Plotting Arrays

In this lesson, we will learn how to plot arrays using a Python library called "Matplotlib".

Just like large tables or lists of numbers, looking at large arrays isn't that insightful. Visualising them helps us quickly get an idea of the general meaning. One way of plotting 2-dimensional arrays of numbers is to think of them as flat 2-dimensional surfaces, colored according to the value of each cell in the array. You can choose how you turn a value inside a cell into a color. You might choose to simply turn the value into a color according to a color scale, or you might color everything white except values above a certain threshold which would be black.

Let's try plotting the small three by two array we created above. Before we can do this, we need to extend Python's abilities to plot graphics. We do this by *importing* additional Python code that others have written. You can think of this as borrowing food recipe books from your friend to add to your own bookshelf so that your bookshelf now has extra content enabling you to prepare more kinds of dishes than you could before.

The following shows how we import graphics plotting capability.

```
import matplotlib.pyplot
```

The `matplotlib.pyplot` is the name of the new "recipe book" that we're borrowing. You might come across phrases like "import a module" or "import a library". These are just the names given to the additional Python code you're importing.

If you get into Python, you'll often import additional capabilities to make your life easier by reusing other's useful code. You might even create your own code to share with others!

We're ready to plot that array now. Take a look at the code below. The

instruction to create a plot is `imshow()`, and the first parameter is the array we want to plot.

That last bit “interpolation” is there to tell Python not to try to blend the colors to make the plot look smoother, which it does by default, trying to be helpful and `savefig()` is just a simple function to output the plot that we created in the first line. Let’s run the cod and see what we get.

```
import matplotlib.pyplot
matplotlib.pyplot.imshow(a, interpolation='nearest')
matplotlib.pyplot.savefig("output/samplePlot.png")
```

Exciting! Our first plot shows the three by two sized array as colors. You can see that the array cells which have the same value also have the same color. Later we’ll be using this very same `imshow()` instruction to visualize an array of values that we feed our neural network.

The Python packages have a very rich set of tools for visualizing data. You should explore them to get a feel for the wide range of plots possible, and even try some of them. Even the `imshow()` instruction has many options for plotting for us to explore, such as using different color palettes.