Nathon Tadeo
Student ID: 801265462
12/9/24
Homework 7
Github: https://github.com/nathon-tadeo/Intro-to-ML/blob/main/homework_7_intro_to_ml.ipynb

Problem 1 (50 pts):

a. Build a Convolutional Neural Network, like what we built in lectures to classify the images across all 10 classes in CIFAR 10. You need to adjust the fully connected layer at the end properly concerning the number of output classes. Train your network for 200 epochs. Report your training time, training loss, and evaluation accuracy after 200 epochs. Analyze the results in your report and compare them against those of a fully connected network (homework 2) regarding training time, achieved accuracy, and model size.

Using the Neural network model from Homework 6, it was easily modified to incorporate Problem 1a. Since CIFAR 10 went through the same preprocessing. The model design training and structure remained consistent, however, transitioning from a fully connected network to a convolutional neural network had some key differences. The model's training was around 3742.66 seconds while the previous homework's was only 2056.88 seconds. The Convolutional Neural Network trained for 1685.78 seconds longer than the fully connected network. This could be because a fully connected network processes flattened inputs directly. At the same time, a convolutional model involves multiple sliding windows over the input image, which retains the image's structure. The Convolutional NN did achieve a higher accuracy of 0.7079 while the fully connected NN had a lower accuracy of 0.5198. This equates to an accuracy difference of 0.1881. This lower accuracy of the fully connected network could be attributed to the flattening of the input images. Due to the convolutional filters, the CNN model is a slightly bigger model than the fully connected NN. Training loss and validation accuracy between both models can be found in Figures 1.1 and 1.2.

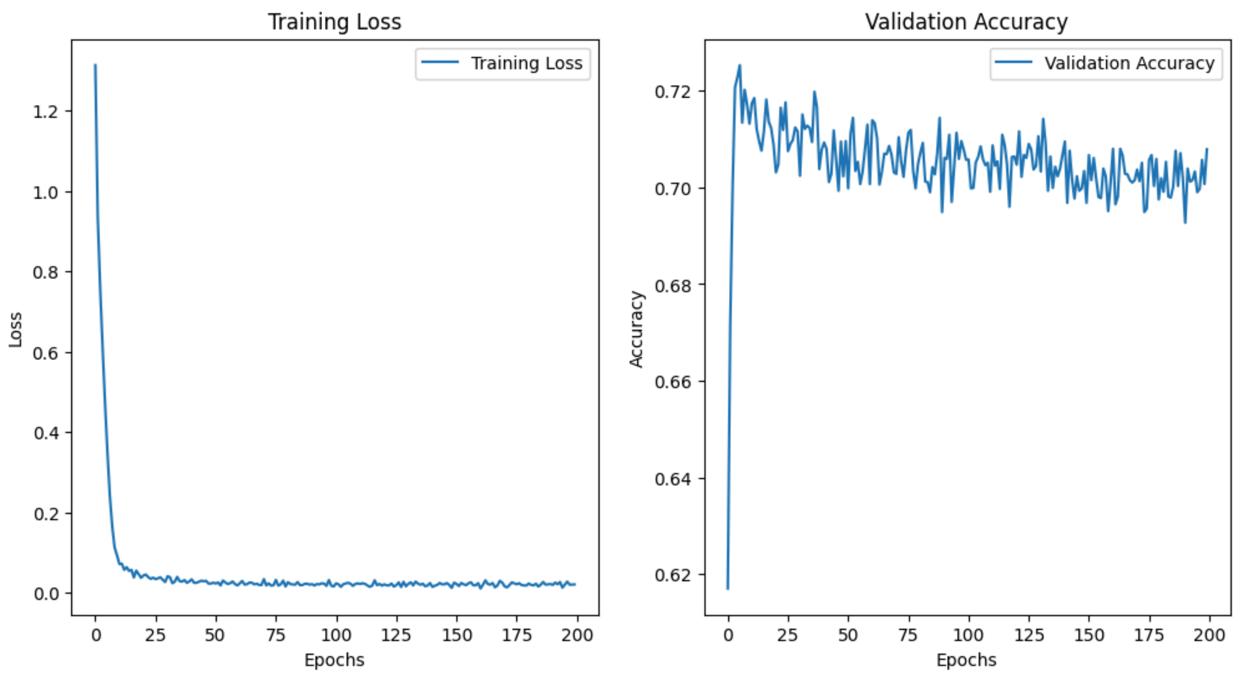

Figure 1.1 CIFAR-10 Training Loss and Validation Accuracy (Convolutional Neural Network)

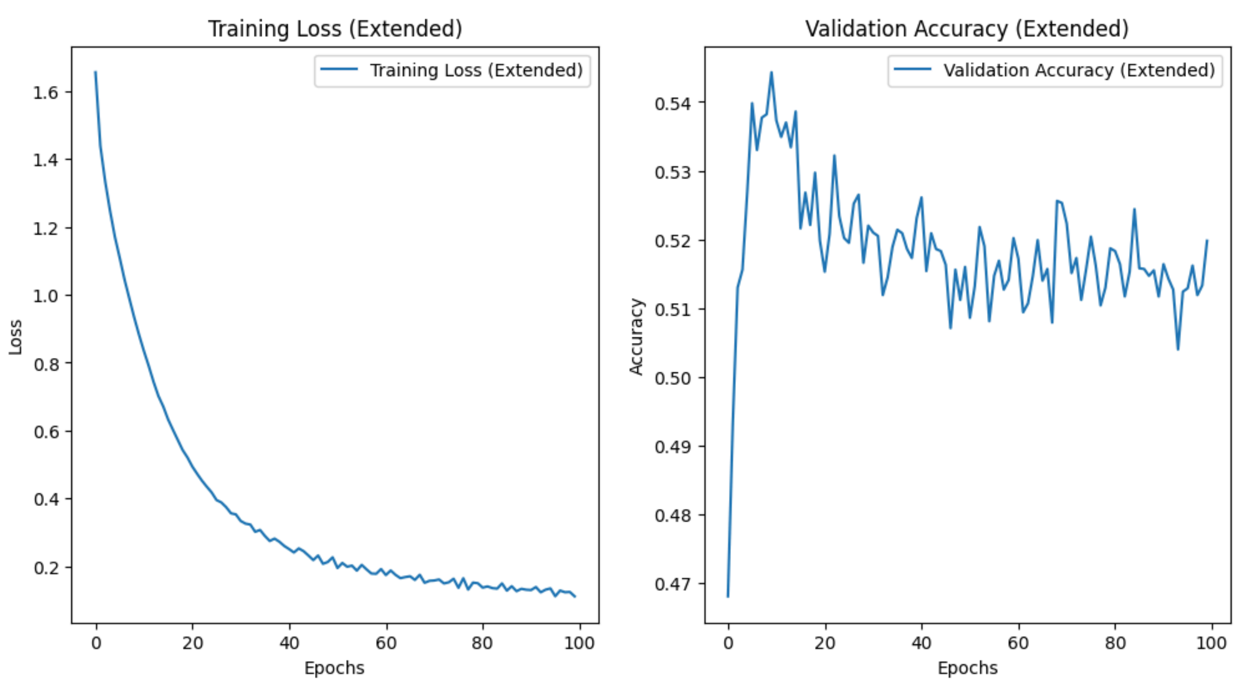

Figure 1.2 CIFAR-10 Training Loss and Validation Accuracy (256 Nodes 3 Hidden Layers)

b. Extend your CNN by adding one more additional convolution layer followed by an activation function and pooling function. You also need to adjust your fully connected layer properly with respect to intermediate feature dimensions. Train your network for 200 epochs. Report your training time, loss, and evaluation accuracy after 200 epochs. Analyze your results in your report and compare your model size and accuracy over the baseline implementation in Problem1.a. Do you see any over-fitting?

Similar to problem 1a, in terms of preprocessing the CIFAR-10, except an additional convolution layer and an activation and pooling function were added. This modification of the model equated to 3910.61 seconds of training while 1a only took 3742.66 seconds. These 167 more seconds of training allowed for the model to obtain a higher accuracy of 0.7445, only 0.0336 higher than the CNN model from 1a. The extended CNN will have a slightly bigger model size. No over-fitting in both model. The training loss and validation accuracy of both models can be seen in Figures 2.1 and 2.2.

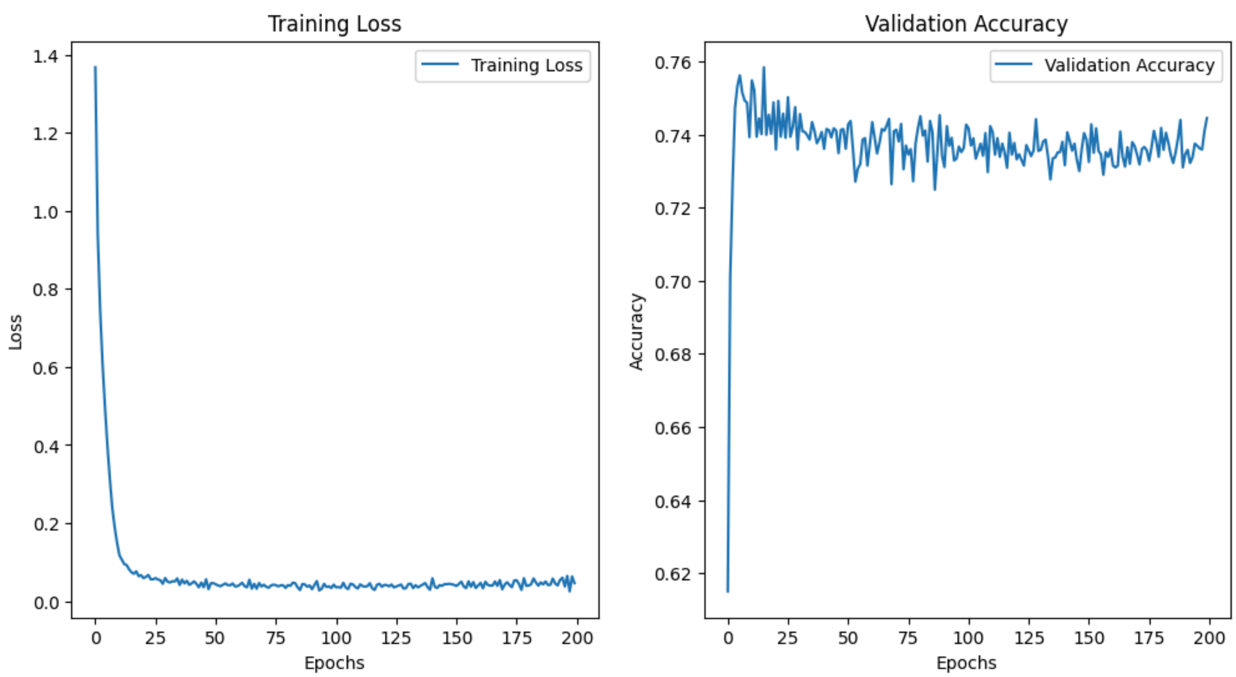

Figure 2.1 CIFAR-10 Training Loss and Validation Accuracy (Convolutional Neural Network added layer w/activation function and pooling function)

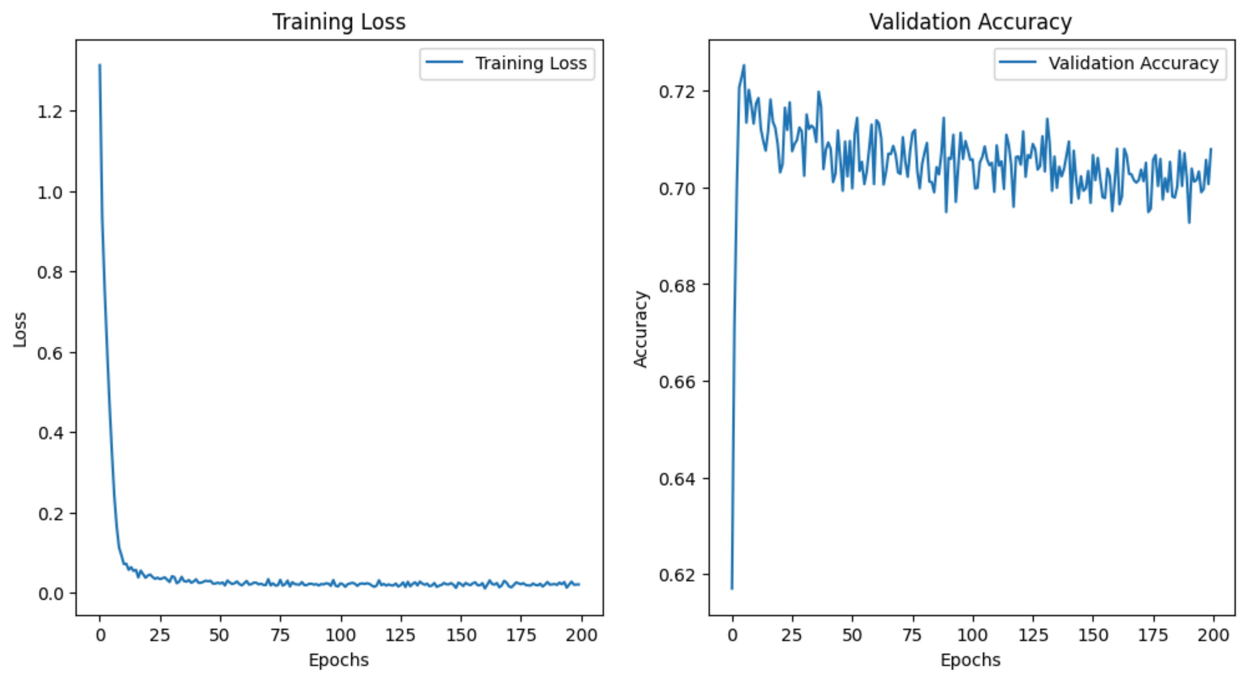

Figure 2.2 CIFAR-10 Training Loss and Validation Accuracy (Convolutional Neural Network)

Problem 2 (50pts)

a. Build a ResNet-based Convolutional Neural Network, like what we built in lectures (with skip connections), to classify the images across all 10 classes in CIFAR 10. For this problem, let's use **10** blocks for ResNet and call it ResNet-10. Use similar dimensions and channels as we need in lectures. Train your network for 200 epochs. Report your training time, training loss, and evaluation accuracy after 200 epochs. Analyze your results in your report and compare them against problem 1.b on training time, achieved accuracy, and model size.

The preprocessing remained the same, and the CIFAR-10 dataset was classified across all 10 classes while using the ResNet-based architecture. The model was designed with 10 residual blocks, with skip connections. The ResNet-based model completed training in 11802.05 seconds which was 7891.44 seconds longer than the extended CNN in Problem 1b (3910.61 seconds). This long run time fortunately improved the accuracy to 0.8595 this is a 0.115 accuracy increase compared to Problem 1b (0.7445). The model size of the ResNet-based CNN is once again, bigger than the previous model (Problem 1b). Considering the runtime, it would only be worthwhile to use this model if our hardware is more than capable in order to have a reasonable training time. The training loss and validation accuracy of both models can be seen in Figures 3.1 and 3.2.

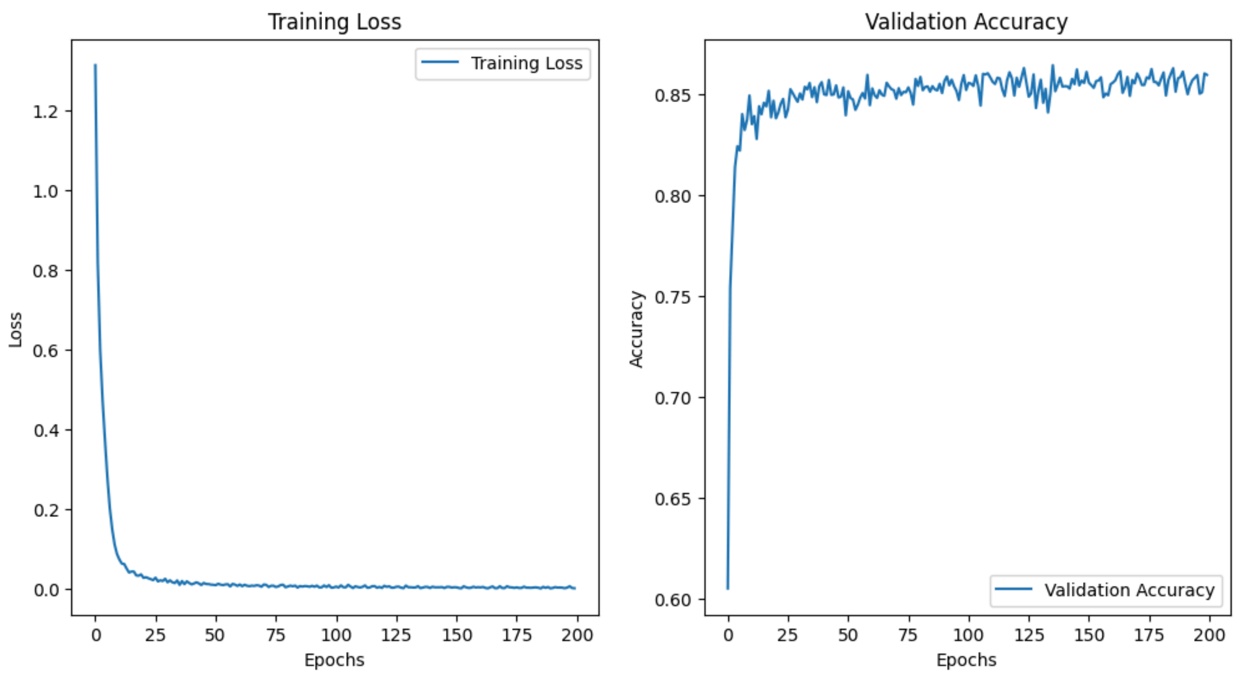

Figure 3.1 CIFAR-10 Training Loss and Validation Accuracy (ResNet-based)

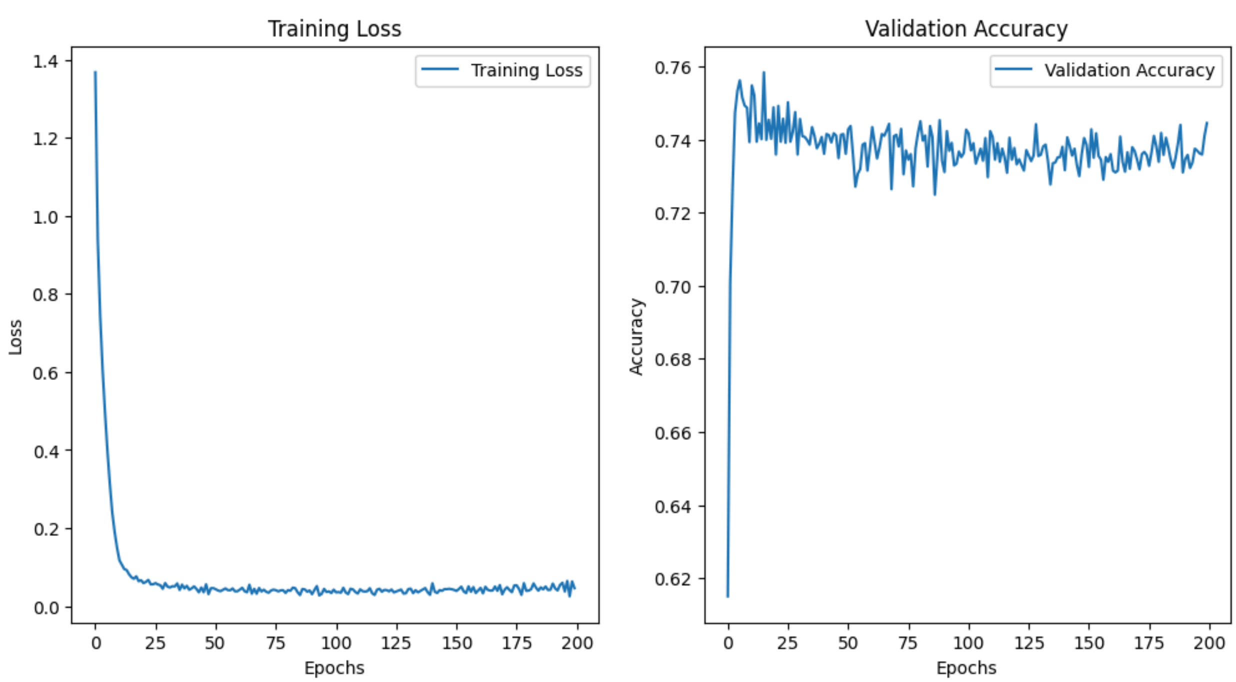

Figure 3.2 CIFAR-10 Training Loss and Validation Accuracy (Extended Convolutional Neural Network)