

# Deep Learning Assignment - 1

## Report

---

### **Question 1.**

Link of code

(<https://colab.research.google.com/drive/1PLUmidcMTshL29n0T-atacGZbQFI-3os?authuser=1#scrollTo=quSI8iWhLOL>)

Implement a neural network and utilize the CIFAR-10 dataset for the analysis.

#### **Different parameters used in the Problem:**

Length of training data is : 50000

Length of testing data is : 10000

After further splitting the data into training and validation data set

Length of Training data is :49000

Length of Validation data is :1000

Batch size is : 4000

Learning rate is = 0.0001

Loss function is Cross Entropy Loss

Using 5 layers :

1 input layer

3 hidden layer

1 output layer

Total no. of Epoch used is = 10

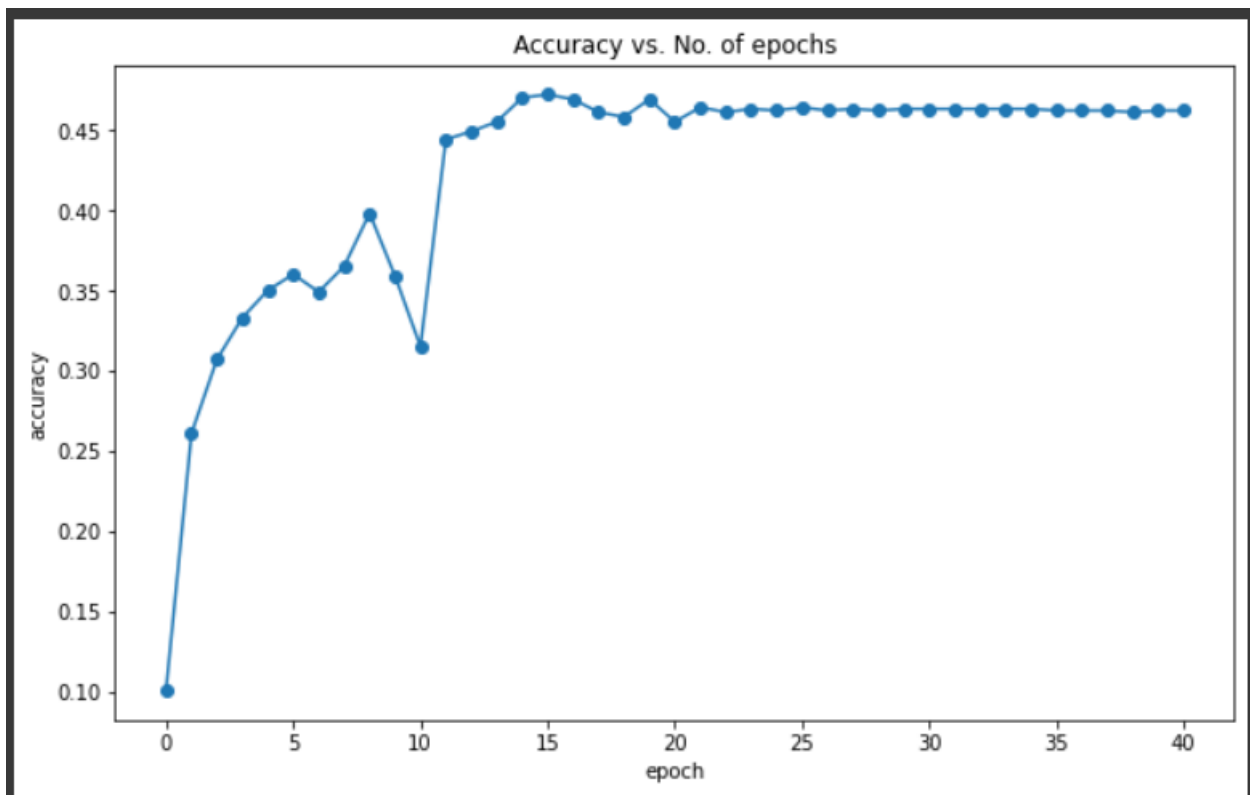
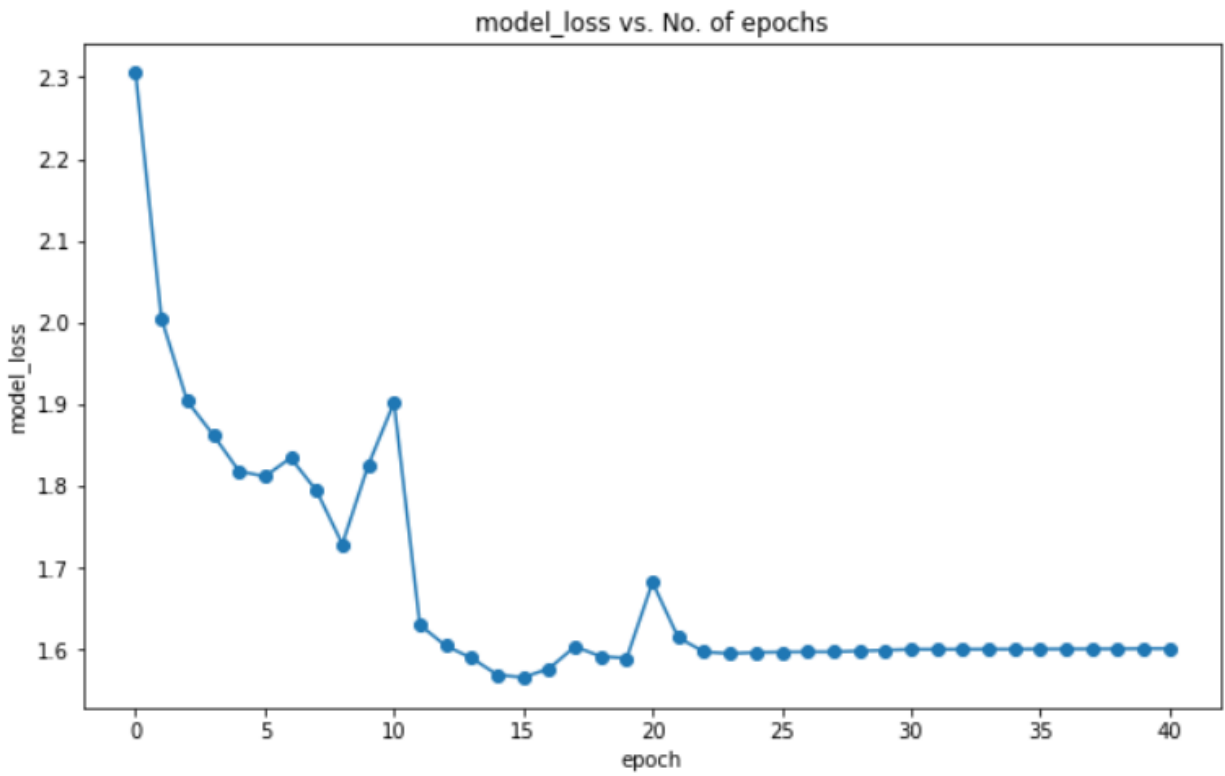
1. ) Utilize various activation functions like sigmoid, tanh and critique the performance in each case.

#### **Using “tanh” activation function**

Final training accuracy is = 46.20%

Testing accuracy is = 47.68%

And the plots of the Error and Accuracy are given below:

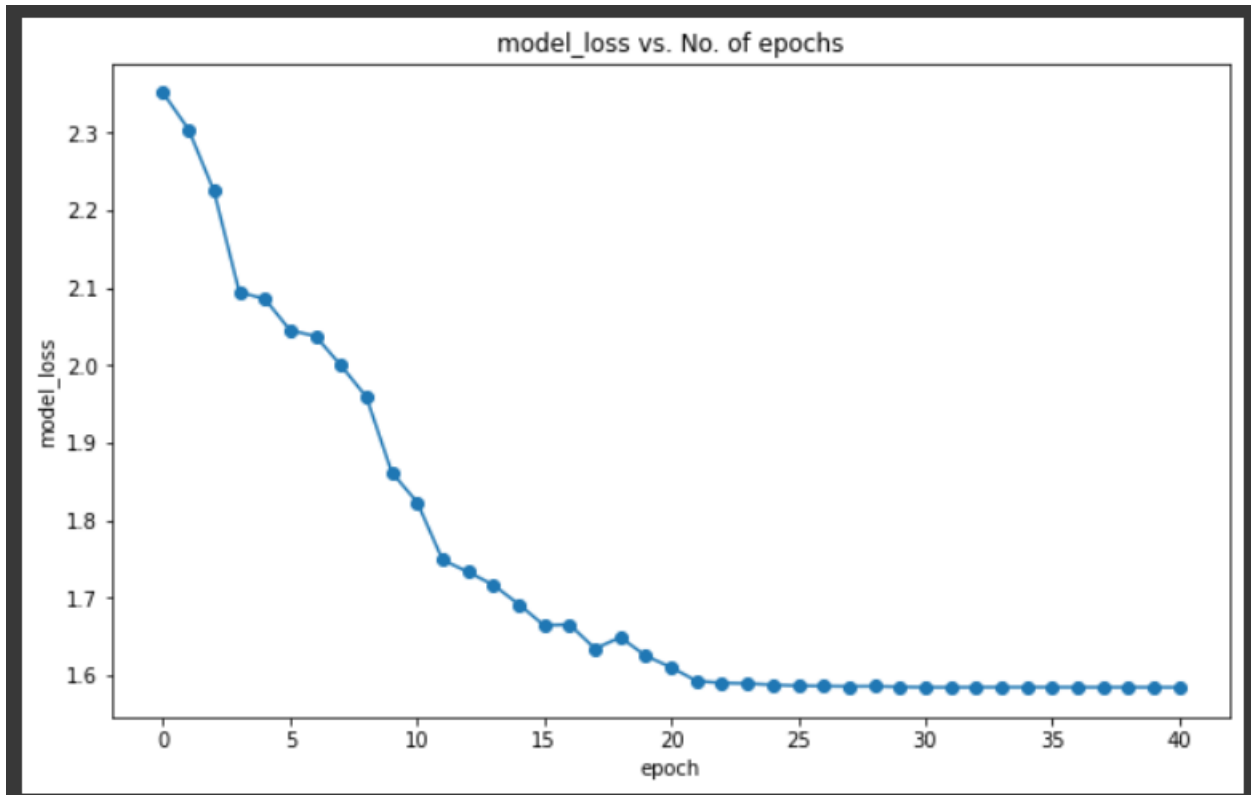


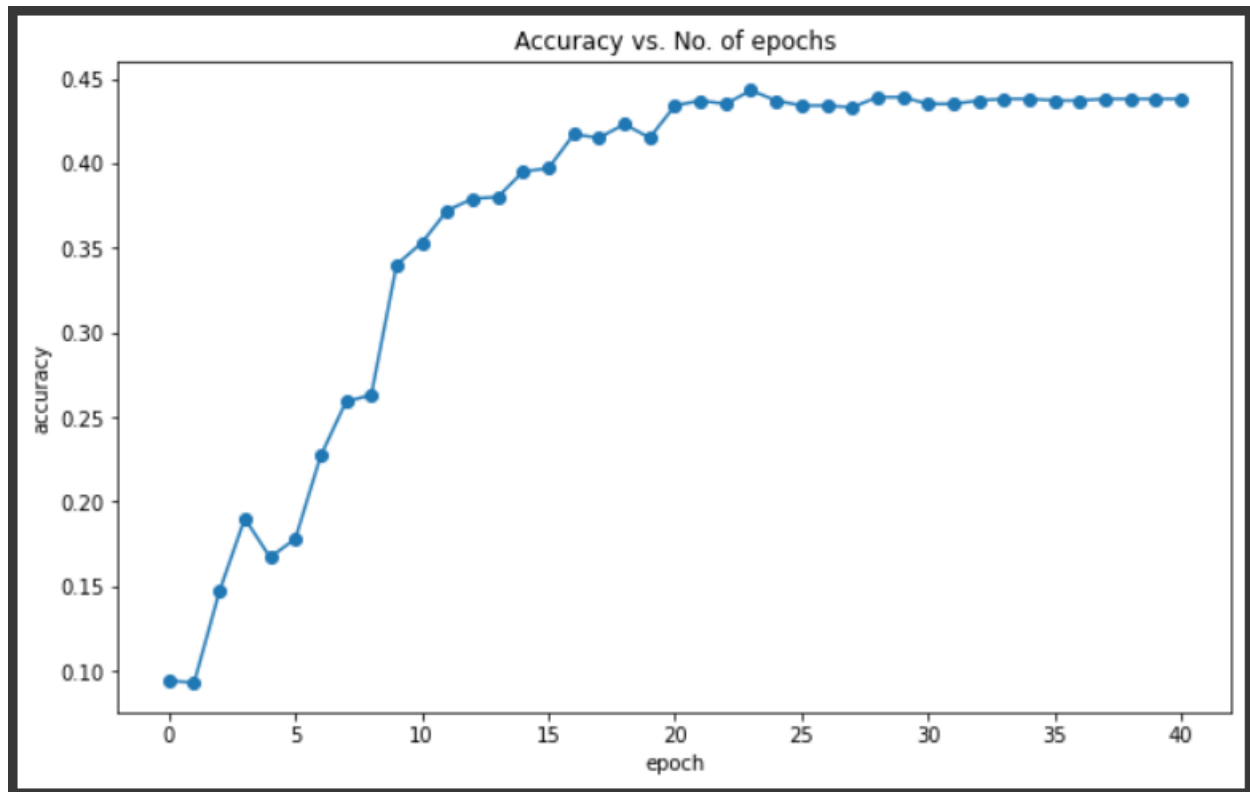
### Using “Sigmoid” activation function

Final training accuracy is = 43.80%

Testing accuracy is = 44.33%

And the plots of the Error and Accuracy are given below:





2.) Increase the depth of the given network by adding more Fully-Connected layers till the point you encounter the vanishing gradient problem. With the help of the results, mention how to identify it.

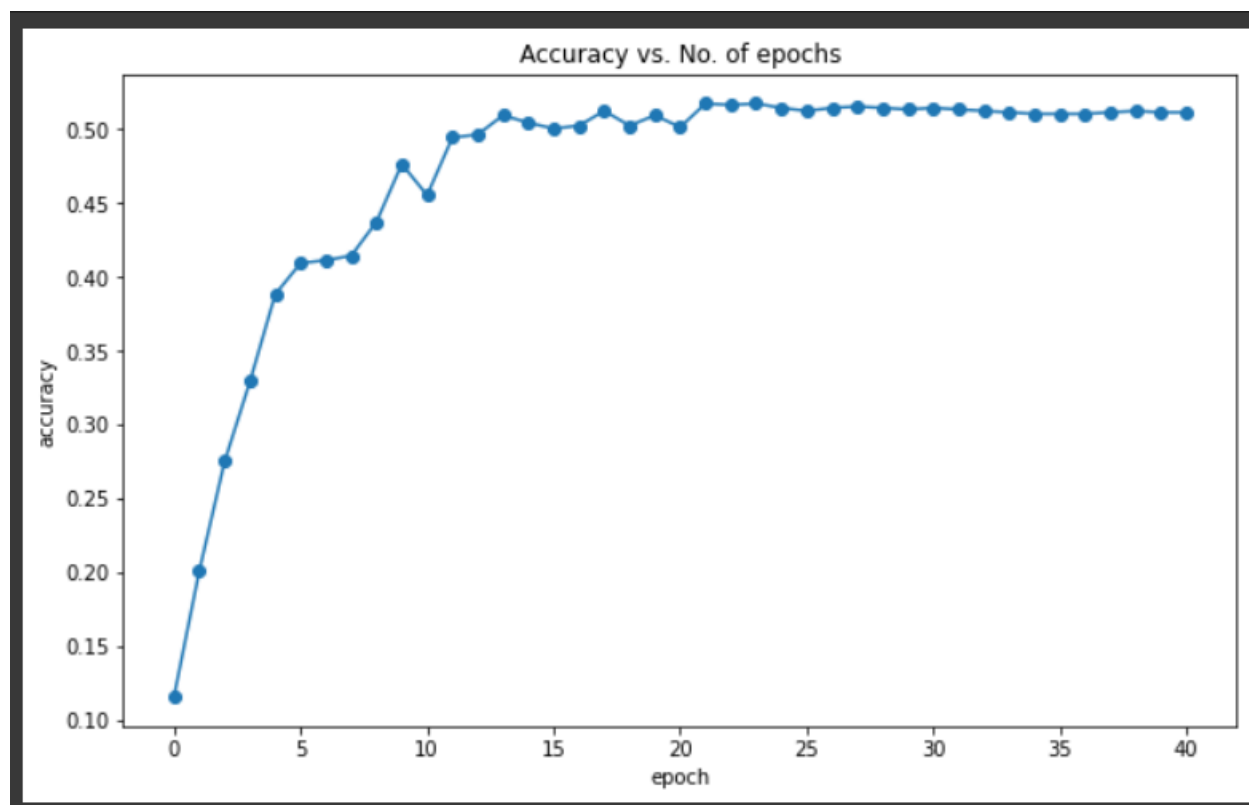
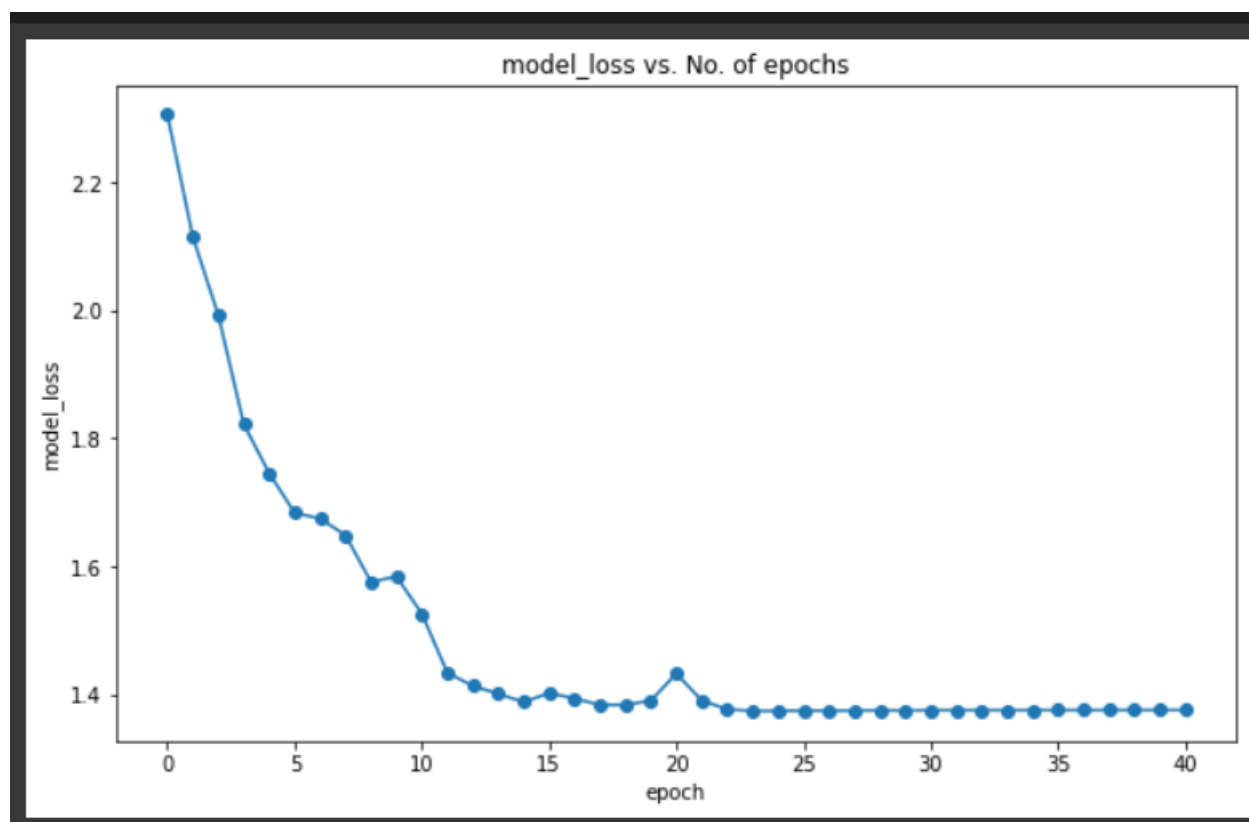
3.) Suggest and implement methods to overcome the above problem.

Answer : To get rid of the vanishing gradient problem we will be using the “Relu” Activation Function and the result after implementing the “Relu” activation function we will get the following results :

Final training accuracy is = 51.11%

Testing accuracy is = 54.51%

And the plots of the Error and Accuracy are given below:



## **Question 2.**

Link of

code(<https://colab.research.google.com/drive/1ns3gH84SRZE1pj9CO96UWo83HoJgLUhb?authuser=1#scrollTo=wTIPL9Bd2Plm>)

Implement a neural network on the Gurmukhi dataset and implement the following regularization techniques from scratch:

1. L-1 regularization
2. L-2 regularization
3. Dropout

Compare the performance of the above techniques and mention reasons to support your answer. You are free to utilize PyTorch's inbuilt functions for implementing activation and loss functions. However, various regularization techniques must be implemented from scratch without the support of any library. Also, implement gradient checking (from scratch) to verify the values of gradients during backpropagation. (refer to this hint for gradient checking)

### **Different parameters used in the Problem:**

Length of training data is : 1000

Length of testing data is : 178

After further splitting the data into training and validation data set

Length of Training data is :900

Length of Validation data is :100

Batch size is : 50

Learning rate is = 0.0001

Loss function is Cross Entropy Loss

Using 5 layers :

1 input layer

3 hidden layer

1 output layer

Total no. of Epoch used is = 20

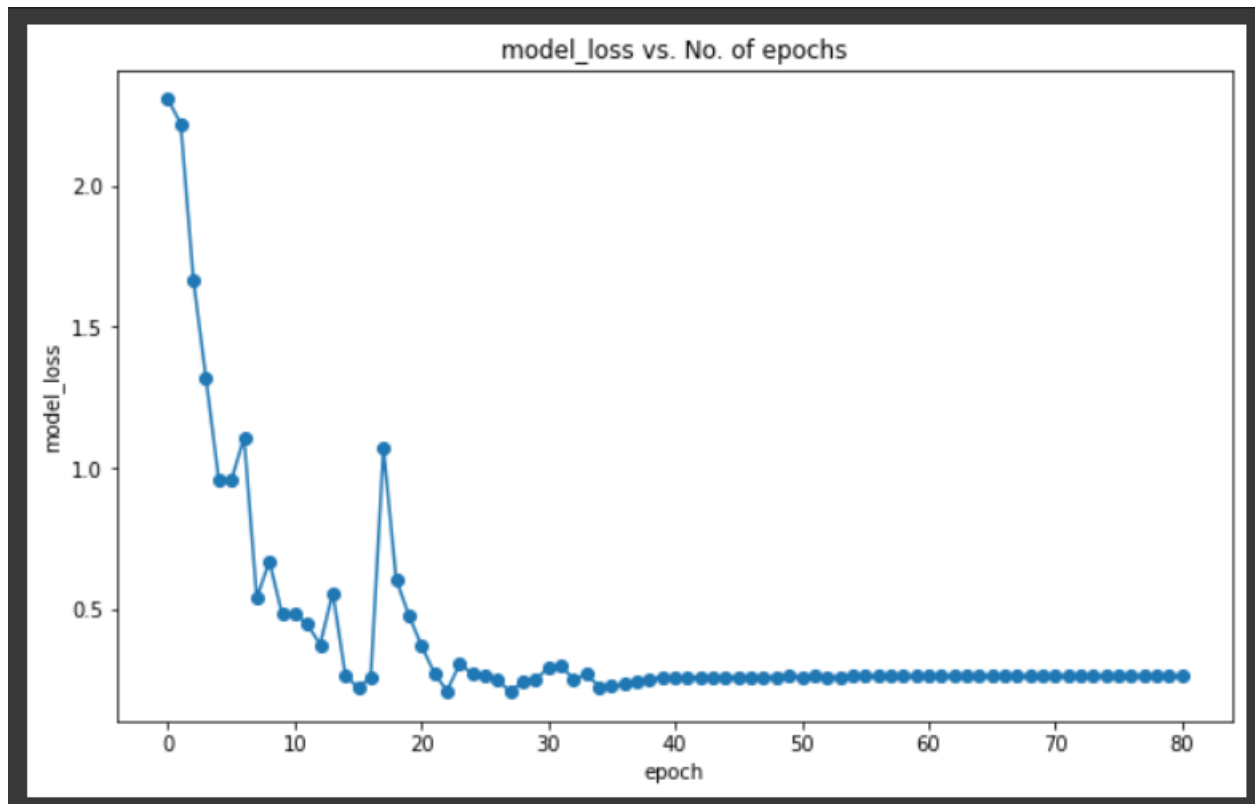
### **1. L-1 regularization**

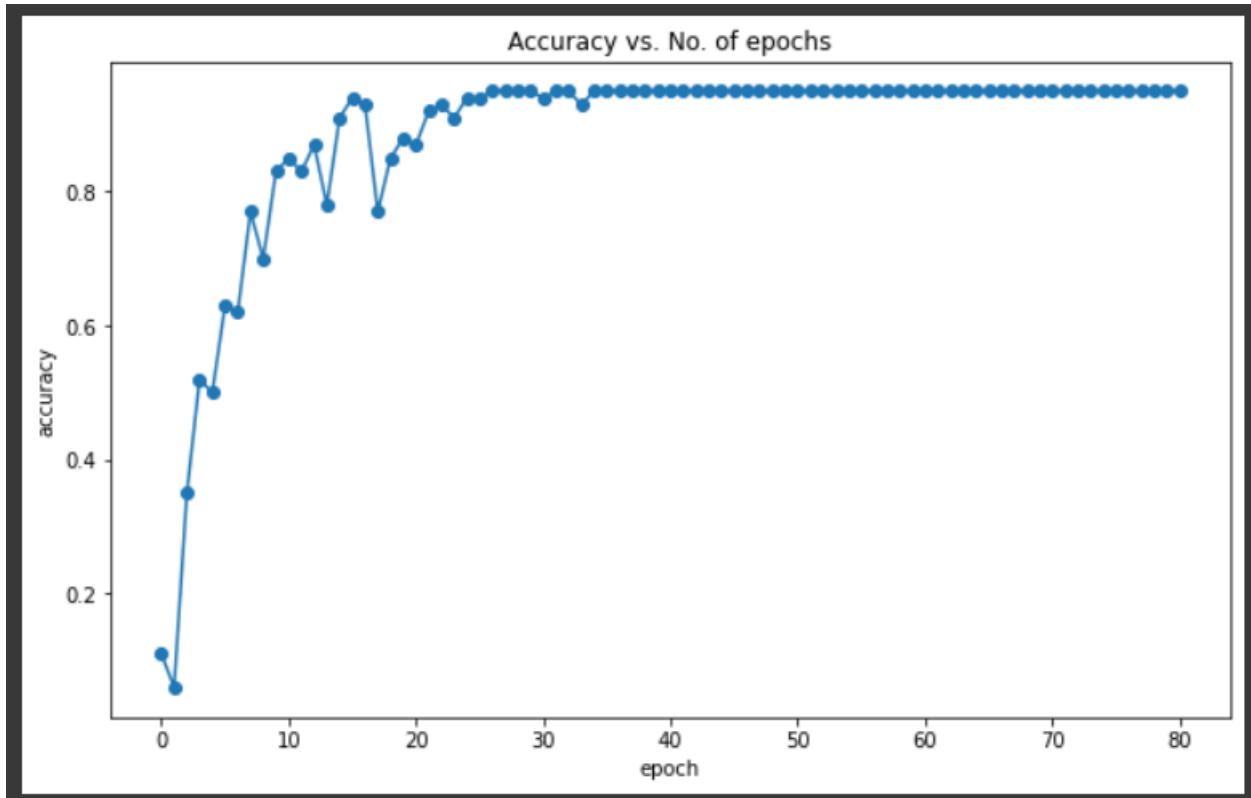
After using the L1 regularization we get the following results :

Final training accuracy is = 95%

Testing accuracy is = 94.71%

And the plots of the Error and Accuracy are given below:





## 2. L-2 regularization

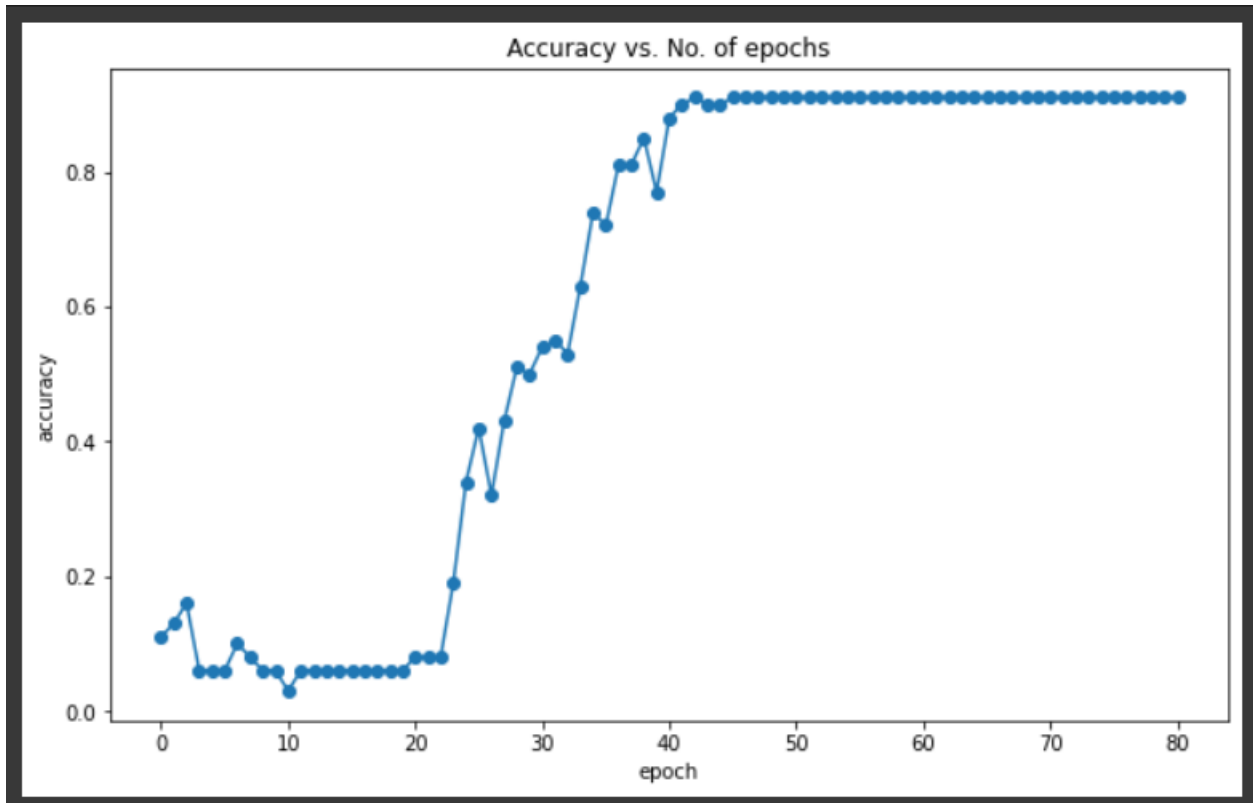
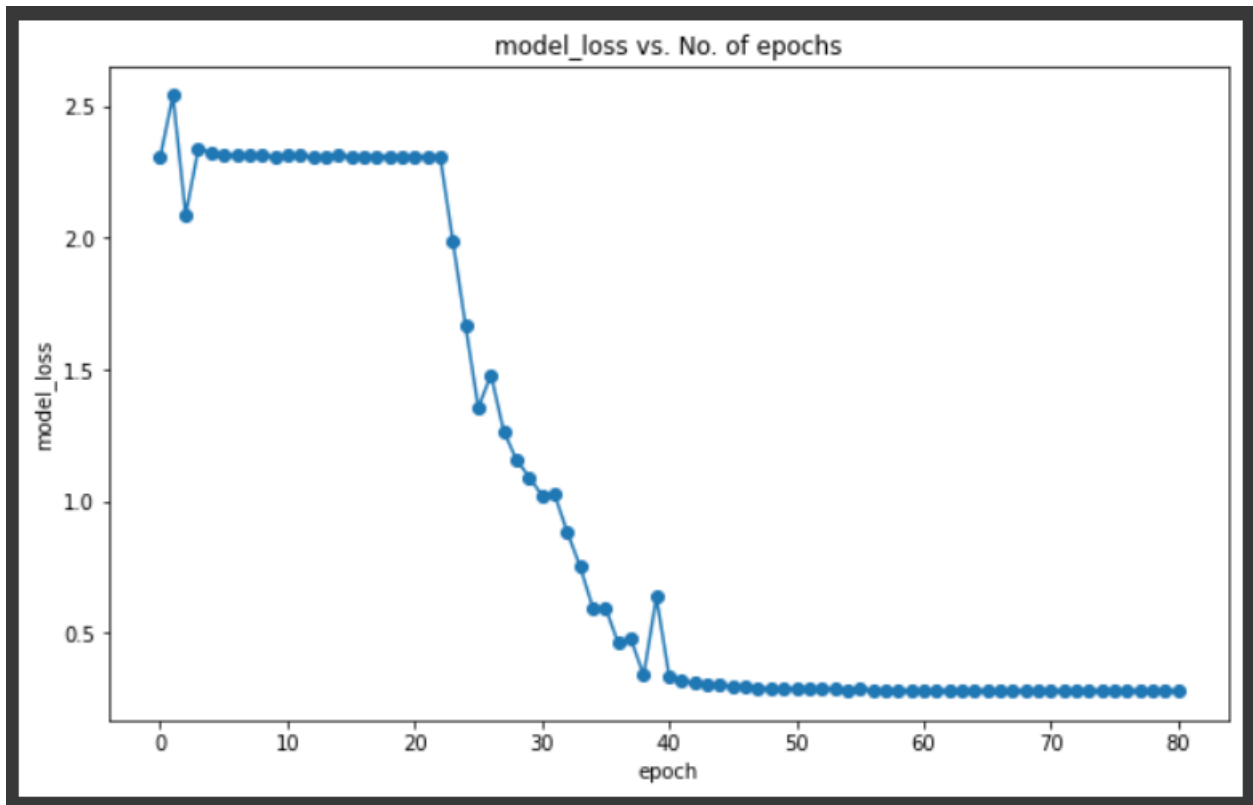
After using the L2 regularization we get the following results :

Final training accuracy is = 91%

Testing accuracy is = 81.51%

And the plots of the Error and Accuracy are given below:





### 3. Dropout technique

After using the L1 regularization we get the following results :

Final training accuracy is = 91%

Testing accuracy is = 88.43%

And the plots of the Error and Accuracy are given below:

