The Datakirk - Older Learners Curriculum Overview

The following is an overview of the proposed curriculum for the older learners at the Datakirk. The course takes place mainly through the use of Jupyter notebooks, with concepts introduced through the framework of programming in Python, however no prior knowledge is assumed. The only prerequisite is a Nat5/GCSE level of mathematics.

The course is quite ambitious, aiming to take the learners all the way to some of the most cutting edge techniques used in real machine learning and data science.

1. Introduction to data science

This is an introductory lesson where we discuss the concept of data science and machine learning, with the following aims:

1. To understand data science as a general process of converting raw data into valuable insight, and be able to frame real world problems in these terms.
2. Understand the different areas that make up the field of data science including statistics, pure maths and programming. Understand, on a basic level, the the different pieces of maths, programming languages, and technologies used in practice.
3. Understand what the term "Machine Learning" means and be able to name and describe two or three important issues within this subject.
4. Be able to explain the K-Nearest neighbours algorithm.

2. Programming Basics

The plan is to spend six lessons introducing the learners to programming in Python. This is split into three notebooks, each of which is designed to be completed over the course of two lessons.

1. Notebook 1 introduces the concept of a programming language, and introduces specifically Python. We look an Jupyter notebooks, how to open and use them. We then cover variables, data types, simple math expressions, conditions and simple array data structures.
2. Notebook 2 continues with Python basics. We look specifically at loops and functions and introduce new concepts such as error messages.
3. Notebook 3 introduces the concept of packages and libraries. We specifically look at NumPy and Matplotlib.

4. Regression

Next, I propose to spend roughly 8-10 lessons on the topic of regression, also picking up some new programming techniques along the way. This will be split into several notebooks, following a structure something like this:

1. In notebook 1 we put programming on hold temporarily and explore interactive graphs, created in matplotlib, that graphically explain concepts surrounding regression. We introduce the concept of simple linear regression with one and two variables and begin to explore the idea of loss functions, gradients and optimisation.
2. Notebook 2 is about implementation. We introduce some basics of linear algebra, specifically the idea of a matrix for holding our data, and look at the numpy functions that

can help us solve linear regression problems.
3. In notebook 3 we start thinking some extensions of simple linear regression. We look at polynomial regression and implement a solver.
4. In notebook 4 we look at some ways to generalise regression further with basis functions. We look at fitting 1D functions and discuss the curse of dimensionality. We look at spline methods and RBFs.
5. Finally we introduce the concept of overfitting and discuss regularisation techniques for dealing with it. We look at L2 and L1 norms and other penalty methods. We also introduce the idea of a training, validation and test data as well as cross validation.

5. Classification

Following on from regression, I propose to spend around 6-8 weeks on the topic of classification, mainly through the lens of logistic regression.

1. Notebook 1 introduces the idea of classification and describes its difference from regression. In this first notebook we postpone LR in favour of KNN. We implement a simple KNN algorithm and evaluate its performance. We discuss the idea of balanced targets and look at confusion matrices.
2. Notebook 2 introduces logistic regression. We discuss the idea of exponentials, loss functions and binary classification. We plot some logistic functions in 1 and 2D.
3. We explore the idea of gradient descent and why it is necessary in the case of LR. We revisit loss landscapes and think about algorithms for traversing them.
4. We build a logistic regression model and evaluate it on a real dataset, comparing its performance with KNN.

6. Neural Networks

The final stage of this course will be to look into neural networks. In theory, by this point, there is very little additional content to learn. Neural networks pull together everything that has been learnt so far with only a few extensions. However, they are certainly complex and covering just the basics if definitely ambitious. This section could in theory last indefinitely but I think a good target would be to cover the basics over the course of 15 weeks or so. I haven't begun thinking about this in great detail yet so ideas are rough at this stage.

1. What are neural networks? Where are they used? Why are people excited about them. For this section I want to keep it discursive and open with little to no maths or programming.
2. What is the structure of a neural network? What operations are going on? How can we represent these operations mathematically?
3. What parameters can we vary? What is the loss function and gradient descent?
4. Backpropogation
5. Neural network extensions. CNNs RNNs, GANs.