Intro Deep Learning: Homework 3

Daniel Novikov

March 13th, 2022

Problem 1

You have a convolutional neural network that takes as an input image of size 512 $\times$ 512 $\times$ 3 and passes it through a layer that convolves the image using 3 filters of dimensions 5 $\times$ 5 $\times$ 3 with a valid padding. (a) List all learnable parameters of this convolution layer. (b) What if you want to replicate the behavior of this convolutional layer using a fully connected layer? How many parameters would that fully connected layer have?

Answer:

(a) The model has a parameter for each element of the kernels. Each kernel has $5 \times 5 \times 3 = 75$ parameters and there are 3 kernels, which means there are 225 parameters in the convolution layer.

(b) We would have a fully connected layer with an output neuron for each position the kernel traverses during convolution. The number of parameters is given by the number of image positions of the kernel's traversal * kernel size * number of kernels.

Number of kernel positions = Size of image - size of kernel + 1

$$= (512 * 512 * 3 - 5 * 5 * 3 + 1) * (5 * 5 * 3) * 3 = 176,930,550$$

parameters.

And that's my final answer because that's what I think was being asked. However, I can't help but notice that to keep the same functionality, most of the parameters for the FC layer should be 0, except at the 5 $\times$ 5 $\times$ 3 features seen by the kernel at the position. Furthermore, these parameters should be shared for every neuron but just stored in different positions, the ones that match the input features seen by the kernel at that image position. So really the number of learnable parameters is the same, 225, if we restrict that the FC layer does the same thing the conv layer does and we only count non-0, learnable weights.

Problem 2

Given a binary input image of diagonal streaks (see example in Figure 2) and two filters (see Figure 1a) describe how would you build a detector for finding the location of pattern shown in Figure 1b on the input image. Allowed operations are convolution, summation, and argmax.

Answer:

I would convolve the image in Figure 2 with the kernels in Figure 1A and receive two output matrices, one for each kernel. Then I add those matrices together. Finally, since each index in this matrix corresponds to the kernel at a certain position in the image, taking the argmax gives the index that corresponds to the kernel being at the position most closely matching the desired pattern.

Problem 3

Demonstrate that convolution is translation invariant for 1D convolution (Note:this can be extended to N-D convolutions as well).

Answer:

To convolve a 1D vector with a kernel, we slide the kernel over the vector, moving the kernel over by 1 element each time. The kernel passes over the entire vector. If the pattern we wish to match is translated elsewhere in the vector, the kernel will still find it, since it passes over the entire vector.

Ex: Kernel = $[1, 2, 3]$

Vector = [6, 7, 8, 1, 2, 3, 5]

The kernel will match the pattern beginning at index 3 and ending at index 5 and output a high value here.

If we translate the pattern in the vector

Vector = [1, 2, 3, 6, 7, 8, 5]

The kernel will match the pattern beginning at index 0 and ending at index 2 and output a high value here.

In conclusion, convolution is translation invariant because no matter where in the data we move the pattern, the kernel moves over all of the data and will eventually find it.