ELEC 475 Lab 1

Ethan Parliament

20220776

September 21, 2024

Model Details

The model used for lab 1 was an MLP autoencoder consisting of 4 fully connected layers. The encoder side of the model took a 1 by 784 input, then a fully connected layer of 392, and a bottleneck of size 8. The decoder side of the model was symmetrical to the encoder, taking the bottleneck and passing it to a fully connected layer of size 392, and then to the output layer of 784. Between each layer, a ReLU activation function was applied, except for the last layer, where a sigmoid activation function was applied. A diagram of the model can be seen below in Figure 1.

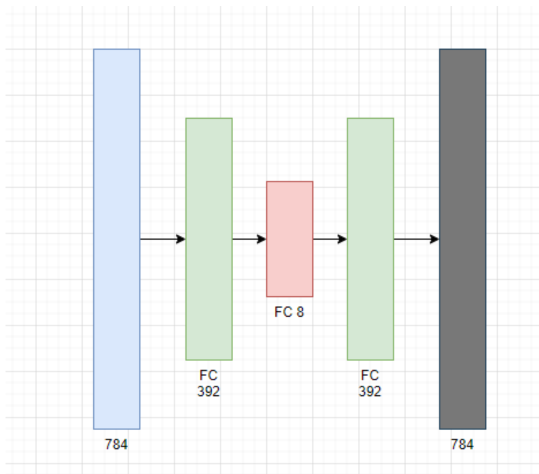

Figure 1: Diagram of the model

Training Details

The model was trained for 50 epochs, using the Adam optimizer with the initial learning rate set at 10^{-3} and a weight decay of 10^{-5}. The learning rate scheduler used was ReduceLROnPlateau, which will reduce the learning rate once learning stagnates. The loss function that the model was trained with was a mean squared error loss function. The model was trained with a batch size of 2048.

Results

The loss curve plot shown in Figure 2 shows how the model improved in each epoch. It starts off with a large loss value and quickly decreases the loss in the first few epochs. In the later stages of training, the loss starts to plateau, and the learning rate schedule begins decreasing the learning rate in order to find the minimum value.

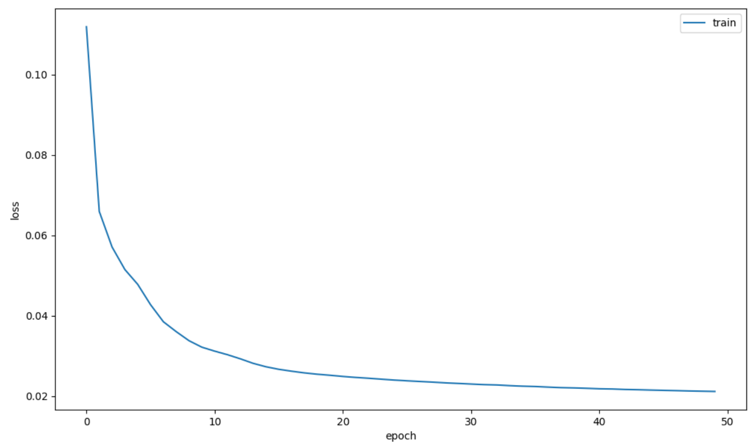

Figure 2: Loss curve plot from training the model

Overall the model worked well at recreating the images from the MNIST dataset, being able to recreate clear images with high accuracy. An example of the outputs can be seen below in Figure 3.

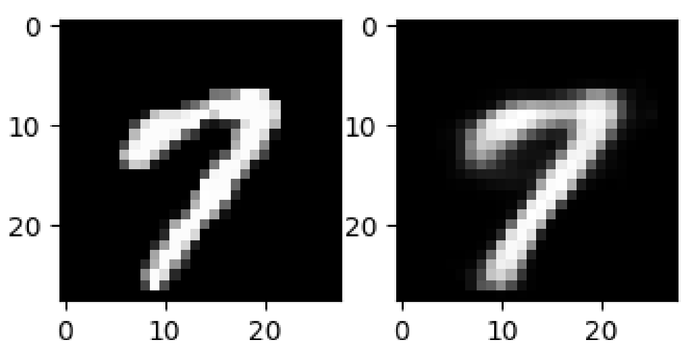

Figure 3: Good sample output from the model. Input image is on the left, output image on the right

The model performed well with image denoising. Adding noise to the images had very little effect on the performance of the model. Even when increasing the noise to a level where humans are not able to determine what digit was written, the model is able to somewhat successfully remove the noise so only the digit remains. Examples of denoising can be seen below in Figure 4 and Figure 5.

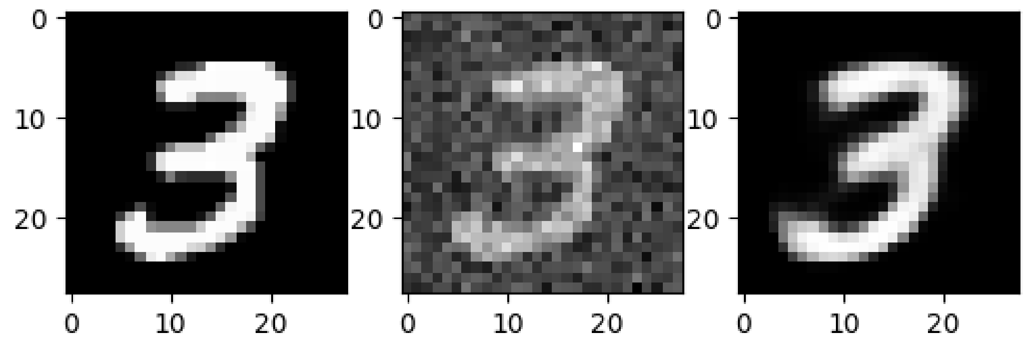

Figure 4: Example of denoising. Original image is on the left, noisy image in the middle, and denoised image on the right

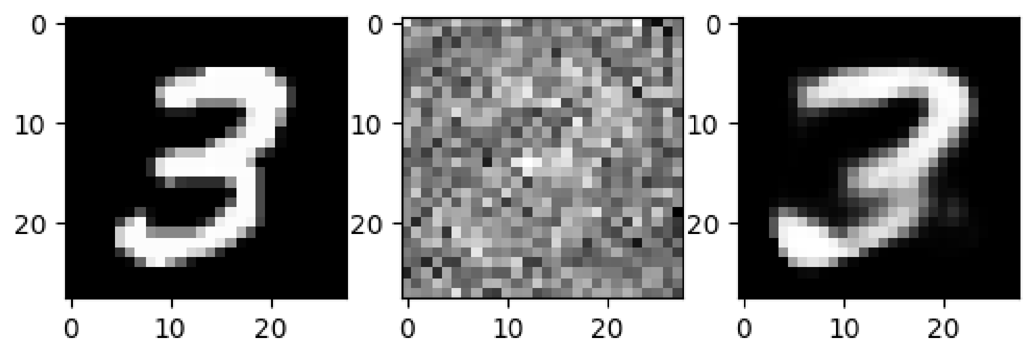

Figure 5: Example of denoising with lots of noise. Original image is on the left, noisy image in the middle, and denoised image on the right

Bottleneck interpolation was a part of the lab that was a bit more difficult. At first, the interpolation between 2 images was coming out with very thick digits, and not much detail on them. Once the input images were normalized, the results became much better. An example of the linear interpolation is shown below in Figure 6.

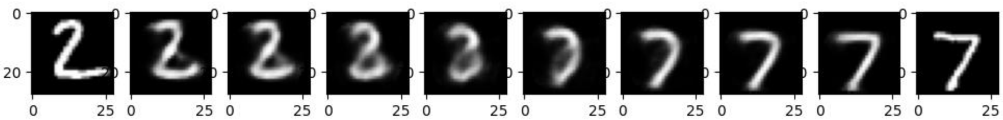

Figure 6: Interpolation between 2 images