

Setup and Training

I used a random seed 1234 in PyTorch, and the Iris dataset was split into 80% training and 20% test data. I trained the network using 500 iterations and a 0.1 learning rate (eta).

Results

After 500 iterations, the network achieved an accuracy of 100% on the test data. Below is the confusion matrix for the test dataset:

$$\begin{bmatrix} 9 & 0 & 0 \\ 0 & 13 & 0 \\ 0 & 0 & 8 \end{bmatrix}$$

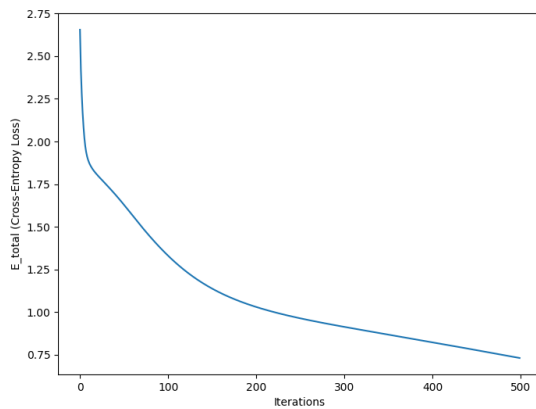
All of the off-diagonal elements are zero, and the diagonal with $9 + 13 + 8 = 30$ which matches the test data sample size ($150 \cdot 20\% = 30$). This shows that the network classified all training samples correctly (no misclassification).

Cross-Entropy Loss

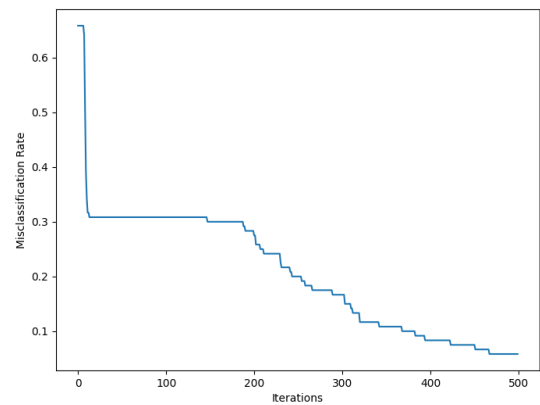
From the plot in Figure 1 (a), we can see that the loss decreased fast in the first iterations, and then it converged to a lower value. This should indicate that the network was fitting the data well.

Misclassification Rate

The misclassification rate dropped steadily throughout the training process. It started off around 65% but was lowered very quickly as the network learned. Finally, it reached a misclassification rate of 0% by the 500th iteration. The plot in Figure 1 (b) shows how the misclassification rate changed over iterations.



(a) Cross-Entropy Loss over 500 iterations



(b) Misclassification Rate over 500 iterations

Figure 1: Cross-Entropy Loss and Misclassification Rate over 500 iterations