Congratulations! You passed!

Grade received 80% | Latest Submission Grade 80% | To pass 80% or higher

Go to next item

1. Which of the following do you typically see in a ConvNet? (Check all that apply.) **0 / 1 point**

- [ ] FC layers in the first few layers
- [x] Multiple CONV layers followed by a POOL layer

✓ **Correct**
True, as seen in the case studies.

- [x] Multiple POOL layers followed by a CONV layer

! **This should not be selected**
No, this is not a common practice.

- [x] FC layers in the last few layers

✓ **Correct**
True, fully-connected layers are often used after flattening a volume to output a set of classes in classification.

Expand

✗ **Incorrect**
You chose the extra incorrect answers.

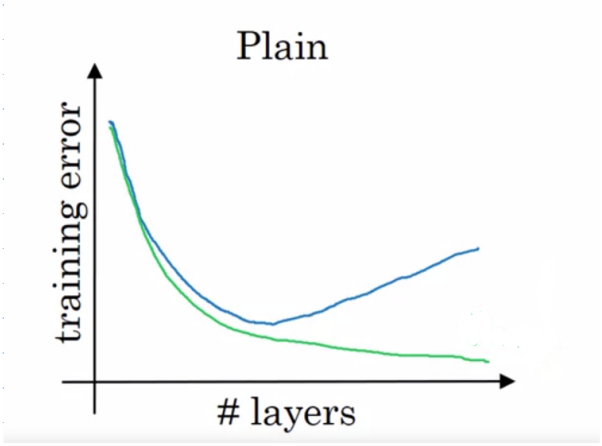
2. LeNet - 5 made extensive use of padding to create valid convolutions, to avoid increasing the number of channels after every convolutional layer. True/False? **1 / 1 point**

- ( ) True
- (•) False

Expand

✓ **Correct**
Yes, back in 1998 when the corresponding paper of LeNet - 5 was written padding wasn't used.

3. Based on the lectures, in the following picture, which curve corresponds to the expected behavior in theory, and which one corresponds to the behavior we get in practice? This when using plain neural networks. **1 / 1 point**



- ( ) The blue one depicts the theory, and the green one the reality.
- ( ) The blue one depicts the results in theory, and also in practice.
- ( ) The green one depicts the results in theory, and also in practice.
- (•) The green one depicts the results in theory, and the blue one the reality.

Expand

Correct
Yes, in theory, we expect that as we increase the number of layers the training error decreases; but in practice after a certain number of layers the error increases.

4. The following equation captures the computation in a ResNet block. What goes into the two blanks above? **1 / 1 point**

$$a^{[l+2]} = g(W^{[l+2]}g(W^{[l+1]}a^{[l]} + b^{[l+1]}) + b^{l+2} + \_\_\_\_\_) + \_\_\_\_\_$$

- 0 and $z^{[l+1]}$, respectively
- $a^{[l]}$ and 0, respectively
- 0 and $a^{[l]}$, respectively
- $z^{[l]}$ and $a^{[l]}$, respectively

Expand

Correct
Correct

5. Adding a ResNet block to the end of a network makes it deeper. Which of the following is true? **1 / 1 point**

- The number of parameters will decrease due to the shortcut connections.
- It shifts the behavior of the network to be more like the identity function.
- The performance of the networks is hurt since we make the network harder to train.
- The performance of the networks doesn't get hurt since the ResNet block can easily approximate the identity function.

Expand

Correct
Yes, as noted in the lectures in a ResNet block the computations are given by $a^{[l+2]} = g(W^{[l+2]}a^{[l+1]} + b^{[l+2]} + a^{[l]})$ thus if $W^{[l+2]}$ and $b^{[l+2]}$ are zero then we get the identity function.

6. Suppose you have an input volume of dimension n_H x n_W x n_C. Which of the following statements do you agree with? (Assume that the "1x1 convolutional layer" below always uses a stride of 1 and no padding.) **1 / 1 point**

- [x] You can use a 1x1 convolutional layer to reduce n_C but not n_H and n_W.

  ✓ Correct
  Yes, a 1x1 convolutional layer with a small number of filters is going to reduce n_C but will keep the dimensions n_H and n_W
- [ ] You can use a 1x1 convolutional layer to reduce n_H, n_W, and n_C.
- [x] You can use a 2D pooling layer to reduce n_H, n_H,
- [ ] You can use a 2D pooling layer to reduce $$n_H$$, $$n_W$$, and $$n_C$$.

Expand

Correct
Great, you got all the right answers.

7. Which of the following are true about the inception Network? (Check all that apply) **0 / 1 point**

- [x] Inception blocks allow the use of a combination of 1x1, 3x3, 5x5 convolutions and pooling by stacking up all the activations resulting from each type of layer.

  ✓ Correct
  Correct. The use of several different types of layers and stacking up the results to get a single volume is at the heart of the inception network.
- [ ] One problem with simply stacking up several layers is the computational cost of it.
- [x] Making an inception network deeper won't hurt the training set performance.

  ! This should not be selected
  Incorrect. As seen in the lectures in practice when stacking more layers the training performance might start increasing instead of decreasing.
- [ ] Inception blocks allow the use of a combination of 1x1, 3x3, 5x5 convolutions, and pooling by applying one layer after the other.

Expand

Incorrect
You didn't select all the correct answers

8. Which of the following are common reasons for using open-source implementations of ConvNets (both the model and/or weights)? Check all that apply. **1 / 1 point**

- [ ] The same techniques for winning computer vision competitions, such as using multiple crops at test time, are widely used in practical deployments (or production system deployments) of ConvNets.
- [x] Parameters trained for one computer vision task are often useful as pre-training for other computer vision tasks.

Correct
True

- [ ] A model trained for one computer vision task can usually be used to perform data augmentation for a different computer vision task.
- [x] It is a convenient way to get working with an implementation of a complex ConvNet architecture.

Correct
True

Expand

Correct
Great, you got all the right answers.

9. Which of the following are true about Depth wise-separable convolutions? (Choose all that apply) **1 / 1 point**

- [ ] They are just a combination of a normal convolution and a bottleneck layer.
- [x] They have a lower computational cost than normal convolutions.

Correct
Yes, as seen in the lectures the use of the depthwise and pointwise convolution reduces the computational cost significantly.

- [ ] The result has always the same number of channels n_c as the input.
- [x] They combine depthwise convolutions with pointwise convolutions.

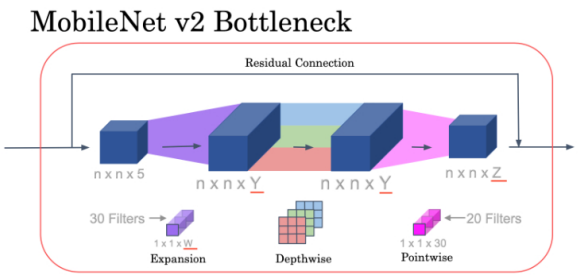
Correct
Correct, this combination is what we call depth wise separable convolutions.

Loading [MathJax]/jax/output/CommonHTML/jax.js

Expand

Correct
Great, you got all the right answers.

10. Fill in the missing dimensions shown in the image below (marked W, Y, Z). **1 / 1 point**



- ( ) W = 30, Y = 30, Z = 5
- ( ) W = 5, Y = 20, Z = 5
- (•) W = 5, Y = 30, Z = 20
- ( ) W = 30, Y = 20, Z =20

Expand

Correct