

### **Mini Batch Size:**

Minibatch size – 50

### **Parameter Initialization:**

I chose 'He Normalization' method with normal distribution to initialize my weight and bias parameters with a mean = 0 and standard deviation =  $\sqrt{2 / (\text{Number of neurons in previous layer})}$

### **Activation Function & Learning Rates :**

The learning rate values chosen were

Activation: ReLU

- 0.01
- 0.001
- 0.09

Activation: tanH

- 0.05
- 0.02
- 0.1

### **Plots for different configurations:**

#### **Plot 1:**

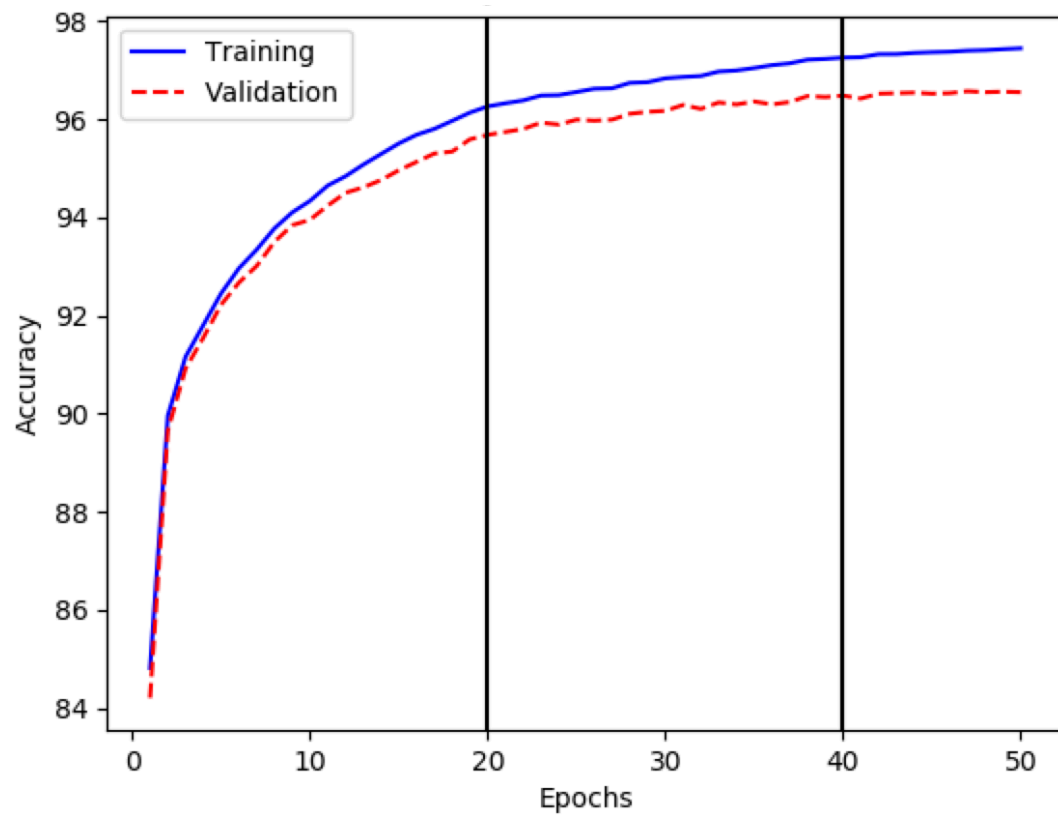
Model – 784, 512, 10

Activation – ReLu

Learning Rate – 0.01

Training Accuracy: 97.44

Validation Accuracy: 96.55



Plot 2:

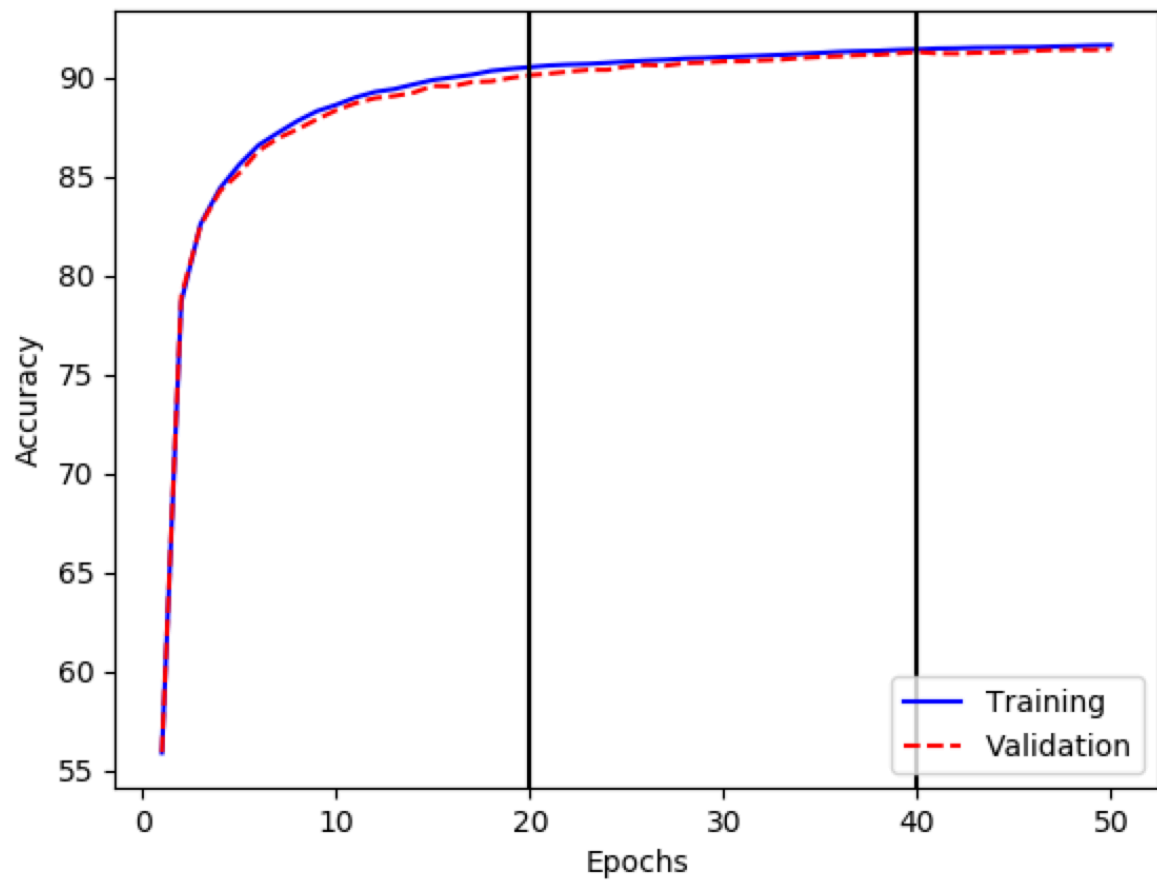
Model – 784, 512, 10

Activation – ReLu

Learning Rate – 0.001

Training Accuracy: 91.61

Validation Accuracy: 91.42



Plot 3:

N = 0.09

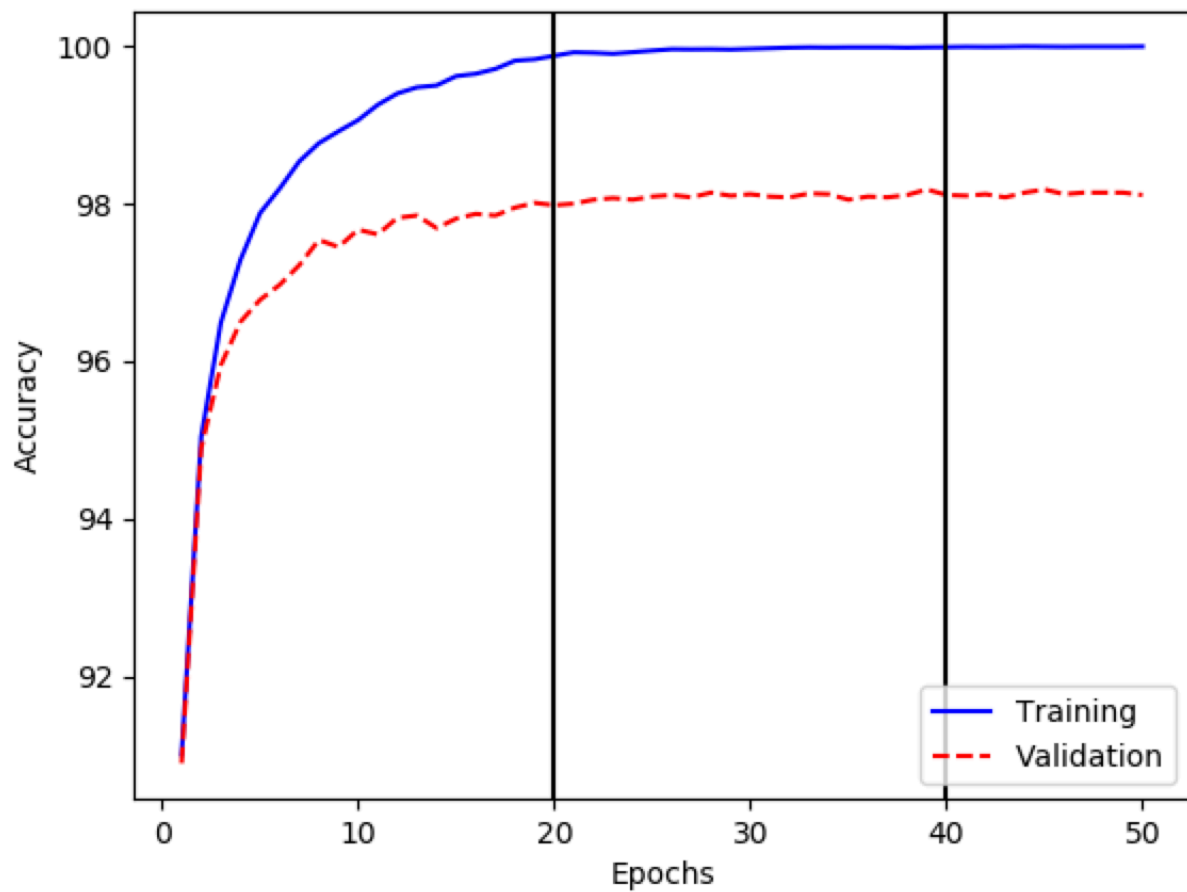
Batch = 50

Epoch = 50

Act – ReLu

Training Accuracy : 99.99

Validation Accuracy: 98.11



Plot 4:

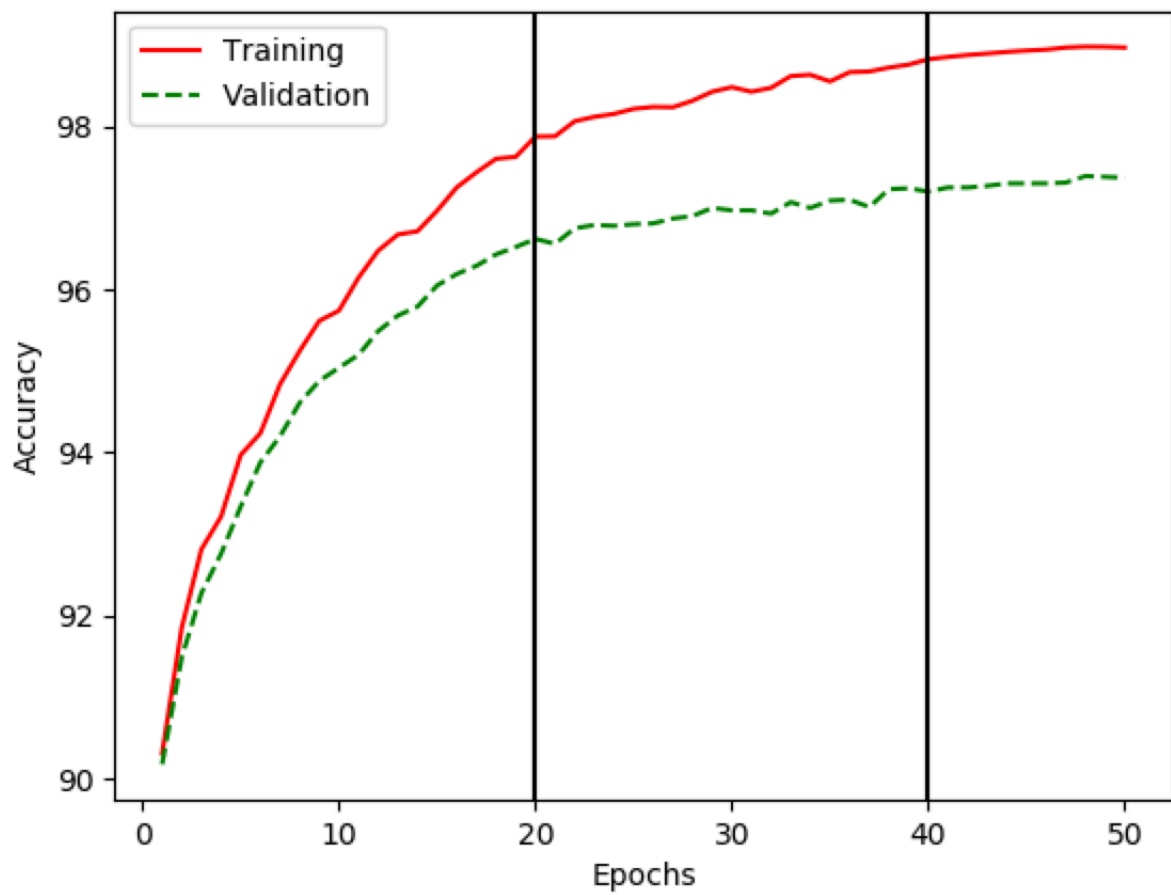
Model – 784, 512, 10

Activation – tanH

Learning Rate – 0.05

Training Accuracy: 98.97

Validation Accuracy: 97.37



Plot 5:

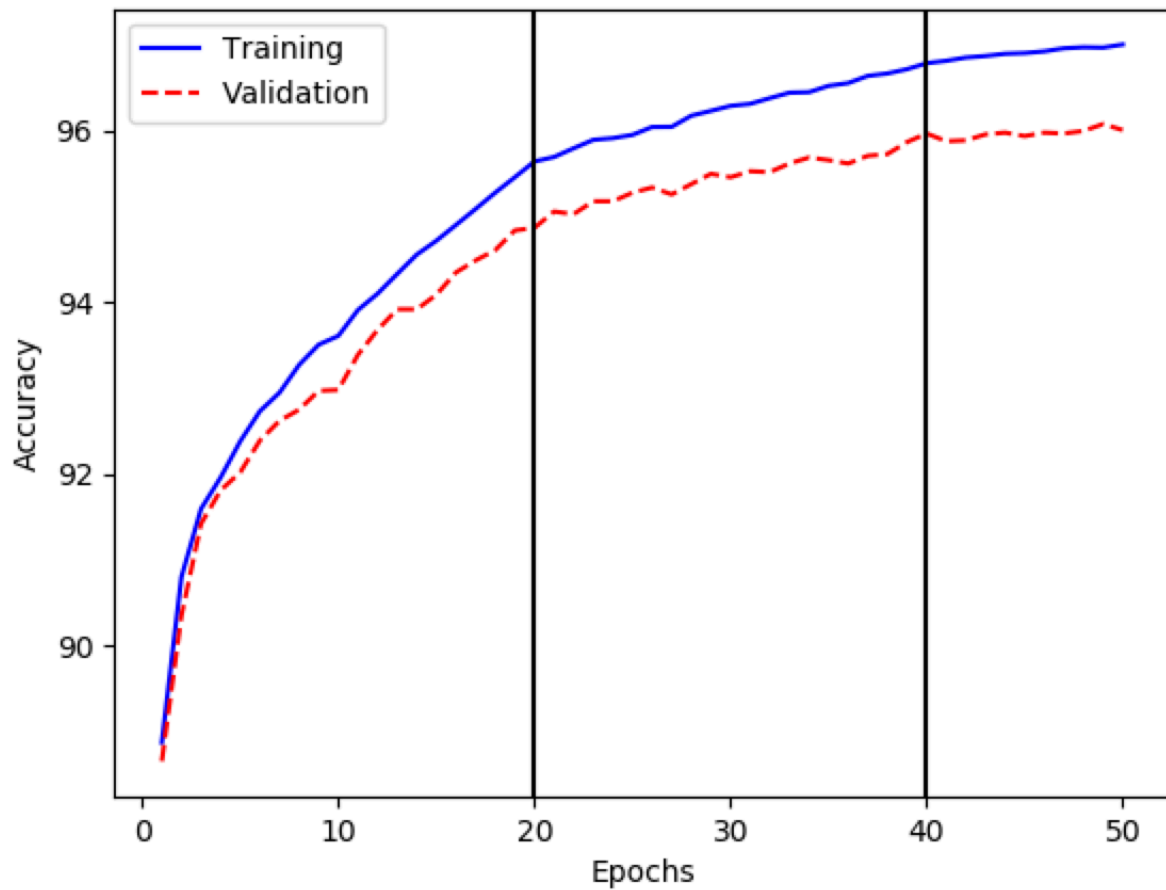
Model – 784, 512, 10

Activation – tanH

Learning Rate – 0.02

Training Accuracy: 97.00

Validation Accuracy: 96.00



Plot 6:

Model – 784, 512, 10

Activation – tanh

Learning Rate – 0.1

N = 0.1

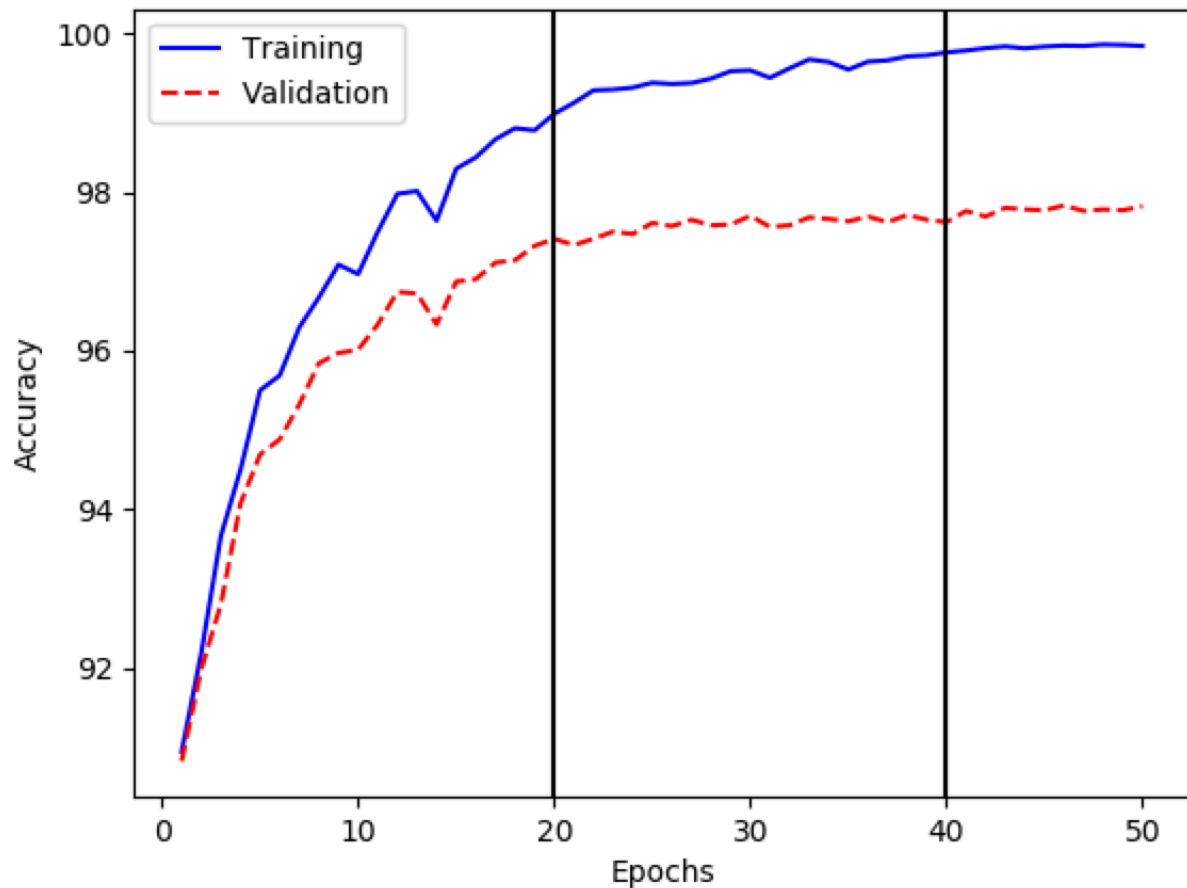
Batch = 50

Epoch = 50

Act – tanH

Training Accuracy : 99.84

Validation Accuracy: 97.82



### **Final Configuration:**

The final configuration chosen was a input layer of 784 neurons, a hidden layer of 512 neurons and an output layer of 10 neurons (Plot 3)

Batch Size – 50

Epoch – 50

Learning Rate -0.09

Activation in Intermediate Layers – ReLu

Activation in Output Layer – Softmax

Parameter Initialization – He Normalization

Final Training Accuracy : 99.99

Final Test Accuracy : 98.26