Deliverable #1

Karl Satchi E. Navida
Deep Learning - MSCS23A
April 3, 2025

I. Dataset

The dataset is from the **UCI ML Repository** – An **Optical Recognition of Handwritten Digits**. It contains 43 sets of all digits: 30 for training and 13 for testing data.

The images were preprocessed using programs made available by NIST to extract normalized bitmaps of handwritten digits from preprinted form. Each digit is originally a **32x32 pixel** image, reduced into **8x8 grid** after dividing it into a **4x4 blocks**. Each block counts the number of "on" pixels (dark areas), giving values from **0 to 16**. (Alpaydin, 1998)

II. Classification Results

The classification result is separated into two sections after testing several configurations such as using Sigmoid and SoftMax activation functions, and the use of Batch Normalization as a regularization technique.

Along the three configurations done, three tests were conducted to, at least, get some variations in the output. The images that will be provided will only show the highest and lowest accuracies within the said test. However, all images will still be included in the zipped file.

1. Sigmoid

    Using Sigmoid as an activation function, the result sits around 88 to 90%. Three test runs were done with varying degree of accuracy using the confusion matrix: with 90.04% being the highest.

    The Sigmoid function is technically inappropriate for this due to the fact that the output is a classification output; spitting the digits (0-9) as it is trying to predict whether the handwritten image input is one of those digits.

    Figures 1 and 2 shows the two of three confusion matrices created, showing the 88.93% and 90.04% accuracy rate.

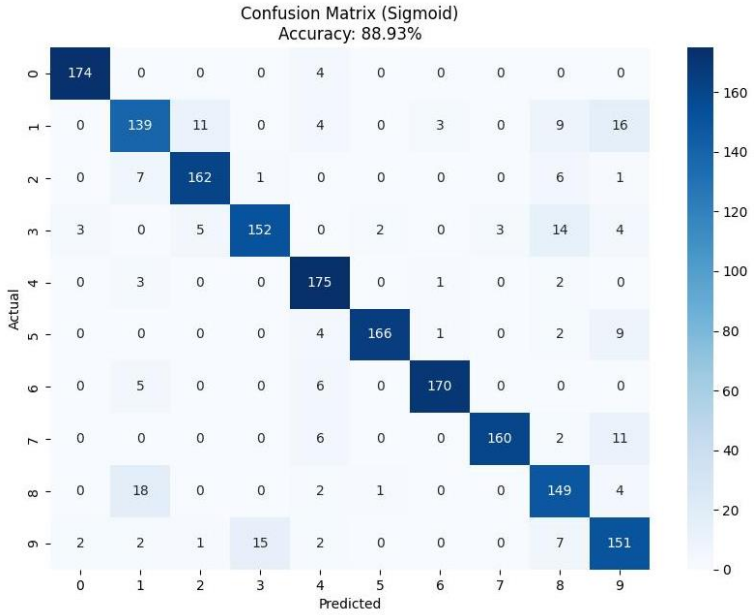

Figure 1: 1st Sigmoid Confusion Matrix, showing an 88.93% accuracy.

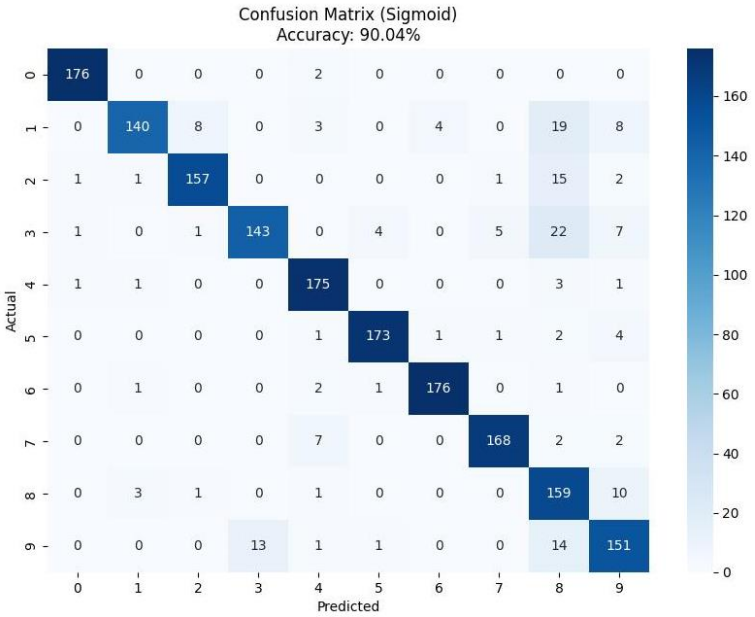

Figure 2: The 3rd Sigmoid Confusion Matrix, showing a 90.04% accuracy.

2. SoftMax

SoftMax is used for multi-classification purposes. This means that this activation function is the perfect fit for the current dataset as the output is a classification of digits zero (0) to nine (9).

i. Without Batch Normalization

In the first test, the neural network was ran without a batch normalization regularization technique, which renders nearly the same accuracy as the Sigmoid function, albeit slightly higher. It floats in a steady 89.09% to 89.93%.

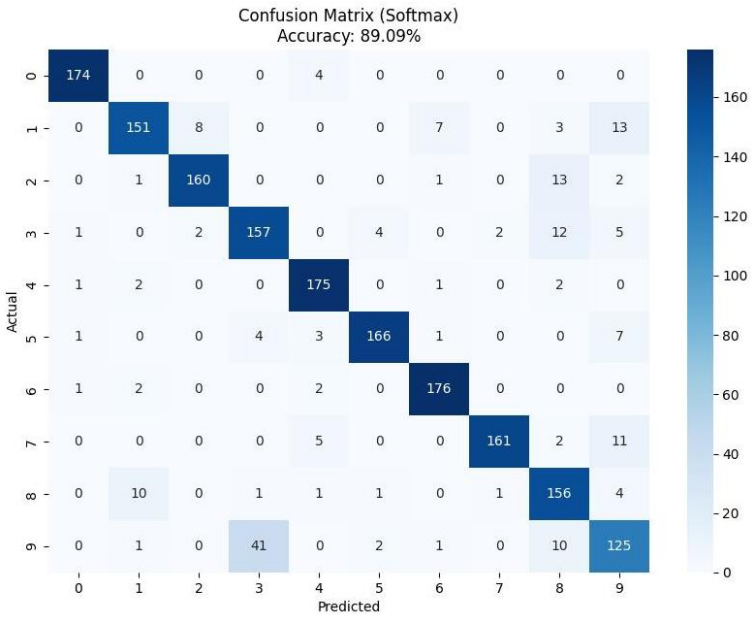

Figure 3: 3rd SoftMax Confusion Matrix, showing an 88.09% accuracy.

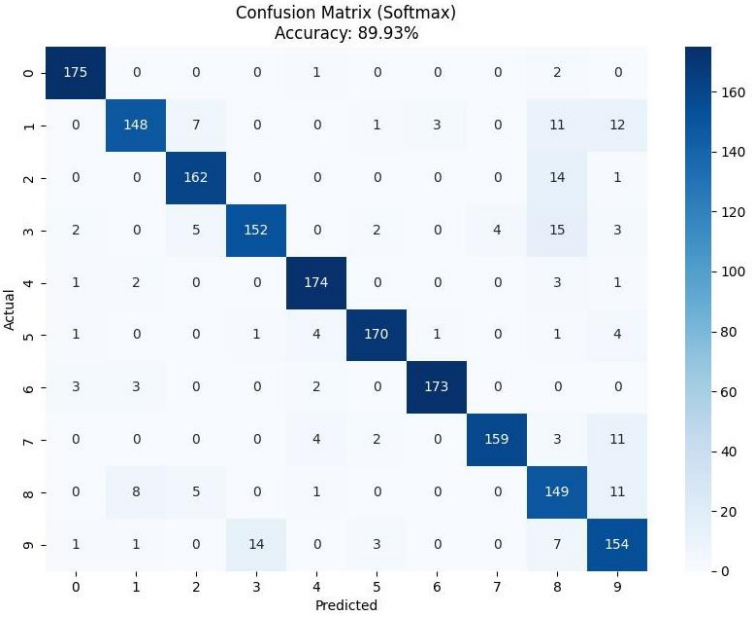

Figure 4: 2nd SoftMax Confusion Matrix, showing an 89.93% accuracy.

ii. With Batch Normalization

In the second test, Batch Normalization was added as a regularization technique. This resulted in a much better accuracy yield of around 90.54% to 90.87%, proving that the said regularization technique does help.

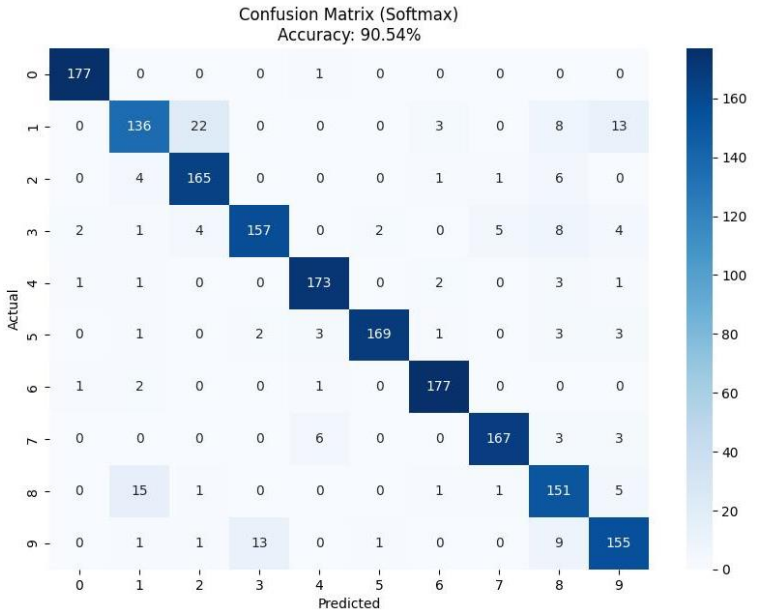

Figure 5: 2nd SoftMax Confusion Matrix with Batch Normalization, showing a 90.54% accuracy.

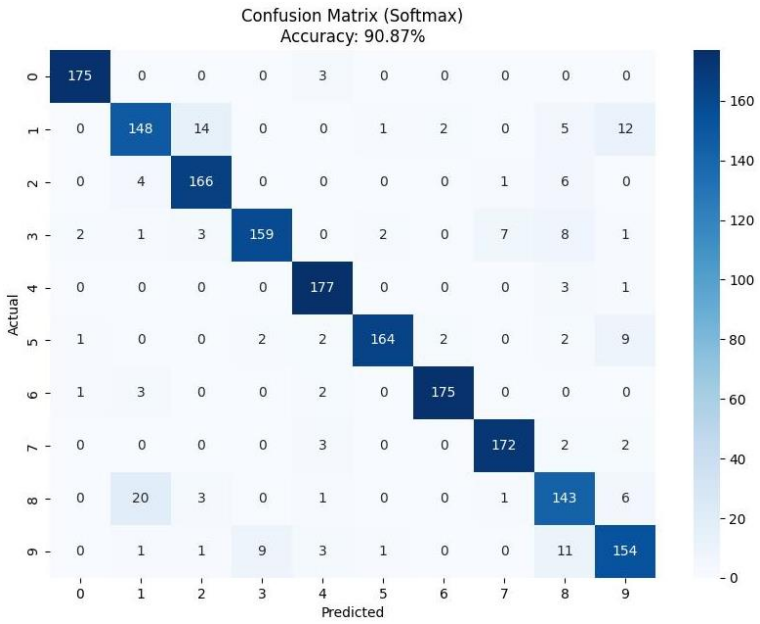

Figure 6: 1st SoftMax Confusion Matrix with Batch Normalization, showing a 90.87% accuracy.

The accuracy formula used the basic percentage formula to get the numbers provided in the classification report:

$$accuracy = \frac{y_{true}}{y_{test}} * 100$$

The code base of this deliverable is also published in GitHub (https://github.com/Virus5600/Deep-Learning).

Works Cited

Alpaydin, E. &. (1998). *Alpaydin, E. & Kaynak, C.* Retrieved from UCI Machine Learning Repository: https://doi.org/10.24432/C50P49