CS 559 Neural Networks – Homework 3

Zohair Hashmi | UIN: 668913771 | zhashm4@uic.edu

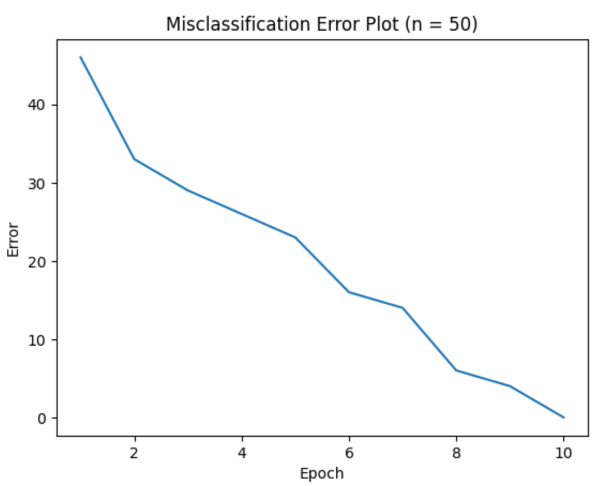
NOTE:

- Libraries used in python code:
  - Numpy
  - Pandas
  - Gzip
  - Matplotlib.pyplot
- Python code requires user to input the following four parameters for training the model. For each part of the homework, input parameters were adjusted accordingly and results were obtained.
  - Number of training samples (n)
  - Epsilon (∈) – Misclassification Error Ratio
  - Learning Rate (η)
  - Random Seed
- Random Seed set to 42 for Parts F-H.

Part (F)

For $n = 50$, $\eta = 1$ & $\epsilon = 0$, the plot converges with an accuracy of 60.44% on test set.

Number of errors: 3956 | Accuracy: 60.44 %



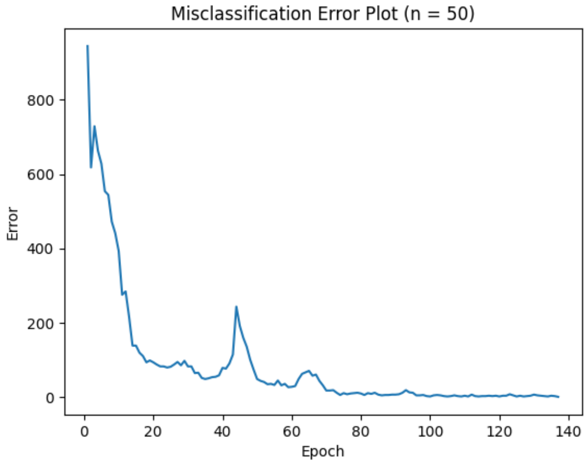
Percentage of misclassified samples = 39.56%

[Comment] The disparity in the misclassification rates between the training and test datasets can be attributed to the limited number of samples employed during model training. With a small training sample size, the model struggles to grasp the intricate patterns present in the diverse range of input images. This deficiency in understanding ultimately results in a high rate of incorrect predictions when applied to the test data.

Part (G)

For $n = 1000$, $\eta = 1$ & $\epsilon = 0$, the plot converges with an accuracy of 83.30% on test set.

Number of errors: 1670 | Accuracy: 83.30 %



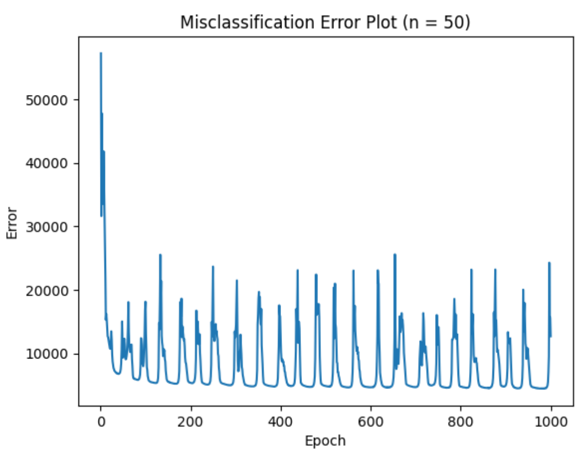
Percentage of misclassified samples = 16.70 %

[Comment] In this step, we expanded the training sample size while maintaining the same parameters as in step (f). It becomes evident that increasing the number of training samples enables the model to capture a wider range of variations in the input images. Consequently, when the model undergoes testing on the test data, we observe substantial enhancements in accuracy, with fewer samples being misclassified compared to the previous scenario.

Part (H)

For $n = 60000$, $\eta = 1$, $\in = 0$ & an epoch threshold of 1000, the plot does not converge with an accuracy of 78.18% on test set.

Number of errors: 2182 | Accuracy: 78.18 %

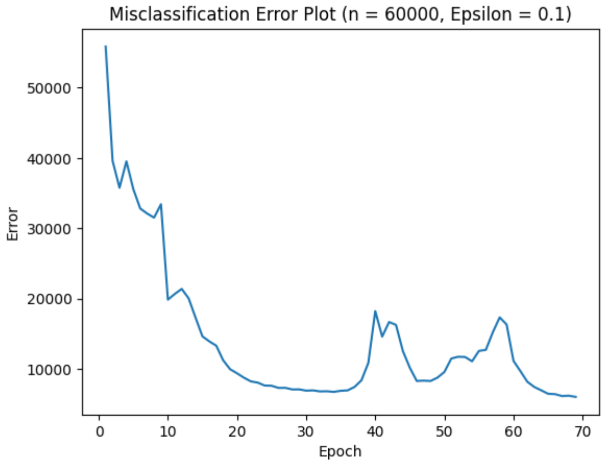


[Comment] In this step, we observe that the plot fails to converge. This phenomenon occurs due to the chosen parameters and a substantial training sample size. Notably, we find that the ratio of misclassified samples never approaches zero. This outcome is primarily attributed to training the model on a large number of samples, which introduces the extent of variation for the neural network to comprehend during the training process. Consequently, the model struggles to achieve convergence. Additionally, as the number of epochs increases, the risk of overfitting the model grows, ultimately resulting in reduced accuracy when applied to the test dataset.

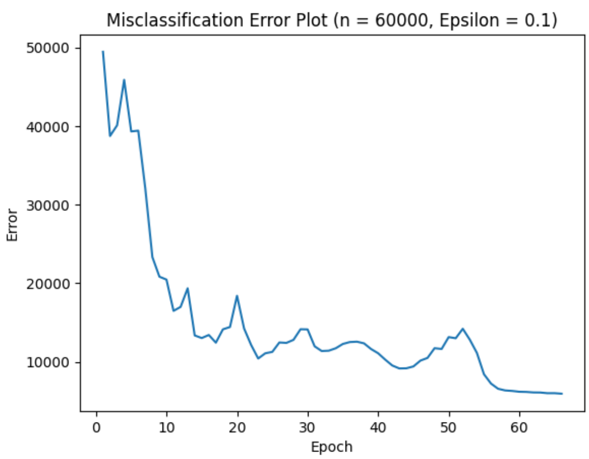
Part (I)

The following experiments are use different seed numbers and the parameters: $n = 60000$, $\eta = 1$, $\in = 0.1$

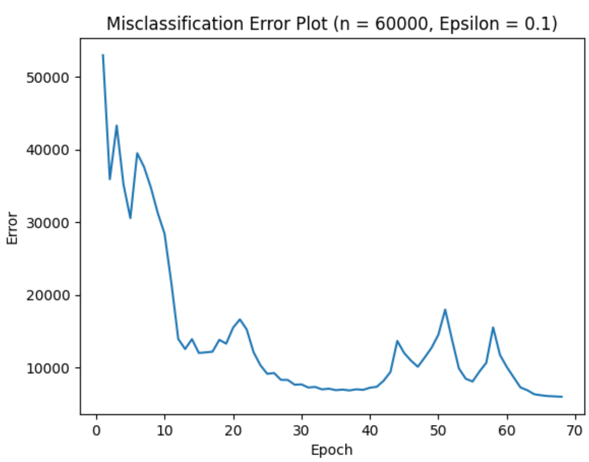
Experiment 1: Random Seed: 100 | Number of errors: 933 | Accuracy: 90.67 %



Experiment 2: Random Seed: 200 | Number of errors: 933 | Accuracy: 90.67 %



Experiment 3: Random Seed: 300 | Number of errors: 943 | Accuracy: 90.57 %



[Comment] In this experiment, we initiated different sets of weights while maintaining consistent parameters: $n = 60000, \eta = 1,\ \varepsilon = 0.1$. It's notable that each experiment concluded training in approximately the same number of epochs and yielded comparable results in terms of accuracy and misclassification error. This observation highlights that, for a fixed training sample size, learning rate, and misclassification ratio threshold, the neural network's weight values tend to converge to very similar values, even when starting from different weight initializations.