WHAT IS CONVOLUTION AND POOLING?

In the previous example, you saw how you could create a neural network called a deep neural network to pattern match a set of images of fashion items to labels. In just a couple of minutes, you're able to train it to classify with pretty high accuracy on the training set, but a little less on the test set. Now, one of the things that you would have seen when you looked at the images is that there's a lot of wasted space in each image. While there are only 784 pixels, it will be interesting to see if there was a way that we could condense the image down to the important features that distinguish what makes it a shoe, or a handbag, or a shirt. That's where convolutions come in. So, what's convolution? You might ask. Well, if you've ever done any kind of image processing, it usually involves having a filter and passing that filter over the image in order to change the underlying image. The process works a little bit like this. For every pixel, take its value, and take a look at the value of its neighbors. If our filter is three by three, then we can take a look at the immediate neighbor, so that you have a corresponding three by three grid. Then to get the new value for the pixel, we simply multiply each neighbor by the corresponding value in the filter. So, for example, in this case, our pixel has the value 192, and its upper left neighbor has the value zero. The upper left value and the filter is negative one, so we multiply zero by negative one. Then we would do the same for the upper neighbor. Its value is 64 and the corresponding filter value was zero, so we'd multiply those out. Repeat this for each neighbor and each corresponding filter value, and would then have the new pixel with the sum of each of the neighbor values multiplied by the corresponding filter value, and that's a convolution. It's really as simple as that. The idea here is that some convolutions will change the image in such a way that certain features in the image get emphasized. So, for example, if you look at this filter, then the vertical li

nes in the image really pop out. With this filter, the horizontal lines pop out. Now, that's a very basic introduction to what convolutions do, and when combined with something called pooling, they can become really powerful. But simply, pooling is a way of compressing an image. A quick and easy way to do this, is to go over the image of four pixels at a time, i.e, the current pixel and its neighbors underneath and to the right of it. Of these four, pick the biggest value and keep just that. So, for example, you can see it here. My 16 pixels on the left are turned into the four pixels on the right, by looking at them in two-by-two grids and picking the biggest value. This will preserve the features that were highlighted by the convolution, while simultaneously quartering the size of the image. We have the horizontal and vertical axes.

https://www.tensorflow.org/api\_docs/python/tf/keras/layers/MaxPool2D

https://www.tensorflow.org/api\_docs/python/tf/keras/layers/Conv2D