Your grade: 100%

Your latest: **100%** • Your highest: **100%** • To pass you need at least 80%. We keep your highest score.

Next item →

1. What do you think applying this filter to a grayscale image will do?

1 / 1 point

$$\begin{bmatrix} 0 & 1 & -1 & 0 \\ 1 & 3 & -3 & -1 \\ 1 & 3 & -3 & -1 \\ 0 & 1 & -1 & 0 \end{bmatrix}$$

- ○ Detect 45 degree edges
- ○ Detect image contrast
- ◉ Detect vertical edges
- ○ Detect horizontal edges

Expand

Correct

Correct! As you can see the difference between values from the left part and values from the right of this filter is high. When convolving this filter on a grayscale image, the vertical edges will be detected.

2. Suppose your input is a 128 by 128 grayscale image, and you are not using a convolutional network. If the first hidden layer has 256 neurons, each one fully connected to the input, how many parameters does this hidden layer have (including the bias parameters)?

1 / 1 point

- (selected) 4194560
- 4194304
- 12582912
- 12583168

Expand

Correct

Correct, the number of inputs for each unit is 128×128 since the input image is grayscale, so we need $128 \times 128 \times 256$ parameters for the weights and 256 parameters for the bias thus $128 \times 128 \times 256 + 256 = 4194560$.

3. Suppose your input is a 256 by 256 color (RGB) image, and you use a convolutional layer with 128 filters that are each 7×7. How many parameters does this hidden layer have (including the bias parameters)?

1 / 1 point

- ( ) 18816
- (•) 18944
- ( ) 1233125504
- ( ) 6400

Expand

Correct

Yes, you have $7 \times 7 \times 3 + 1$ weights per filter with the bias. Given that you have 128 filters, you get $(7 \times 7 \times 3 + 1) \times 128 = 18944$.

4. You have an input volume that is $121 \times 121 \times 16$, and convolve it with 32 filters of 4×4, using a stride of 3 and no padding. What is the output volume?

1 / 1 point

- ○ $118 \times 118 \times 32$
- ◉ $40 \times 40 \times 32$
- ○ $118 \times 118 \times 16$
- ○ $40 \times 40 \times 16$

Expand

Correct

Correct, using the formula $n_H^{[l]} = \frac{n_H^{[l-1]} + 2 \times p - f}{s} + 1$ with $n_H^{[l-1]} = 121, p = 0, f = 4$, and $s = 3$ we get 40

5. You have an input volume that is 31x31x32, and pad it using “pad=1”. What is the dimension of the resulting volume (after padding)?

1 / 1 point

- ○ 31x31x34
- ◉ 33x33x32
- ○ 33x33x33
- ○ 32x32x32

Expand

Correct

Yes, if the padding is 1 you add 2 to the height dimension and 2 to the width dimension.

6. You have a volume that is $64 \times 64 \times 32$, and convolve it with 40 filters of 9×9, and stride 1. You want to use a "same" convolution. What is the padding?

1 / 1 point

- ○ 6
- ◉ 4
- ○ 0
- ○ 8

Expand

Correct

Yes, when using a padding of 4 the output volume has $n_H = \frac{64-9+2\times4}{1} + 1$.

7. You have an input volume that is 32x32x16, and apply max pooling with a stride of 2 and a filter size of 2. What is the output volume?

1 / 1 point

- [ ] 32x32x8
- [x] 16x16x16
- [ ] 16x16x8
- [ ] 15x15x16

Expand

Correct

Correct, using the following formula: $n_H^{[l]} = \frac{n_H^{[l-1]} + 2 \times p - f}{s} + 1$

8. Because pooling layers do not have parameters, they do not affect the backpropagation (derivatives) calculation.

1 / 1 point

- ◯ True
- ◉ False

Expand

Correct

Everything that influences the loss should appear in the backpropagation because we are computing derivatives. In fact, pooling layers modify the input by choosing one value out of several values in their input volume. Also, to compute derivatives for the layers that have parameters (Convolutions, Fully-Connected), we still need to backpropagate the gradient through the Pooling layers.

9. Which of the following are true about convolutional layers? (Check all that apply)

1 / 1 point

- [ ] It speeds up the training since we don't need to compute the gradient for convolutional layers.
- [x] Convolutional layers provide sparsity of connections.

Correct

Yes, this happens since the next activation layer depends only on a small number of activations from the previous layer.

- [x] It allows a feature detector to be used in multiple locations throughout the whole input volume.

Correct

Yes, since convolution involves sliding the filter throughout the whole input volume the feature detector is computed over all the volume.

Expand

Correct

Great, you got all the right answers.

10. The sparsity of connections and weight sharing are mechanisms that allow us to use fewer parameters in a convolutional layer making it possible to train a network with smaller training sets. True/False?

1 / 1 point

- [ ] False
- [x] True

Expand

Correct

Yes, weight sharing reduces significantly the number of parameters in a neural network, and sparsity of connections allows us to use a smaller number of inputs thus reducing even further the number of parameters.