CSIC 5011: Exploring High Dimension Data with MDS, PCA, AutoEncoder and VAE

Liang Zhicong 20485672
Department of Mathematics, Hong Kong University of Science and Technology

Introduction

Dimension reduction is one of the most important topic in data science. Meaningful dimension reduction of the original data not only can speed up computation but also can help extract essential information out of the redundant one, lending hand to applications like classification, regression and visualization, etc. In this report, we will use four unsupervised learning methods to reduce the dimension of hand written digit image (MNIST) and visualize the their embedding spaces, which includes Multidimensional Scaling (MDS), Principle Component Analysis (PCA), autoencoder (AutoEncoder) and Variational Autoencoder (VAE).

Methodology

MDS takes the “distance” between objects as input and tries to place each object in N-dimensional space that the distance are preserved as much as possible. Each object is then assigned coordinates in each of the N dimensions.

PCA is one of the most widely used dimension reduction method. It uses an orthogonal transformation to convert a set of observation of possibly correlated variables into a set of values of linearly uncorrelated variables called principle components, while retaining as much as possible variance of the data.

AutoEncoder is a neural network that used to learn efficient data codings. On one hand, it aims to learn a low dimension representation (Z) for a set of data by ignoring the noise (encoding). On the other hand, it tries to reconstruct the input as close as possible to the original one using the representation learned before (decoding).

VAE is an extension of AutoEncoder that achieves great success in generating data. It combine the idea of AutoEncoder with statistical inference. In VAE, the high dimension data X and low representation Z are random variable, which makes it possible to sample new X from the distribution $p(x|z)$[1].

In this report, we will use the projected data on the embedding space of MDS and PCA, and the hidden low dimension representation of AutoEncoder and VAE as our output. For convenience of visualization, we set the dimension of embedding space and hidden representation to 2.

We will finally evaluate these methods by how well they separate the different classes of images in the 2-dim space.

Dataset

MNIST dataset consists of image of handwritten digits, labeled by 0, 1, 2…, 8, 9. All the image are of shape (1, 28, 28) [2].

Experiment

In this experiment, we firstly normalize the data to have mean 0 and standard deviation 1. To avoid data points clustering together in the figures, we will only plot 2000 points in the embedding space.

As for AutoEncoder and VAE, we use a 6-layer fully-connected neural network with structure 784-400-200-2-200-400-784.

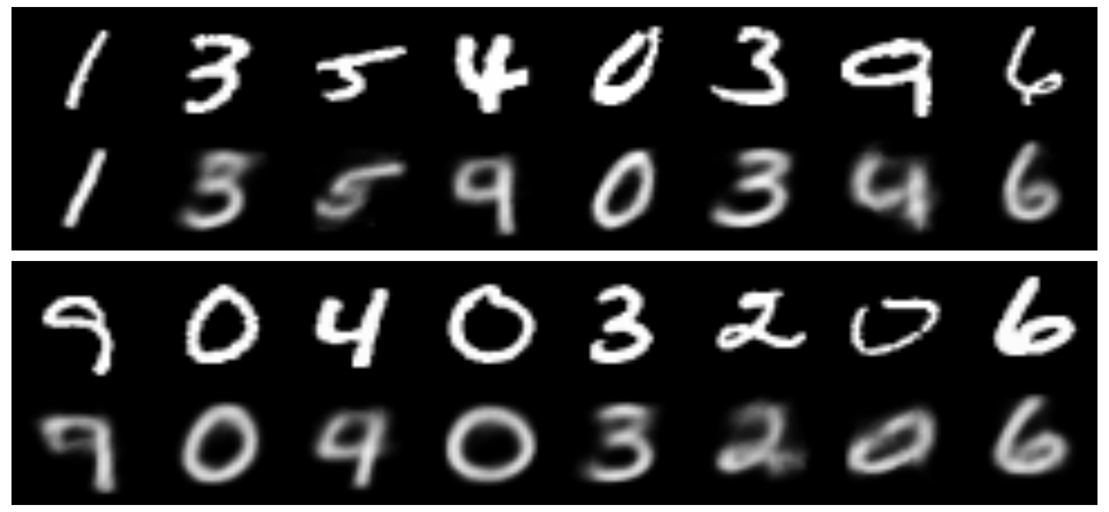
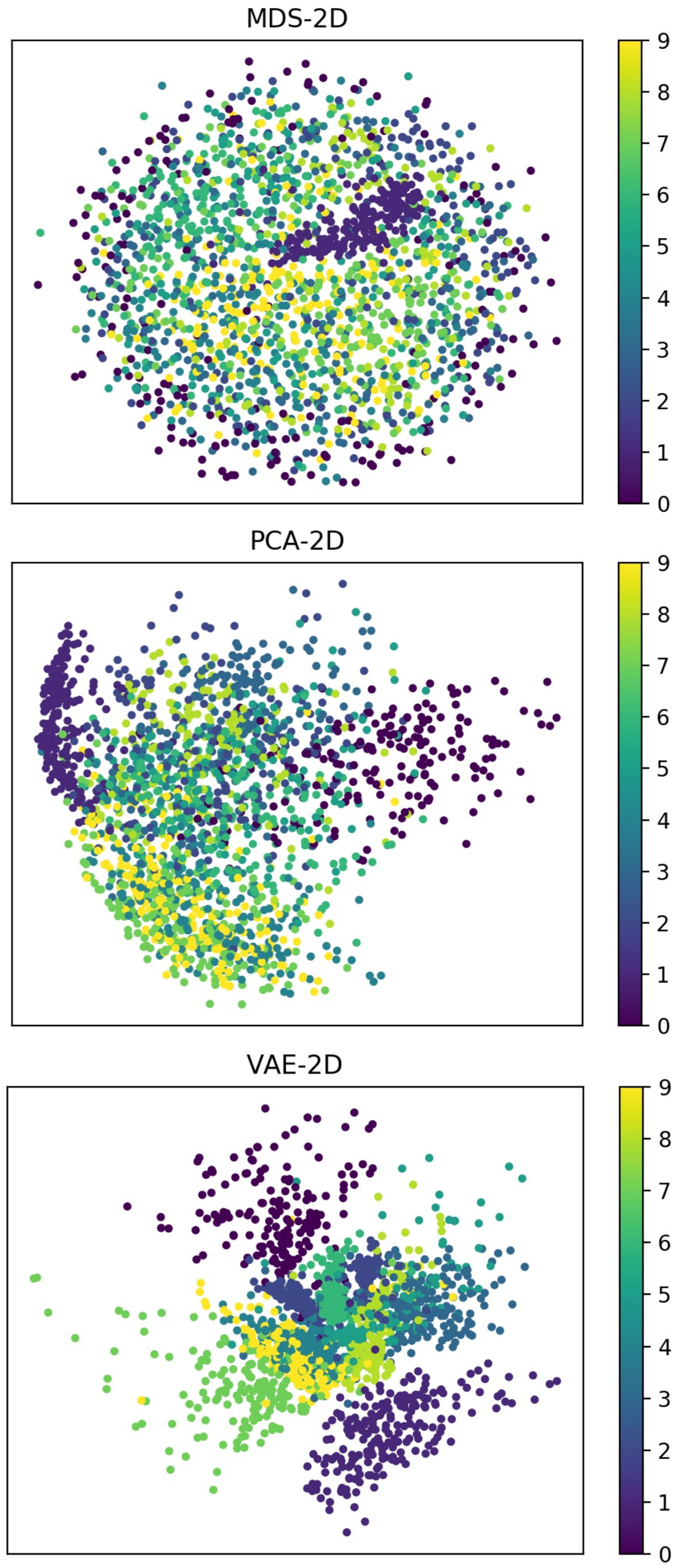


Figure 1. Comparison between original image and reconstructed image. The upper one is generated by AutoEncoder while the bottom one is generated by VAE.



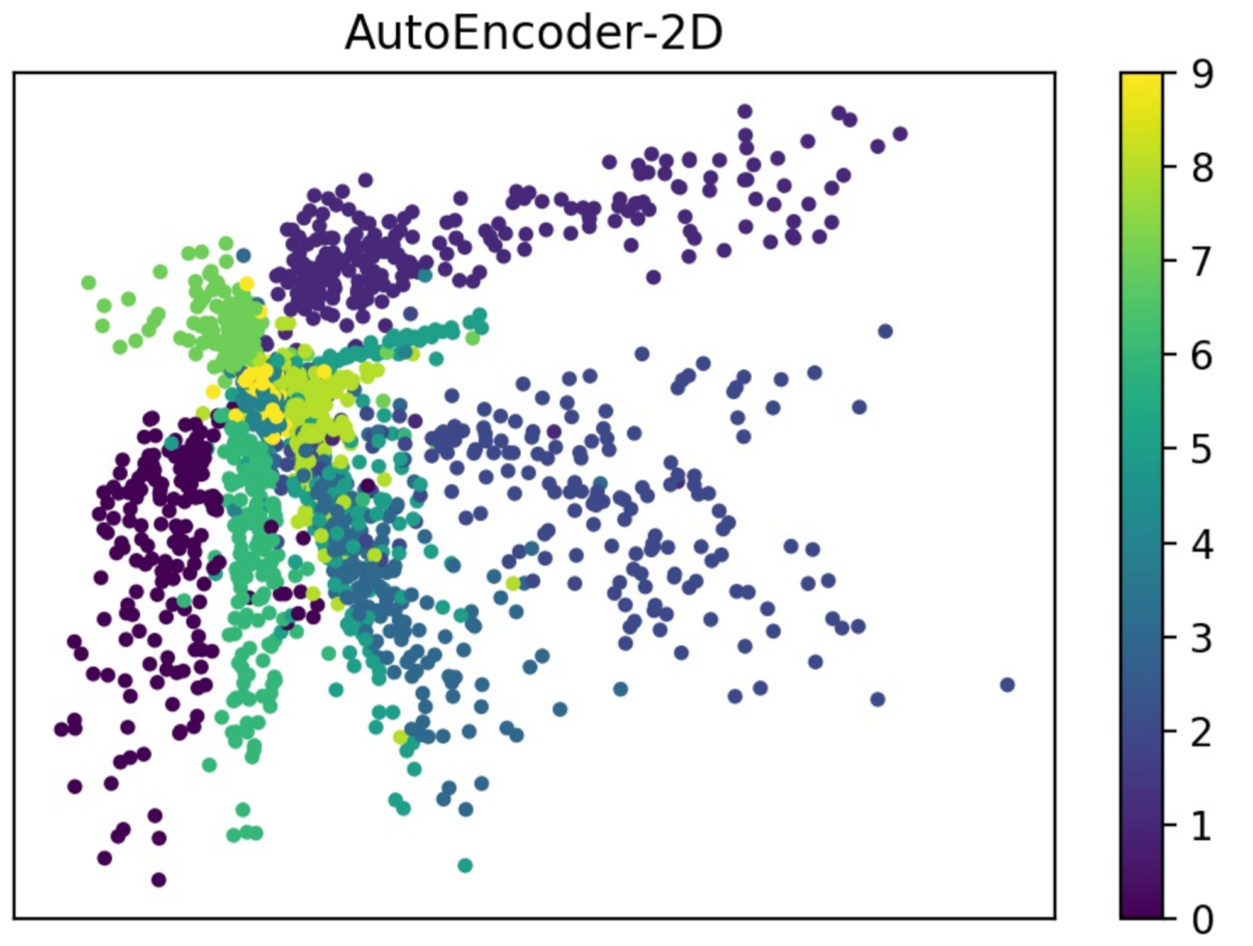


Figure 2. Visualization of embedding space of MDS, PCA and hidden representation of AutoEncoder and VAE.

Analysis

In Figure 1, we show the original images and the reconstructed ones of AutoEncoder and VAE. The result is surprising since the dimension of the hidden representation is only 2.

In Figure 2, we notice that MDS performs the worst, and then PCA. A reasonable explanation is that the image data lies in nonlinear space that could not be well separated by linear transformation. AutoEncoder performs the best while VAE follows. As we can see, AutoEncoder not only well separate the different class of data but also maintains “cone” pattern for each class, where data clouds are compact. As for VAE, data are relatively well separated. However, due to the stochastic nature, they are not as compact as the ones in AutoEncoder.

References

[1] Kingma, D.P. and Welling, M., 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

[2] LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P., 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), pp.2278-2324.