

Session 1/2 Quiz

Due Mar 11, 2019 at 5:30am**Points** 260**Questions** 15**Available** Mar 4, 2019 at 2pm - Mar 11, 2019 at 5:30am 7 days**Time Limit** 30 Minutes

This quiz was locked Mar 11, 2019 at 5:30am.

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	28 minutes	226.19 out of 260

Score for this quiz: **226.19** out of 260

Submitted Mar 6, 2019 at 4:01pm

This attempt took 28 minutes.

Question 1

5 / 5 pts

Let us assume, we have 100 images in our dataset, and 20 total batches. How many times would be performing a forward prop before we perform 1 backprop?

☐ 1☐ 100

Correct!☐ 20☒ 5**Question 2****5 / 5 pts**

Let us assume, we have 100 images in our dataset, and 20 total batches. In one (1) epoch, how many total backprops would we perform?

☐ 100☐ 5**Correct!**☒ 20☐ 1**Question 3****10 / 10 pts**

Let us assume the network below:

Layer 1 | input(244, 244, 3)

Layer 2 | conv2D(32, 3, 3, act=ReLU)

Layer 3 | conv2D(64, 3, 3, act=ReLU)

Layer 4 | conv2D(128, 5, 5, act=ReLU)

Layer 5 | conv2D(32, 1, 1, act=ReLU)

Layer 6 | MaxPooling(2)

Layer 7 | conv2D(64, 3, 3, act=ReLU)

...

What is the global receptive field for each kernel/filter in the Layer 7?

☐ 11x11

☒ 20x20

☐ 18x18

☐ 16x16

Correct!

Question 4

20 / 20 pts

Why do we add 1x1 layers? (select all which apply)

☒ To reduce the number of channels

Correct!

Correct!☒ It is computationally much efficient as compared to a 3x3 to reduce the number of channels☐ To increase the number of channels**Correct!**☒ Combine a large number of channels into smaller relevant ones**Question 5****20 / 20 pts**

Why do we apply MaxPooling? (select all which apply)

Correct!☒ To reduce the number of layers required in a network**Correct!**☒ To reduce the resolution of the layers**Correct!**☒ To increase the effective receptive field☐ To reduce the number of channels**Question 6****10 / 10 pts**

How many layers should we add to a DNN to make an efficient network for object detection?

Correct!

- ☒ As many as required for us to reach the required receptive field
- ☐ Does not matter
- ☐ Minimum 21 layers
- ☐ As many as we can add, as more layers are always good

Question 7**10 / 20 pts**

What among these things are not used anymore? (select all which apply)

Correct!

- ☒ DropOuts

Correct Answer

- ☐ Adding as many layers as possible irrespective of the image size

Correct Answer

- ☐ Sigmoid as activation function in CNNs

Correct!

- ☒ Fully Connected Layers

Question 8**10 / 10 pts**

Assume this short DNN below:

Layer 0 | model = Sequential()

Layer 1 | model.add(Convolution2D(34, 3, 3, activation='relu', input_shape=(28,28,1)))

Layer 2 | model.add(Convolution2D(32, 1, activation='relu'))

Layer 3 | model.add(Convolution2D(32, 3, 3, activation='relu'))....

What is the **total number of parameters** we have added in the above 3 layers? (remember we have by default added biases).

i.e. Convolution2D(34, 3, 3) has $3 \times 3 \times 34 + 34 = 340$ parameters.

Correct!

☒ 10708

☐ 9652

☐ 724

☐ 10610

Question 9

20 / 20 pts

Consider the following network:

input(200, 200, 1)

conv2D(128, 5, 5)

MaxPooling(2)

conv2D(128, 11, 11)

conv2D(32, 11, 11)

conv2D(128, 1, 1)

...

What all statements apply below?

Correct!

☒ Bad network, we should not have used 1x1 to increase the number of channels

Correct!

☒ Bad network, we should not have performed MaxPooling so early

Correct!

☒ Bad network, we should have used 3x3 kernels for best performance

☐ Nothing can be said about this network, as it depends on how long we train it.

Question 10

20 / 20 pts

(Please select all which apply)

When we select a batch from the dataset, we should make sure that

Correct!

☒ it represents all the classes equally

Correct!

☒ it is never from the "test" dataset

Correct!

☒ it is utilizing full GPU resources

Correct!☒ it is randomly shuffled**Question 11****13.33 / 20 pts**

Why do we not use Fully Connected (FC) layers these days? (Select all which apply)

Correct!

If we use FC we force our network to use only a specific size of input image, but we want a network which can process images of all sizes

Correct!

FC layers loses all the spatial information especially required in vision domain

Correct!

FC layers add a lot more parameters than Convolution layers and hence are slower to process

You Answered

We need different kind of activation functions for FC layers which are not efficient

Question 12**20 / 20 pts**

Adding a bias in a network is useful.

Select the ones which apply:

Correct!☒ it is not useful, and people are today cautious about removing them**Correct!**☒ it is useful for simple, fully connected networks, where the problem we are solving is also linear/simple**Correct!**☒

it is not useful, as in vision domain we have too many parameters, and 1 bias variable cannot help resolve non-linear models we have

Correct!☒ it is not helpful, but sticking with us because of it's historicity**Question 13****20 / 20 pts**

Select all which are true below:

We use squared loss because:

Correct!☒ it is a continuous function**Correct!**☒ we want to punish larger values more (especially when our network has just been initialized)☐ there are no other loss functions**Correct!**☒ it always gives us positive loss/cost values

Question 14**20 / 20 pts**

Would you add MaxPooling just before final prediction layers? (Select all which apply)

- ☐ no, because it will reduce the number of parameters
- ☐ yes, as it helps reduce the resolution of the channels
- ☐ yes, because it helps reduce the total number of parameters
- ☒ no, because it will reduce the rich information in the last layers required for accurate predictions

Correct!**Question 15****22.86 / 40 pts**

Select all which are true below:

- ☐ Accuracy of a model is dependent on the total number of layers in a model
- ☐ It is guaranteed to reach global minima for our DNN if we select right number of layers and kernels
- ☐ CNN cannot be used for audio or text related problems

Correct!

If GPU resources allow, we would rather prefer to make our batch size equal to total number of images for backpropagation



A model with a larger number of parameters will always have higher accuracy than a fewer number of parameters



Total number of kernels is dependent on the total number of input channels

Correct!

We initialize all our kernels/filters randomly



Total number of output channels is dependent on the total number of input channels

Correct!

We mostly use ReLU as activation function, because it is efficient, and accelerated. In all other activation functions, increase in computation requirements does not justify minimal increase in accuracy



Total number of channels in the kernel is dependent on the total number of output channels

Correct!

We always add 2x2 MaxPooling, else we would loose too much of the information

Correct!

Softmax does not improves the accuracy of the network, but just creates large separation in prediction values, and hence might be misleading

Correct!

We always use 3x3 kernels (with the exception of 1x1 sometimes)

You Answered

☒ A model with large number of kernels will always have higher accuracy than a fewer number of kernels

You Answered

☒ We use 1x1 to increase the total number of layers

☐ Softmax improves the accuracy of a network

☐ It is possible to get exact same accuracy everytime we train a network from scratch

☐ 1x1 is mostly used to increase the number of channels

Correct!



Our interception point (location where we add MaxPooling and 1x1 layers), are different for each dataset, and depends on the size of the image as well as the features we want to extract

You Answered

☒ We add as many layers as possible, GPU resources being the only constraint

Quiz Score: **226.19** out of 260