✓ **Congratulations! You passed!**
TO PASS 80% or higher

[Keep Learning]

Retake the assignment in **7h 57m**

GRADE
100%

The Basics of ConvNets

LATEST SUBMISSION GRADE

100%

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 vertical edges

    ○ Detect image contrast

    ○ Detect horizontal edges

    > ✓ **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 300 by 300 color (RGB) image, and you are not using a convolutional network. If the first hidden layer has 100 neurons, each one fully connected to the input, how many parameters does this hidden layer have (including the bias parameters)?

    1 / 1 point

    ○ 27,000,001

    ○ 9,000,001

    ○ 9,000,100

    ◉ 27,000,100

    > ✓ **Correct**
    >
    > Correct, the number of weights is $300 \times 300 \times 3 \times 100 = 27,000,000$, when you add the bias terms (one per neurons) you get $27,000,100$.

3. Suppose your input is a 300 by 300 color (RGB) image, and you use a convolutional layer with 100 filters that are each 5x5. How many parameters does this hidden layer have (including the bias parameters)?

    1 / 1 point

    ◉ 7600

    ○ 2600

    ○ 7500

    ○ 2501

    > ✓ **Correct**
    >
    > Correct, you have $25 \times 3 = 75$ weights and 1 bias per filter. Given that you have 100 filters, you get 7,600 parameters for this layer.

4. You have an input volume that is 63x63x16, and convolve it with 32 filters that are each 7x7, using a stride of 2 and no padding. What is the output volume?

    1 / 1 point

    ○ 16x16x16

    ◉ 29x29x32

    ○ 29x29x16

    ○ 16x16x32

✓ **Correct**

Yes, $\frac{63-7+0\times2}{2}+1=29$ and the number of channels should match the number of filters.

---

5. You have an input volume that is 15x15x8, and pad it using "pad=2." What is the dimension of the resulting volume (after padding)?

⟳ 1 / 1 point

○ 17x17x10

⦿ 19x19x8

○ 19x19x12

○ 17x17x8

✓ **Correct**

Correct, padding is applied over the height and the width of the input image. If the padding is two, you add 4 to the height dimension and 4 the width dimension.

---

6. You have an input volume that is 63x63x16, and convolve it with 32 filters that are each 7x7, and stride of 1. You want to use a "same" convolution. What is the padding?

⟳ 1 / 1 point

⦿ 3

○ 1

○ 7

○ 2

✓ **Correct**

Correct, you need to satisfy the following equation: $n_H - f + 2 \times p = n_H$ as you want to keep the dimensions between the input volume and the output volume.

---

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

○ 16x16x8

○ 32x32x8

○ 15x15x16

⦿ 16x16x16

✓ **Correct**

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

---

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

⟳ 1 / 1 point

○ True

⦿ False

✓ **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. In lecture we talked about "parameter sharing" as a benefit of using convolutional networks. Which of the following statements about parameter sharing in ConvNets are true? (Check all that apply.)

⟳ 1 / 1 point

☐ It allows parameters learned for one task to be shared even for a different task (transfer learning).

☑ It reduces the total number of parameters, thus reducing overfitting.

✓ **Correct**

Yes, a convolutional layer uses parameter sharing and has usually a lot less parameters than a fully-connected layer.

☑ It allows a feature detector to be used in multiple locations throughout the whole input image/input volume.

✓ **Correct**

Yes, by sliding a filter of parameters over the entire input volume, we make sure a feature detector can be used in multiple locations.

☐ It allows gradient descent to set many of the parameters to zero, thus making the connections sparse.

10. In lecture we talked about "sparsity of connections" as a benefit of using convolutional layers. What does this mean? `1 / 1 point`

○ Each filter is connected to every channel in the previous layer.

○ Each layer in a convolutional network is connected only to two other layers

◉ Each activation in the next layer depends on only a small number of activations from the previous layer.

○ Regularization causes gradient descent to set many of the parameters to zero.

✓ **Correct**

Yes, each activation of the output volume is computed by multiplying the parameters from **only one filter** with a volumic slice of the input volume and then summing all these together.