Fariba Khoshghalbvash
Report 65
10/08/2018

Summary

As it was demonstrated in my previous report, it seems that the NNMF weight initialization method has a more constructive impact on autoencoder as oppose to the discriminative cases. So during the past week I implemented most of my experiments on training and improving the results of an AE. Also since my target conference is CVPR, I decided to work on reconstruction of the input (images) and evaluate my proposed method in terms of better reconstruction as well as loss drop. I designed two experiments, one for training and AE using large amount of samples which are also distributed uniquely among different classes so it can be count as the dataset in a perfect world. The other one however contains very low and imbalanced number of samples in each class which is more similar to real world datasets. I further recorded the output of each experiment in two stages: (1) after 10 epochs, (2) after 100 epochs.

Data

I used four numbers of MNIST data (0, 1, 2, 4) to run my experiments. For the large dataset I used 5500 samples of each class. For the smaller dataset I chose 226 sample from number 0, 99 from number 1, 70 from number 2, 18 from number 4. The purpose of this was to avoid the benefits of a well-formed data. %20 of the data remained untouched and used for testing reconstruction and out of the %80 of the training set, %33 was used for validation and illustrating the loss drop.

For the imbalanced set, random over sampling and noising were applied to training set to build a data set of 4 class with the same number of samples in each class.

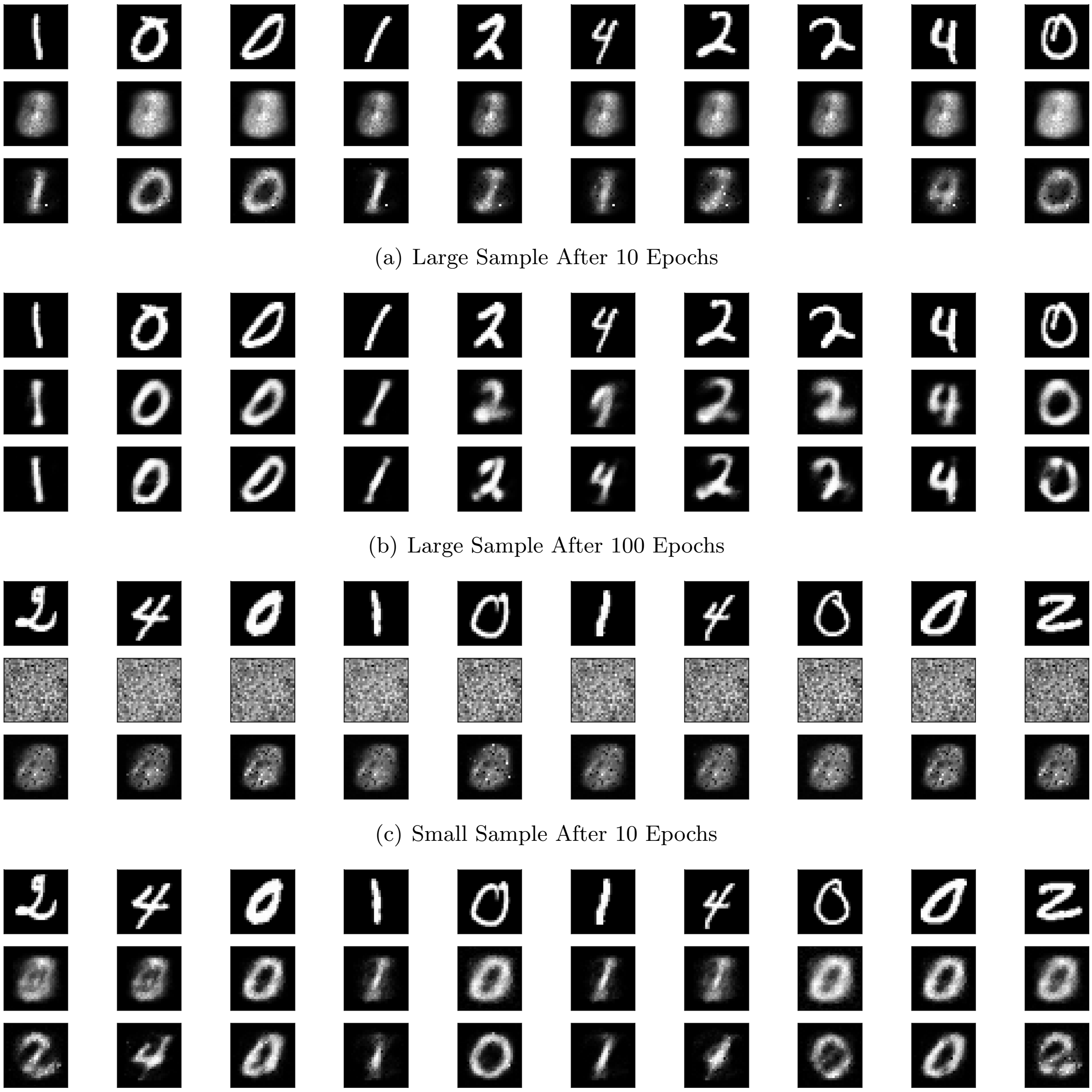
Table 1: Dataset distribution in details

Set/Class		**0**	**1**	**2**	**4**
Large	Whole	5500	5500	5500	5500
	Test	1100	1100	1100	1100
	Train	4400	4400	4400	4400
Small	Whole	226	99	70	18
	Test	45	20	14	4
	Train	181	79	56	14
	Train after oversampling	181	181	181	181

Results and Conclusions

The result of reconstruction of 10 test images (Fig. 1) and loss drop (Fig. 2) indicates that using NNMF as a weight initialization method significantly improves training and AE. Specially, when the dataset has small number of samples and imbalanced class distribution which is very common among most dataset. As we can see in Fig. 1c, the randomly initialized AE after 10 epochs has trained barely nothing as opposed to NNMF initialized AE.

This research answers the question *"How to reduce the effect of random initialization in deep networks?"*, and the result promises that it can be benefitial in terms of reducing both the number of training epochs therefore lower computations and the number of required samples.



(a) Large Sample After 10 Epochs

(b) Large Sample After 100 Epochs

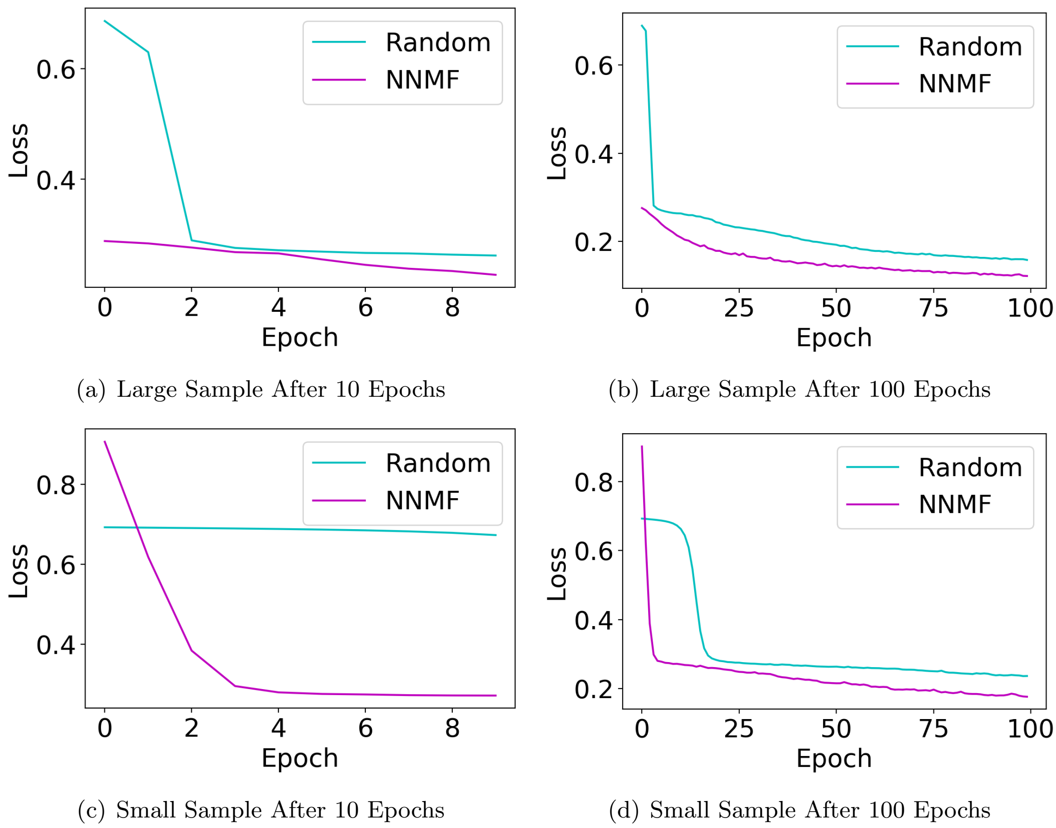
(c) Small Sample After 10 Epochs

(d) Small Sample After 100 Epochs

Figure 1: Comparison of original image *(top)*, reconstructed image by an AE using random initialization *(middle)* and reconstructed image by an AE using NNMF initialization *(bottom)*.

What to do next week?

During the following week I plan to start writing the paper and find other well-known initialization mathods to perform a more solid evaluation. The deadline for CVPR is Nov. 16th and I think it would give more time to review my writings if I start working on the paper now.



(a) Large Sample After 10 Epochs

(b) Large Sample After 100 Epochs

(c) Small Sample After 10 Epochs

(d) Small Sample After 100 Epochs

Figure 2: Comparison of loss drop while training an AE using random and NNMF.

Future work

I read a couple of abstracts from last year CVPR conference and the majority of papers are about Generative Adversarial Nets (GAN). So in future I have two research goals to work on:

- Start studying and working on GAN.
- Continue my previous work (on weighted multi-modal integration) and develope it using hyper-parameter optimization.