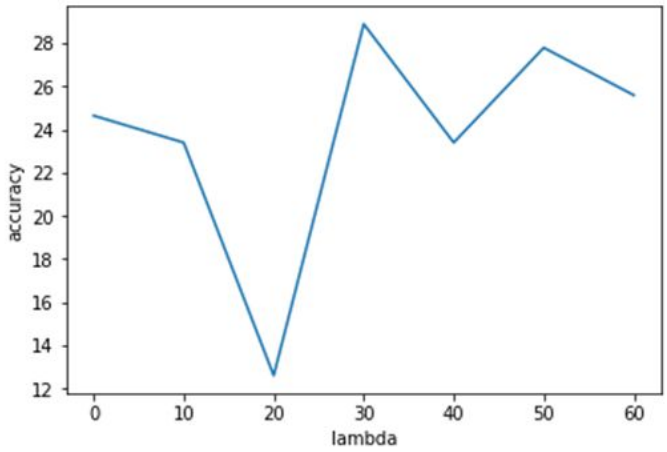
Project 2

CSE 474
Group:23

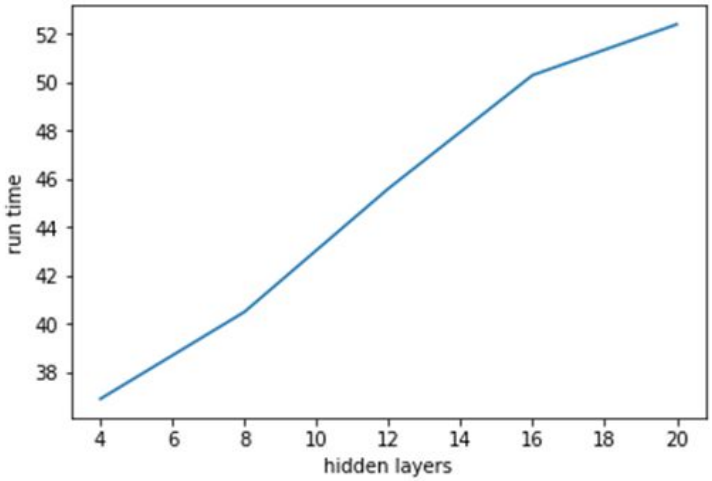
Anthony Rubin
Vikram Singh
Abhijeet Verma

Regularization in the neural network



To understand the regularization parameters such as accuracy, run time and hidden layers we have plotted the graph taking different values and checking the output for the mnist_all i.e the handwritten data over the digits 0-9.

In the above graph we can see that the accuracy decreases at lambda=20 and then increased at lambda=30 and then decreases from there.Therefore we can predict that lambda is attaining maximum accuracy near lambda=30.



Note: Runtime in seconds

The runtime vs hidden layers clearly depicts that as we increase the hidden layer, the run time increases accordingly. This happens due to the increased complexity of computation between the hidden layers of the neural networks. The runtime is increasing when we choose the lambda is equal to 30.

The common inference in the graph can be seen that the graph reaches the peak and decreases. This happens due to the hyper parameters trying to fit the data accordingly. We have to choose the regularization parameters in such a way that it neither overfits the model nor it underfits it.

Therefore we found that the optimal selection of hyper parameters can be chosen as lambda=30 and hidden layers=20, when we choose 20 as the upper limit for the hidden layers.

– Accuracy of classification method on the hand written digit set:

Test set Accuracy: 32.1%
Runtime 40.237738847732544
Tunned hyper-parameter for neural network
n_hidden =28
lambdaval= 30

– Accuracy of classification method on the AI Quick Draw data set:

Test set Accuracy: 22.512%
Runtime: 34.53272318840027

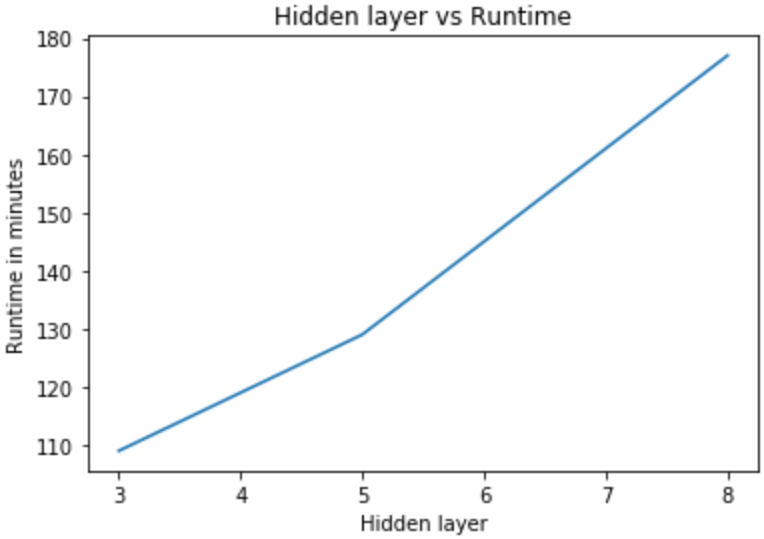
Tunned hyper-parameter for neural network
n_hidden= 12
lambdaval= 40

Note: each time we are running with tunned hyper-parameter we see fluctuate in accuracy with ±10%

Tensorflow tests on Quick Draw data set.

The following tensorflow run times are specific to a 2012 Macbook Pro with a Samsung 860 Evo solid state drive. The accuracy of neural networks with 3,5, and 7 hidden layers were all very similar, but of course the runtimes differed greatly.

Three Hidden Layers
Runtime 109 minutes, Train_loss 1.18750, train_accuracy 0.565, test_loss 1.14687, test_accuracy: 0.61484
Five Hidden Layers
Runtime 129 minutes, train_loss 1.106253, train_accuracy 0.655, test_loss 1.16743, test_accuracy 0.60796
Seven Hidden Layers
Runtime 177 minutes, train_loss 1.09476, train_accuracy 0.63, test_loss 1.19184 test_accuracy 0.60128



As the graph shows the learning time of the neural networks increases as we increase the hidden layers from 3,5 and 7. This happens due to the fact that the complexity of the problem increases with increase in the number of hidden units which increases the time of execution.

The fact that the test accuracy didn't change by more than 1.5% shows the is proof of the difficulty of the given data set. As expected, the more hidden layers there were the higher the runtime was.