Luleå University of Technology

D7047E – Advanced Deep Learning

5 April 2023

Exercise 3 | Group 1 | Berezin Ilya, Meenakshi Subhash Chippa

Contents

PCA and t-SNE visualization

Downloading and preparing MNIST dataset.

*Code was developed in Google Colab IDE due to availability of GPU's computational resources, please see the results in shared *.ipynb notebooks with generated outputs, including training epochs statistics and metrics plots.

ConvNet model architecture

conv1: a 2D convolutional layer with 1 input channel, 32 output channels, and a kernel size of 3x3 with padding of 1.

bn1: a batch normalization layer for conv1.

conv2: a 2D convolutional layer with 32 input channels, 32 output channels, and a kernel size of 3x3 with padding of 1.

bn2: a batch normalization layer for conv2.

pool1: a max pooling layer with a kernel size of 2x2 and stride of 2.

dropout1: a dropout layer that randomly zeroes out 25% of the elements of the input tensor.

conv3: a 2D convolutional layer with 32 input channels, 64 output channels, and a kernel size of 3x3 with padding of 1.

bn3: a batch normalization layer for conv3.

conv4: a 2D convolutional layer with 64 input channels, 64 output channels, and a kernel size of 3x3 with padding of 1.

bn4: a batch normalization layer for conv4.

pool2: a max pooling layer with a kernel size of 2x2 and stride of 2.

dropout2: a dropout layer that randomly zeroes out 25% of the elements of the input tensor.

fc1: a fully connected layer with 6488 input features and 512 output features.

bn5: a batch normalization layer for fc1.

dropout3: a dropout layer that randomly zeroes out 50% of the elements of the input tensor.

fc2: a fully connected layer with 512 input features and 10 output features (one for each class).

activation_function: an activation function used in all convolutional and fully connected layers.

The network takes a 1-channel grayscale image as input and outputs a vector of size 10 with a score for each class. The input image is first passed through two sets of convolutional and pooling layers, and then through two fully connected layers. Batch normalization is applied after each convolutional and fully connected layer, and dropout is applied after each max pooling and fully connected layer. The ReLU activation function is used throughout the network.

ConvNet model training and corresponding accuracies

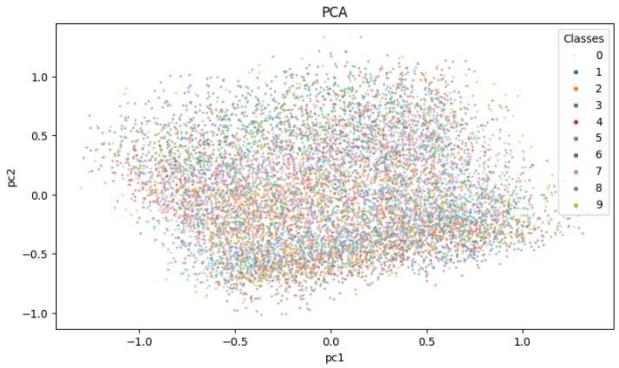
```
Training model ConvNet with learning rate: 1e-06, optimizer: SGD and activation function: ReLU


Epoch 1/1
----------
train progress: 100% 375/375 [00:28<00:00, 14.26it/s, acc=tensor(0.1092, device='cuda:0', dtype=torch.float64), loss=2.54]
                     loss: 2.5413 accuracy: 10.92%
valid progress: 100% 94/94 [00:05<00:00, 16.48it/s, acc=tensor(0.1240, device='cuda:0', dtype=torch.float64), loss=2.29]
                     loss: 2.3079 accuracy: 12.40%

 ------------------------  -------
Model                     ConvNet
Best Validation Accuracy  12.40%
Validation Loss           2.3079
Learning Rate             1e-06
Optimizer                 SGD
Activation                ReLU
Epoch                     1
------------------------  -------

Best model saved to /content/drive/MyDrive/D7047E/MNIST/ConvNet_acc0.1240_lr1e-06_SGD_ReLU_epoch1.pth
```

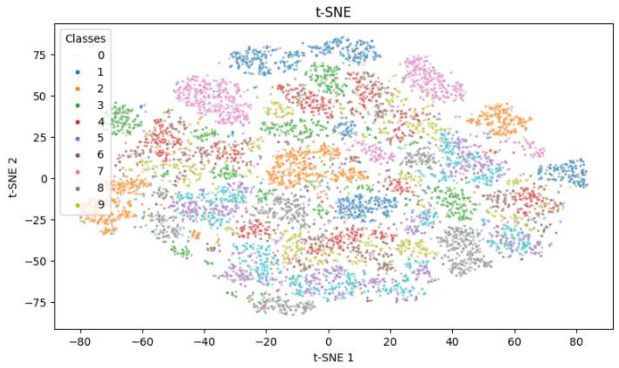
Figure.1.1 Model parameters, trained on MNIST for 1 epoch with learning rate of 1e-6.

```
Training model ConvNet with learning rate: 0.01, optimizer: SGD and activation function: ReLU


Epoch 1/50
----------
train progress: 100% 375/375 [04:56<00:00, 1.37it/s, acc=tensor(0.8527, dtype=torch.float64), loss=0.245]
                     loss: 0.4411 accuracy: 85.27%
valid progress: 100% 94/94 [00:26<00:00, 3.40it/s, acc=tensor(0.9523, dtype=torch.float64), loss=0.064]
                     loss: 0.1544 accuracy: 95.23%

Epoch 00027: reducing learning rate of group 0 to 1.0000e-04.
Epoch 28/50
----------
train progress: 100% 375/375 [04:28<00:00, 1.59it/s, acc=tensor(0.9854, dtype=torch.float64), loss=0.0776]
                     loss: 0.0469 accuracy: 98.54%
valid progress: 100% 94/94 [00:26<00:00, 4.35it/s, acc=tensor(0.9884, dtype=torch.float64), loss=0.0115]
                     loss: 0.0402 accuracy: 98.84%

Early stopping triggered at epoch 28. No significant improvement in the validation metric for 5 epochs.

 ------------------------  -------
Model                     ConvNet
Best Validation Accuracy  98.90%
Validation Loss           0.0390
Learning Rate             0.01
Optimizer                 SGD
Activation                ReLU
Epoch                     23
------------------------  -------

Best model saved to /content/drive/MyDrive/D7047E/model__ConvNet_acc0.9890_lr0.01_SGD_epoch23_ReLU.pth
```

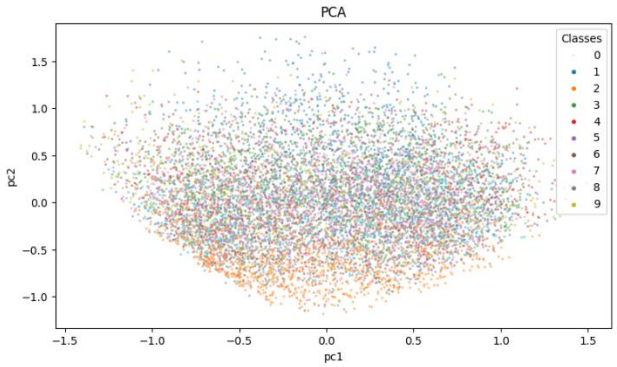
Figure.1.2 Model parameters, trained on MNIST for 23 epoch with adaptive learning rate.

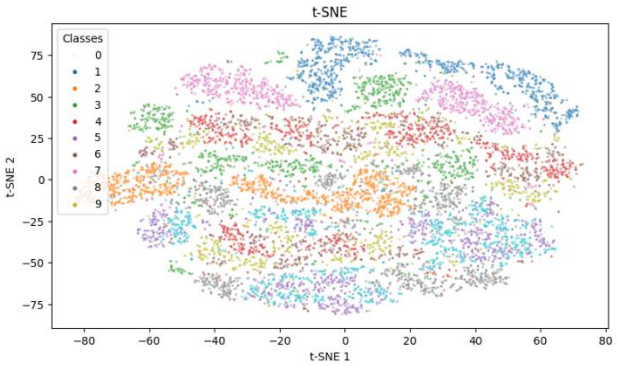
Extracting features after training on the MNIST data with learning rate of 1e-6 for 1 epoch.





Extracting features after training on the MNIST data with learning rate of 1e-3 for 20 epochs.





Extracting features after training on the MNIST data with adaptive learning rate for 23 epochs.