The cifar10 dataset is a dataset of 60,000, 32x32 images, with 10 classes and 6,000 images per class. These classes are all mutually exclusive, meaning that they only contain the class they are labeled with, and no other class. For the network I created, I used an architecture found in an online tutorial for creating a cifar10 classifier. This is a convolutional neural network, that outputs the relative likelihoods for the 10 different classes in the dataset.

The network consists of an initial convolutional layer that outputs 32 filters. The first convolutional layer uses a 3x3 kernel size and a step size of 1. This then feeds into another convolutional layer, with the same parameters. Both layers use a relu activation function. Then, this second convolutional layer is flattened, and fed into a fully connected dense layer. This dense layer outputs 128 values, again using a relu activation layer. These values are then fed into a final fully connected dense layer, outputting 10 values, which are the relative probabilities of the different classes. This final dense layer uses a softmax activation function to standardize the values between 0 and 1.

The network was trained over 64 epochs, and the model accuracy and cross entropy loss were recorded for each epoch for both the training and testing data. These graphs are shown below. As can be seen in these graphs the model appears to not be deep enough, and is as a result, overtraining itself past ~5 epochs. Additionally, the highest accuracy reached was at epoch ~10, and plateaued from there, with an accuracy of ~65%. This is in comparison to the state-of-the-art accuracy, by ViT-H/14, of 99.5%. Training these 64 epochs locally on my surface pro took about 70 minutes.