Homework 5 – Report

**Question 1**

****

****

**Question 2**A white screen with a white background

Description automatically generated with medium confidence

**A white screen with black text

Description automatically generatedA white background with text

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated**

I observe that the best structures were [784, 1024, 512, 10] and [784, 256, 128, 10] since they both had approximately 88.74% test accuracy. ReLU also performed the best because it had 88.74% test accuracy.

**Question 3**

**Implementation Parts a-e**

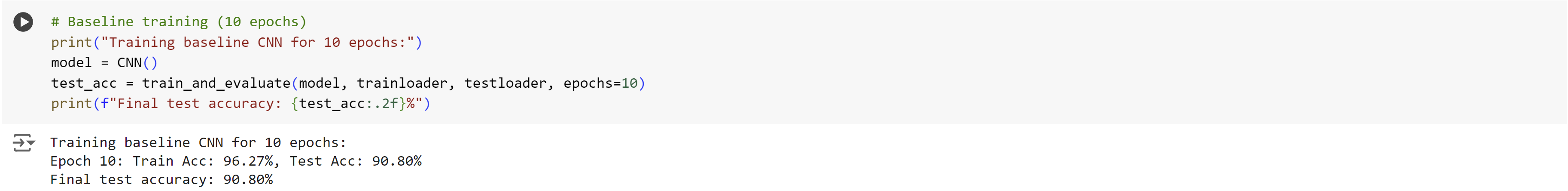
I commented specifically the parts of the code that have to do with each part. There’s some experimentation code as well.

**A white screen with black text

Description automatically generatedA white screen with black text

Description automatically generatedA white background with black text

Description automatically generated**

****

**Experimentation:**

**Part a**

**A close-up of a computer screen

Description automatically generated**

**A graph with a red line and blue line

Description automatically generated**

There is a steady increase in training accuracy that converges around 100% accuracy. However there is not steady improvement in the validation accuracy. A possible reason for this is that the model is overfitting the data and can’t seem to generalize.

**Part b**

**A close-up of a computer screen

Description automatically generatedA graph with a blue line and red line

Description automatically generated  
Part c**

**A close-up of a computer screen

Description automatically generated**

**Part d**

**A close-up of a computer screen

Description automatically generated**

**Analysis:**

**Part a**

I notice that without dropping, the training accuracy reaches 100% accuracy really fast and that the testing accuracy does not increase steadily. However, with dropout, the training accuracy increases much slower and reaches 97% instead of 100% and the testing accuracy seems to be a little more stable but still only reaches around 92%

**Part b**

The single layer definitely shows more overfitting than the CNN with two layers. I can prove this because the training accuracy for one layer was 96.32% while the training accuracy for two layers was 90.89% and the testing accuracy for the single layer was 91.15% and the testing accuracy for two layers was 90.72% which is much closer to its training accuracy than the single layer. However, I did notice that two layers performed slightly worse.

**Part c**

When the learning rate was 0.001, the test accuracy was 88.23% while the test accuracy of the 0.1 learning rate was only 10%, which is significantly worse. I think we could increase the test accuracy by choosing something a little higher, yet closer to 0.001 like 0.03. This will be more effective learning without overshooting the way 0.1 did.

Written Exercises

**Question 1**

**Part a**

Yes, a multilayer perceptron with one hidden layer of ReLU units represent any function that an MLP with linear hidden units can represent and you would need either the same or less amount of nodes since linear networks can be redundant.

**Part b**

No, MLP with linear units can’t represent the mappings that an MLP with ReLU units can represent no matter how many hidden nodes you add.