

Name : Yashas Nagaraj, Udupa, 01647695

Group Partner : Keerthi Datta, Konanur, Ramanna, 01647641

4. Optimizing Deep Neural Networks

This is a neural network classification problem, where objective is to optimize the cross-entropy error of the network by checking with a greater number of hidden layers, different Gradient Descent Optimizers, different non-linearities, suitable learning rates if it helps in improving the performance of classification with respect to the previous task's performance. Subsequently, to compare the newly found architecture with ResNet architecture to observe if there is further advantage.

- a) Neural network with 9 hidden layers and 40 neurons each layer were being adjusted within the network. Three different learning rates were being considered i.e, $\{1.0e-3, 1.0e-4, 1.0e-5\}$ with two different non-linearities ReLu & tanh. And previous task's stopping criteria is being reused i.e 34 epochs. Following are the training/test losses & test accuracies that were obtained for corresponding learning rates.

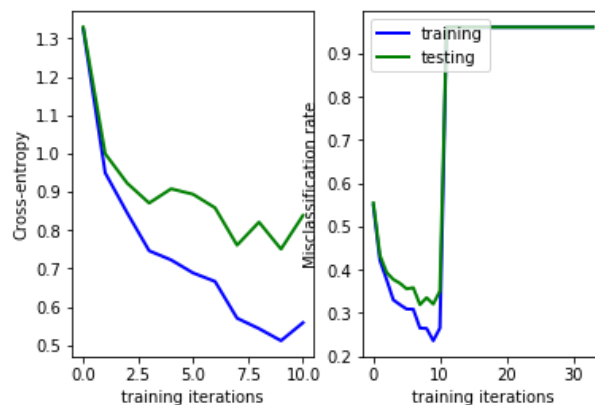
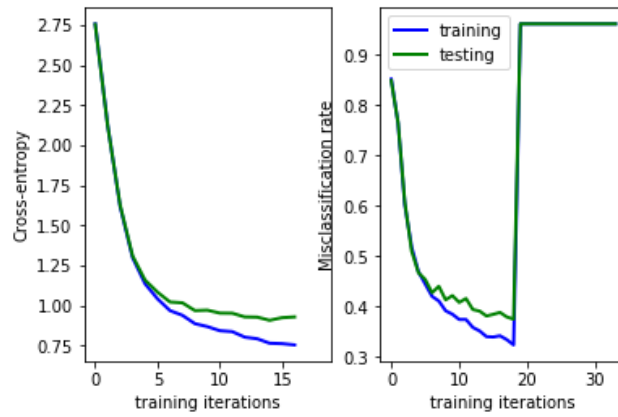
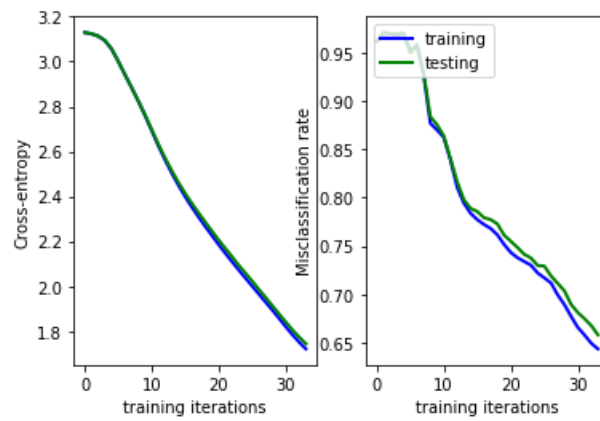
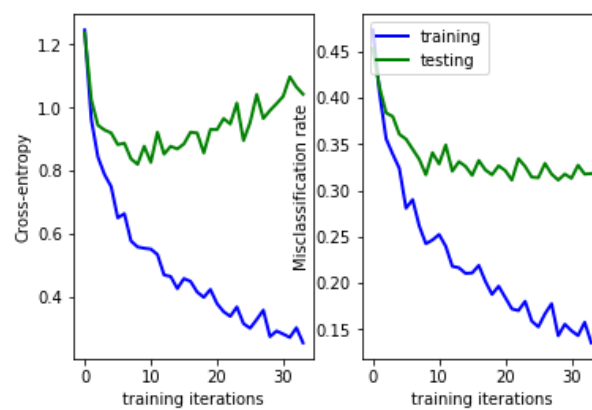


Fig 1: Learning Rate: 1.0e-3 (ReLu)

Fig 2: Learning Rate: $1.0e-4$ (ReLU)Fig 3: Learning Rate: $1.0e-5$ (ReLU)Fig 4: Learning Rate: $1.0e-3$ (tanh)

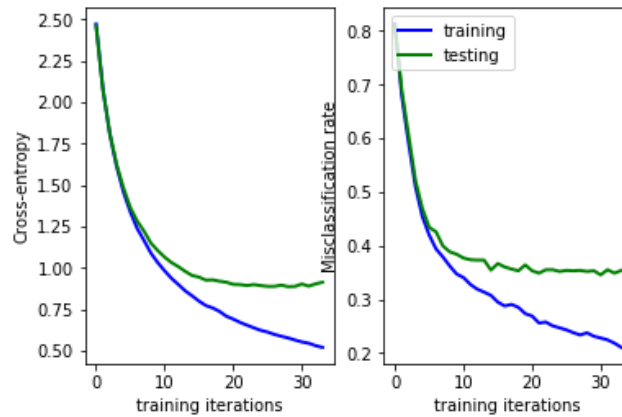


Fig 5: Learning Rate: 1.0e-4 (tanh)

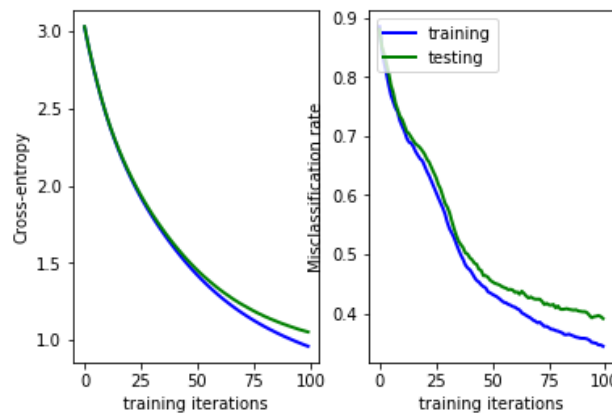


Fig 6: Learning Rate: 1.0e-5 (tanh)

From the plots with different learning rates it is observed that for some learning rates the Gradient suddenly started spiking to high “nan” once it reached to around 68% test accuracy. When compared to tanh and ReLu both were reaching up to 68-69% accuracy. And as per, the results and observations of task-2, if more than 200 neurons are being considered in the hidden layer, the accuracy to start decreasing which, really made sense in this task while observing with more hidden neurons but distributed over multiple hidden layers. However, neural network with the later architecture, test accuracy has decreased compared to architecture of task2(Implies that the network is too deep). But in comparison to ReLu & tanh, with 3 different learning rates, with some combinations it is being observed that it started occurring to “nan” state (resulting least accuracy) over the iterations. And with learning rate 1.0e-3, there observed “overfitting” to training data. The fairly best-found combinations were learning rate - 1.0e-4 & hidden activation function – ReLu.

- b) Now with the ResNet architecture, the performance is being measured. Following are the plots that were obtained for different learning rates with 34 epochs.

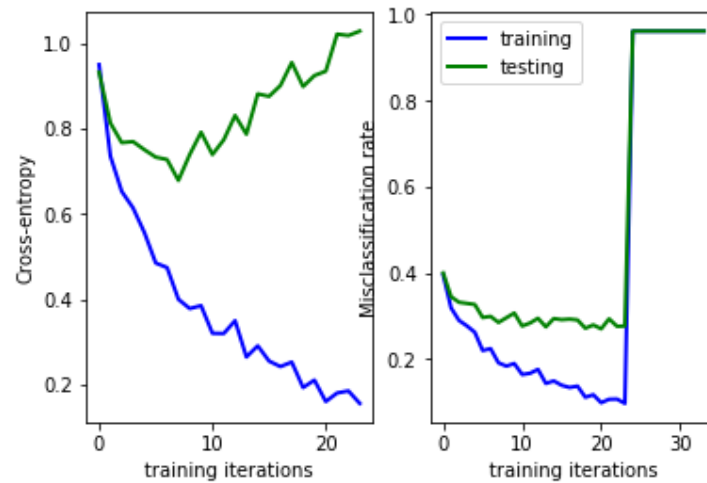


Fig 7: Learning Rate: 1.0×10^{-3}

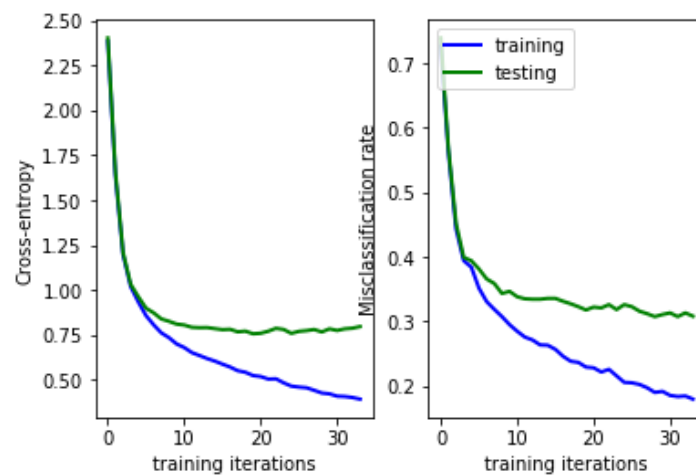


Fig 8: Learning Rate: 1.0×10^{-4}

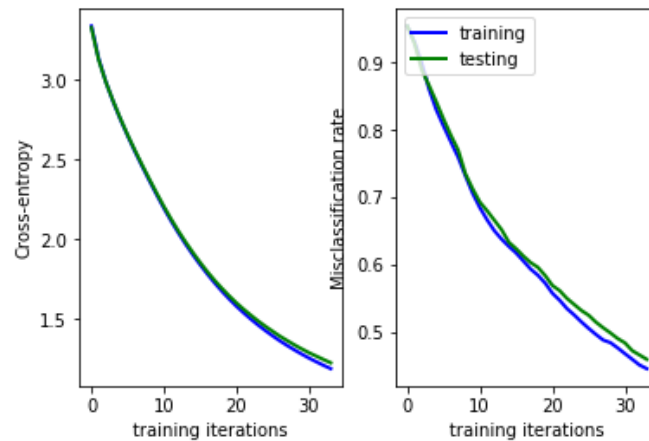


Fig 9: Learning Rate: 1.0e-5

From the above plots, it was observed that with learning rate 1.0e-4 fits better in performing deep neural network classification problem. Where, it manages to reach up to 70% test accuracy with training loss: 0.404 & test loss: 0.781 without overfitting the training data.