Data Analyst
Nanodegree Program Syllabus

Built in partnership with MODE kaggle

Level: Intermediate **Duration:** 4 months (10 hours/week)

Before You Start

Educational Objectives: This program prepares you for your job as a data analyst by helping you learn to organize data, uncover patterns and insights, draw meaningful conclusions, and clearly communicate critical findings. You'll develop proficiency in Python and its data analysis libraries (Numpy, pandas, Matplotlib) and SQL as you build a portfolio of projects to showcase in your job search.

Length of Program: The program covers one term of three month (approx. 13 weeks). If you spend about 10 hours per week working through the program, you should finish the term within 13 weeks.

Prerequisites: In order to succeed in this program, we recommend having experience working with data in Python (Numpy and Pandas) and SQL.

How to Prepare: If you are new to programming, try the Programming for Data Science Nanodegree program.

Nanodegree Program Overview Page: click here

Nanodegree Program Info

Advance your programming skills and refine your ability to work with messy, complex datasets. You'll learn to manipulate and prepare data for analysis, and create visualizations for data exploration. Finally, you'll learn to use your data skills to tell a story with data.

Module 1: Introduction to Data Analysis

Learn the data analysis process of questioning, wrangling, exploring, analyzing, and communicating data. Learn how to work with data in Python using libraries like NumPy and Pandas.

Project 0: Explore Weather Trends

This project will introduce you to the SQL and how to download data from a database. You'll analyze local and global temperature data and compare the temperature trends where you live to overall global temperature trends.

Project 1: Investigate a Dataset

In this project, you'll choose one of Udacity's curated datasets and investigate it using NumPy and pandas. You'll complete the entire data analysis process, starting by posing a question and finishing by sharing your findings.

Lesson Title	Learning Outcomes
ANACONDA	Learn to use Anaconda to manage packages and environments for use with Python.
JUPYTER NOTEBOOKS	Learn to use this open-source web application to combine explanatory text, math equations, code, and visualizations in one shareable document.
DATA ANALYSIS PROCESS	Perform the entire data analysis process on a dataset. Learn to use NumPy and Pandas to wrangle, explore, analyze, and visualize data.
PANDAS AND NUMPY: CASE STUDY 1	Perform the entire data analysis process on a dataset. Learn more about NumPy and Pandas to wrangle, explore, analyze, and visualize data.
PANDAS AND NUMPY: CASE STUDY 2	Implement k-means clustering to break an image up into parts. Find the contours and edges of multiple objects in an image. Learn about background subtraction for video.
PROGRAMMING WORKFLOW FOR DATA ANALYSIS	Learn about how to carry out analysis outside Jupyter notebook using IPython or the command line interface.

Module 2: Practical Statistics

Learn how to apply inferential statistics and probability to important, real-world scenarios, such as analyzing A/B tests and building supervised learning models.

Project 2: Analyze Experiment Results

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.

Lesson Title	Learning Outcomes
SIMPSON'S PARADOX	Examine a case study to learn about Simpson's Paradox.
PROBABILITY	Learn the fundamental rules of probability.
BINOMIAL DISTRIBUTION	Learn about binomial distribution where each observation represents one of two outcomes. Derive the probability of a binomial distribution.
CONDITIONAL PROBABILITY	Learn about conditional probability, i.e., when events are not independent.
BAYES RULE	Build on conditional probability principles to understand the Bayes rule. Derive the Bayes theorem.
STANDARDIZING	Convert distributions into the standard normal distribution using the Z-score. Compute proportions using standardized distributions.
SAMPLING DISTRIBUTIONS AND CENTRAL LIMIT THEOREM	Use normal distributions to compute probabilities. Use the Z-table to look up the proportions of observations above, below, or in between values.
CONFIDENCE INTERVALS	Estimate population parameters from sample statistics using confidence intervals.
HYPOTHESIS TESTING	Use critical values to make decisions on whether or not a treatment has changed the value of a population parameter.
T-TESTS AND A/B TESTS	Test the effect of a treatment or compare the difference in means for two groups when we have small sample sizes.
REGRESSION	Build a linear regression model to understand the relationship between independent and dependent variables. Use linear regression results to make a prediction.
MULTIPLE LINEAR REGRESSION	Use multiple linear regression results to interpret coefficients for several predictors.
LOGISTIC REGRESSION	Use logistic regression results to make a prediction about the relationship between categorical dependent variables and predictors.

Module 3: Data Wrangling

Learn the data wrangling process of gathering, assessing, and cleaning data. Learn how to use Python to wrangle data programmatically and prepare it for deeper analysis.

Project 3: Wrangle and Analyze Data

Real-world data rarely comes clean. Using Python, you'll gather data from a variety of sources, assess its quality and tidiness, then clean it. You'll document your wrangling efforts in a Jupyter Notebook, plus showcase them through analyses and visualizations using Python and SQL.

Lesson Title	Learning Outcomes
INTRO TO DATA WRANGLING	Identify each step of the data wrangling process (gathering, assessing, and cleaning). Wrangle a CSV file downloaded from Kaggle using fundamental gathering, assessing, and cleaning code.
GATHERING DATA	Gather data from multiple sources, including gathering files, programmatically downloading files, web-scraping data, and accessing data from APIs. Import data of various file formats into pandas, including flat files (e.g. TSV), HTML files, TXT files, and JSON files. Store gathered data in a PostgreSQL database.
ASSESSING DATA	Assess data visually and programmatically using pandas. Distinguish between dirty data (content or “quality” issues) and messy data (structural or “tidiness” issues). Identify data quality issues and categorize them using metrics: validity, accuracy, completeness, consistency, and uniformity.
CLEANING DATA	Identify each step of the data cleaning process (defining, coding, and testing). Clean data using Python and pandas. Test cleaning code visually and programmatically using Python

Module 4: Data Visualization with Python

Learn to apply visualization principles to the data analysis process. Explore data visually at multiple levels to find insights and create a compelling story.

Project 4: Communicate Data Findings

In this project, you will use Python's data visualization tools to systematically explore a selected dataset for its properties and relationships between variables. Then, you will create a presentation that communicates your findings to others.

Lesson Title	Learning Outcomes
DATA VISUALIZATION IN DATA ANALYSIS	Understand why visualization is important in the practice of data analysis. Know what distinguishes exploratory analysis from explanatory analysis, and the role of data visualization.
DESIGN OF VISUALIZATIONS	Interpret features in terms of level of measurement. Know different encodings that can be used to depict data in visualizations. Understand various pitfalls that can affect the effectiveness and truthfulness of visualizations.
UNIVARIATE EXPLORATION OF DATA	Use bar charts to depict distributions of categorical variables. Use histograms to depict distributions of numeric variables. Use axis limits and different scales to change how your data is interpreted.
BIVARIATE EXPLORATION OF DATA	Use scatter plots to depict relationships between numeric variables. Use clustered bar charts to depict relationships between categorical variables. Use violin and bar charts to depict relationships between categorical and numeric variables. Use faceting to create plots across different subsets of the data.
MULTIVARIATE EXPLORATION OF DATA	Use encodings like size, shape, and color to encode values of a third variable in a visualization. Use plot matrices to explore relationships between multiple variables at the same time. Use feature engineering to capture relationships between variables.
EXPLANATORY VISUALIZATIONS	Understand what it means to tell a compelling story with data. Choose the best plot type, encodings, and annotations to polish your plots. Create a slide deck using a Jupyter Notebook to convey your findings.
VISUALIZATION CASE STUDY	Apply your knowledge of data visualization to a dataset involving the characteristics of diamonds and their prices.