

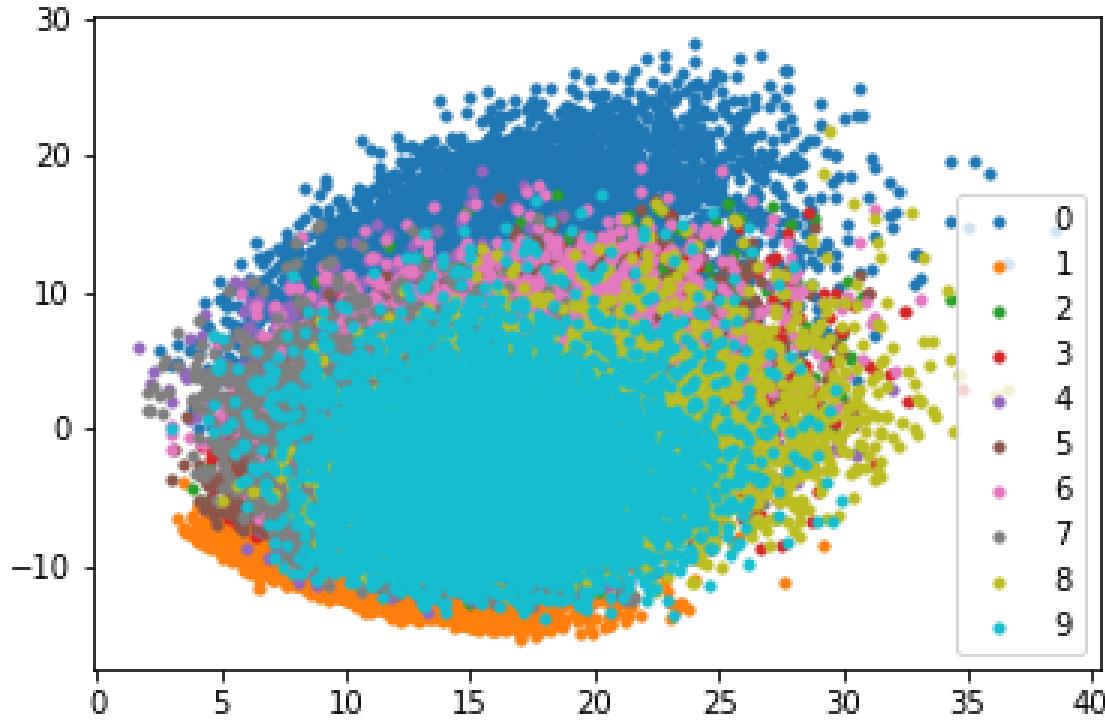
**Problem 1**

FIGURE 1. MNIST Data Projection by PCA

PCA did a great job separating 1's and the rest of the data. The same can be said about 0's and the rest of the data. Data points with other labels tend to overlap with each other.

Note that I used  $\frac{1}{n-1} X^T X$  formula for the covariance. However, Python's `numpy.cov(X, rowvar = False)` could also be used. Python's `numpy.cov` gives better results in terms of separating data points for using PCA in this problem but I didn't trust it.

**Problem 2**

Here is the result of Autoencoder training:

```
Epoch 3 --- > loss = 0.6891043014427026
Epoch 4 --- > loss = 0.6863332463006179
Epoch 5 --- > loss = 0.6849041216323773
Epoch 6 --- > loss = 0.6835595601350068
Epoch 7 --- > loss = 0.6824035375863313
Epoch 8 --- > loss = 0.6812644581745069
Epoch 9 --- > loss = 0.6804489429195721
Epoch 10 --- > loss = 0.6796802424589793
Epoch 11 --- > loss = 0.6792065823177497
Epoch 12 --- > loss = 0.6788403040915728
Epoch 13 --- > loss = 0.6778402211169402
Epoch 14 --- > loss = 0.6778000818639993
```

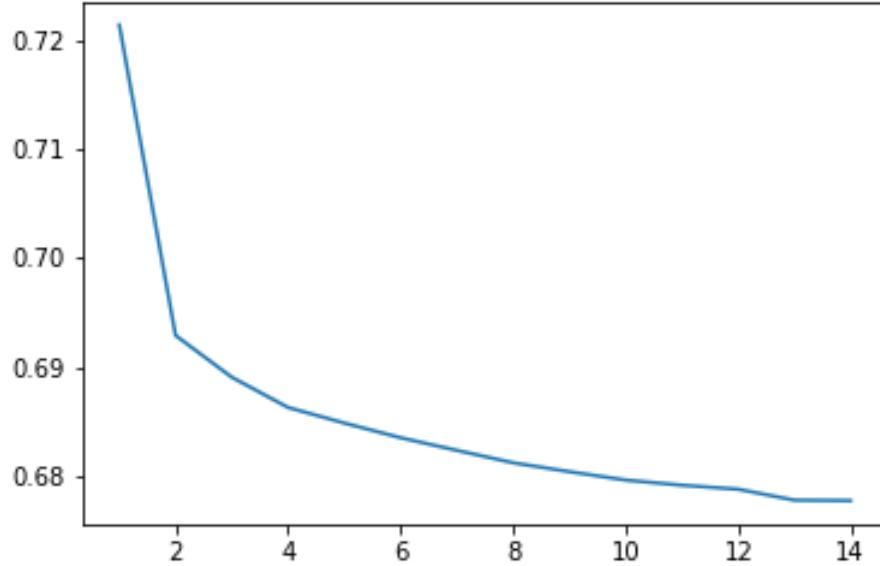


FIGURE 2. Training Loss by Epoch Number

I defined a threshold as 0.0002. If the difference between the training loss at the last epoch and the previous epoch is less than this threshold, I stop the learning. In Figure 3., you see the

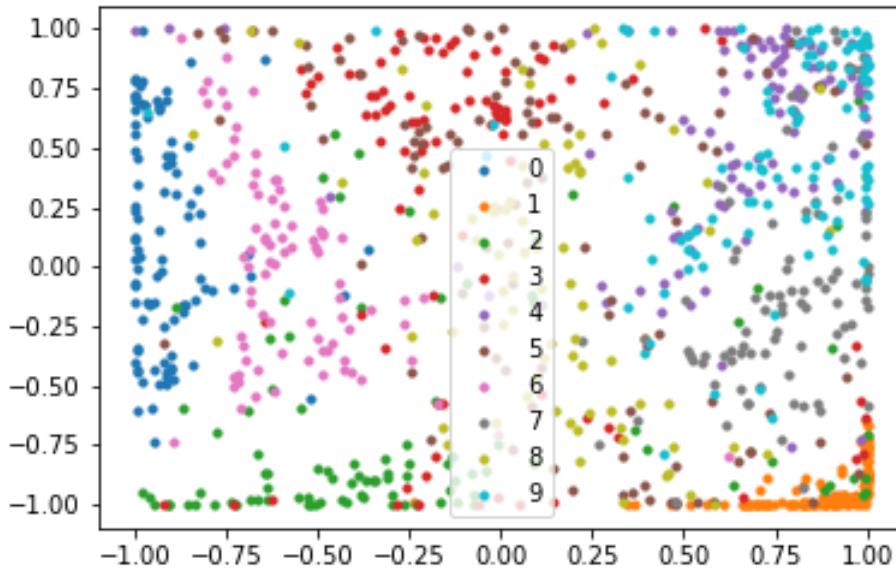


FIGURE 3. 1000 Random Data Projection by Autoencoder

distribution of 1000 random data. In PCA, we saw that it was very successful with distinguishing 0's and 1's and not that successful for other labels. Here, Autoencoder is much more effective than PCA especially for the labels other than 0 and 1. Autoencoder is easily able to differentiate between 0 and 1. 4's and 9's kind of overlapped. 3's and 5's kind of overlapped. There are also other overlapping issues but these are not as much as PCA results.

### Problem 3

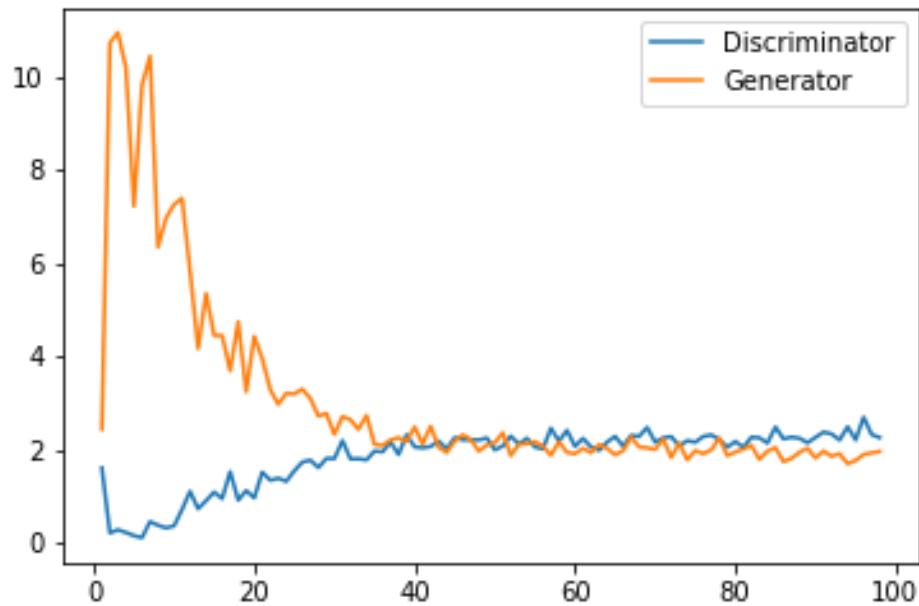


FIGURE 4. Training Loss by Epoch Number

If the absolute value of difference between the last loss value and the average of the last 40 loss values is less than the predefined threshold, I stop the learning. It converged at the 98th epoch. You can change threshold from 0.03 to a larger number for faster convergence or you can also use the last 10 loss values and not 40. Also, algorithms kind of converge around epoch 40 based on Figure 4.

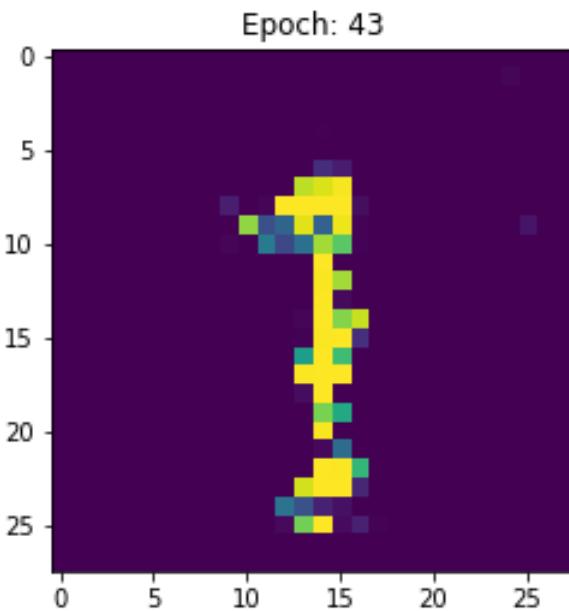


FIGURE 5. Epoch 43: Looks like 1

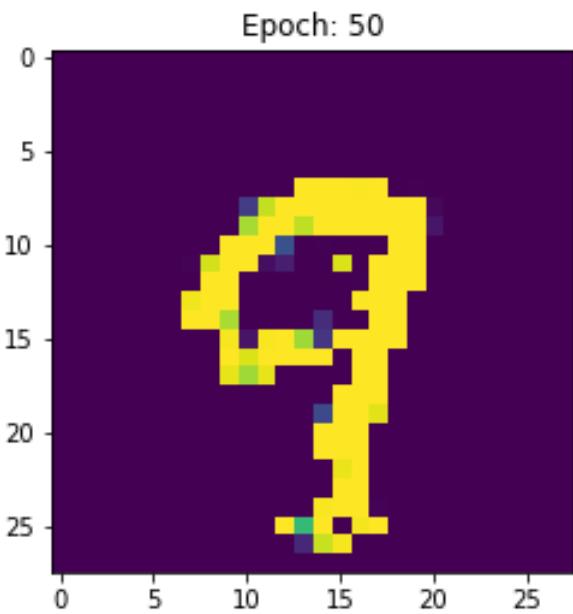


FIGURE 6. Epoch 50: Looks like 9

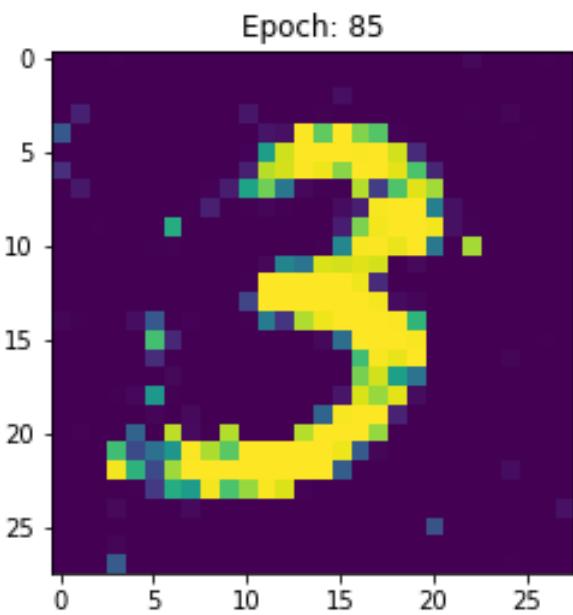


FIGURE 7. Epoch 85: Looks like 3

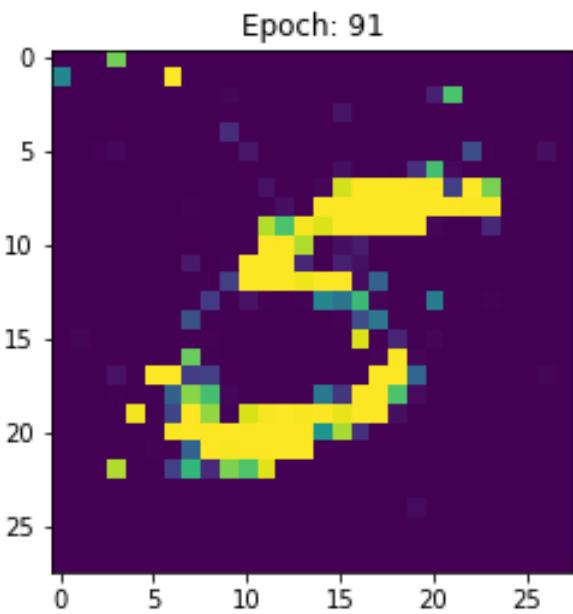


FIGURE 8. Epoch 91: Looks like 5

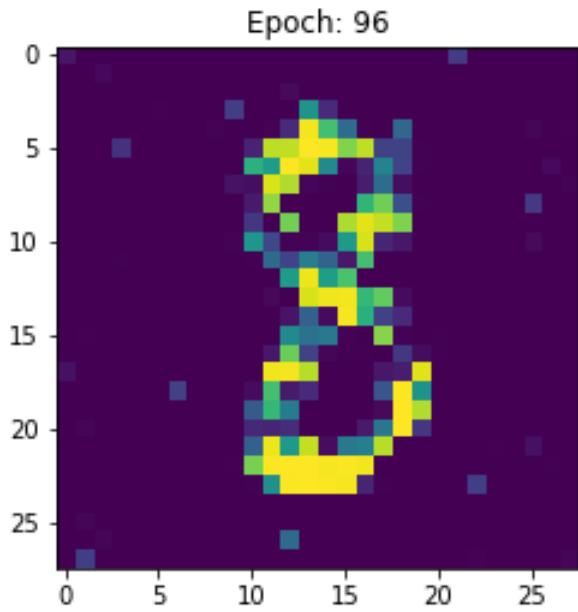


FIGURE 9. Epoch 96: Looks like 8

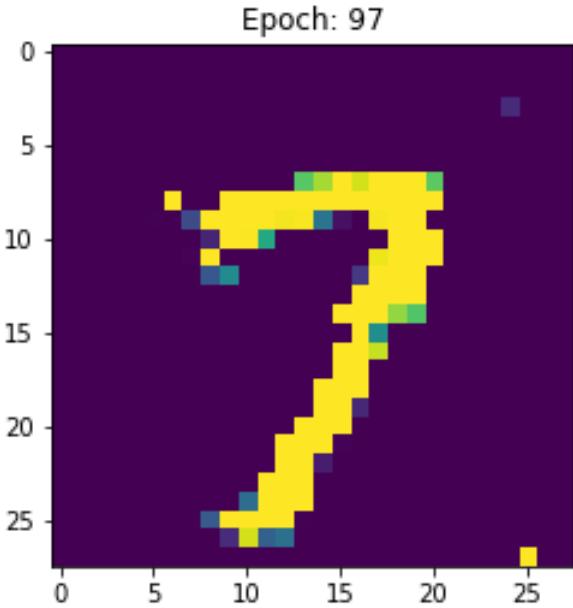


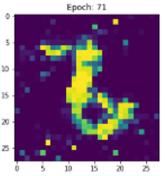
FIGURE 10. Epoch 97: Looks like 7

I plotted all images at all epochs. You can find them in my submission. I only plotted some images that are after epoch 40 as we can see training losses converge around epoch 40. However, there are also full numbers being formed before epoch 40. Epochs all the way through epoch 98 show more improvement as we can also imagine from the training loss plot in Figure 4. At the beginning, epoch 0 shows a fuzzy image. You can't even tell there is a number in that image. We also see these fuzzy images in later epochs. However, overall, with increasing epoch number, we see more improvement.

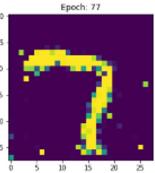
Following pages show all the images at all epochs in order. ——>



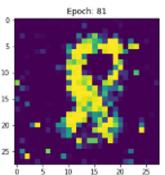




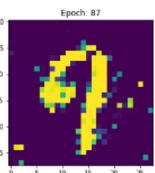
p3\_Image\_at\_Epoch\_71.png p3\_Image\_at\_Epoch\_72.png p3\_Image\_at\_Epoch\_73.png p3\_Image\_at\_Epoch\_74.png p3\_Image\_at\_Epoch\_75.png



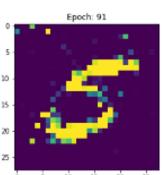
p3\_Image\_at\_Epoch\_76.png p3\_Image\_at\_Epoch\_77.png p3\_Image\_at\_Epoch\_78.png p3\_Image\_at\_Epoch\_79.png p3\_Image\_at\_Epoch\_80.png



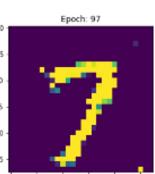
p3\_Image\_at\_Epoch\_81.png p3\_Image\_at\_Epoch\_82.png p3\_Image\_at\_Epoch\_83.png p3\_Image\_at\_Epoch\_84.png p3\_Image\_at\_Epoch\_85.png



p3\_Image\_at\_Epoch\_86.png p3\_Image\_at\_Epoch\_87.png p3\_Image\_at\_Epoch\_88.png p3\_Image\_at\_Epoch\_89.png p3\_Image\_at\_Epoch\_90.png



p3\_Image\_at\_Epoch\_91.png p3\_Image\_at\_Epoch\_92.png p3\_Image\_at\_Epoch\_93.png p3\_Image\_at\_Epoch\_94.png p3\_Image\_at\_Epoch\_95.png



p3\_Image\_at\_Epoch\_96.png p3\_Image\_at\_Epoch\_97.png p3\_Image\_at\_Epoch\_98.png