINING NEURAL NETWORKS	Learn about techniques for how to improve training of a neural network, such as: early stopping, regularization and dropout.
DEEP LEARNING WITH PYTORCH	Learn how to use PyTorch for building deep learning models.

Project: Image Classifier

In the next few years software developers will need to know how to incorporate deep learning models into everyday applications. Any device with a camera will be using image classification, object detection and face recognition, all based on deep learning models. In this project you will be implementing an image classification application. This application will train a deep learning model on a dataset of images. It will then used the trained model to classify new images. First you will develop your code in a Jupyter notebook to ensure your training implementation works well. Afterwich you will convert your code into a python application that you will be able run from the command line of your system