MNIST Digits Classification with PyTorch

Introduction

In this homework, I compare the MLP and CNN models with and without Batch Normalization.

Hyperparameter Selection

In theory, to select the right hyperparameters, a full exhaustive search that iterates through every combination should be done with a **validation dataset**. However, this is done mainly on an ad hoc basis, where the parameters work well enough in combination.

Table1: Hyperparameters used for all models

Loss	CrossEntropy
Learning rate	0.01
Batch size	100
Weight decay	0
Momentum	0.9
Max epoch	50

Results Summary

Table2: Network details

Models	CNN with BN	CNN without BN	MLP with BN	MLP without BN
Architecture	Conv-BN-ReLU-AvgPool-Conv-BN-ReLU-AvgPool-Reshape-Linear	Conv-ReLU-AvgPool-Conv-ReLU-AvgPool-Reshape-Linear	Linear-BN-ReLU-Linear-BN-ReLU-Linear	Linear-ReLU-Linear-ReLU-Linear
>98% test acc	9 epochs	10 epochs	3 epochs	5 epochs
Time per epoch	15s	13s	13s	13s
Time to reach >98% test acc	~130s	~130s	~40s	~65s
Training loss	10^{-2}	10^{-2}	10^{-3}	10^{-3}
Test loss	10^{-3}	10^{-3}	10^{-3}	10^{-3}
Training acc	98.50%	98.87%	100%	100%
Test acc	98.51%	98.46%	98.30%	98.27%

Comments and Discussion

Training time & Convergence:

CNN and MLP take about the same time per epoch. MLP converges (defined as >% test acc) quicker. Adding BatchNormalization layers does not affect run time appreciably.

No. of Parameters:
MLP has more parameters, CNN does parameter sharing and tying. BatchNormalization does not affect no. of parameters.

Accuracy:
CNN achieves slightly better acc. BatchNormalization does not affect this.

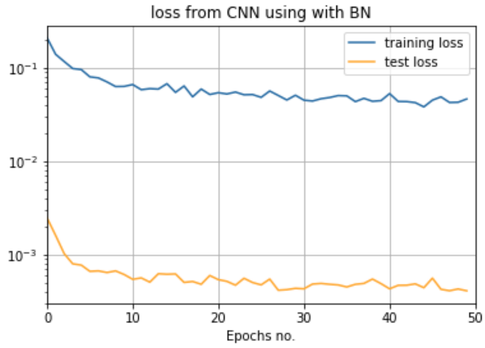
Overfitting:
Overfitting plagues MLP but not so much for CNN. BatchNormalization does not affect this.

Effects of BN:
The use of Batch Normalization is to be more **robust** to bad parameter initialization. Here, we don't see much difference with or without BN. We would perhaps see the effects of BN if initialization is badly done.

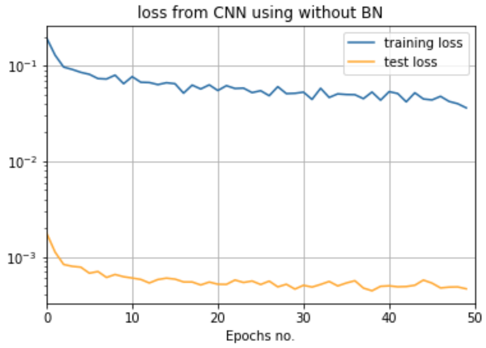
Plots

The loss (training & test) and accuracy (training & test) of each setup are plotted.

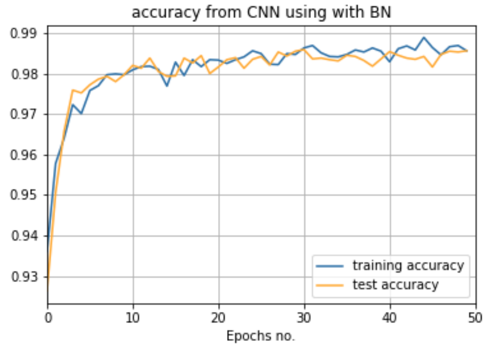
CNN with BN loss



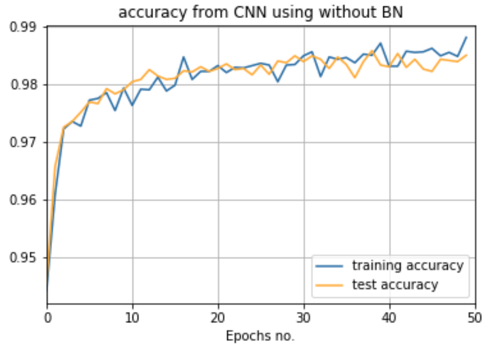
CNN without BN loss



CNN with BN acc

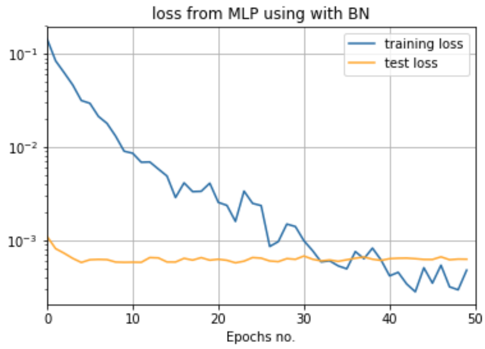


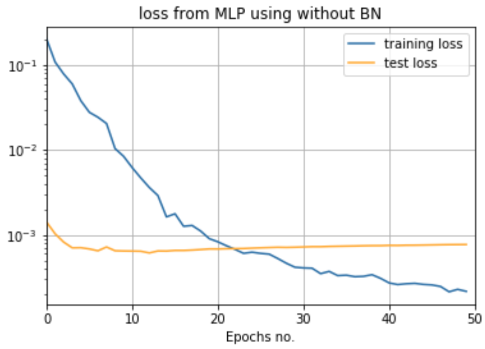
CNN without BN acc



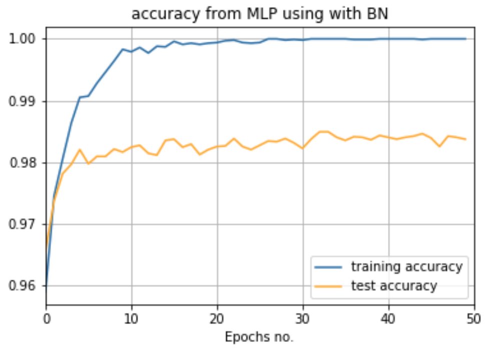
MLP with BN loss

MLP without BN loss

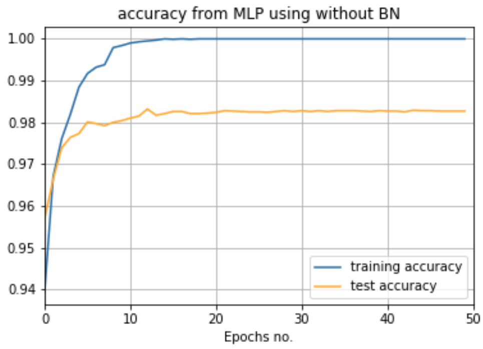




MLP with BN acc



MLP without BN acc



Reference:

* [Deep Learning](http://www.deeplearningbook.org/)
* [CS231n](http://cs231n.github.io/neural-networks-2/)
* [Ioffe, S., Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift](https://arxiv.org/abs/1502.03167)