Autoencoder on MNIST

Gonsoo Moon
Post Master Student of Computer Science
University of Texas at Dallas,
gonsoomoon@gmail.com

1. Introduction

This project is for showing how much Autoencoder is effective from accuracy perspective using MNIST image data set. Accordingly, it displays the original input image compared to the output image using the Autoencoder. Also, the L2 norm cost of the training and validation is shown for whether a model is a good standing. The toolkit Tensorflow 1.0.0 and python are used on the Mac pro.

2. Data & Architecture

MNIST image data set is 59,950 and each image is 28 * 28. The architecture has three encode layers, one code layer and three decode layers. The size is 1,000, 500, 250 and 250, 500, 1,000 for the encode and decode layer respectively. The 4 neurons of code layer is used. Each layer uses the batch normalization that states three steps: "1. Grab the vector of logits incoming to a layer before they pass through the nonlinearity 2. Normalize each component of the vector of logits across all examples of the minibatch by subtracting the mean and dividing by standard derivation 3. Given normalized inputs, (^X) use an affine transformation to restore representational power with two vectors of parameters, gamma * ^X + beta"

3. Training & Loss Function

The batch size is 100 so that the total of batches is 599. The number of training epochs is 200 and the learning rate is 0.01. The loss function, L2 norm, is

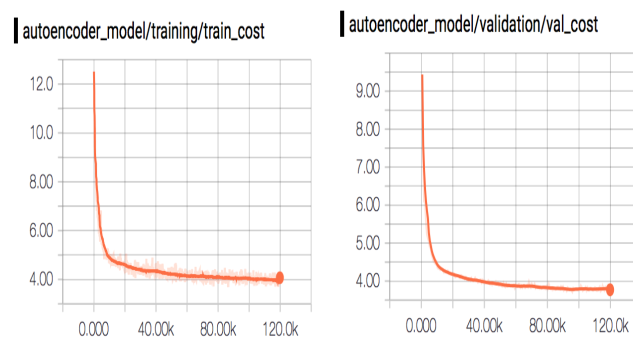
$$\sqrt{\sum_{i=0}^{n}(I - O)^2}$$

where I is input vector and O is reconstruction vector. The L2 norm is the difference from the input and reconstruction vector.

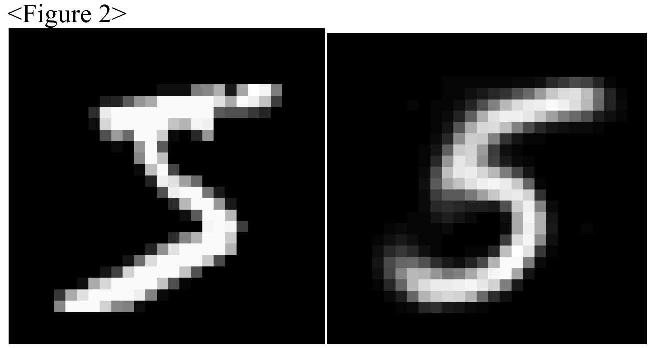
4. Experiment & Analysis

The <Figure1> shows the L2 norm cost, reaching to, at an epoch of 200, 3.973 training cost and 3.771 validation cost. Finally, test cost is 4.249. <Figure2> compares the original input and reconstruction images respectively.

<Figure1>



<Figure 2>



5. References

[1] Nikhil Buduma (2017). Fundamentals of deep learning. Sebastopol, CA: O'Reilly Media