Assignment #1 -- Feed-Forward Neural Network

*Statement of Purpose*

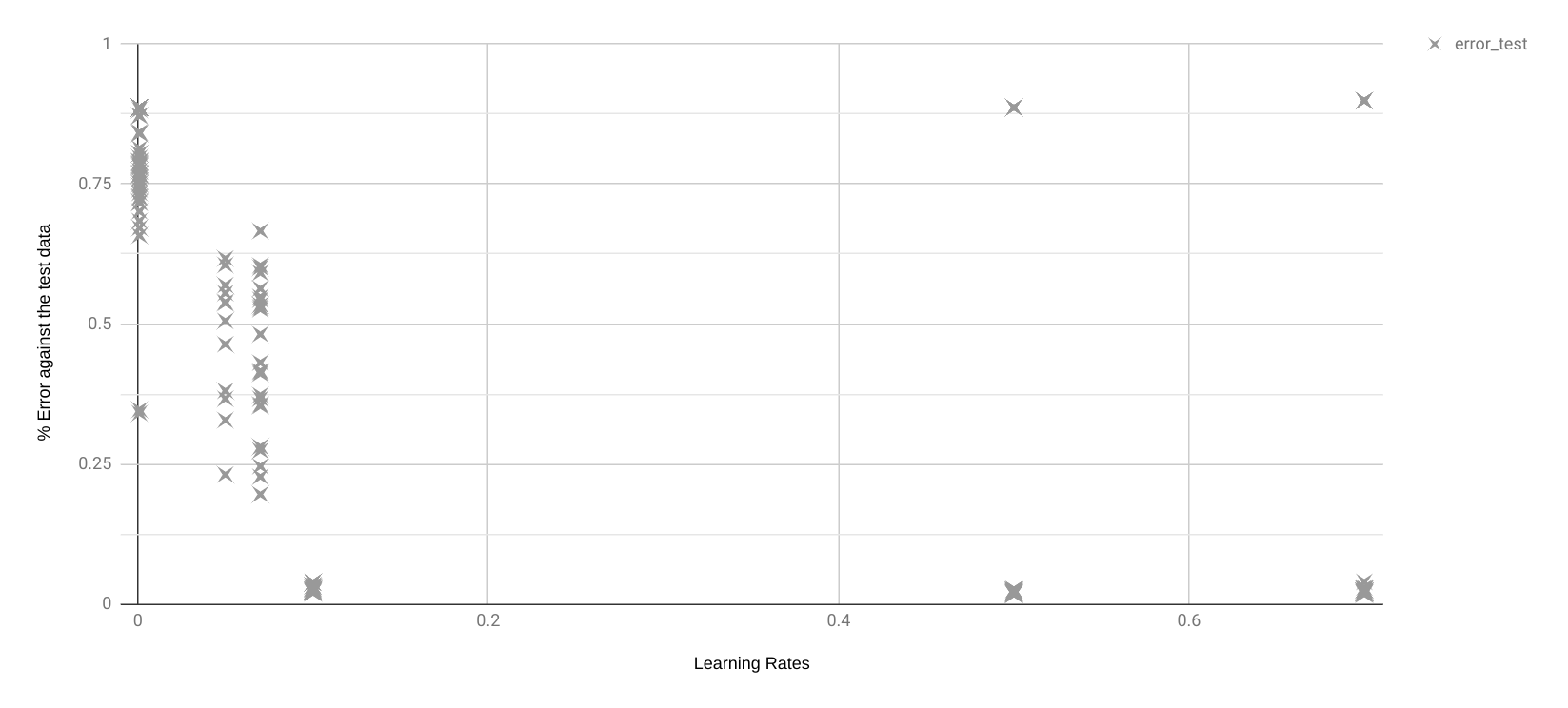
The provided skeleton for a class base neural network, available in <https://github.com/aisrobots/dl-lab-2018>, allowed the creation of n-hidden layer neural networks, with tanh, sigmoid or ReLu activation. Figure 1 describes the architecture and denotes the complication induced by the object abstraction of the neuronal class with layers as objects. A NN (Neural network class for short), receives a group of layers, or class architecture, to learn how to decipher human handwriting. There are three types of layers: Input (which just propagates the inputs to the NN so the user can predict what number does the handwriting represent), Output Layer (softmax output with one hot encoding for easier training) and hidden layers (which are parametrizable objects that can learn through the weights and bias attributes of the class).



Figure 1. Architecture of the neural network communicating layer objects

Particularly of interest is that the NN would learn how to predict handwriting, by modifying the W (weights of each layer) and b (bias of each layer) through their respective gradients (dW and db). To accomplish, validation data was taken on a testbench that ran combinations with the following criteria:

* learning rate from 0.001 to 0.7
* Permutations of 2 layers that could be {relu, }



|  |  |  |  |
| --- | --- | --- | --- |
|  | Second Layer | | |
| First Layer | relu | sigmoid | tanh |
| relu | 0.28410 | 0.26040 | 0.24000 |
| sigmoid | 6.41580 | 0.32890 | 0.28880 |
| tanh | 0.27460 | 0.27580 | 0.26150 |

|  |
| --- |
| epoch 30.0000, loss 0.0001, train error 0.0000 epoch 30.0000, loss 0.0889, valid error 0.0164  Duration: 295.7s    Error in test 0.0173 |