Congratulations! You passed!

TO PASS 80% or higher

Keep Learning

GRADE
87.5%

Week 3 Quiz

LATEST SUBMISSION GRADE
87.5%

1. If I put a dropout parameter of 0.2, how many nodes will I lose? — 1 / 1 point

- (•) 20% of them
- ( ) 2% of them
- ( ) 20% of the untrained ones
- ( ) 2% of the untrained ones

2. Why is transfer learning useful? — 1 / 1 point

- ( ) Because I can use all of the data from the original training set
- ( ) Because I can use all of the data from the original validation set
- (•) Because I can use the features that were learned from large datasets that I may not have access to
- ( ) Because I can use the validation metadata from large datasets that I may not have access to

✓ **Correct**

3. How did you lock or freeze a layer from retraining? — 0 / 1 point

- ( ) tf.freeze(layer)
- ( ) tf.layer.frozen = true
- (•) tf.layer.locked = true
- ( ) layer.trainable = false

4. How do you change the number of classes the model can classify when using transfer learning? (i.e. the original model handled 1000 classes, but yours handles just 2) — 1 / 1 point

- ( ) Ignore all the classes above yours (i.e. Numbers 2 onwards if I'm just classing 2)
- ( ) Use all classes but set their weights to 0
- (•) When you add your DNN at the bottom of the network, you specify your output layer with the number of classes you want
- ( ) Use dropouts to eliminate the unwanted classes

✓ **Correct**

5. Can you use Image Augmentation with Transfer Learning Models? — 1 / 1 point

- ( ) No, because you are using pre-set features
- (•) Yes, because you are adding new layers at the bottom of the network, and you can use image augmentation when training these

6. Why do dropouts help avoid overfitting? — 1 / 1 point

- (•) Because neighbor neurons can have similar weights, and thus can skew the final training
- ( ) Having less neurons speeds up training

✓ **Correct**

- (•) The network would lose specialization to the effect that it would be inefficient or ineffective at learning, driving accuracy down
- ( ) Training time would increase due to the extra calculations being required for higher dropout

✓ **Correct**

8. Which is the correct line of code for adding Dropout of 20% of neurons using TensorFlow — 1 / 1 point

- ( ) tf.keras.layers.Dropout(20)
- ( ) tf.keras.layers.DropoutNeurons(20),
- (•) tf.keras.layers.Dropout(0.2),
- ( ) tf.keras.layers.DropoutNeurons(0.2),