축하합니다! 통과하셨습니다!

받은 학점 100% 최신 제출물 학점 100% 통과 점수: 80% 이상

다음 항목으로 이동

1. When building a ConvNet, typically you start with some POOL layers followed by some CONV layers. True/False? 1 / 1점

- (●) False
- ( ) True

더 보기

맞습니다
Correct. It is typical for ConvNets to use a POOL layer after some Conv layers; sometimes even one POOL layer after each CONV layer; but is not common to start with POOL layers.

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 / 1점

- ( ) True
- (●) False

더 보기

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

3. Training a deeper network (for example, adding additional layers to the network) allows the network to fit more complex functions and thus almost always results in lower training error. For this question, assume we're referring to "plain" networks. 1 / 1점

- ( ) True
- (●) False

더 보기

맞습니다
Correct, Resnets are here to help us train very deep neural networks.

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

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

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

, respectively

$$z^{[l]}$$ and $$a^{[l]}$$, respectively

더 보기

맞습니다
Correct

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

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

더 보기

맞습니다
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 / 1점

- [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

- [x] You can use a 2D pooling layer to reduce n_H, n_W, but not n_C.

  Correct
  This is correct.

- [ ] You can use a 1x1 convolutional layer to reduce

  n_H

  ,

더 보기

맞습니다
Great, you got all the right answers.

7. Which ones of the following statements on Inception Networks are true? (Check all that apply.) 1 / 1점

- [x] Inception blocks usually use 1x1 convolutions to reduce the input data volume's size before applying 3x3 and 5x5 convolutions.

  Correct

- [x] A single inception block allows the network to use a combination of 1x1, 3x3, 5x5 convolutions and pooling.

  Correct

- [x] Making an inception network deeper (by stacking more inception blocks together) can improve performance, but can also lead to overfitting and increase in computational cost.

✓ Correct

☐ Inception networks incorporate 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.

↙↗ 더 보기

맞습니다
Great, you got all the right answers.

8. When having a small training set to construct a classification model, which of the following is a strategy of transfer learning that you would use to build the model? 1 / 1점

○ It is always better to train a network from a random initialization to prevent bias in our model.

○ Use an open-source network trained in a larger dataset. Use these weights as an initial point for the training of the whole network.

◉ Use an open-source network trained in a larger dataset freezing the layers and re-train the softmax layer.

○ Use an open-source network trained in a larger dataset, freeze the softmax layer, and re-train the rest of the layers.

↙↗ 더 보기

맞습니다
Yes, this is a strategy that can provide a good result with small data.

9. In Depthwise Separable Convolution you: 1 / 1점

☑ For the "Depthwise" computations each filter convolves with only one corresponding color channel of the input image.

✓ Correct

☑ You convolve the input image with $$n_c$$ number of $$n_f$$ x $$n_f$$ filters ($$n_c$$ is the number of color channels of the input image).

✓ Correct

☐ Perform one step of convolution.

☐ For the "Depthwise" computations each filter convolves with all of the color channels of the input image.

☐ You convolve the input image with a filter of $$n_f$$ x $$n_f$$ x $$n_c$$ where $$n_c$$ acts as the depth of the filter ($$n_c$$ is the number of color channels of the input image).

☑ Perform two steps of convolution.

✓ Correct

☑ The final output is of the dimension $$n_{out}$$ x $$n_{out}$$ x $$n^{'}_{c}$$ (where $$n^{'}_{c}$$ is the number of filters used in the pointwise convolution step).

✓ Correct

☐ The final output is of the dimension $$n_{out}$$ x $$n_{out}$$ x $$n_{c}$$ (where $$n_{c}$$ is the number of color channels of the input image).

↙↗ 더 보기

맞습니다
Great, you got all the right answers.

10. Suppose that in a MobileNet v2 Bottleneck block the input volume has shape $64 \times 64 \times 16$. If we use 32 filters for the expansion and 16 filters for the projection. What is the size of the input and output volume of the depthwise convolution, assuming a pad='same'?

1 / 1점

- (●) $64 \times 64 \times 32$ $64 \times 64 \times 32$
- ( ) $64 \times 64 \times 32$ $64 \times 64 \times 16$
- ( ) 32 × 32 × 32 $32 \times 32 \times 32$
- ( ) Processing math: 100% imes 16$$ $$64 \times 64 \times 32$$

더 보기

맞습니다

Correct, the size of the input and output volume of the depthwise convolution is determined by the number of filters in the expansion.