MASC 520 Project 1 Report

Handwritten Digit Recognition using Multi Layer Perceptron (MLP) on MNIST Dataset

Ruoxi Li
5817808450

Abstract

A multilayer perceptron (MLP) is a class of feedforward artificial neural network that can be used to distinguish data that is not linearly separable. In this project, I trained a MLP to classify handwritten digit images in MNIST dataset with 10 different labels from 0 to 9. The neural network has two hidden layers, each followed by a ReLU nonlinearity which has non-saturing property, and a final 10-way softmax. Dropout, a regularization technique, is used to reduce overfitting. On the test data, I achieved 98% accuracy after 14 epochs.

1 Method

The architecture of the model consists of one input layer, two hidden layers and one output layer, i.e. [784 - 512 128 10] units in each hidden layer, as shown in Figure 1. I use Rectified Linear Units(ReLUs) as the activation function followed by each hidden layer. This non-saturating nonlinearity has been proven to be faster to train neural networks than the saturating ones which the gradients are often approaching zero in the limiting cases. When it comes to the larger datasets, learning faster would have a great impact on the performance of the model.

The final layer is a 10-way softmax which produces a probability distribution over the 10 class labels. I use a cross entropy loss for the optimization of the neural network, which minimizes the negative log-probability of the correct label under the predicted probability distribution over the training examples.

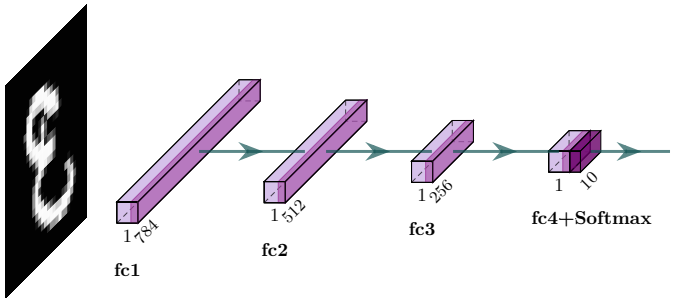

Figure 1: Schematic of the neural network architecture to recognize handwritten digit

To combat with overfitting problem, I use a regularization technique, called Dropout. In short, it keeps a neuron unit active with some probability p, which is a hyperparameter. Alternatively, it

Preprint. Under review.

can also be viewed as sampling a neural network from the full neural network where the parameter updates only contribute to the sampled neural network.

The Adam optimizer is used for gradient descent. The Adam optimization algorithm is an extension to stochastic gradient descent that is faster and more robust than other optimizer. It has been widely used in a lot of neural networks.

2 Result

We need to do data pre-processing before start training. The reason is that activation functions (ReLU) works around zero. If input data is biased toward positive or negative, randomly initialized layers produce biased output. The training data are normalized by the mean and standard deviation computed on the training set.

Adam optimizer with 0.01 learning rate and a minibatch of 64 images is used for training. The dropout probability p is set to be 0.5, i.e. 50% of neurons are turned off. The model is trained for 14 epochs.

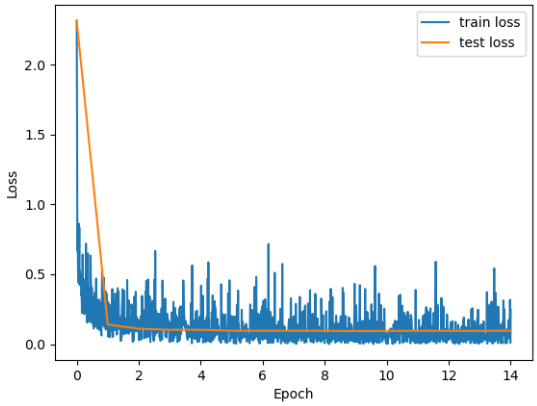

Figure 2: Training and testing loss

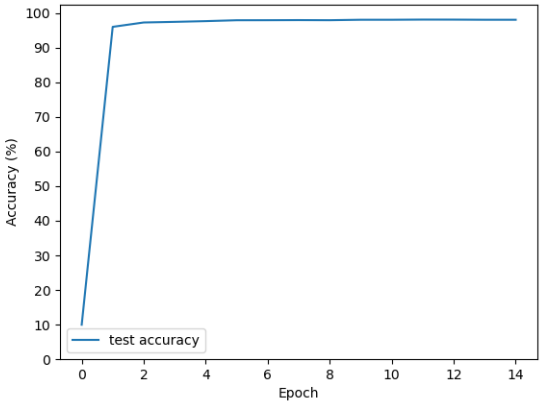

Figure 3: Testing accuracy for recognizing handwritten digit

The MLP is implemented by Pytorch. The training and testing loss is shown in Figure 2, and the testing accuracy is shown in Figure 3. The loss decreases from 2.7 to 0.15 after 14 epochs. The training loss is evaluated every iteration and the testing loss is calculated every epoch. After only 14 training epochs, the accuracy on training set is 99%, and the accuracy on test set is 98%.

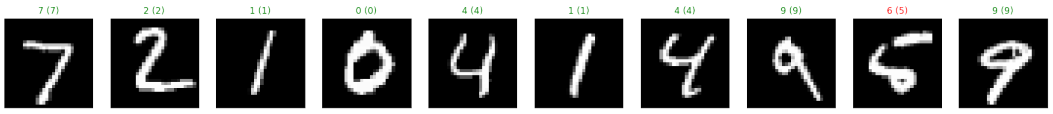

Figure 4: Classification result

To illustrate the result, we sample 10 classified images from test set. The result is shown in Figure 4. The MLP only incorrectly classify the second last figure, which is a little hard to distinguish whether it is 5 or 6.

3 Conclusion

The MLP is a very poweful tool for classification. In this project, I trained a two layer MLP on the MNIST dataset for the handwritten digits classification task. ReLU are used as the activation function and Adams are used as the optimizer. To overcome the overfitting issue, dropout is used for each layer. After tuning some hyperparameters, the model achieves 98% test accuracy after only 14 epochs. The performance can be improved further when the convolutional layer involves.