REPORT- Assignment 3

Approach followed for model.py-

In this approach we tried defining a feed-forward with a maximum of 3 hidden layer.
Each of the layer is densely connected to its previous and later layers (Fully Connected Network).

The neural network object is defined in the main training scripts, hene command are given to add hidden layer of given width, output layer and training and predicting the output on the test data. The training function takes the input and labels and then either train and validate or predict depending on the type of input specified. The predicted output is then sent back to the training script and is is then printed and save in form of a .csv file for evaluation.

The class defined in mdel.py contains various member functions to perform action such as - adding a hidden layer, output_layer, predicting, evaluating loss and optimising.

Optimiser used- GradientDescent()

**There are two train function in the model.py file. First one is for the mnist.py and second one is for speech.py. While training for either of the two the training function of other should be commented.

Optimum value of hyperparameters-

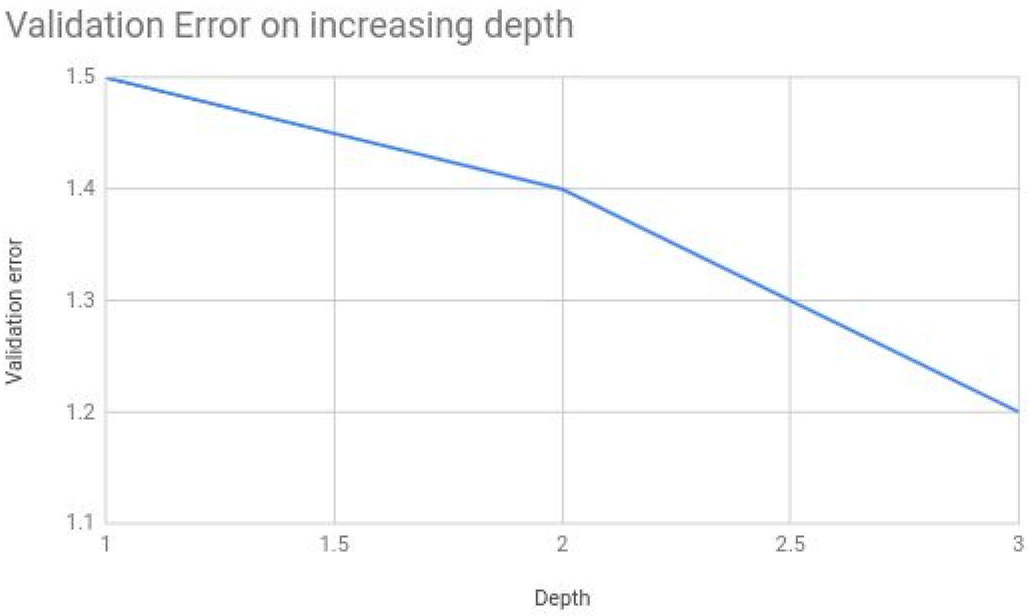
1) No of Nodes in hidden layers

In all the cases it has been observed that that the error decreases with increasing the number of nodes in the the hidden layer. We have chosen the optimum value of layer width to be 1000 as there is no significant improvement in accuracy on increasing it further.

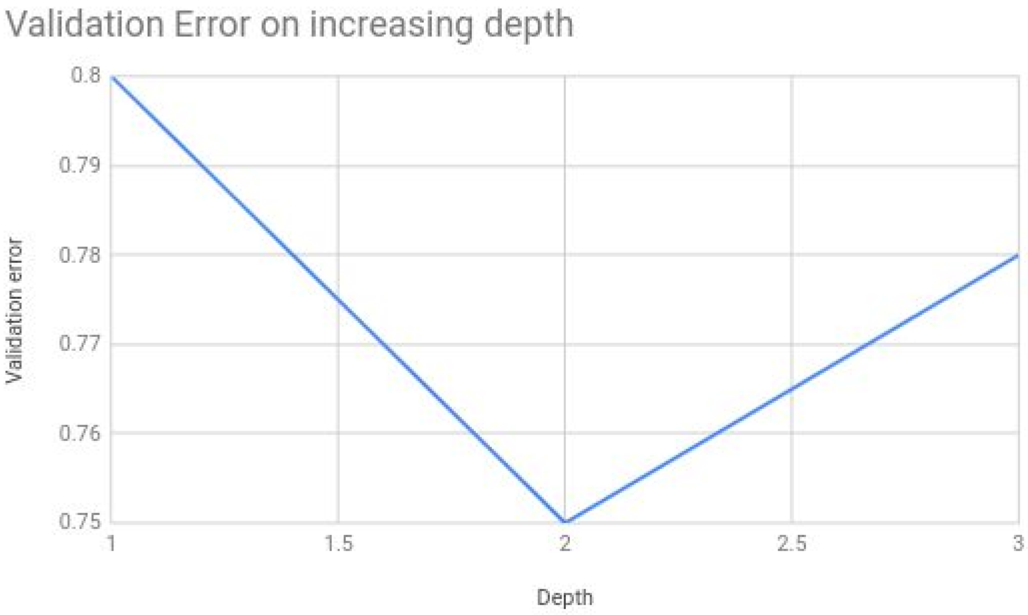
2) No of Hidden Layers(Depth of the network)

The error value decreases on increasing the number of hidden layer in the network. This decrease was significant in Mnist.py but not so significant in Speech.py. Hence the optimum value of network depth is 3 for mnist and 1 for speech recognition.

Following the reduction of validation error with depth for mnist

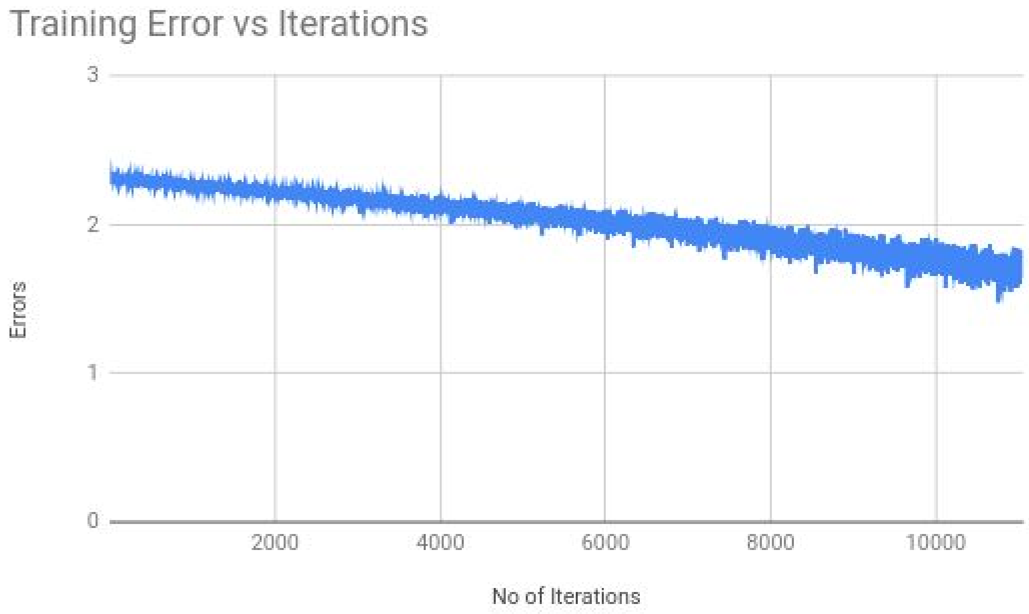


Validation Error almost remains constant for the case of speech

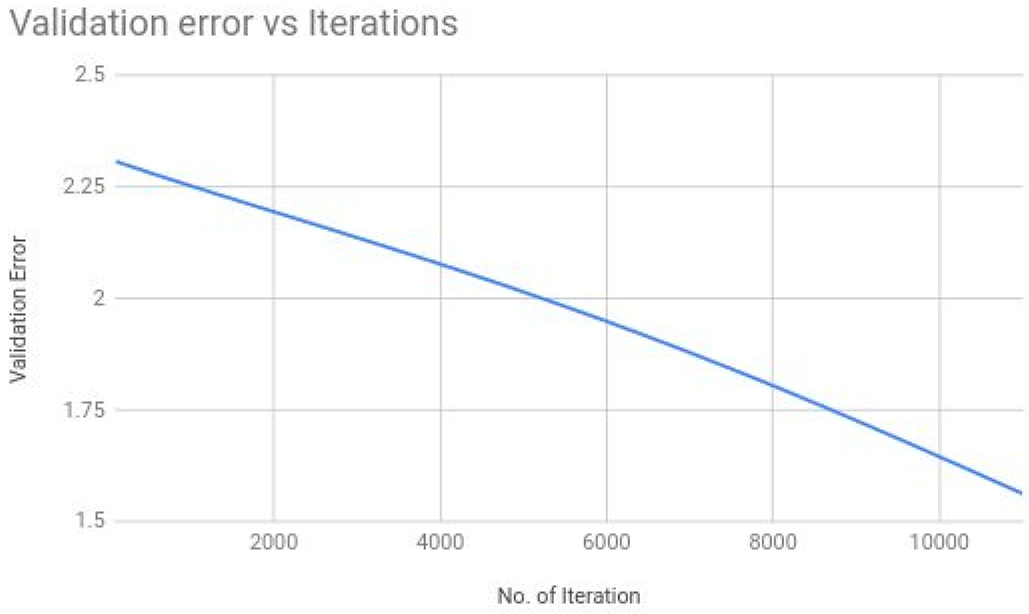


Graph for variation of training error with time→

Following is the variation of Training Error and Validation Error with theNo. of Iternations-
This error report is for the case of mnist.py training



Graph for variation of test error with time→



Here number of Iterations= Max Epoches * No. of steps per Epoches

As it can be seen that the values of both test and tran error is is decreasing with increase in number of iterations. It will soon reach an optimum value and continue to oscillate about the mean.

*Similar are the graphs for Speech where the Training and Validation is decreasing with time
The outputs of both the files have been attached the folder.