CMU 10-715: Homework 1

Perceptron Algorithm on Handwritten Digits

Abishek Sridhar (Andrew Id: abisheks).

1 Report

1.1 MNIST Binary Classification Data

Part (c): Plotting 5x5 grid of samples images from training data

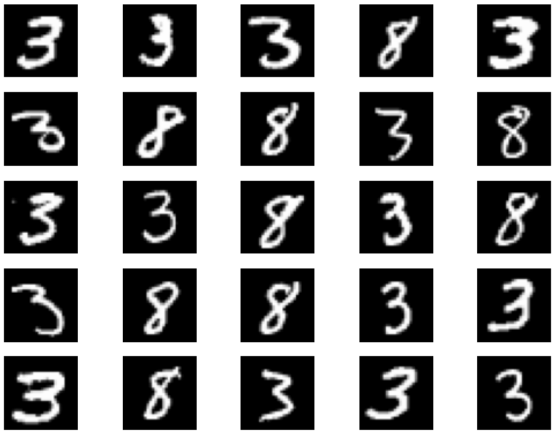

Figure 1: First 25 samples from the filtered MNIST training data

Part (e): Plotting histogram to show the amount of threes and eights in filtered training data

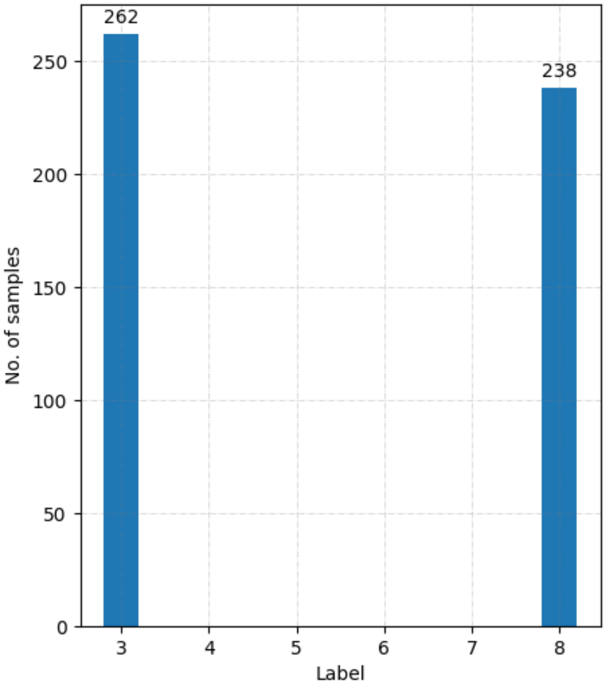

Figure 2: Histogram showing the label counts in filtered training data

1.2 Perceptron Algorithm

Part d.(a) Plotting the train and test accuracy trajectories

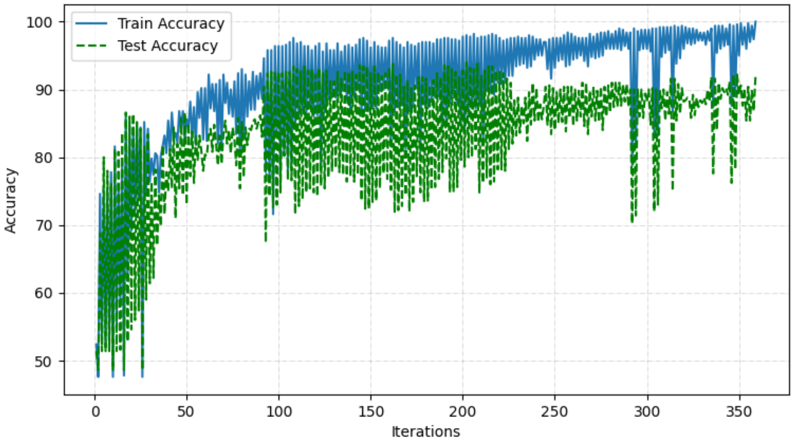

Figure 3: Trajectories of the train and test accuracy during the course of the training

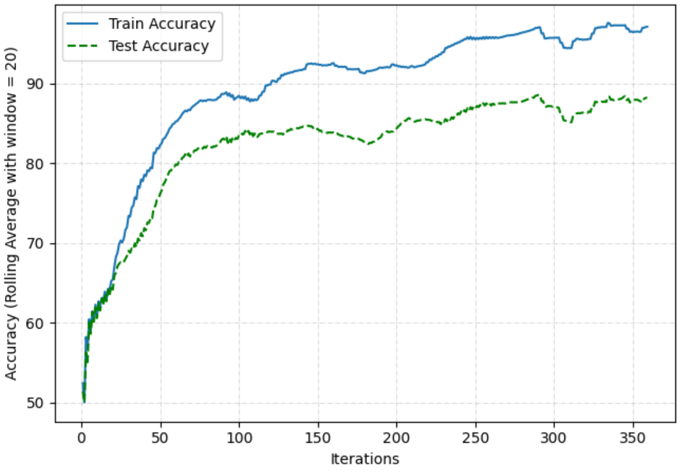

Figure 4: Trajectories of the train and test accuracy with rolling average of window size 20, during the course of the training

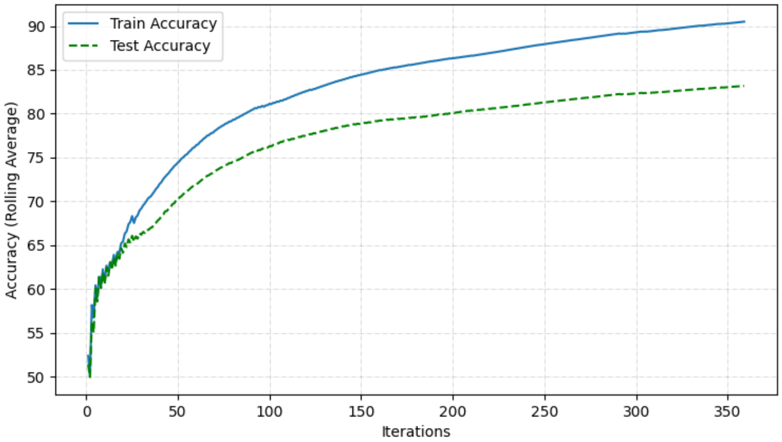

Figure 5: Trajectories of the rolling average train and test accuracy during the course of the training, with rolling average calculated using all values till that point

Part d.(b): Reporting the final train accuracy
The train accuracy the end of the training was obtained to be **100%**.

Part d.(c): Reporting the final test accuracy
The test accuracy the end of the training was obtained to be **92%**.

Part (e): Description of train and test trajectories and intuitive reasoning for generalization gap

From figures 4 and 5, we see that train and test accuracy trajectories (wrt iterations) have a nearly increasing plot with very similar local trends that occur at an increasing generalization gap. But as evident from the figure 3, both train and test absolute accuracy trajectories oscillate quite a bit, with the test accuracy's oscillations being marginally greater. Infact, the train accuracy achieves its maximum absolute value of 100% at the final iteration, but the test accuracy does achieve comparable or even slightly higher absolute accuracies in earlier iterations due to the noise. Hence, we can say the maximum test accuracy saturates after an initial few iterations while the maximum train accuracy still increases, increasing the generalization gap and suggesting overfitting.

I checked that the train data and test data together are linearly separable (the train accuracy ran to 100% when trained on the two datasets together with the perceptron learning algorithm). But the perceptron model does not generalize well to the test dataset when trained on the train dataset alone, even though it is linearly separable and simple. The earlier observation of generalization gap existence and overfitting corroborates this fact. Also, we can observe from the trajectory plots that the algorithm converges well within 400 iterations, which suggests that not all points in the training set were looked at. This could be one reason for poor generalizability since the algorithm might greedily pick a valid, but poor decision boundary among all possible solutions without visiting some points. The algorithm doesn't try to maximize the margin or have any regularization considerations - it can choose a poor decision boundary that lies right next to a correctly classified point that it probably did not visit. This could be a reason for the overfitting, and the consequent increasing generalization gap.

Intuitively, the perceptron learning algorithm can be seen to be analogous to the stochastic gradient descent with mini-batch size 1, since we are considering one misclassified sample to update the weights and bias. This can bring in noise to the accuracy values and the path taken by the parameter values to converge, resulting in parameters not converging to the exact optimum but to a neighborhood surrounding it. This might explain the noise in absolute accuracy values at the end of each iteration, with the noise in the test accuracy values being slightly amplified due to the greedy nature of the learning algorithm wrt train datapoints and the higher variance with unseen samples.

2 Collaboration Acknowledgement

I discussed with Kousik Rajesh regarding what kind of rolling average to use (considering all accuracies till a given instant or considering a smaller window of points preceding the given instant).