Deep convolutional models

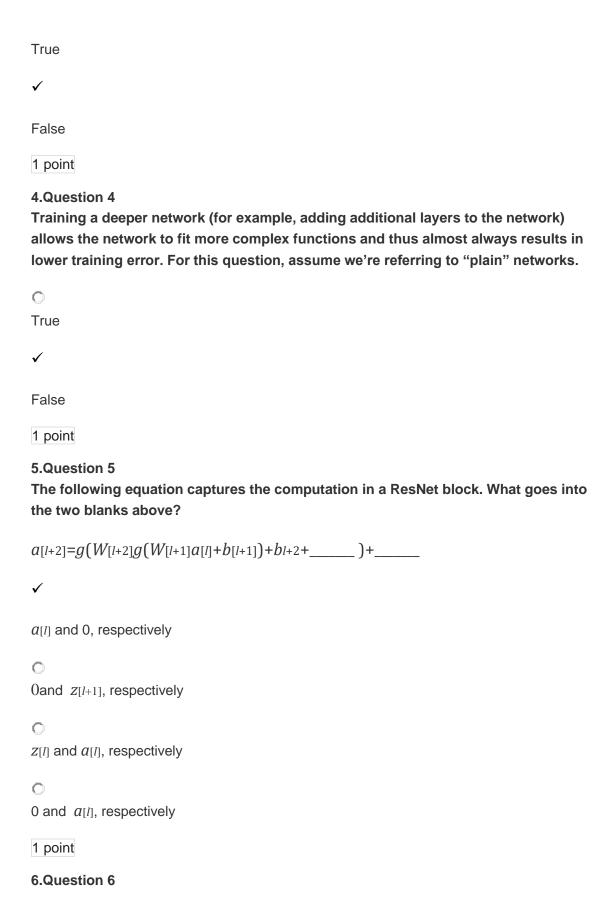
Which of the following do you typically see as you move to deeper layers in a ConvNet?

$n_H n_H$ and $n_W n_W$ increases, while $n_C n_C$ decreases
\square n_H n_H and n_W n_W increases, while n_C n_C also increases
✓
$n_H n_H$ and $n_W n_W$ decrease, while $n_C n_C$ increases
${f C}$ n_ ${f H}n{m H}$ and n_ ${f W}n{m W}$ decreases, while n_ ${f C}n{m C}$ also decreases
1 point
2.Question 2 Which of the following do you typically see in a ConvNet? (Check all that apply.)
Multiple CONV layers followed by a POOL layer
Multiple POOL layers followed by a CONV layer
Multiple POOL layers followed by a CONV layer

3.Question 3

In order to be able to build very deep networks, we usually only use pooling layers to downsize the height/width of the activation volumes while convolutions are used with "valid" padding. Otherwise, we would downsize the input of the model too quickly.

0



that apply.)
The skip-connections compute a complex non-linear function of the input to pass to a deeper layer in the network.
Using a skip-connection helps the gradient to backpropagate and thus helps you to train deeper networks
The skip-connection makes it easy for the network to learn an identity mapping between the input and the output within the ResNet block.
$\hfill \Box$ A ResNet with L layers would have on the order of $L^{\wedge}2L_2$ skip connections in total.
1 point
7.Question 7 Suppose you have an input volume of dimension 64x64x16. How many parameters would a single 1x1 convolutional filter have (including the bias)?
© 4097
© 2
O 1
✓
17
1 point
8. Question 8 Suppose you have an input volume of dimension $n H_{nH} \times n W_{nW} \times n C_{nC}$. Which

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

You can use a 1x1 convolutional layer to reduce n_Cnc but not n_HnH, n_Wnw.
\square You can use a 1x1 convolutional layer to reduce n_H n H, n_W n w, and n_C n c.
You can use a pooling layer to reduce n_Hn_H , n_Wn_W , and n_Cn_C .
You can use a pooling layer to reduce n_Hn_H , n_Wn_W , but not n_Cn_C .
1 point
9.Question 9 Which ones of the following statements on Inception Networks are true? (Check all that apply.)
Inception blocks usually use 1x1 convolutions to reduce the input data volume's size before applying 3x3 and 5x5 convolutions.
A single inception block allows the network to use a combination of 1x1, 3x3, 5x5 convolutions and pooling.
Inception networks incorporates a variety of network architectures (similar to dropout, which randomly chooses a network architecture on each step) and thus has a similar regularizing effect as dropout.
Making an inception network deeper (by stacking more inception blocks together) should not hurt training set performance.
1 point
10.Question 10 Which of the following are common reasons for using open-source implementations of ConvNets (both the model and/or weights)? Check all that apply.

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.

A model trained for one computer vision task can usually be used to perform data augmentation even for a different computer vision task.

~

It is a convenient way to get working an implementation of a complex ConvNet architecture.

~

Parameters trained for one computer vision task are often useful as pretraining for other computer vision tasks.

1 point