Practical Python for Data Science

# Description

Artificial Intelligence is playing a larger role in our everyday lives; from early detection of cancer to predicting customer behaviour, AI is proving to be a powerful tool in understanding the world around us. But how does it work? And how can we be sure we’re developing correct and robust models for good rather than evil? In this introductory Python course, we will learn how to use the Python programming language to write simple programs and start to build models that help use solve real problems. By the end of this course, you will have a portfolio of AI projects based on real world problems, and the understanding to interpret and apply more specific or advanced techniques to your field of interest.

# Prerequisites

* GCSE Level Maths
* A Computer and an Internet connection

# General Notes

* Weeks 5-7 all take a similar format, with the structure of the session as follows:

1. Learn the theory of the model
2. Apply the theory to abstract data (unlabelled x,y)
3. Apply the model to a “real world example” (fake generated data, but more complicated – perhaps more dimensions, or with entries missing)

* After that, the students would be given a worksheet with open-source data and asked to apply that method to an actual real-world problem, to build a portfolio.

|  |  |  |
| --- | --- | --- |
| Week Number | Title | Description |
| Week 0 | Introduction to tools, software and resources | Optional activity delivered via video for setup and a more thorough explanation of the course  **Aims:**   * Understand why we would want to learn how to code, and why we are choosing Python (second most popular programming language, easy to learn, can run online etc) * Understand what AI and Machine Learning is   Go over course material with examples of what we’re going to do   * Get students to either install Anaconda to run Python locally, or log onto Google Colab to run code in the cloud.   *Note: A top-spec computer here shouldn’t be necessary, we’re not using any GPU boosted libraries here so Colab should be fine.* |
| Week 1 | Python Crash Course | This session is about learning the core Python functionality we need to engage in the following content. It’s not a software engineering course so we won’t be covering topics such as object orientated programming – we won’t need it for data science. (However, it may be included in an extension task if time permits!)  **Aims**   * Learn all the necessary basics of Python for this course, including variables, data types, functions, lists, dictionaries, control flow and loops. * Solve some “simple programming problems” using these ideas. |
| Week 2 | Data Wrangling | We start with the first part of “Data Science” – the data! Before we can learn how to analyse data, we need to know how to import it into Python and manipulate it to our needs.  **Aims**   * Learn to use the numpy and pandas libraries for high performance numerical computation and tabular data management respectively. * Import data from the web using pandas and learn the basic of data cleaning (removal of missing data etc). Discussion of different strategies of this – could removing data points make our models worse/unfair? * Time permitting, learn about the scipy and/or BeautifulSoup libraries for statistic and web scraping functionality. |
| Week 3 | Data Visualisation | One of the most useful things to do before analysing our data is exploring what the data looks like. This process might answer some questions straight away or generate more for further investigation – it also serves as a “sanity check” for our more advanced methods and empowers us to visualise the models we create.  **Aims**   * Get comfortable with the matplotlib library – this is the “grandfather” library for Python data visualisation and a lot of libraries are built on top of this (see below). Understand the difference and pros/cons of the functional approach vs the OOP approach. * Introduce the seaborn and bokeh libraries for more “out of the box” plots, as well as using pandas directly. Understand that both seaborn and pandas use matplotlib to work, so it’s worth it to master matplotlib! * Look at some of the other data visualisation options out there, for example shiny with R or d3.js with JavaScript. Data visualisation on it’s own is a job if done well! |
| Week 4 | Building Models the Right Way | We take a break in coding to have a theory session. We talk about the different types of models, different training metrics and practices, and some common problems with models that can occur during training. We also briefly talk about AI ethics here – this is a fast-growing field identifying unethical models and poor practice. We talk about why these models are not only unethical, but why they might actually be broken.  *Note: This might be a bit of a sensitive topic because it brings the content into the context of the wider world and politics. However, it’s very important to look at some of these “evil” models to raise awareness of this problem. A good resource for this is* [*here*](https://github.com/daviddao/awful-ai)*.*  **Aims**   * Learn the different types of machine learning models and the kind of problems machine learning can solve. * Learn about some common metrics used in data science and what they mean – accuracy, precision, F1 score, Kappa score (and AUC if time permits) * Discuss good training practices with ideas such as train-test-split, cross validation and overfitting. * Talk about how these principles amongst others can translate into real world problems if not followed, and how some AI is being used incorrectly. |
| Week 5 | Regression | We start by looking at looking at the relationship between two variables with regression and how it applies to the real world. This is intended to be an easy introduction into the world of data science to explain how the relevant libraries work.  **Aims**   * Learn the theory of linear and logistic regression, and how to apply these principles using Python and the scikitlearn library. * Also learn the statsmodels library, a more statistical approach to the same problem (if time allows) |
| Week 6 | Classification/Clustering (part one) | Classification is one of the most typical problems a data scientist encounters and has a number of methods for both supervised and unsupervised approaches. In this session we’ll take a look at some approaches for this problem and the pros and cons of the resulting models.  **Aims**   * Learn the theory of the k means clustering, k nearest neighbours and decision tree algorithms, and apply these to problems. * Visualise the results of these models with some clever plotting. |
| Week A (Optional, time permitting) | Classification/Clustering (part two) and Dimensionality Reduction | There are a lot of classification algorithms so we’ll start this session by looking at one more, as well as the concept of dimensionality reduction; which is a powerful method that can be used to learn more about our data as well as “power up” our models in general.  **Aims**   * Learn the theory of Principle Component Analysis (PCA) and Support Vector Machines (SVM), and apply these to some real problems. |
| Week B  (Optional, time permitting) | Natural Language Processing (NLP) | It’s very easy to build models to work with numeric data, such as time spent on a website or sales per customer, but sometimes our data isn’t in a “nice” form to use for data science – in this session we introduce the idea of natural language processing, and why it is useful.  *Note: Week 8 and 9 are flexible – the course is quite intense so if these weeks need to be left out it won’t impact the final outcome; they are just two bits of “further reading” people can go off and investigate on their own!*   * Introduce some of the NLP libraries and apply them to some simple cases. * Potentially add webscraping with BeautifulSoup if not covered in week 2 |
| Week C  (Optional, time permitting) | Neural Networks | Neural networks and deep learning are the “fashionable” models of the moment and are what are most commonly used in “Big Data” environments. Neural networks often require more data to work than previous models, but often have a higher “ceiling” due to their ability to “learn” abstract patterns given enough data.  **Aims**   * Understand the theory behind neural networks, including back propagation and gradient decent. (This will have to be more “intuition based” as it requires a university level of mathematics to understand fully) * Look at some of the different types of neural network models and how they work for different use cases. * Apply these models to a very simple problem, such as digit recognition. |
| Week 8 | Final Project | By this point students should feel confident to approach problems on their own and know where to find resources to help out when problems occur. This week, students are asked to produce a project using data they have found and find interesting, with the tools we have learnt during the course. It’s intended as a portfolio building exercise for students to spread their wings and learn what interests them.  Some example projects would include: interactive data visualisation (data journalism), neural network image classifiers, or something else relevant to the students interests.  Throughout the course we’ll be looking through the documentation and reading articles related to the methods we’re applying. One of the most important skills for data scientists is pushing through the documentation to create something seemingly previously impossible, and that independence is what this session aims to achieve. |