

ECE/CS/ME 539 – Fall 2024 — Activity 24

Consider the following code

This program defines a simple convolutional neural network by stacking convolutional and max pooling layers as follows:

```
1 from torch import nn
2 model = nn.Sequential(
3     nn.Conv2d(3, 32, kernel_size=(5, 5), stride=(1, 1)),
4     nn.ReLU(),
5     nn.MaxPool2d(kernel_size=(2, 2), stride=(2, 2)),
6     nn.Conv2d(32, 64, kernel_size=(3, 3), stride=(1, 1)),
7     nn.ReLU(),
8     nn.MaxPool2d(kernel_size=(2, 2), stride=(2, 2)),
9     nn.Conv2d(64, 64, kernel_size=(3, 3), stride=(1, 1)),
10    nn.ReLU(),
11    nn.Flatten(),
12    nn.Linear(1024, 10)
13 )
```

- (a) Suppose we feed this model with an input tensor of shape $N \times 3 \times 32 \times 32$ matrix. After the first Conv2D layer (line 3), what is the size of the output activation tensor? *Hint: Check the documentation of `nn.Conv2d()`.*
- (b) How many trainable parameters (number of elements in the weights and bias tensors) does this convolution layer have? *Hint: Inspect `model[0].weight` and `model[0].bias`.*
- (c) Computationally, the costliest operation is multiplication. How many multiplications should the model perform to compute the output for the first layer? (This is often referred to as FLOPs) *Hint: It might be easier to first consider a convolution with a 1×1 kernel size and only 1 output filter.*
- (d) There are _____ filters in the second Conv2D layer (line 6). The size of each Kernel filter (a tensor) is _____ \times _____ \times _____.
- (e) The Flatten layer (line 11) reshapes the output activation tensor of the third Conv2D layer (line 9) into a vector. The shape of the third Conv2d's output is ($H \times W \times C$): _____ \times _____ \times _____. The length of the output vector of the Flatten layer (line 11) is _____.