Fall-2018
Chirag Khurana (2016CSB1037)

Multi-layer perceptrons and Self Driving Cars

Introduction

A multi-layer neural network with 1024 - 512 - 64 - 1 architecture is implemented with sigmoid as the the activation function in each layer except the output layer. The weights of each layer are initialized by uniformly spacing over the range [-0.01, 0.01]. The bias terms are initialized to 0.

Loss Function:

$$\frac{1}{2}\sum_{i}(y_i - o_i)^2$$

Input/Output:

A grayscale image of size 32x32 is given as input and prediction of the steering angle is given as output.

Input grayscale image is converted to array of size 1024.

Data Normalization is done by:

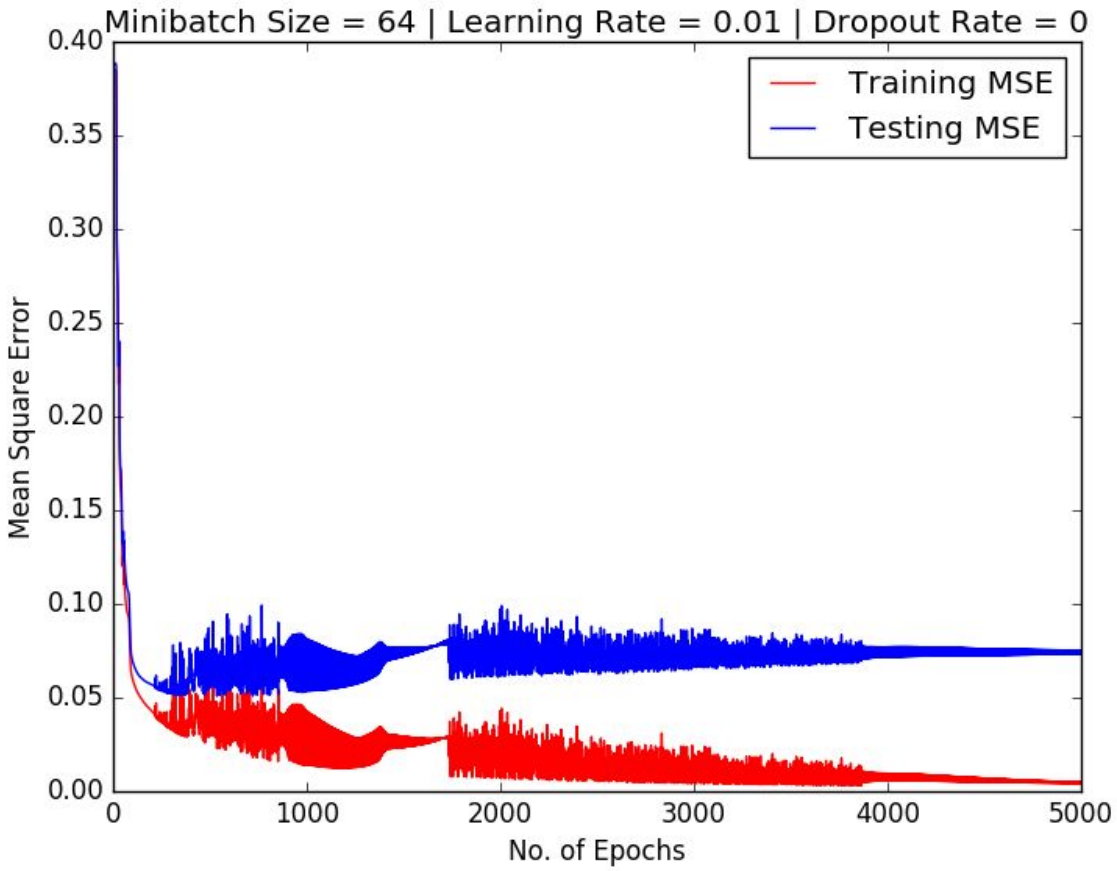
$$X' = \frac{X - min(X)}{max(X)}$$

Training/Validation Split Ratio = 80:20

Experiment - 1

(Epochs: 5000, Learning Rate: 0.01, Minibatch Size: 64)

Plot of Mean Squares Error on the training and validation set as a function of training iterations:

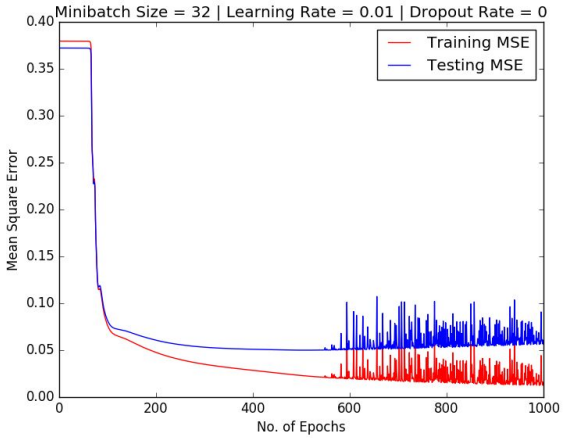


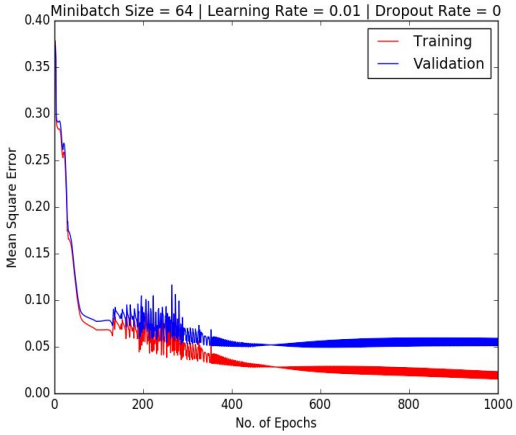
Observations: It is clearly visible that as number of epochs increases large, it will lead model to overfit the training data because training error decreased and validation error increased.

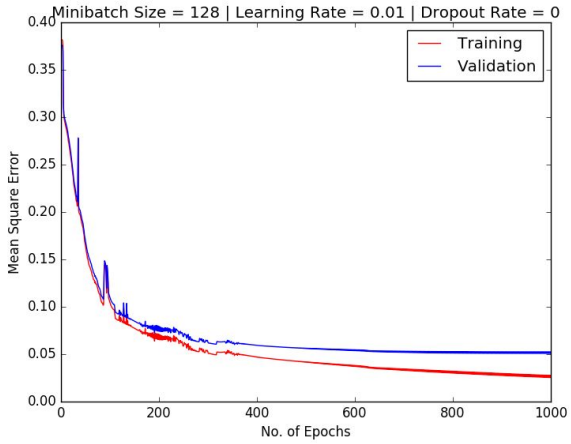
Experiment - 2

(Epochs: 1000, Learning Rate: 0.01, Minibatch Size: 32, 64, 128)

Plot of Mean Square Error on the training and validation set as a function of training iterations:





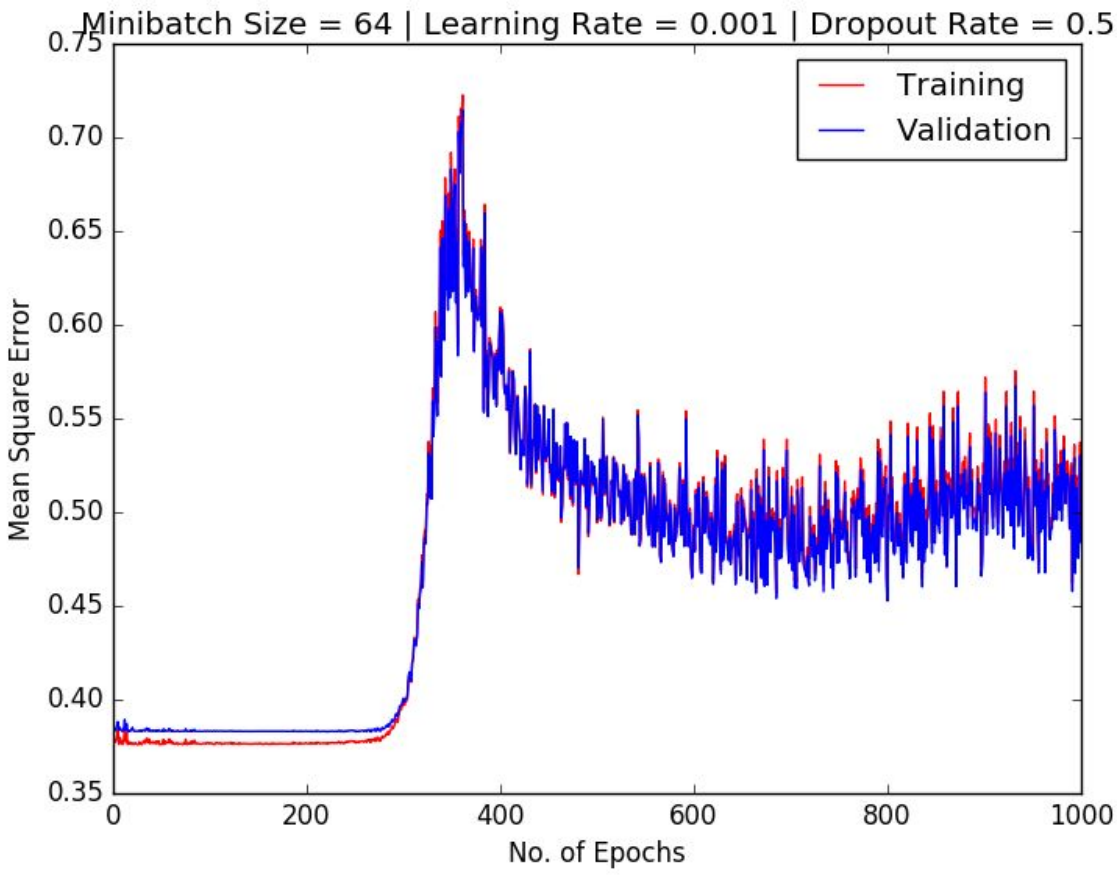


Observations: As minibatch size increased, error saturated(means less fluctuation in consecutive epochs) in less number of epochs. As we can see fluctuation in 32 started around 600 epochs, in 64 started around 125 epochs, and in 128 finished around 300 epochs.

Experiment - 3

(Epochs: 1000, Learning Rate: 0.001, Minibatch Size: 64, Dropout Rate: 0.5)

Plot of Mean Square Error on the training and validation set as a function of training iterations:



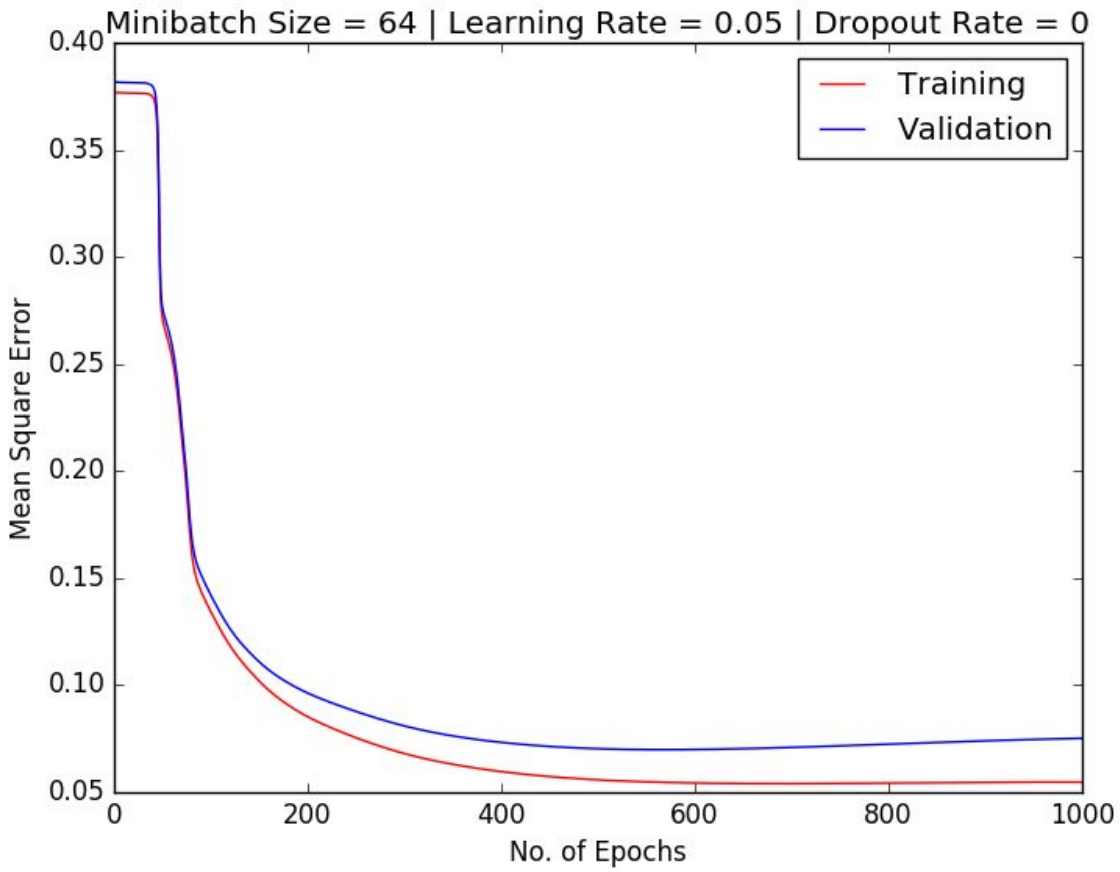
Observations: Overfitting can be avoided with dropouts, in this case training and validation errors are approximately equal.

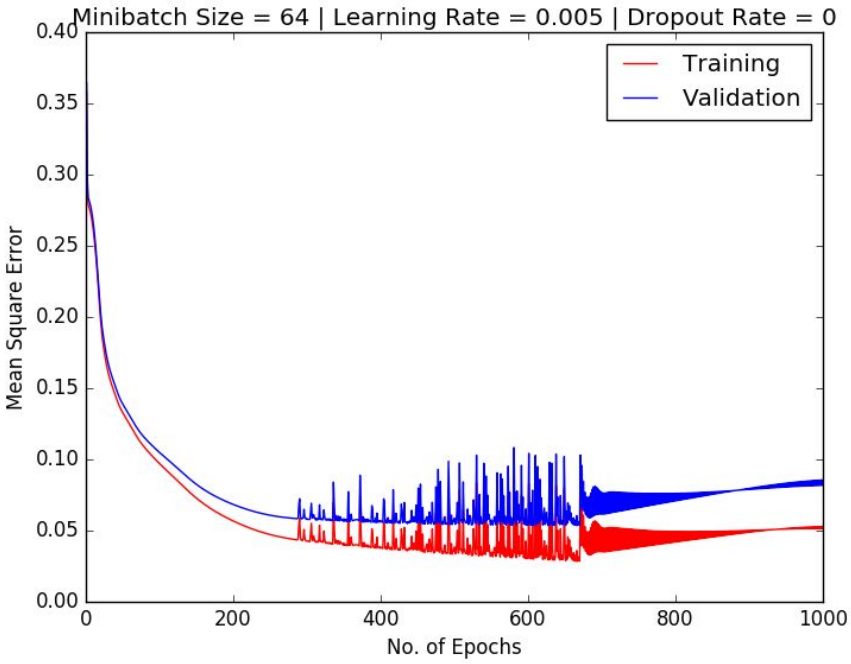
Experiment - 4

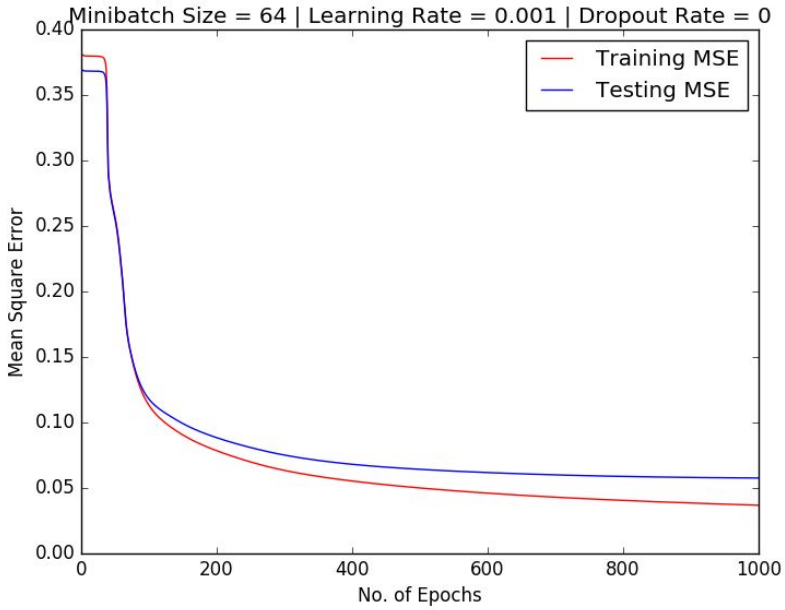
(Epochs: 1000, Learning Rate: 0.05, 0.005, 0.001, Minibatch Size: 64)

Note: For Learning Rate = 0.05, model started diverging, therefore in this case mean square error loss function is used instead of sum of square error indirectly learning rate become very low and hence curve is very smooth without error fluctuations.

Therefore observations based on other learning rates for this experiment are given below.





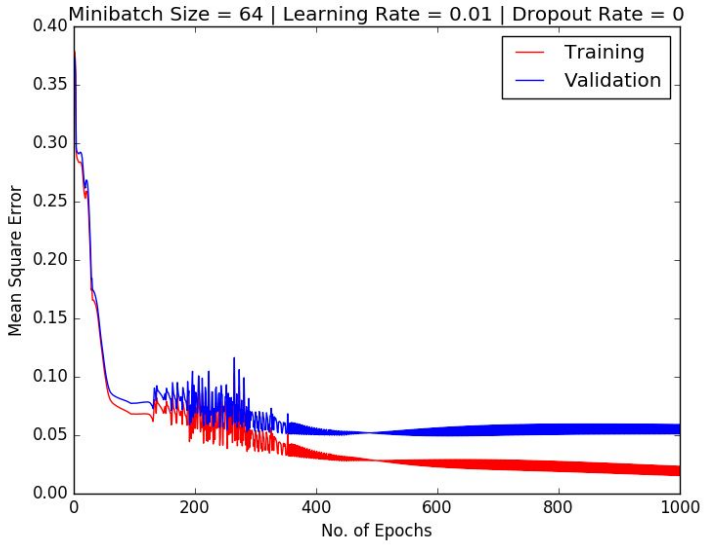


Observations: As learning rate decreases-
1. Error fluctuations becomes less.
2. Converges slowly
3. Less overfitting for same number of epochs.

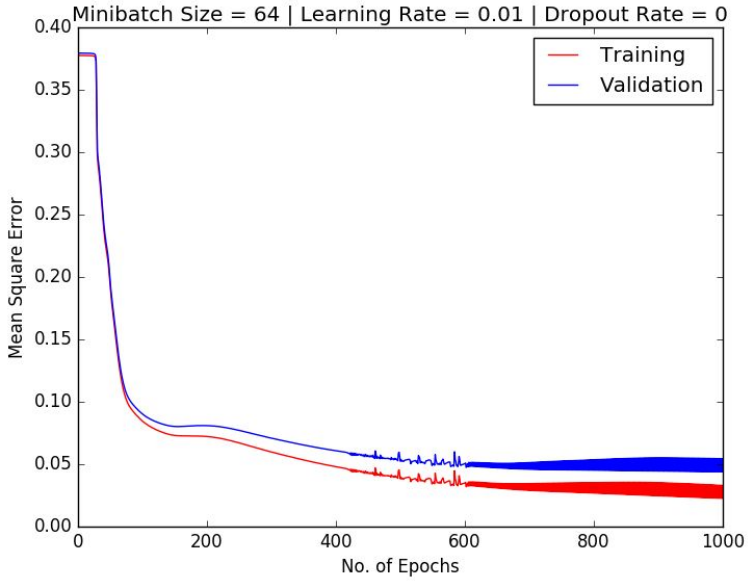
Bonus question:

(Epochs: 1000, Learning Rate: 0.01, Minibatch Size: 64)

Architecture: 1024 - 512 - 64 - 1



Architecture: 1024 - 512 - 256 - 64 - 1



Observation: Fluctuation of errors decreased as number of layers is increased.