Deep Learning Lab

Autonomous Intelligent Systems

Exercise 1

Author: Eduardo ALVARADO

Matriculation No. 4454388

1st November 2017

1. Introduction

In the following exercise, we have implemented a Neural Network (MLP) to perform a classification task for the dataset MNIST.

In summary, a MLP works as follows: Every layer requires different arguments such as the activation function type (*relu*, *tanh* or *sigmoid*), or layer size. The layer uses this information to compute more important paraments like the dimensions of the layer. As a result, the layer provides us three outputs: *fprop()* (forward pass), *bprop()* (backward pass) and *output_shape()* (shape of the next layer in the network).

2. Neural Network Design

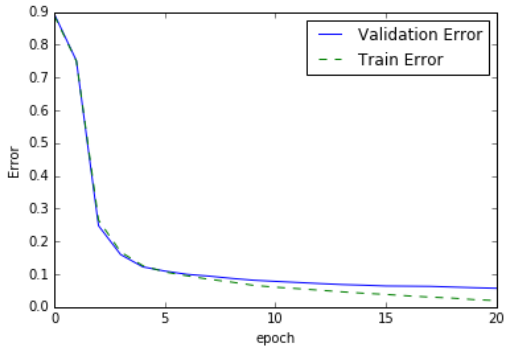
1.1. Default Case

The Network Design provided by the exercise consists of 3 Fully Connected Layers. The first two Layers use *Relu* as Activation Function, and each of them has *num_units = 100*. The third layer uses a *Linear* Activation Function, with *num_units = 10.*

Moreover, the training of the network is made in based on the following paraments: *max_epochs = 20, batch_size = 64, learning_rate = 0.1*

The following results were obtained for the first and last Epoch:

Epoch	Training error	Validation error
0	88.90%	89,36%
20	1,94%	5,69%




```
Duration: 76.4s
```

Figure 1: Training and Validation error for default design

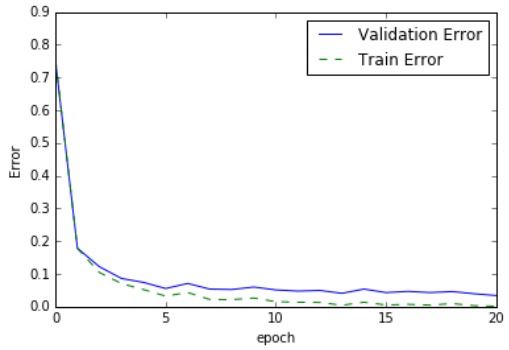
1.2. Improved Case

The improved Network designed by me consists of 4 Fully Connected Layers. The first three Layers use *Relu* as Activation Function. The first two *Relu* have *num_units = 250, and the third one uses num_units = 200*. The fourth layer uses a *Linear* Activation Function, with *num_units = 10,* like in the default case.

The parameters used for the training of the network are: *max_epochs = 20, batch_size = 32, learning_rate = 0.4*

The following results were obtained for the first and last Epoch:

Epoch	Training error	Validation error
0	75.66%	75,17%
20	1,2%	3,39%




```
Duration: 254.2s
```

Figure 2: Training and Validation error for improved design

3. Conclusion

The default design and custom design provide us different results, since the parameters of both differ. Although both designs are based on *Relu* as Activation Functions, there are some main differences which we should emphasize: The improved design has one more layer, 4 in total. Also, the *num_units* increase up to 250 for two of the layers, and up to 200 for the other one.

The training parameters changed as well: We have decreased the *batch_size,* from 64 to 32, and increase the *learning_rate,* from 0,1 to 0,4.

These modifications can be reflected in the Training and Validation error of the designs: At 0 Epochs, both, the Training error and Validation error decreased: From 88.90% to 75.66% in the first case, and from 89,36% to 75,17% in the second case. At 20 Epochs, both errors decreased in the following way: For the Training error, from 1,94% to 1,2%. For the Validation error, from 5,69% to 3,39%.

With this information, we can certainly establish that a better overall estimation of the MLP can be achieved by increasing the *learning_rate,* and decreasing the *batch_size.* Also, an increment in the number of Fully Connected Layers and *num_units yield* a significant improvement.