Results from the neural network built by hand:

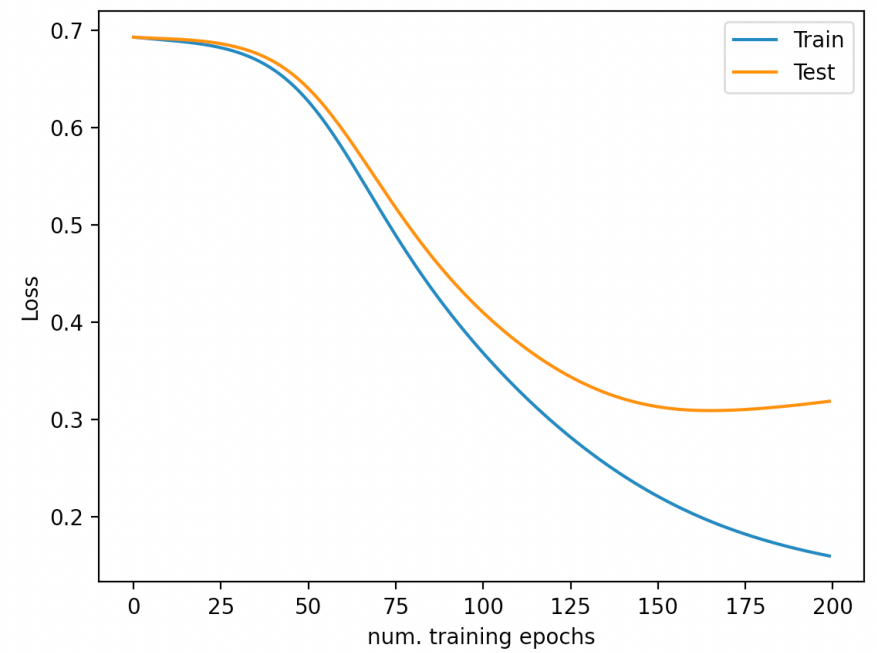

Fig A. Average loss for the train and test set per training epoch

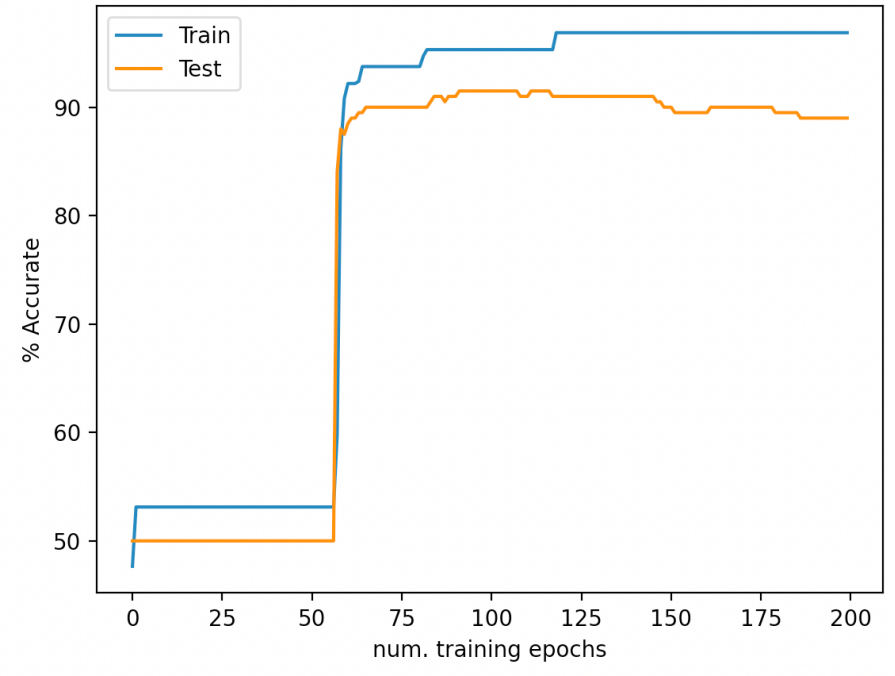

Fig. B. Percentage labels classified correctly per training epoch.

Results from the neural network built using autograd:

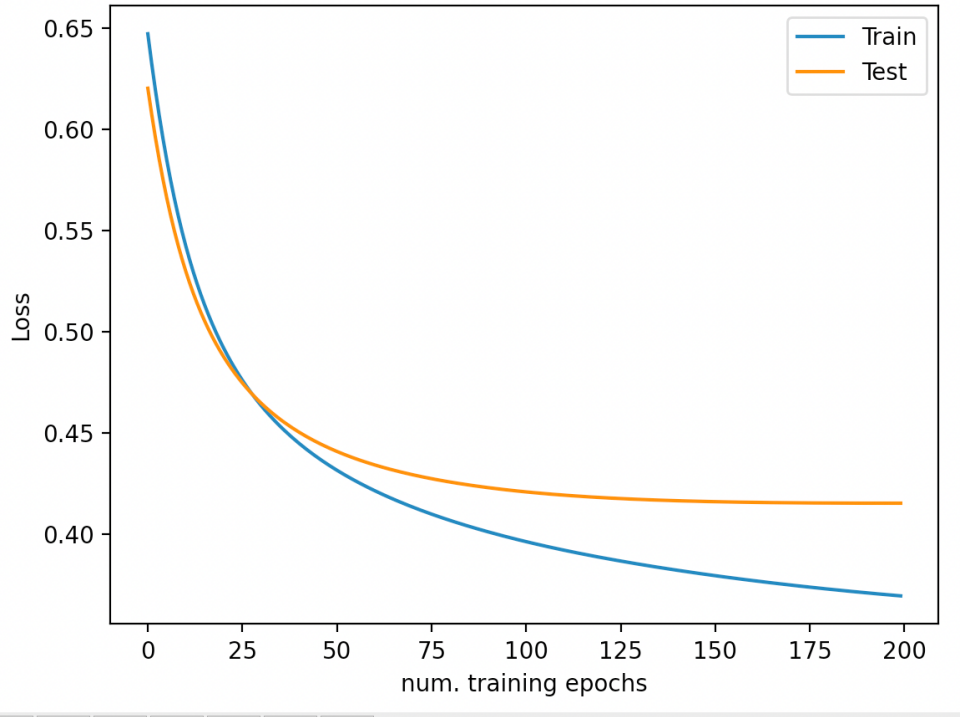

Fig A. Average loss for the train and test set per training epoch

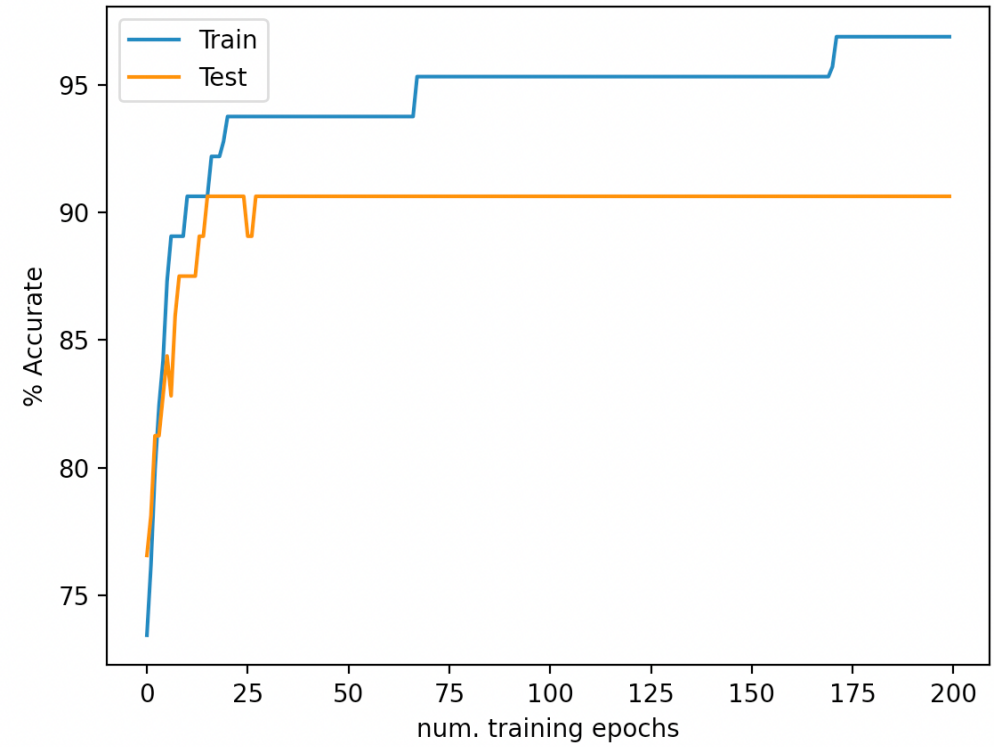

Fig B. Average loss for the train and test set per training epoch

Analysis

Generally speaking, both networks converged well over a 200-epoch training period to produce similarly high levels of accuracy. Interestingly, the network built by hand achieved a smaller loss by the end of 200 epochs than the network computed via autograd in PyTorch. However, the convergence time for the network built by hand was significantly slower than the PyTorch implementation. This may be related to the way PyTorch initializes weights and biases, or the way stochastic gradient descent is optimized in autograd. Both networks are great models of the input dataset, achieving greater than 90% training accuracy.

Both networks generalized well to the left-out test set, achieving close to 90% accuracy on classification of the test set. According to the accuracy plots, it seems both networks became slightly overfit to the training set toward the end of training. Both networks showed a tendency to improve training set classification accuracy while plateauing in their classification accuracy for the test set. This suggests the model is becoming more tuned to the idiosyncratic qualities of the training set than to learning a generalizable structure. This suggests any additional training may reduce generalizability to a test set.