1. Explain convolutional neural network, and how does it work?

Ans: we can perform several operations like padding, filtering, pooling layer and flattening.

Filtering is used to extract features from an image like edges, shapes etc and the pooling layer is used to extract the most important feature and it also helps to reduce the spatial dimensions of the image.

1. How does refactoring parts of your neural network definition favor you?

Ans:

1. **Improved readability and maintainability:** Refactoring can help to simplify and clarify the code for your neural network, making it easier to understand and modify. By breaking down complex sections of the network into smaller, more modular pieces, you can make it easier to debug and update the network as needed.
2. **Better performance:** Refactoring can also help to optimize the performance of your neural network. By reducing the number of redundant operations or unnecessary layers, you can speed up the training and testing of your network, making it more efficient and reducing the time and computational resources required.
3. What does it mean to flatten? Is it necessary to include it in the MNIST CNN? What is the reason for this?

Ans: flattening layer is used to convert higher dimension vector into 1D vector by concatenating all the rows. Yes the images are converted into 1 D in MINST CNN so that can be passed through a fully connected layer.

4. What exactly does NCHW stand for?

Ans:

* N: batch size, which is the number of samples that are processed simultaneously in a single forward or backward pass of the neural network
* C: number of channels, which refers to the number of color channels in an image or the number of feature maps in a convolutional layer
* H: height of the input feature map or image
* W: width of the input feature map or image

1. Why are there 7\*7\*(1168-16) multiplications in the MNIST CNN's third layer?

Ans:

There is one bias for each channel. (Sometimes channels are called features or filters when they are not input channels.) The output shape is 64x4x14x14, and this will therefore become the input shape to the next layer. The next layer, according to summary, has 296 parameters. Let's ignore the batch axis to keep things simple. So for each of 14\*14=196 locations we are multiplying 296-8=288 weights (ignoring the bias for simplicity), so that's 196\*288=56\_448 multiplications at this layer. The next layer will have 7\*7\*(1168-16)=56\_448 multiplications.

6.Explain definition of receptive field?

Ans:

In a convolutional neural network (CNN), the receptive field is the region of the input space that a particular neuron or feature map is sensitive to. More specifically, the receptive field of a neuron is defined as the area of the input feature map that is involved in the computation of the neuron's output.

In the context of CNNs, the receptive field can be viewed as a "window" over the input feature map that is passed through each layer of the network. At each layer, the receptive field size grows as the size of the filter/kernel used in that layer increases. The receptive field of a neuron in one layer is determined by the receptive fields of the neurons in the previous layer that connect to it.

7. What is the scale of an activation's receptive field after two stride-2 convolutions? What is the reason for this?

Ans:

Assuming the stride-2 convolutions have the same filter size and padding, the scale of an activation's receptive field after two stride-2 convolutions would be four times larger than the original receptive field.

8. What is the tensor representation of a color image?

Ans: Height, Width, Channel

9. How does a color input interact with a convolution?

Ans: we perform padding, filtering, pooling layer and flattening