

Week 3 Quiz

LATEST SUBMISSION GRADE

100%

1. If X is the standard notation for the input to an RNN, what are the standard notations for the outputs?

1 / 1 point

- ☐ Y
- ☐ H
- ☒ \hat{Y} and H
- ☐ \hat{H} and Y

✓ Correct

2. What is a sequence to vector if an RNN has 30 cells numbered 0 to 29

1 / 1 point

- ☐ The total \hat{Y} for all cells
- ☐ The \hat{Y} for the first cell
- ☐ The average \hat{Y} for all 30 cells
- ☒ The \hat{Y} for the last cell

✓ Correct

3. What does a Lambda layer in a neural network do?

1 / 1 point

- ☒ Allows you to execute arbitrary code while training
- ☐ Pauses training without a callback
- ☐ There are no Lambda layers in a neural network
- ☐ Changes the shape of the input or output data

✓ Correct

4. What does the axis parameter of `tf.expand_dims` do?

1 / 1 point

- ☐ Defines the axis around which to expand the dimensions
- ☐ Defines the dimension index to remove when you expand the tensor
- ☐ Defines if the tensor is X or Y
- ☒ Defines the dimension index at which you will expand the shape of the tensor

✓ Correct

5. A new loss function was introduced in this module, named after a famous statistician. What is it called?

1 / 1 point

- ☐ Hyatt loss
- ☒ Huber loss
- ☐ Hawking loss
- ☐ Hubble loss

✓ Correct

6. What's the primary difference between a simple RNN and an LSTM

1 / 1 point

- ☒ In addition to the H output, LSTMs have a cell state that runs across all cells
- ☐ LSTMs have a single output, RNNs have multiple
- ☐ LSTMs have multiple outputs, RNNs have a single one
- ☐ In addition to the H output, RNNs have a cell state that runs across all cells

✓ Correct

7. If you want to clear out all temporary variables that tensorflow might have from previous sessions, what code do you run?

1 / 1 point

- ☐ `tf.cache.clear_session()`
- ☐ `tf.keras.clear_session`
- ☐ `tf.cache.backend.clear_session()`
- ☒ `tf.keras.backend.clear_session()`

✓ Correct

8. What happens if you define a neural network with these two layers?

1 / 1 point

```
tf.keras.layers.Bidirectional(tf.keras.layers.LSTM(32)),
```

```
tf.keras.layers.Bidirectional(tf.keras.layers.LSTM(32)),
```

```
tf.keras.layers.Dense(1),
```

- ☐ Your model will fail because you have the same number of cells in each LSTM
- ☐ Your model will compile and run correctly
- ☒ Your model will fail because you need `return_sequences=True` after the first LSTM layer
- ☐ Your model will fail because you need `return_sequences=True` after each LSTM layer

✓ Correct