**Chapter 2 Quiz**

*Before we begin: the mathematical building blocks of neural networks*

**Questions**

1. How are layers and data distillation related?
2. What is another way to say “densely connected”?
3. What does a 10-way softmax layer do in the book’s example on page 28?
4. What does a loss function do?
5. What is an optimizer?
6. What is 1 reason why a model performs much worse on new data when it has very high accuracy on test data?
7. What is a tensor?
8. How many axes does a scalar have?
9. If a tensor has 3 axes, how many ranks does it have?
10. How do you find the dimensions of a Numpy array ‘y’?
11. What is the difference between a 3-D vector and a 3-D tensor?
12. Name the 3 key attributes of a tensor.
13. Write the number of dimensions for each type of tensor: images, timeseries data, vector data, video
14. Places the following words in the correct order for the Keras layer instance.

\_\_\_\_\_\_\_.\_\_\_\_\_\_\_.\_\_\_\_\_\_\_(512, activation=\_\_\_\_\_\_\_)

Dense keras layers ‘relu’

1. What is a dot product?
2. What is a simple explanation for the purpose of the ‘relu’ operation?
3. What type of operations useful for vectorized implementations?
4. What happens with addition when the shapes of two tensors being added differ?
5. The product between a 3-D vector and another 3-D vector results in a 3-D vector. (True or False)
6. When taking the dot product of a matrix and vector, the operation will auto adjust to any dimension variants. (True or False).
7. If we are taking the dot-product of matrix x (2x3) and y, what must be the dimensions of y?
8. Write the code to reshape 5x3 array ‘w’ into a 15x1 array. What if we want a 3x5 array?
9. What is it called when weight matrices are filled with small random values?
10. A gradient is a derivative of what?
11. Computing the gradient of the loss with regard to the network’s parameters is also called \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.
12. What is a forward pass?
13. SGD stand for? A step in SGD is also called what? What happens if your steps are too large?
14. What is the different between a mini-batch SGD and a batch SGD. What are the pros and cons of using a batch SGD?
15. Make sure your mini-batches are constant over all types of experiments (True or False).
16. What has historically been a source of issues in the world of deep-learning research?
17. Name other optimization methods or optimizers.
18. What two things does momentum address in SGD? Why are these important?
19. What is backpropagation?
20. What is symbolic differentiation?

**Answers**

1. (28) Layers implement “progressive data distillation” by filtering data and extracting representations from the data.
2. (28) Fully connected
3. (29) Returns an array of 10 probability scores assigning probabilities to each digit.
4. (29) Measures the performance of the model on training data
5. (29) The mechanism through which the network will update tself based on the data it sees and its loss function
6. (30) overfitting
7. (31) data stored in a multidimensional Numpy array; container for data
8. (31) 0
9. (31) 3
10. (31) y.ndim
11. (31) 3-D vector has 3 entries. Dimensions for a vector refer to the number of entries.
12. (32) # of axes(rank), shape, data type
13. (35) Images=4, timeseries data=3, vector data=2, video=5
14. (38) keras.layers.Dense(512, activation='relu')
15. The result of multiplying 2 arrays.
    1. <https://docs.scipy.org/doc/numpy/reference/generated/numpy.dot.html>
    2. <https://hackernoon.com/learning-ai-if-you-suck-at-math-p6-math-notation-made-easy-1277d76a1fe5>
16. To speed up training.
    1. https://stats.stackexchange.com/questions/226923/why-do-we-use-relu-in-neural-networks-and-how-do-we-use-it
17. (38) Element-wise operations
18. (39) Axes (called *broadcast axes*) are added to the smaller tensor to match the ndim of the larger tensor. 2 The smaller tensor is repeated alongside these new axes to match the full shape of the larger tensor.
19. (41) False. Results in a scalar.
20. (41) False. The first dimension of the matrix must be the same as the dimension of the vector. The operation results in a vector with the same shape as the vector in the operation.
21. (42) 3x2
22. (42-43) w = w.reshape((15, 1)), w = w.reshape((3x5)) or w = np.transpose(w)
23. (46) random initialization
24. (48) A tensor operation
25. (49) Backward pass
26. (46) A step to obtain predictions.
27. (49) Stochastics Gradient Descent. Learning rate. Your updates may end up taking you to completely random locations on the curve; jumping around instead of gradually decreasing loss rate.
28. (50) Batch takes all data. Mini-batch takes a mini group of data. Pros: more accurate. Cons: more expensive computationally.
29. (50) False. Find a reasonable size depending on the data and experiment.
30. (50) Intuitions developed through low-dimensional representations may not always be accurate in practice.
31. (50) Adagrad. RMSProp.
32. (50) convergence speed and local minima. The optimization process could get stuck at a local minimum instead of moving towards the global minimum so implement a constant momentum factor.
33. (52) Start with the final loss value and work backwards i.e. bottom layers to top layers, and use the chain rule to assess the contribution of each parameter on the loss value AKA reverse-mode differentiation.
34. (52) Computing a gradient function for the chain that maps network parameter values to gradient values. Given a chain of operations with a known derivative.