



Indian Institute of Technology Madras

M.Tech (Computer Science and Engineering)

DEEP LEARNING - Course 2019

Programming Assignment 1

SUBMITTED TO:
Prof. Mitesh Khapra

SUBMITTED BY:
CS18M038, Ojas Mehta
CS18M052, Avani Shukla

Team Name: 101010101

1 Task

Built a Feedforward neural network for image classification task. The dataset used is "A-Z Handwritten Alphabets dataset" having 10 different handwritten english alphabets each represented as 784 pixel image. The train, validation and test split of 55000, 5000 and 10000 images respectively.

2 Plots

1. One hidden layer with sigmoid activation function, cross entropy loss, Adam optimization and batch size 20.
varying the size of the hidden layer (50, 100, 200, 300)

- Hidden layer size: 50

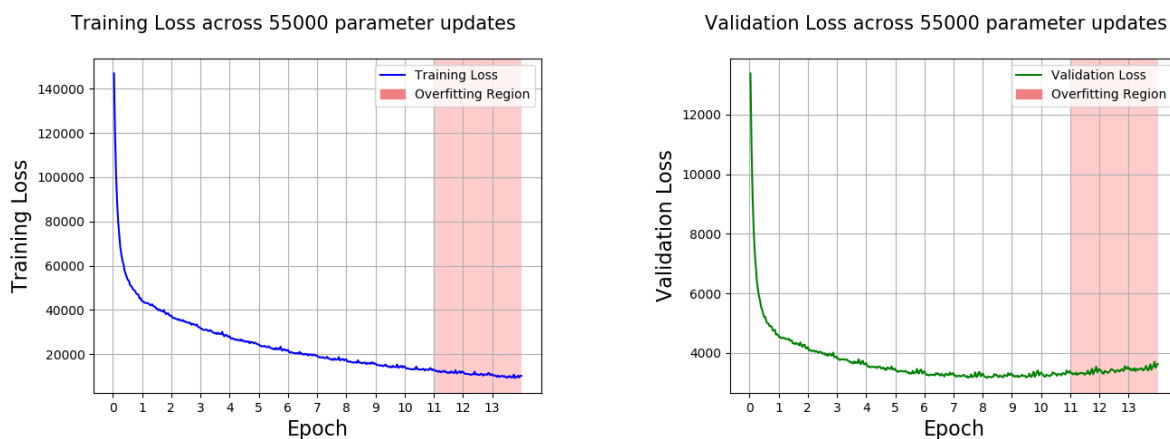


Figure 1: Plot of Loss vs Epoch for learning rate 0.001

- Hidden layer size: 100

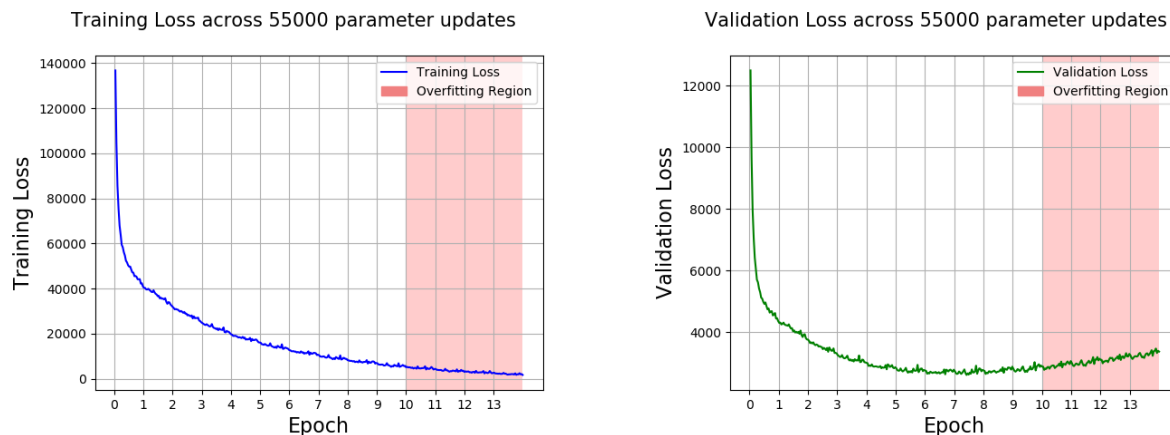


Figure 2: Plot of Loss vs Epoch for learning rate 0.001

- Hidden layer size: 200

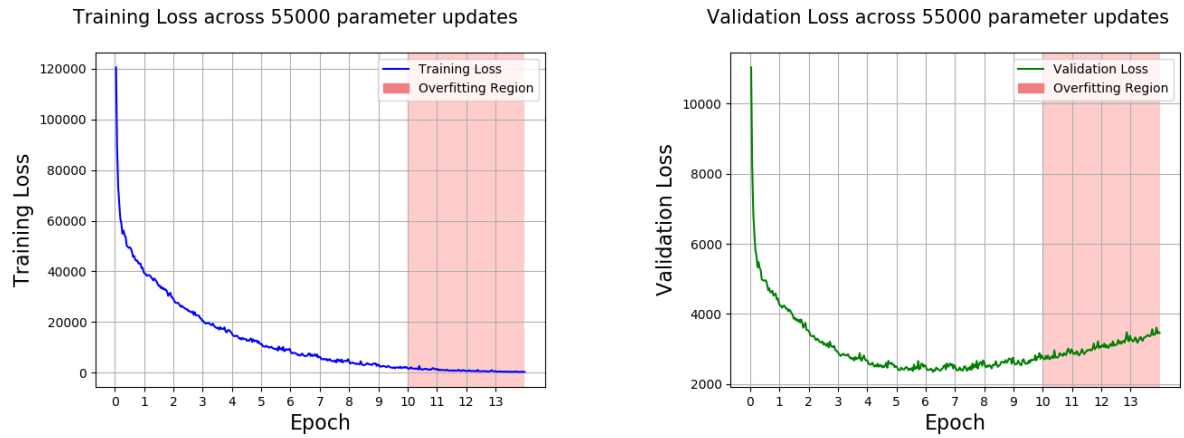


Figure 3: Plot of Loss vs Epoch for learning rate 0.001

- Hidden layer size: 300

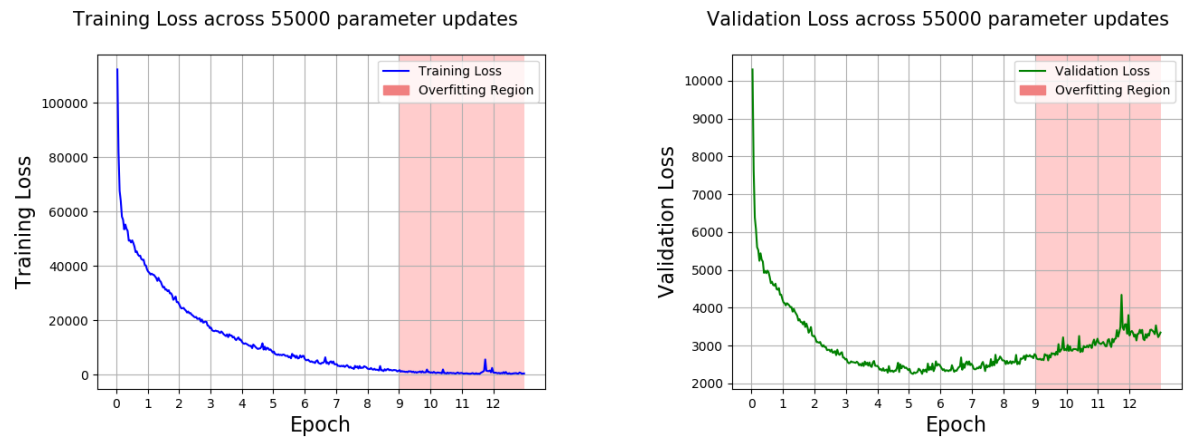


Figure 4: Plot of Loss vs Epoch for learning rate 0.001

2. Two hidden layers and the same size for each hidden layer with sigmoid activation function, cross entropy loss, Adam optimization and batch size 20.
varying the size of the hidden layer (50, 100, 200, 300)

- Each hidden layer size: 50

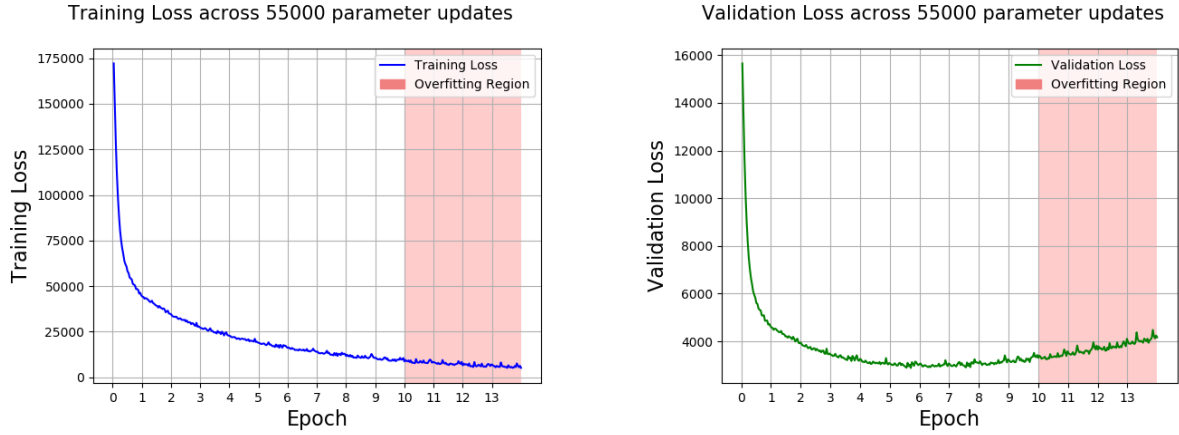


Figure 5: Plot of Loss vs Epoch for learning rate 0.001

- Each hidden layer size: 100

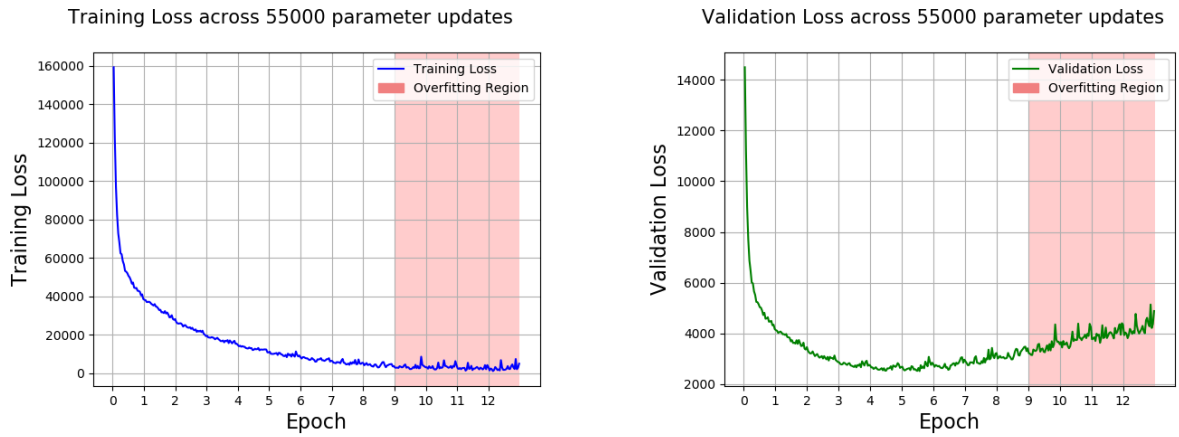


Figure 6: Plot of Loss vs Epoch for learning rate 0.001

- Each hidden layer size: 200

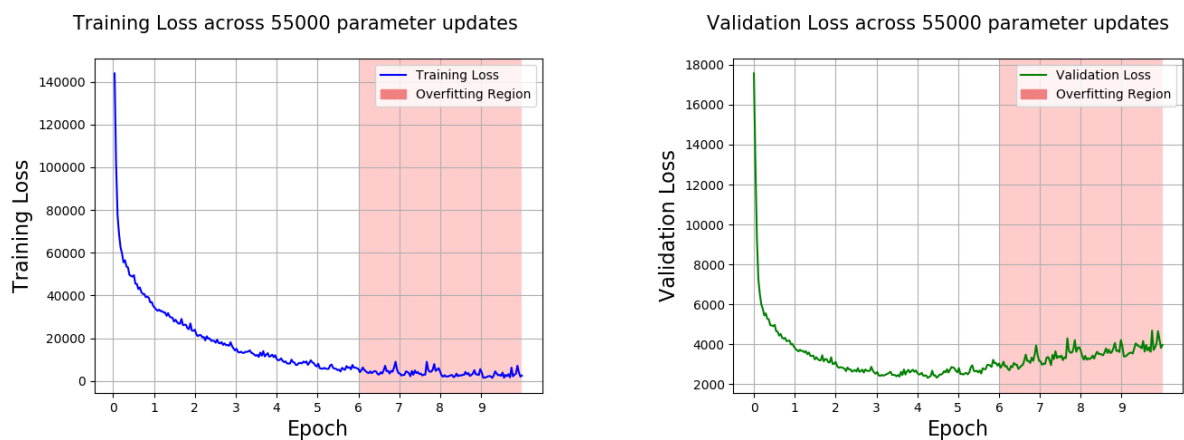


Figure 7: Plot of Loss vs Epoch for learning rate 0.001

- Each hidden layer size: 300

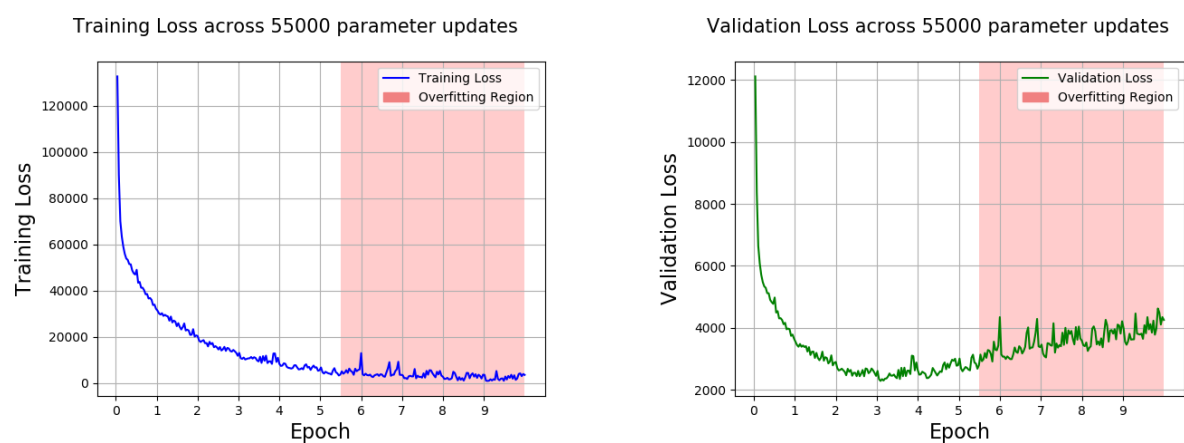


Figure 8: Plot of Loss vs Epoch for learning rate 0.001

3. Three hidden layers and the same size for each hidden layer with sigmoid activation function, cross entropy loss, Adam optimization and batch size 20.
varying the size of the hidden layer (50, 100, 200, 300)

- Each hidden layer size: 50

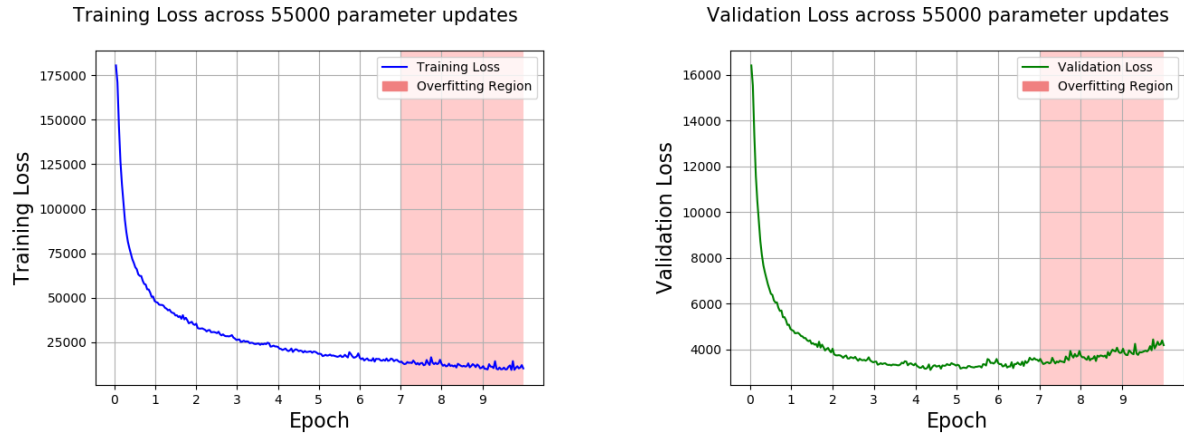


Figure 9: Plot of Loss vs Epoch for learning rate 0.001

- Each hidden layer size: 100

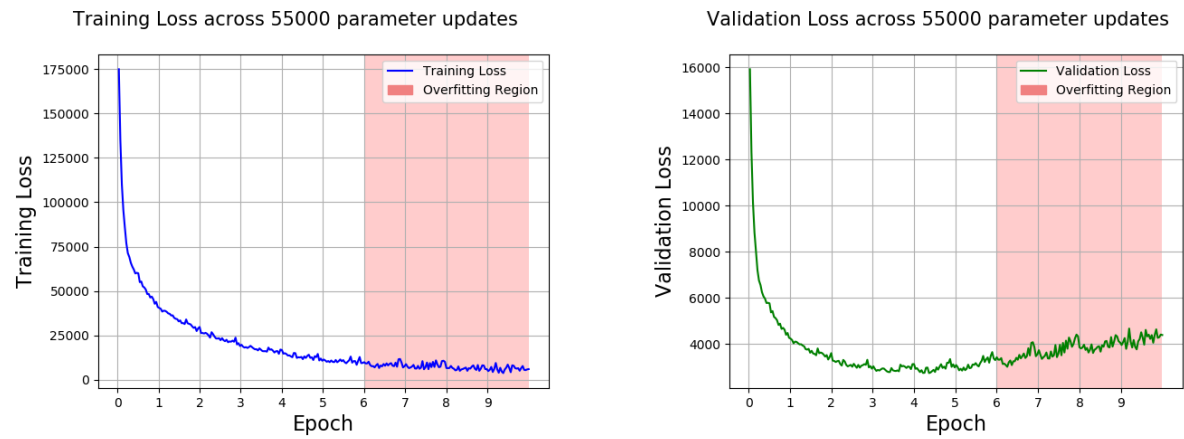


Figure 10: Plot of Loss vs Epoch for learning rate 0.001

- Each hidden layer size: 200

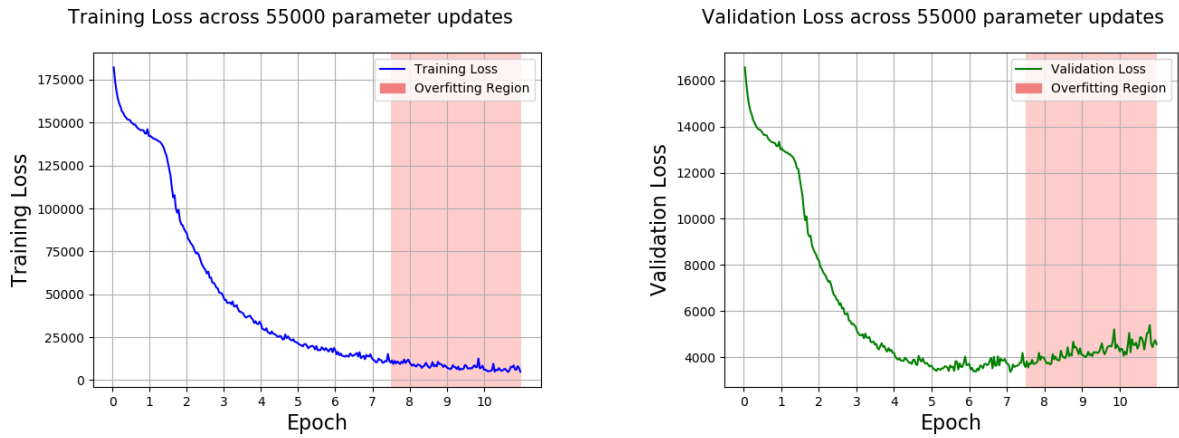


Figure 11: Plot of Loss vs Epoch for learning rate 0.001

- Each hidden layer size: 300

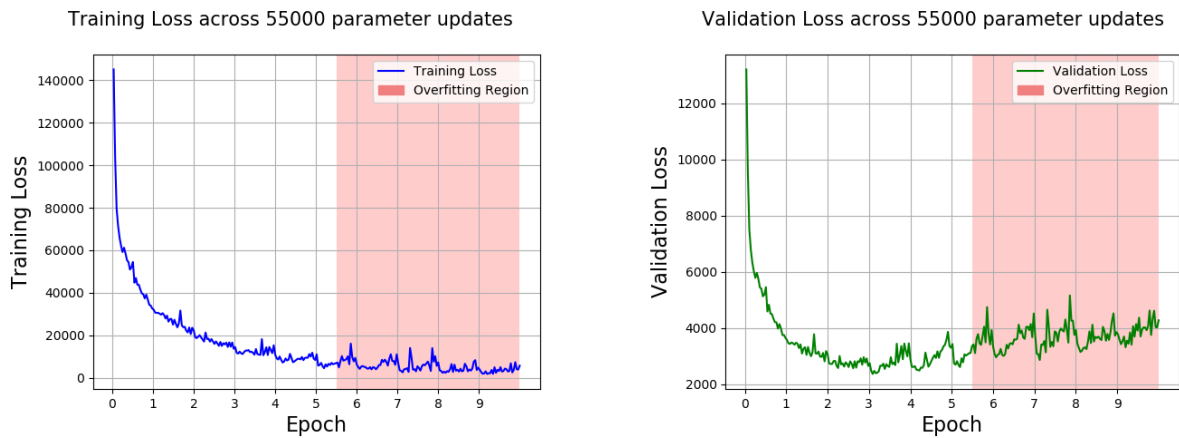


Figure 12: Plot of Loss vs Epoch for learning rate 0.001

4. Four hidden layers and the same size for each hidden layer with sigmoid activation function, cross entropy loss, Adam optimization and batch size 20.
varying the size of the hidden layer (50, 100, 200, 300)

- Each hidden layer size: 50

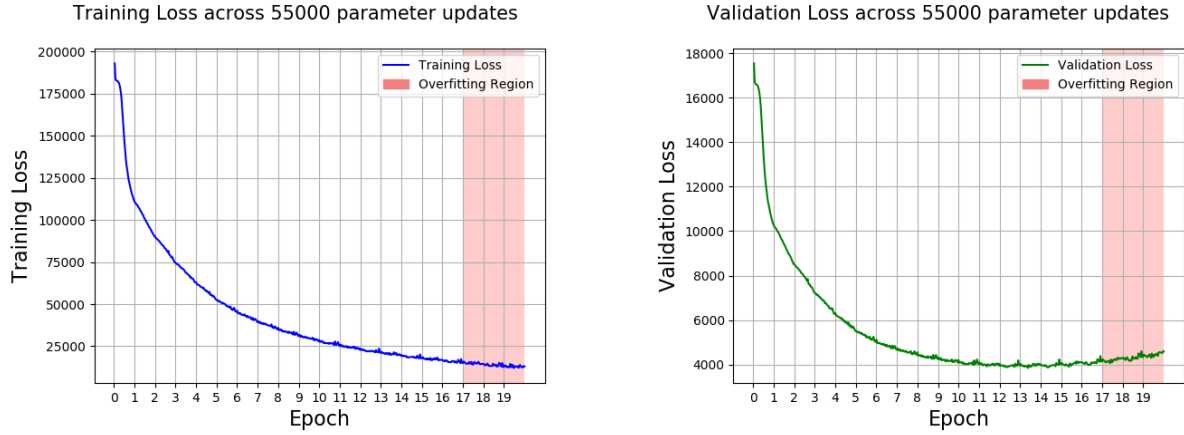


Figure 13: Plot of Loss vs Epoch for learning rate 0.0003

- Each hidden layer size: 100

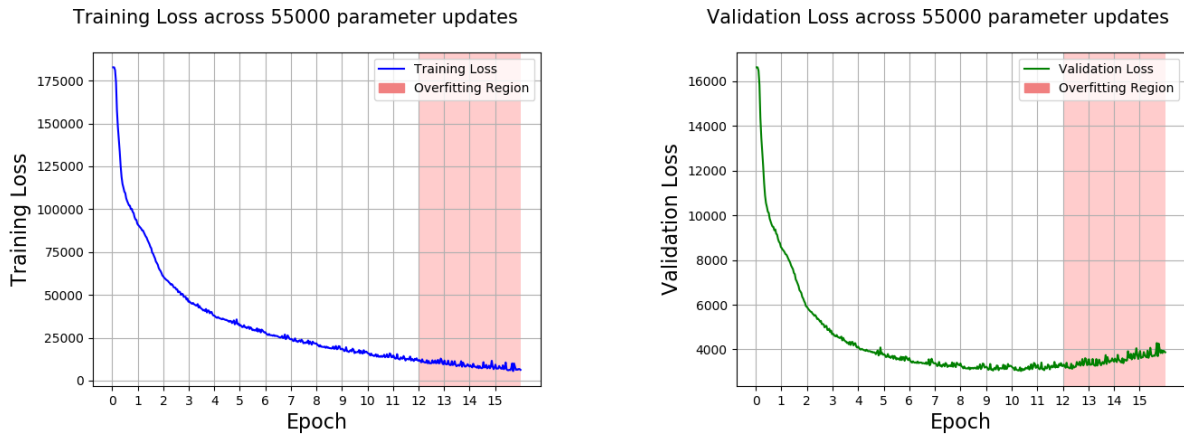


Figure 14: Plot of Loss vs Epoch for learning rate 0.0003

- Each hidden layer size: 200

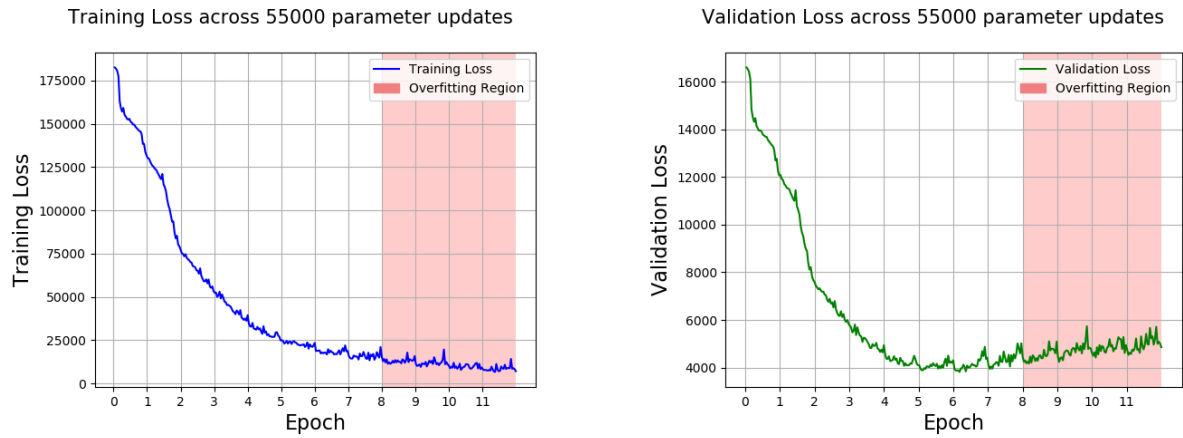


Figure 15: Plot of Loss vs Epoch for learning rate 0.001

- Each hidden layer size: 300

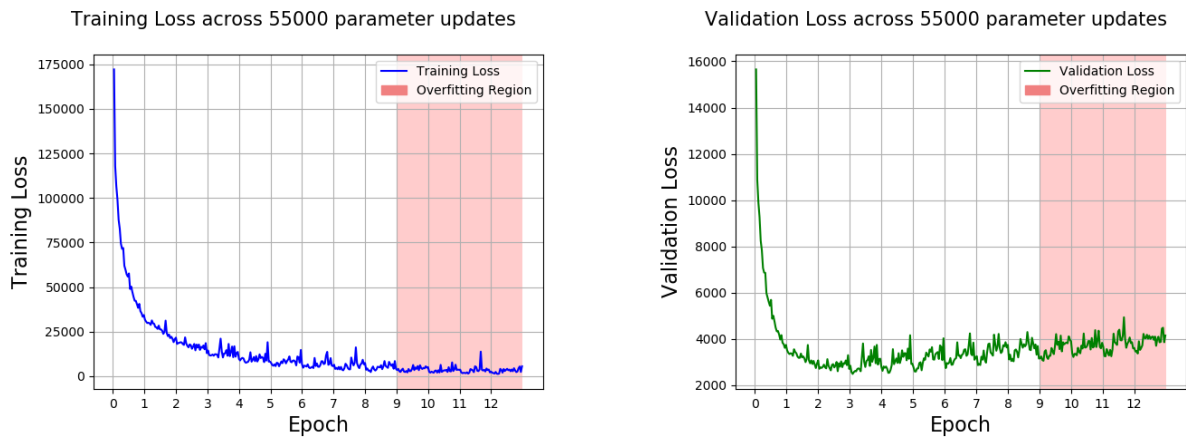


Figure 16: Plot of Loss vs Epoch for learning rate 0.001

5. Adam, NAG, Momentum, GD optimization [with 4 hidden layers and each layer having 300 neurons] (again sigmoid activation, cross entropy loss, batch size 20).

- Adam

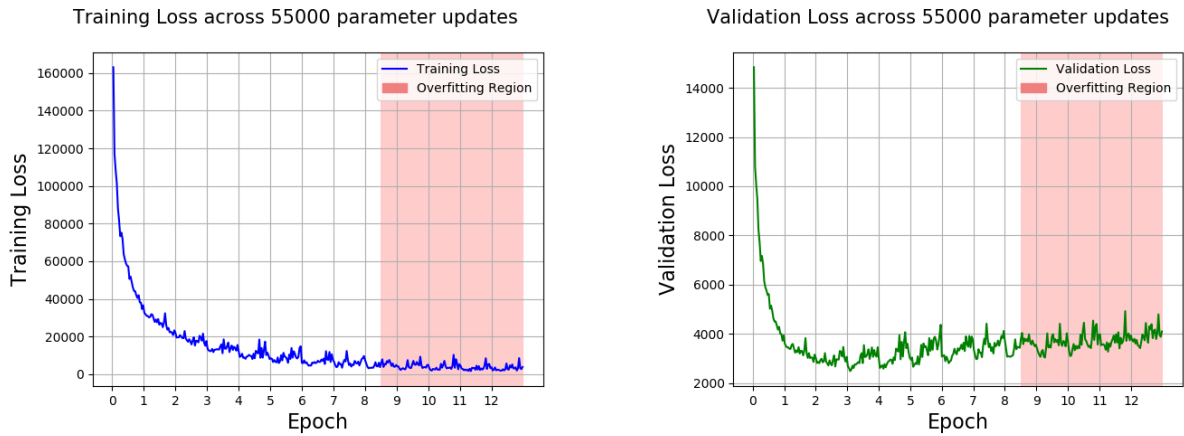


Figure 17: Plot of Loss vs Epoch for learning rate 0.001

- NAG

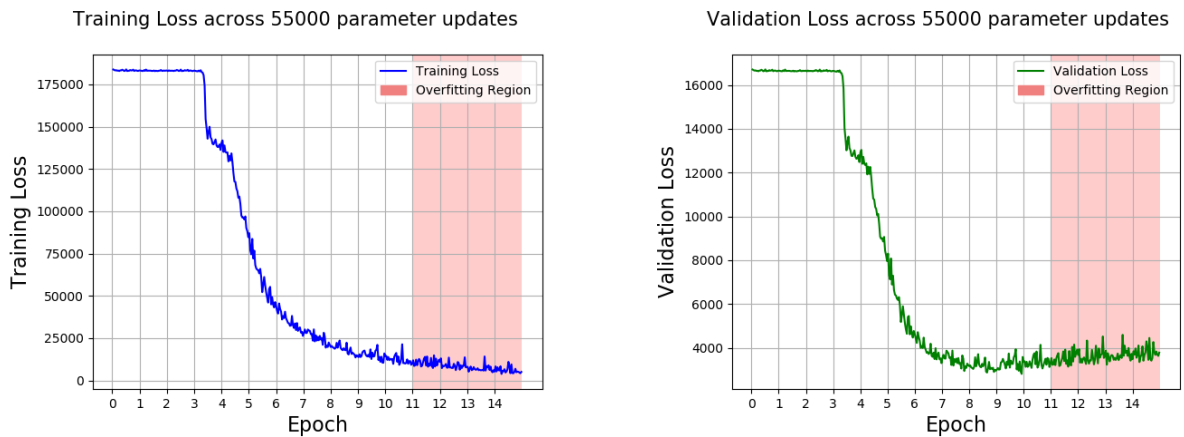


Figure 18: Plot of Loss vs Epoch for learning rate 0.001 and momentum 0.5

- Momentum

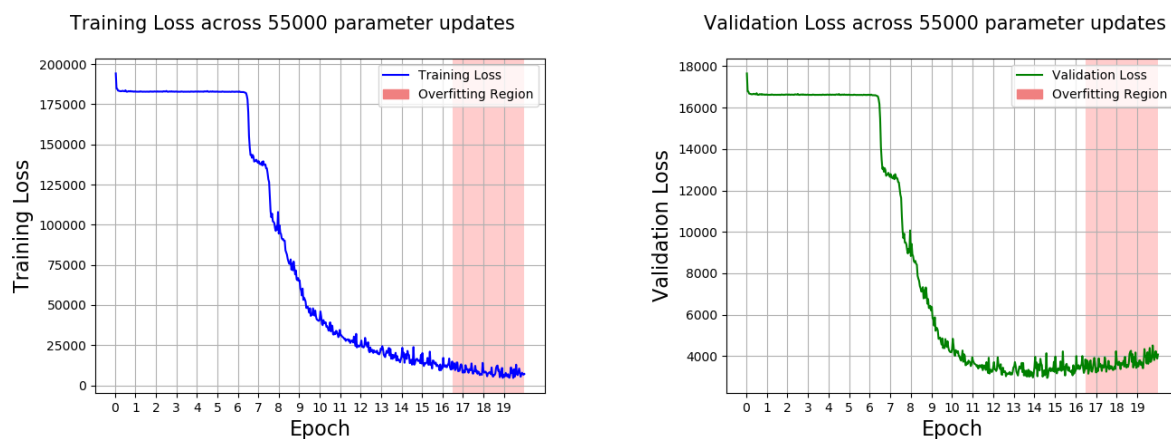


Figure 19: Plot of Loss vs Epoch for learning rate 0.001 and momentum 0.9

- GD

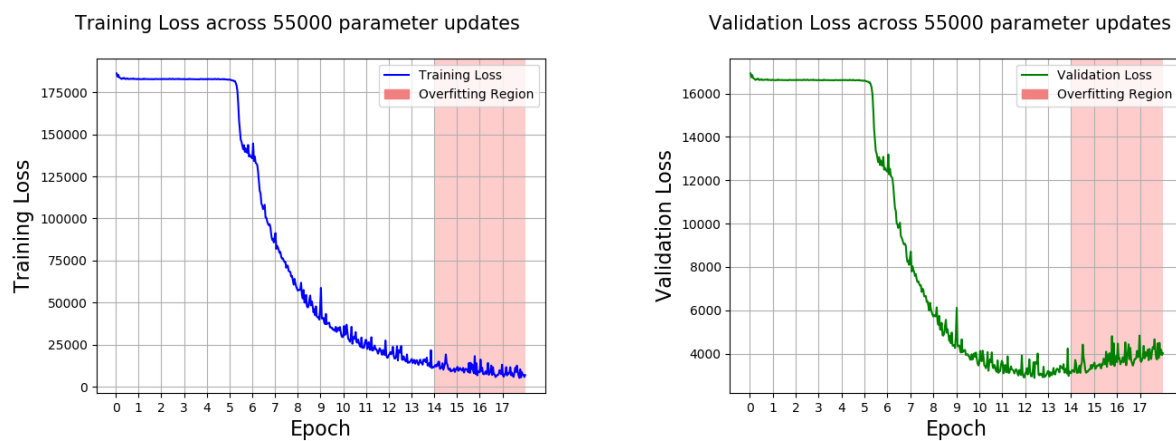


Figure 20: Plot of Loss vs Epoch for learning rate 0.01

6. sigmoid v/s tanh activation [Adam, 2 hidden layers, 100 neurons in each layer, batch size 20, cross entropy loss].

- sigmoid

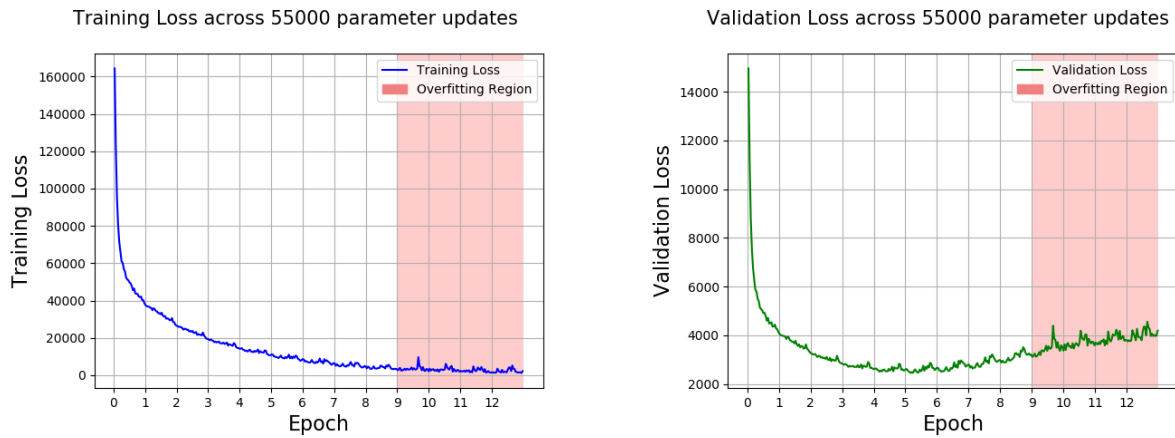


Figure 21: Plot of Loss vs Epoch for learning rate 0.001

- tanh

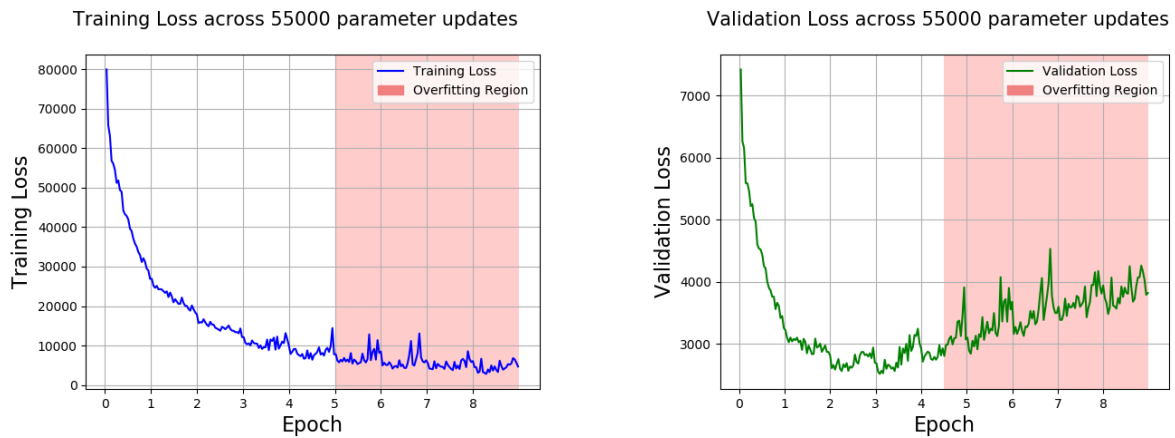


Figure 22: Plot of Loss vs Epoch for learning rate 0.001

7. cross entropy loss v/s squared error loss [Adam, 2 hidden layers, 100 neurons in each layer, batch size 20, sigmoid activation]

- Cross entropy loss

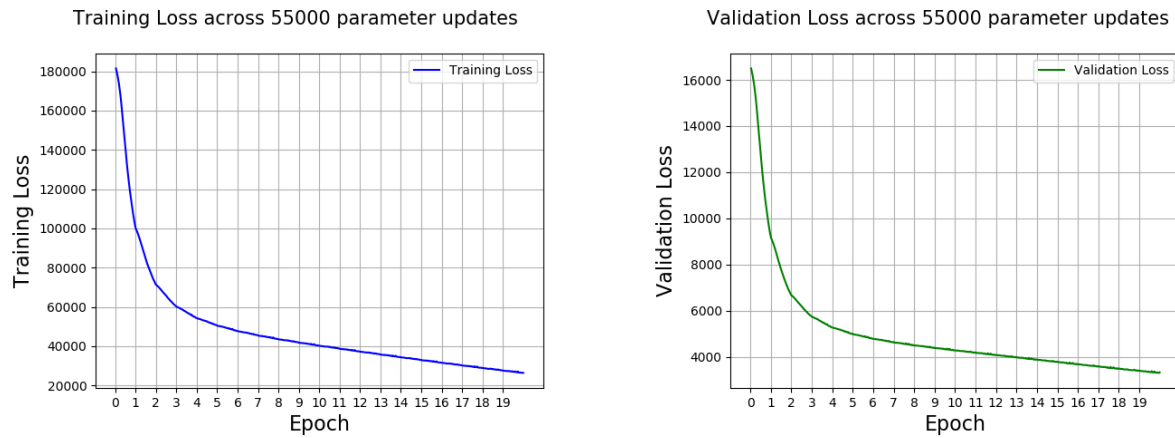


Figure 23: Plot of Loss vs Epoch for learning rate 0.0001

- Squared error loss

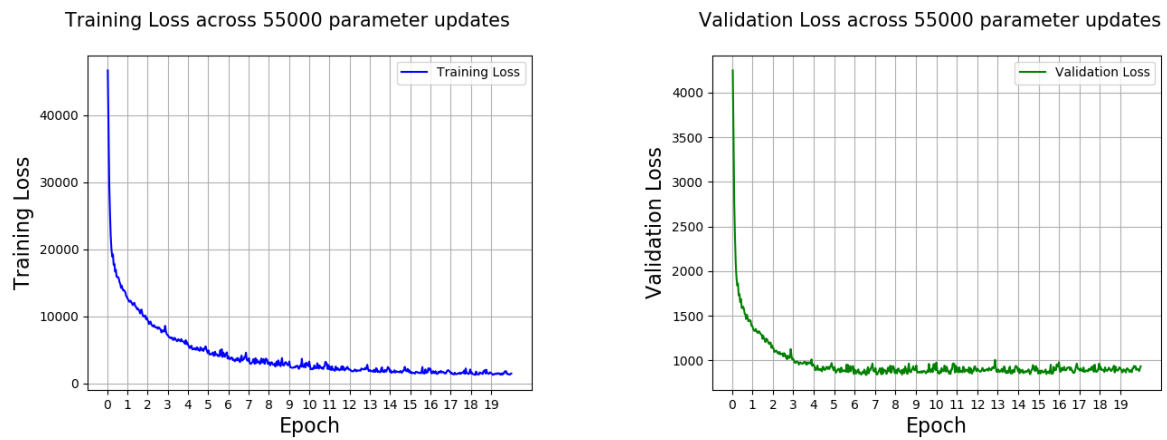


Figure 24: Plot of Loss vs Epoch for learning rate 0.001

8. Batch size: 1,20,100,1000 [Adam, 2 hidden layers, 100 neurons in each layer, sigmoid activation, cross entropy loss]

- Batch size : 1

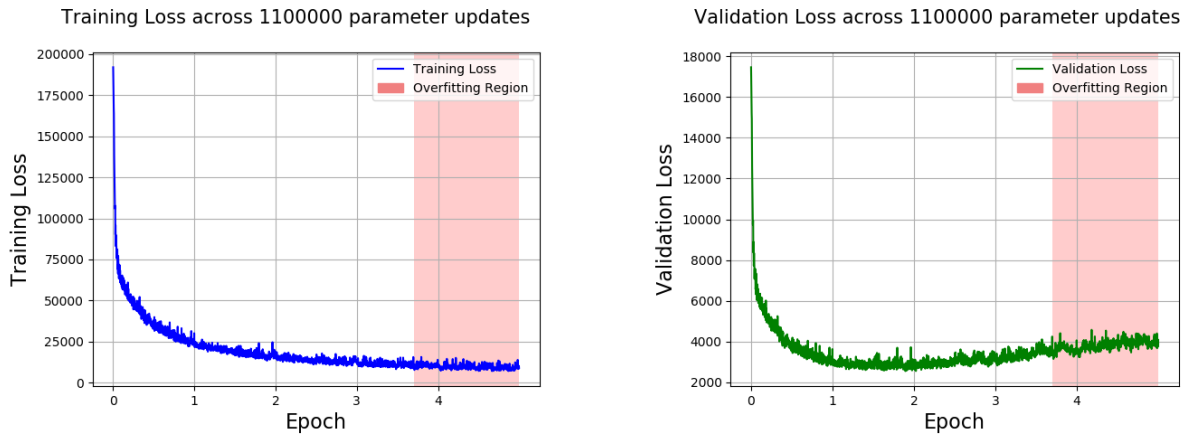


Figure 25: Plot of Loss vs Epoch for learning rate 0.001

- Batch size : 20

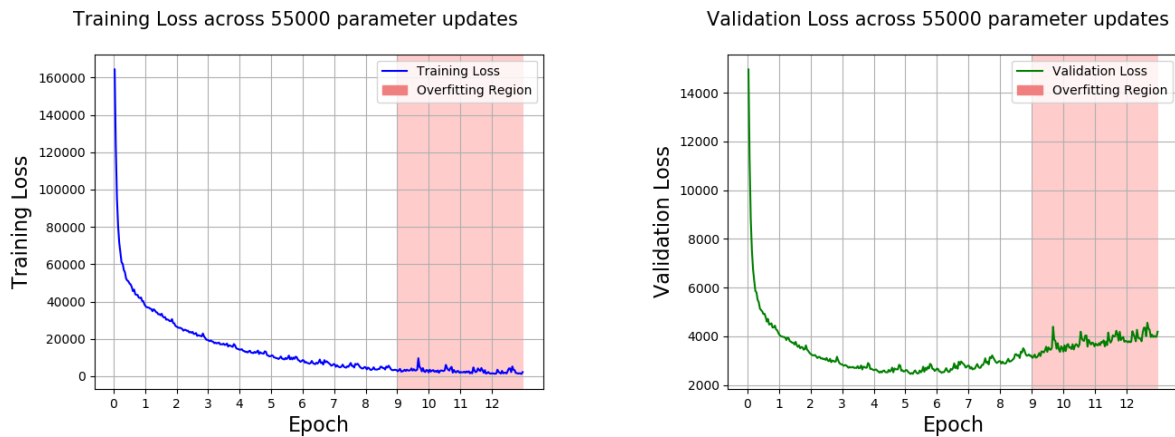


Figure 26: Plot of Loss vs Epoch for learning rate 0.001

- Batch size : 100

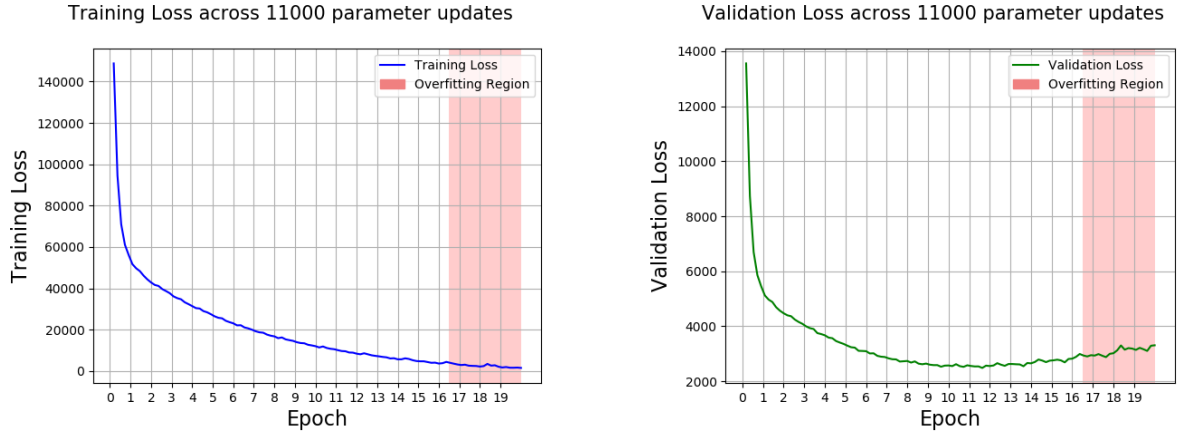


Figure 27: Plot of Loss vs Epoch for learning rate 0.001

- Batch size : 1000

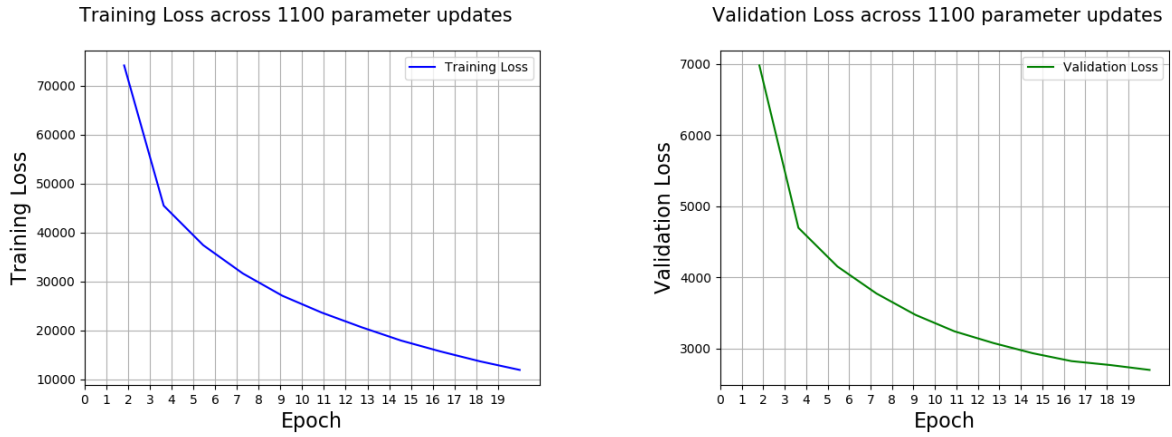


Figure 28: Plot of Loss vs Epoch for learning rate 0.001

3 Observations

- For models with same number of hidden layers, as the number of neurons increases in the hidden layer, the model converges faster(Fig. 1-16).
- For models with same number of hidden layers, mostly the models with higher number of neurons per layer achieved better accuracy.(Fig. 1-16).
- For models with same number of neurons per hidden layers, the models with 3 hidden layers achieved best accuracy.(Fig. 1-16).

- Loss curve for simpler models is consistently smoother than the ones for complex models(Fig. 1-16).
- Adam turns out to be the best optimization algorithm and gave the fastest convergence with best accuracy(Fig. 17-20).
- tanh as activation function shows faster convergence and better accuracy but has rougher loss curve than the sigmoid activation function(Fig. 21,22).
- For larger batch size, the training takes less number of parameter updates for the model to converge although it is taking more number of epochs at the same time to get to the convergence point(Fig. 25,28).