Report for assignment 1

Mahdieh Sajedi Pour, Nima Taheri

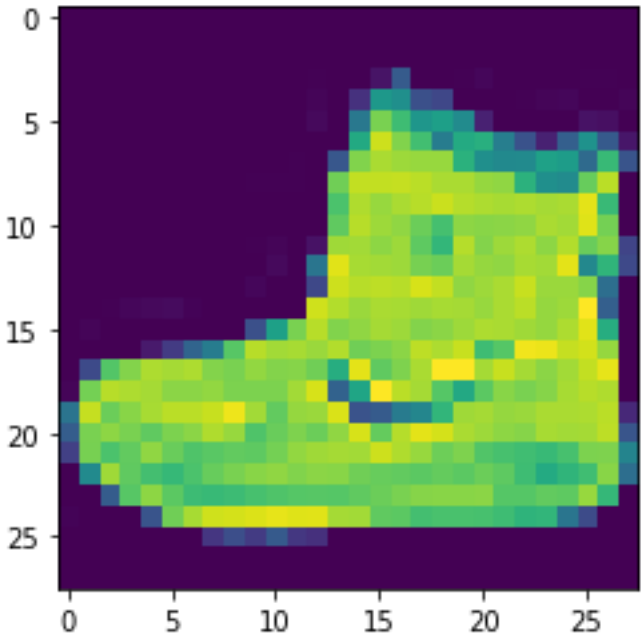
11/7/2022

1) Overview and Problem Statement

We worked with Fashion-MNIST which consists of 10 classes with 60,000 samples as training set and 10,000 as testing set. Each image in the dataset has size of 28 * 28 pixels and 1 channel (gray scale). Our goal is designing a simple MLP to classify these images.

2) Data Preparation

For dataset loading we used ‘torchvision’ module which is a sub-module of PyTorch used for images. We split our dataset into train, validation and test sets. We considered 50,000 examples as train set, 10,000 as validation set and 10,000 as test set. We didn’t use any specific augmentation for images but we converted them to Tensor. Also, we shuffled the datasets and used batch size of 64 for batching the data. We checked for Cuda availability and used GPU to see if it is available. Also, we set random seed for reproducible results.



3) Methodology

We changed many features in our neural network to check the results. They are listed below:

A) The effect of different hidden layers
Comparing three simple neural networks with 1 hidden layer, 3 hidden layers, and 5 hidden layers.

B) Using dropout method
Comparing two neural networks consisting of 3 hidden layers, with and without a dropout layer.

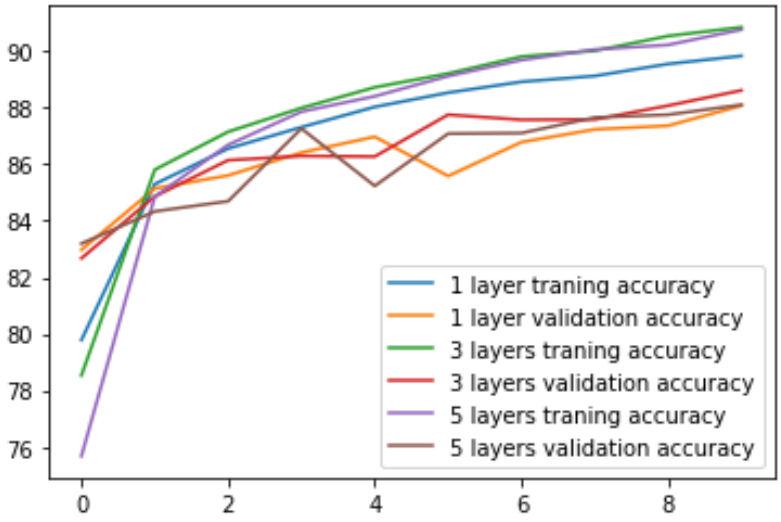
C) Early stopping
We track a metric and if that metric doesn't improve during training we stop training.

D) Batch normalization
Comparing two neural networks consisting of 3 hidden layers, with and without normalization.

E) Regularization
Comparing two neural networks consisting of 3 hidden layers, with L1 and L2 regularization methods.

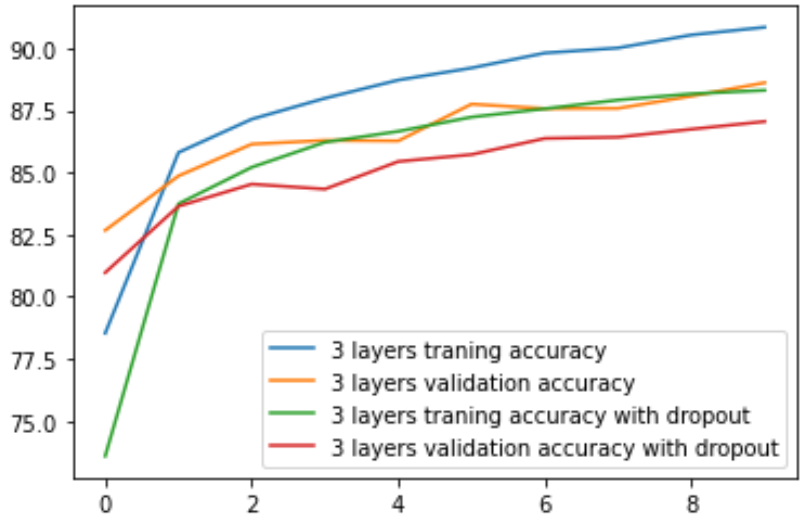
4) Result

A) The effect of different hidden layers



With one hidden layer, we get 88% accuracy. It will increase up to 88.7% when we use 3 layers. When we use 5 layers, we can see that overfitting happens (accuracy = 88.1%). So increasing the number of hidden layers cause accuracy in the test set to decrease. It will cause our network to overfit to the training set. It will learn the training data, but it won't predict new data correctly.

B) Using dropout method

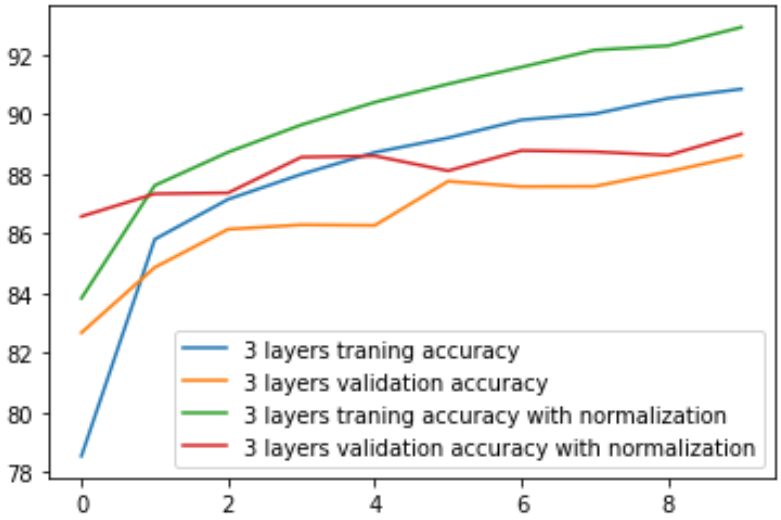


In each training iteration, each node in the network is associated with a probability p whether to keep in the network or to deactivate it (dropout) out of the network with probability 1-p. That means the weights associated with the nodes got updated only p fraction of times because nodes are active only p times during training.
So it supposed to improve the network by preventing overfitting. But in this case, as the network is not that complex, the result is not what we expected.

C) Early stopping

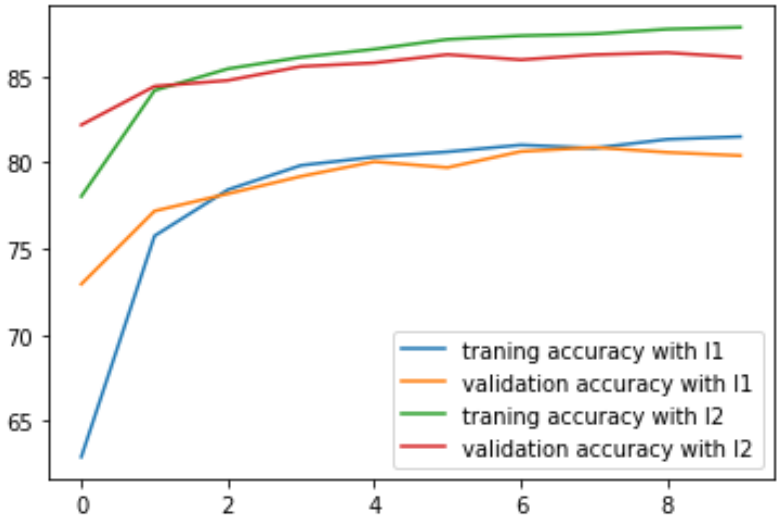
Early stopping is a form of regularization used to avoid overfitting when training a learner with an iterative method.
Here we can see that the model has the least loss at epoch 18 and that's where we save our model. For the next 5 epochs, loss increases and we have an early stopping.

D) Batch normalization



To avoid the learning algorithm, spend much time swinging in the plateau, we normalize the input features such that all the features would be on the same scale. Since our inputs are on the same scale, the weights associated with them would also be on the same scale. This would help the network to train faster. From the plot, we see that the model has been trained faster and better.

E) Regularization



Regularization is a set of techniques that can prevent overfitting in neural networks and thus improve the accuracy of a Deep Learning model when facing completely new data from the problem domain. L2 works better on this dataset.