Session 4 - Quiz

- Due Nov 17 at 11:30pm
- Points 190
- Questions 15
- Available Nov 9 at 12:30pm - Nov 17 at 11:30pm
- Time Limit 31 Minutes

Instructions

You have **30 minutes** to solve this quiz!

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	30 minutes	147.86 out of 190

Score for this quiz: 147.86 out of 190
Submitted Nov 16 at 7:29am
This attempt took 30 minutes.

Question 1
10 / 10 pts

PyTorch does not provide debugging tools because?

- Debugging is difficult.
- PyTorch does not create as many bugs as Tensorflow 😄

Correct!

- It is literally Python, so we can se any python based debugging tools
- PyTorch is not focused on debugging, just writing models 😎

Question 2
22.86 / 40 pts

Match the following

Correct!
PyTorch

Dynamic Graphs

Correct!
Tensorflow (without Keras)

Static Graphs

Correct!
Large dependencies

TensorFlow

You Answered
Pythonic

torch.optim

PyTorch
Correct!
Better GPU/CPU control

PyTorch

You Answered
Closer to Numpy

Tensor

PyTorch
You Answered
Numpy 3D array should be called

Matrix

Tensor
Other Incorrect Match Options:

- Matrix

- torchvision
- torch.optim

Question 3
10 / 10 pts

Which of these are **not** a tensor?

- a 2d matrix

Correct!

- All are tensors
- a scalar component
- a 1d vector

Question 4
3.33 / 10 pts

What is **not true** about torch.Tensor?

Correct!

- It uses a default float64 tensor

Correct Answer

- It produces a new tensor with same dtype
- Other tensors can inherit from it

Correct Answer

- It shares the underlying memory with numpy

Question 5
10 / 10 pts

What is **not true** about torch.tensor?

Correct!

- It uses a default float64 tensor

Correct!

- It shares the underlying memory with numpy
- It produces a new tensor with same dtype
- it always copies the data

Question 6
10 / 10 pts

What all is **not true** about torch.as_tensor?

- [ ] It shares the underlying memory with numpy

Correct!

- [x] it always copies the data
- [ ] It produces a new tensor with same dtype

Correct!

- [x] it cannot accept numpy data

Correct!

- [x] It uses a default float64 tensor

Question 7
10 / 10 pts

What all is **not true** about torch.from_numpy?

- [ ] It shares the underlying memory with numpy
- [ ] It produces a new tensor with same dtype

Correct!

- [x] it always copies the data

Correct!

- [x] It uses a default float64 tensor

Question 8
5 / 5 pts

Consider this Python code and tell whether the **id(x)** before and after the operation would be the same or different.

```
x = 14
print(id(x))
x *= 2
print(id(x))
```

- ( ) may be same
- ( ) may be different
- ( ) same

Correct!

different

Question 9

23.33 / 35 pts

Consider this Pytorch code:

```
x =torch.randn(5, 5) #requires_grad = False by defaults
y =torch.randn(5, 5) #requires_grad = False by defaults
z =torch.randn((5, 5), requires_grad=True)
a = x + y
b = a + z
```

Correct Answer

- [ ] gradient for b is constant

Correct!

- [x] a.requires_grad is False

Correct Answer

- [ ] only if all inputs don't require gradient, the output also won't require it

Correct!

- [x] If there's a single input to an operation that requires gradient, its output will also require gradient.

Correct!

- [x] Backward computation is never performed in the subgraphs, where all Tensors didn't require gradients.

Correct!

- [x] b.requires_grad is True

Question 10

6.67 / 10 pts

Which among the below **can be** made from a tensor of shape **[9, 40, 6]**?

- [ ] [3, 12, 12, 4]

Correct!

- [x] [1, 2160]

Correct!

- [x] [8, 270]

Correct Answer

- [ ] [16, [27, 5]]

Question 11

5 / 5 pts

We want the value of the largest element in a 1D tensor (t), what would you use?

- t.argmax

Correct!

- t.max

Question 12

5 / 5 pts

We want the position of the largest element in a 1D tensor, what would you use?

Correct!

- t.argmax
- t.max

Question 13

10 / 10 pts

```
t = torch.tensor([
    [1, 0, 0, 2],
    [0, 3, 3, 0],
    [4, 0, 0, 5]
], dtype=torch.float32)
t.max(dim=1) #what does this line print?
```

- tensor([4, 3, 3, 5])

Correct!

- tensor([2, 3, 5])

Question 14

6.67 / 10 pts

Select all that is **true** about this code:

```
sample = next(iter(something))
```

- [ ] (fix) if we execute the code twice, the first output will be different from the second output

Correct Answer

- [ ] If the "something" object has not implemented __getitem__(), exception will be raised

Correct!

- [x] The iter() function creates an object which can be iterated one element at a time.

Correct!

- [x] If "something" was written in such a way that it returns more than 1 objects/images/sentences/etc, sample is a "batch"

Question 15

10 / 10 pts

What does this piece of code do?

```
optimizer.zero_grad( )
```

Correct!

- (●) You have to zero the gradients because PyTorch gives you the possibility to accumulate gradients, eg., when d multiple passes of backprop on the same graph
- ( ) Because it's likely that you want to perform mini-batch gradient descent. Without zeroing you'd end up with (full) batch gradient descent, more or less, since the gradient would keep accumulating over time.

Quiz Score: 147.86 out of 190